# School of Computing and Information Systems The University of Melbourne COMP90042

## NATURAL LANGUAGE PROCESSING (Semester 1, 2022)

Sample solutions: Week 6

#### Discussion

1. Give illustrative examples that show the difference between:

### (a) **Synonyms** and **hypernyms**

- Two words are synonyms when they share (mostly) the same meaning, for example: *snake* and *serpent* are synonyms.
- One word is a hypernym of a second word when it is a more general instance ("higher up" in the hierarchy) of the latter, for example, *reptile* is the hypernym of *snake* (in its animal sense).

#### (b) **Hyponyms** and **meronyms**

- One word is a hyponym of a second word when it is a more specific instance ("lower down" in the hierarchy) of the latter, for example, *snake* is one hyponym of *reptile*. (The opposite of hypernymy.)
- One word is a meronym of a second word when it is a part of the whole defined by the latter, for example, *scales* (the skin structure) is a meronym of *reptile*.
- 2. Using some Wordnet visualisation tool, for example,

http://wordnetweb.princeton.edu/perl/webwn and the Wu & Palmer definition of word similarity, check whether the word *information* is more similar to the word *retrieval* or the word *science* (choose the sense which minimises the distance). Does this mesh with your intuition?

• The word *information* has five different senses in Wordnet; I've reproduce the fragment of the hierarchy above these senses below:

		entity		
		abstraction		
		communication		
		message		entity
entity	entity	statement	entity	abstraction
abstraction	abstraction	pleading	abstraction	measure
communication	psychological	charge	group	system of meas
message	cognition	accusation	collection	information meas

 $\verb"information"$ 

• Here's the corresponding fragment of the three senses above retrieval:

entity	entity	
physical	abstraction	
process	psychological	entity
processing	cognition	abstraction
data process	process	psychological
operation	basic cog	event
computer op	memory	act

retrieval

• To find the Wu & Palmer similarity, we need to find the **lowest common subsumer** — the lowest node in the hierarchy shared by the two senses, and then apply the following formula:

$$sim(c_1, c_2) = \frac{2 \times depth(LCS(c_1, c_2))}{depth(c_1) + depth(c_2)}$$

- The question asks us to choose the senses which minimise the distance, so we need to check them all.
- The "message" sense of *information* lies at depth 5; the "data retrieval" sense of *retrieval* is at depth 8; the lowest node in the hierarchy that they share is entity (the root node) so the similarity is:

sim(information, retrieval) = 
$$\frac{2 \times 1}{5 + 8}$$
  
=  $\frac{2}{13} \approx 0.154$ 

- What about the "message" sense of *information* with the other senses of *retrieval*?
  - The "memory" sense of *retrieval* is at depth 8, and the lowest node shared is abstraction, abstract entity (at depth 2); this means that the similarity is  $\frac{2\times2}{8+5}\approx0.308$ .
  - The "event" sense of *retrieval* is at depth 6, and the lowest node shared is also abstraction, abstract entity, so the similarity is  $\frac{2\times2}{6+5}=0.364$ .
- Let me summarise these results in a table, where I've numbered the senses according to the Wordnet ordering (left-to-right above):

		information						
		1	2	3	4	5		
retrieval	1	0.154	0.154	0.118	0.154	0.143		
	2	0.308	0.615	0.235	0.308	0.286		
	3	0.364	0.545	0.267	0.364	0.333		

• The maximum similarity (in bold in the table above) is 0.615, for the second sense of *information* — "knowledge acquired through study or experience or instruction" — and the second sense of *retrieval* — "the cognitive operation of accessing information in memory" (because they are both cognitive processes).

- I will leave *science* as an exercise (there are only two senses this time), but the maximum similarity is 0.727 for the "knowledge acquired..." sense of *information*, and the "ability to produce solutions in some problem domain" sense of *science*.
- *science* is clearly the more similar word. This does match with my personal expectations, however, this probably isn't the sense of *science* I had in mind!

#### 3. What is word sense disambiguation?

- Word sense disambiguation is the computational problem of automatically determining which sense (usually, Wordnet synset) of a word is intended for a given token instance with a document.
- 4. For the following term co-occurrence matrix (suitably interpreted):

- (a) Find the Point-wise Mutual Information (PMI) between these two terms in this collection.
  - To evaluate PMI, we need the joint and prior probabilities of the two event (in this case, probably w: document contains world and c: document contains cup.
  - We estimate these based on their appearance out of the total number of instances in the collection (2000), and then substitute:

$$P(w) = 280/2000 = 0.14$$

$$P(c) = 370/2000 = 0.185$$

$$P(w,c) = 55/2000 = 0.0275$$

$$PMI(w,c) = \log_2 \frac{P(w,c)}{P(w)P(c)}$$

$$= \log_2 \frac{0.0275}{0.14 \times 0.185}$$

$$\approx 0.0865$$

- (b) What does the value from (a) tell us about distributional similarity?
  - This value is slightly positive, which means that the two events occur together (in documents) slightly more commonly than would occur purely by chance. There is some possibility that world and cup occurring together is somehow meaningful for documents in this collection.
- 5. In the 09-distributional-semantics iPython notebook, a term-document matrix is built to learn word vectors.
  - (a) What is the Singular Value Decomposition (SVD) method used for here? Why is this helpful?
    - We are using the SVD method to build a representation of our matrix which can be used to identify the most important characteristics of words.
    - By throwing away the less important characteristics, we can have a smaller representation of the words, which will save us (potentially a great deal of) time when evaluating the cosine similarities between word pairs.

- 6. What is a word embedding and how does it relate to distributional hypothesis?
  - We're going to have a representation of words (based on their contexts) in a **vector space**, such that other words "nearby" in the space are similar
  - This is broadly the same as what we expect in distributional similarity ( "you shall know a word by the company it keeps.")
  - The row corresponding to the word in the relevant (target/context) matrix is known as the "embedding".
  - (a) What is the difference between a **skip-gram** model and a **CBOW** model?
    - The element in the condition of the posterior probability: skip-gram models analyse the probability of the context words given the target word;
       CBOW models analyse the probability of the target word given the context words.
  - (b) How are the above models trained?
    - The probabilities here are more complicated than just counting some events in a collection; they are based around taking the dot product of the relevant vectors (or average of vectors, in the case of CBOW), and then marginalising. To improve efficiency, we can also use a negative sampling objective.