LING3401 Linguistics and Information Technology

Tutorial: Word meaning and embeddings

Yige Chen

The Chinese University of Hong Kong

February 19, 2025

Acknowledgment

- Some of the tutorial materials are based on: Dan Jurafsky and James H. Martin. *Speech and Language Processing* (3rd ed. draft). 2024.
- GPT-4o and DeepSeek-R1 helped me write more than half of today's codes. Thanks GPT and DeepSeek!

WordNet

- A lexical database of semantic relations between words that links words into semantic relations including synonyms, hyponyms, and meronyms
- Initially for English only, but later extended to other languages as well
  - Synonyms: Words with similar meanings (e.g., *happy*, *joyful*)
  - Hypernyms: More general terms (e.g., *dog* → *animal*)
  - Hyponyms: More specific terms (e.g., *animal* → *dog*)
  - Antonyms: Opposite words (e.g., *hot* vs. *cold*)
  - Meronyms: Part-whole relationships (e.g., *tree* → *branch*)

Have a try!

- Take a look at Part 1 of today's Colab notebook, and see how we can visualize semantic relations between words using WordNet!
  - We are using the English WordNet and have included only selected semantic relations for better visualization.
  - Choose a word and explore its relationships with other words!

Word embeddings

- Word embeddings: dense vector representations of words, capturing their meaning based on context
- Unlike traditional one-hot encoding, embeddings represent words in a low-dimensional space, where similar words have similar vector representations
  - One-hot encoding: with a vocabulary containing n words, it has n dimensions, where only one position (corresponding to the word's index) is 1, and all others are 0
    ```
    ["dog", "cat", "fish", "bird", "lion"]
    dog -> [1, 0, 0, 0, 0], cat -> [0, 1, 0, 0, 0], ...
    ```
  - But word embeddings are dense, unlike the above

Word embeddings

- Word embeddings are trained on raw text data and learn dense vector representations based on word co-occurrence and context
- Each dimension in an embedding does not have a specific, concrete meaning, but collectively, the dimensions capture semantic relationships between words
- Some popular word embeddings
  - Word2Vec
  - GloVe
  - FastText
  - Contextual embeddings (ELMo, BERT, etc.)
- We will use GloVe for today's tutorial
  - A few variants with different vector dimensions are available. We are using the 100-dimensional GloVe today

Have a try!

- Take a look at Part 2 of today's Colab notebook, and see how we can visualize word embeddings!
    - We are using 100-dimensional GloVe embeddings
    - Since visualizing 100 dimensions is impractical, we will reduce the dimensionality to 2 or 3
    - The dimensionality reduction strategies we will try out are: PCA and t-SNE
        - You don't need to figure out how these methods work

Have a try!

- Take a look at Part 3 of today's Colab notebook, and see how clustering can be performed using word embeddings!
    - Do you still recall unsupervised learning? Here we are!
    - We employ k-means to cluster (i.e., to group) the words into several groups based on their embeddings

Have a try!

- Take a look at Part 4 of today's Colab notebook, and see how we can calculate word similarity using word embeddings!
    - In the notebook, I use cosine similarity (recommended), Euclidean distance, and dot product similarity
    - Try different words by replacing the strings of `word1` and `word2`!

Latent semantic analysis

- A technique that analyzes relationships between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms
- It helps uncover hidden (latent) meanings by reducing words and documents into a lower-dimensional space
- It assumes that words that are close in meaning will occur in similar pieces of text (distributional semantics)

Have a try!

- Take a look at Part 5 of today's Colab notebook, and see how we can employ latent semantic analysis for sentence-level or document-level analysis!
  - Note: The performance of latent semantic analysis can be unsatisfactory.
    - It is not using any already pre-trained word embeddings
  - If you want a better performance for sentence-level similarity analysis, talk to me for better solutions
    - Basically you will be looking for some BERT models which we will introduce later this term

Miscellaneous

- Please do not hesitate to ask questions
- We enjoy feedback from you, so please let us know if you feel there's anything we could have done better
- It would be great if you'd bring your laptop to the class every week