Lecture 1 - Introduction and Word Vectors

§1 The Course

§2 Human Language and Word Meaning

- Example: xkcd cartoon
  - information function
  - social function
- human being vs. or orangutan
  - language
  - networking
  - writing

1. **How do we represent the meaning of a word?**

   Definition: **meaning**

   - the idea that is represented by a word, phrase, etc.
   - the idea that a person wants to express by using words, signs, etc.
   - the idea that is expressed in a work of writing, art, etc.

   Linguistic way of thinking of meaning:

   - **Denotational Semantics**: *Signifier (symbol)* $\iff$ *Signified (idea or thing)*
2. **How do we have usable meaning in a computer?**
   - Common solution: Use e.g. **WordNet**, a thesaurus containing lists of **synonym sets** and **hypernyms** ("is a" relationships).
     - **Problems with resources like WordNet**
       - Great as a resource but missing nuance
       - Missing new meanings of words, hard to keep up-to-date
       - Subjective
       - Requires human labor to create and adapt
       - Can't compute accurate word similarity

e.g. synonym sets containing "good":

```
from nltk.corpus import wordnet as wn
poses = { 'n':'noun', 'v':'verb', 's':'adj (s)', 'a':'adj', 'r':'adv'}
for synset in wn.synsets("good"):
    print("{}: {}".format(poses[synset.pos()],
            ", ".join([l.name() for l in synset.lemmas()])))
```

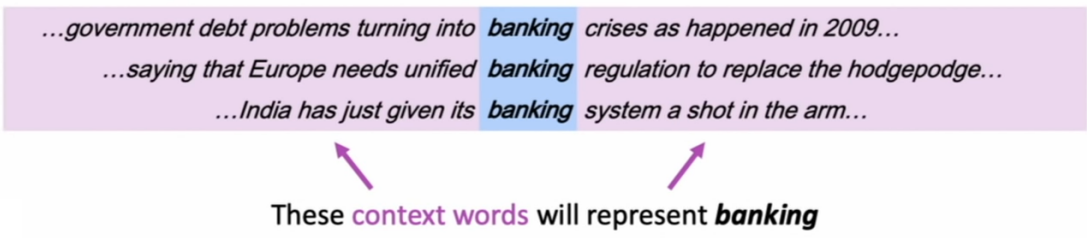
```
noun: good
noun: good, goodness
noun: good, goodness
noun: commodity, trade_good, good
adj: good
adj (sat): full, good
adj: good
adj (sat): estimable, good, honorable, respectable
adj (sat): beneficial, good
adj (sat): good
adj (sat): good, just, upright
…
adverb: well, good
adverb: thoroughly, soundly, good
```

e.g. hypernyms of "panda":

```
from nltk.corpus import wordnet as wn
panda = wn.synset("panda.n.01")
hyper = lambda s: s.hypernyms()
list(panda.closure(hyper))
```

```
[Synset('procyonid.n.01'),
Synset('carnivore.n.01'),
Synset('placental.n.01'),
Synset('mammal.n.01'),
Synset('vertebrate.n.01'),
Synset('chordate.n.01'),
Synset('animal.n.01'),
Synset('organism.n.01'),
Synset('living_thing.n.01'),
Synset('whole.n.02'),
Synset('object.n.01'),
Synset('physical_entity.n.01'),
Synset('entity.n.01')]
```

- **Representing words as discrete symbols**
  - Discrete symbols - a **localist** representation: `Means one 1, the rest 0s`
  - ***One-hot Vectors***
    - e.g. :
      - motel = [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]
      - hotel = [0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]
    - $Vector\ Dimension\ =\ number\ of\ words\ in\ vocabulary\ (e.g., 500,000)$
  - **Problem with words as discrete symbols**
    - The two vectors are **orthogonal**: *no natural notion of **similarity** for one-hot vectors*
    - **Solution**:
      - Try to rely on WordNet's list of synonyms to get similarity?
        - Fail badly: incompleteness
        - ***Instead: learn to encode similarity in the vectors themselves***
- **Representing words by their context**
  - Distributional semantics: **A word's meaning is given by the words that frequently appear close-by**
  - When a word *w* appears in a text, its ***context*** is the set of words that appear nearby (within a fixed-size window).
  - Use the many contexts of *w* to build up a representation of *w*.



- **Word vectors**
  - Build a ***dense*** vector for each word, chosen so that it is similar to vectors of words that appear in similar contexts.
  - Word vectors are sometimes called **word embeddings** or **word representations**. They are **distributed** representations.
  - Visualization - vector space

§3 Word2vec Introduction

- **Word2vec** is a framework for learning word vectors.
- Idea:
  - large corpus of text
  -

§4 Word2vec Objective Function Gradients

§5 Optimization Basics

§6 Looking at Word Vectors