Prof. Dr. Andreas Hotho,
M.Sc. Janna Omeliyanenko
Lecture Chair X for Data Science, Universität Würzburg

12. Exercise for "Sprachverarbeitung und Text Mining"

11.02.2022

1 Knowledge Questions

1. Describe the features and structure of WordNet in your own words.

> WordNet is a lexical database for the English language. WordNet groups English words according to their meaning into synonym groups called synsets. WordNet provides definitions and usage examples for synsets and defines a set of semantic relationships between synsets.

Describe and give an example for WordNets synsets.

> A synset is a set of all synonyms that represent a particular meaning or concept. Example: *Dog*
>
> - Synset1 (First meaning): dog, domestic dog, Canis familiaris
> - Definition: A member of the genus Canis (probably descended from the common wolf) that has been domesticated by man since prehistoric times; occurs in many breeds)
> - Example: "the dog barked all night"
> - Synset2 (Second meaning): andiron, firedog, dog, dog-iron
> - Definition: metal supports for logs in a fireplace
> - Example: "the andirons were too hot to touch"

2. What are potential problems of standard similarity metrics that are based on path length?

   These similarity metrics do not take into account information about:

   - Depth of a concept in the hierarchy
   - network density
   - Meaning of the word

   Name one possible way to fix these problems.

   Use of hierarchy information (e.g. from WordNet) and corpus statistics (e.g. from Brown Corpus) $\rightarrow$ Information Content

3. Define the term *Information Content* in your own words.

   The Information Content of an event indicates how "surprising" the occurrence of the event is, or how much information (in bits) is gained through the occurrence of the event.

4. Explain why the similarity metrics from the lecture that use Information Content use the Information Content of the lowest common subsumer of the words being compared in the concept hierarchy.

   The lower a concept is in the concept hierarchy, the more specific and informative it is (higher Information Content). The words that share a common subsumer low in the hierarchy are more similar to each other than those that share only more abstract subsumers.

2 Word Similarity

2.1 Similarity measures

1. For the following ontology, compute the similarities of the following concepts:

   - sim(Pizza,Pasta)
   - sim(Pizza,Beer)
   - sim(Cola,Coffee)
   - sim(Cookies,Beer)

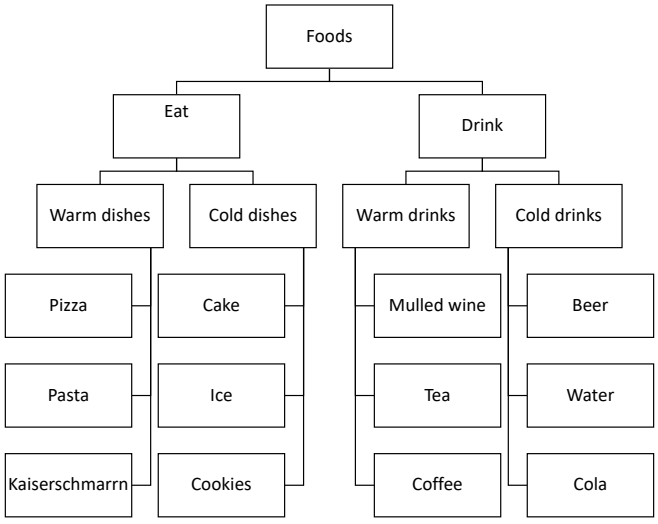
   Use the measures introduced in the lecture for this purpose:

   a) $sim_{path}(c_1, c_2)$

   b) $sim_{jiang-conrath}(c_1, c_2)$

   Use the following text for your calculations, if needed:

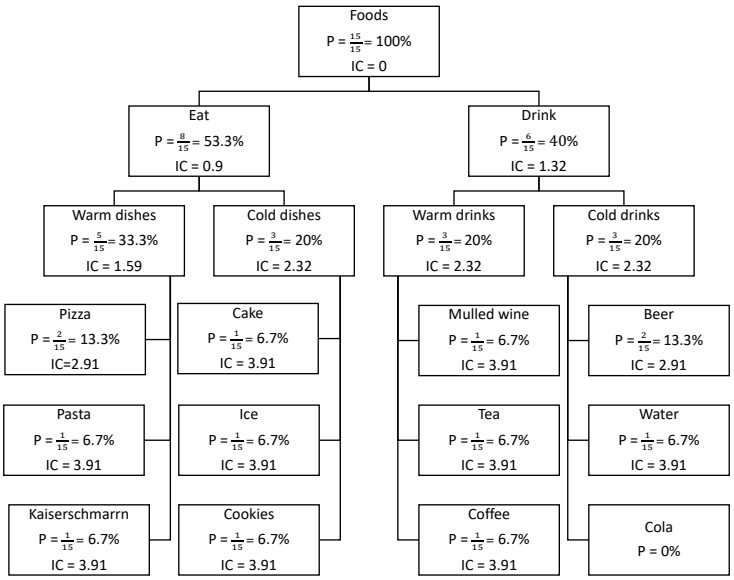
   ```
   My favorite foods are pizza and pasta. I have either
   beer or tea with it. Warm dishes are part of every meal
   for me. With cake I like to have coffee and cookies. The
   water is already frozen to ice when I have mulled wine
   with Kaiserschmarrn at the Christmas market. I would
   rather have pizza and beer, though.
   ```



We define the probability of a word from the taxonomy as:

$$P(c) = \frac{\sum_{w \in \text{words}(c)} \text{count}(w)}{N}$$

where words(c) contains all sub-concepts of the taxonomy from concept c including concept c itself. N is the number of words in the text that occur as a concept in the taxonomy.



Path-Similarity

- $sim(\text{Pizza,Pasta}) = \frac{1}{pathlen(c_1,c_2)} = \frac{1}{3}$
- $sim(\text{Pizza,Beer}) = \frac{1}{7}$
- $sim(\text{Cola,Coffee}) = \frac{1}{5}$
- $sim(\text{Cookies,Bier}) = \frac{1}{7}$

Jiang-Conrath

- $sim(\text{Pizza,Pasta}) = \frac{1}{2\log_2 P(LCS(c_1,c_2)) - \log_2 P(c_1) - \log_2 P(c_2)} = 0.274$

- sim(Pizza,Beer) $= 0.172$
- sim(Cola,Coffee) $=$`Error`
- sim(Cookies,Beer) $= 0.147$

2. Another similarity metric is the weighted path length (wpath) measure, which is defined as follows:

$$sim_{wpath}(c_1, c_2) = \frac{1}{1+pathlen'(c_1,c_2)*k^{IC(LCS(c_1,c_2))}}$$

$$pathlen'(c_1, c_2) = Number\,of\,edges\,in\,graph\,between\,nodes\,c_1\,and\,c_2$$

- Describe the intuition behind wpath in your own words. What types of information that we introduced previously are used in wpath? What is the role of wpath's parameter k?

  wpath combines information about shortest path length and the Information Content of the deepest common subsumer of two concepts. The lower a node is in the hierarchy, the more specific the concept it describes (and the higher the Information Content).

  Two concepts are more similar if they can be described by a more specific super-concept. Additionally, the shorter the shortest path between two concepts, the more similar they are.

  The parameter k specifies how much the Information Content contributes to the weighting of the path length. If $k = 1$, wpath is identical to the path metric.

- Calculate the wpath similarity for the concept pairs from subtask 1.

  Wpath-Similarity

  - sim_{wpath}(Pizza,Pasta) $= \frac{1}{1+pathlen'(c_1,c_2)*0.8^{1.59}} = \frac{1}{1+2*0.8^{1.59}} = 0,416$
  - sim_{wpath}(Pizza,Beer) $= 0.143$
  - sim_{wpath}(Cola,Coffee) $= 0.251$
  - sim_{wpath}(Cookies,Beer) $= 0.143$

3 String Kernel

Calculate the String-Kernel of the words *Text* and *Test*. A completely simplified term depending on λ is sufficient as a result.

Text	Test	$\circ$
T	T	λ^2
e	e	λ^2
x	s	0
t	t	λ^2
Te	Te	λ^4
ex	es	0
xt	st	0
Tx	Ts	0
et	et	λ^6
Tt	Tt	λ^8
Tex	Tes	0
ext	est	0
Tet	Tet	λ^8
Txt	Tst	0

$$Kernel(Text, Test) = 2\lambda^8 + \lambda^6 + \lambda^4 + 3\lambda^2$$

4 Minimum Edit Distance

Determine the minimum edit distance between the following words. We define the costs as triples $w = (w_1, w_2, w_3)$, where w_1 is the cost for Insertion, w_2 the cost for Deletion, and w_3 the cost for Replace.

- Initial-state: `datamining` Goal-state: `textmining` with $w = (1, 1, 2)$
- Initial-state: `gehend` Goal-state: `verstehen` with $w = (3, 7, 9)$

I = Insert, D = Delete, K = Keep (replace for $X[i] = Y[j]$),
R = Replace (replace for $X[i] \neq Y[j]$)

		t	**e**	**x**	**t**	**m**	**i**	**n**	**i**	**n**	**g**
	0	1	2	3	4	5	6	7	8	9	10
d	$^{D}1$	2	3	4	5	6	7	8	9	10	11
a	$^{D}2$	3	4	5	6	7	8	9	10	11	12
t	3	$^{K}2$	$^{I}3$	$^{I}4$	5	6	7	8	9	10	11
a	4	3	4	5	$^{R}6$	7	8	9	10	11	12
m	5	4	5	6	7	$^{K}6$	7	8	9	10	11
i	6	5	6	7	8	7	$^{K}6$	7	8	9	10
n	7	6	7	8	9	8	7	$^{K}6$	7	8	9
i	8	7	8	9	10	9	8	7	$^{K}6$	7	8
n	9	8	9	10	11	10	9	8	7	$^{K}6$	7
g	10	9	10	11	12	11	10	9	8	7	$^{K}6$

		v	**e**	**r**	**s**	**t**	**e**	**h**	**e**	**n**
	0	3	6	9	12	15	18	21	24	27
g	7	$^{R}9$	12	15	18	21	24	27	30	33
e	14	16	$^{K}9$	$^{I}12$	$^{I}15$	$^{I}18$	$^{I}21$	24	27	30
h	21	23	16	18	21	24	27	$^{K}21$	24	27
e	28	30	23	25	27	30	24	27	$^{K}21$	24
n	35	37	30	32	34	36	31	33	28	$^{K}21$
d	42	44	37	39	41	43	38	40	35	$^{D}28$