COMP 6721 Applied Artificial Intelligence (Winter 2021)

Worksheet #10: NLP: Applications, Vector Space Models

Information Extraction. The detection of *Named Entities* (NEs) is a standard NLP application, called *Information Extraction* (IE). A popular Python library for developing NLP applications is *spaCy*,[1] which has an online IE demo at https://explosion.ai/demos/displacy-ent. Try it out on an example text (e.g., a Concordia News article).

Vector dot product. Given the following encoding for $sent_0$ = *"the big dog"*, $sent_1$ = *"the big cat"* and $sent_2$ = *"the big cat and dog"*, compute their similarity as the dot product ($\vec{m} \cdot \vec{n} = \sum_i m_i \cdot n_i$) of their vector representations:

	and	big	cat	dog	the
sent0	0	1	0	1	1
sent1	0	1	1	0	1
sent2	1	1	1	1	1

1. $\vec{sent_0} \cdot \vec{sent_1} =$ ....

2. $\vec{sent_0} \cdot \vec{sent_2} =$ ....

Term Frequency. Fill in the *term frequency* for the two documents (d_1, d_2):

d_1 = *"The big dog barks."*

d_2 = *"The big dog and the big cat."*

Note: ignore words not in the table (we removed so-called *stopwords*).

				d_1			d_2	
token	df	idf	tf	tf.idf	p_i	tf	tf.idf	q_i
dog	50,000							
barks	10,000							
big	100,000							
cat	10,000							

Inverse Document Frequency. Now compute the *inverse document frequency*, $\text{idf} = \log_{10} \frac{N}{\text{df}}$ and add it to the table. Assume N = 10,000,000 (number of documents).

tf-idf Weights. You can now compute the tf-idf weights:

$$w_{t,d} = \begin{cases} (1 + \log \text{tf}_{t,d}) \cdot \text{idf}_t, & \text{if } \text{tf}_{t,d} > 0 \\ 0, & \text{otherwise} \end{cases}$$

(note that we already did the log-scaling for idf above). You now have each document represented as a vector $\vec{d_i} \in \mathbb{R}^{|V|}$ (here $|V| = 4$, the size of our vocabulary).

Cosine Similarity. We can now compute the similarity between the two documents. First, compute the length-normalized vectors $\vec{p}, \vec{q}$ for the two documents and add them to the table above. To normalize a vector, you have to (1) compute its length $\|\vec{v}\| = \sqrt{x_1^2 + \ldots + x_n^2}$, then (2) divide each element by the length: $\frac{x_i}{||\vec{v}||}$. Now you can compute the cosine similarity using the dot product of the normalized vectors, $\text{sim}(d_1, d_2) = \cos(\vec{p}, \vec{q}) = \vec{p} \cdot \vec{q} = \sum_i p_i \cdot q_i$:

- $\cos(\vec{p}, \vec{q}) =$ ....

[1] https://explosion.ai/software#spacy