

Text-based Retrieval of Medical Information

(Teaching Inspired by Research)

Prof. Dr. Marcin Grzegorzek and
the Medical Data Science Team



UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR MEDIZINISCHE INFORMATIK


- ① Introduction
- ② Statistical Language Modelling
- ③ Neural Networks in Language Modelling
- ④ Conclusion

- 1 Introduction
- 2 Statistical Language Modelling
- 3 Neural Networks in Language Modelling
- 4 Conclusion

Contents of the Course

Week	Lecture	Practical Exercises
1	(05/04) Introduction to Medical Information Retrieval (MIR)	(05/04) Introduction to Python
2	(12/04) Main Components and Classification of MIR Systems	(12/04) Introduction to Python
3	(19/04) Metadata in Medical Information Retrieval Systems	(19/04) CBIR in Medical Applications
4	(26/04) No Lecture due to a Business Trip	(26/04) CBIR in Medical Applications
5	(03/05) Set Theoretic Model: Boolean Retrieval	(03/05) CBIR in Medical Applications
6	(10/05) Set Theoretic Model: Fuzzy Retrieval	(10/05) Flask Tutorial
7	(17/05) Vector Space Model: Similarity Measures	(17/05) Flask Tutorial

Contents of the Course

8	(24/05) Vector Space Model: Distance Functions	(24/05) HTML
9	(31/05) Vector Space Model: Latent Semantic Indexing	(31/05) HTML
10	(07/06) Probabilistic Model	(07/06) HTML
11	(14/06) Text-based Retrieval of Medical Information	(14/06) Deep Learning
12	(21/06) Audio-based Retrieval of Medical Information	(21/06) Deep Learning
13	(28/06) Image-based Retrieval of Medical Information	(28/06) Relevance Feedback
14	(05/07) Demonstrators from Current Research Projects	(05/07) Relevance Feedback
15	(12/07) Summary and Conclusions	(12/07) Evaluation 

Text Document Retrieval – Overall Scenario

- 1 Introduction
- 2 Statistical Language Modelling**
- 3 Neural Networks in Language Modelling
- 4 Conclusion

Statistical Language Model – General Concept

- A statistical language model is a probability distribution over sequences of words: $P(w_1, w_2, \dots, w_m)$.
- The language model provides context to distinguish between words that sound similar, e.g., “recognise speech” vs. “wreck a nice beach”.

Statistical Language Model – General Concept

- Data sparsity is a problem in building language models.
- One solution is the assumption that the probability of a word only depends on the previous n words.
- This is known as an n -gram model or unigram model when $n = 1$.

Statistical Language Model – General Concept

- Estimating the relative likelihood of different phrases is useful in many natural language processing applications, especially those that generate text as an output.
- In speech recognition for instance, sounds are matched with word sequences.
- If a separate language model is associated with each document \mathbf{d}_i in a collection, relevant documents can be retrieved based on the probability of the query \mathbf{q} in the document's language model $P(\mathbf{q}|\mathbf{d}_i)$.

Statistical Language Model – Unigram

- The probability for a sequence of three words can be precisely expressed as follows:

$$P(w_1, w_2, w_3) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \quad .$$

- However, the unigram model simplifies it to:

$$P(w_1, w_2, w_3) = P(w_1)P(w_2)P(w_3) \quad .$$

Statistical Language Model – Unigram

Words w_t	Probability in d_i	Probability in d_j
a	0.10	0.30
world	0.20	0.10
likes	0.05	0.03
we	0.05	0.02
share	0.30	0.20
...

$$\sum_{t=1}^m P(w_t | d_i) = 1$$

$$\sum_{t=1}^m P(w_t | d_j) = 1$$

$$P(\mathbf{q} | d_i) = \prod_{\forall w_r \in \mathbf{q}} P(w_r | d_i)$$

$$P(\mathbf{q} | d_j) = \prod_{\forall w_r \in \mathbf{q}} P(w_r | d_j)$$

Statistical Language Model – n -gram

- In an n -gram model, the probability of observing a sentence w_1, \dots, w_m is approximated as:

$$P(w_1, \dots, w_m) = \prod_{t=1}^m P(w_t | w_1, \dots, w_{t-1}) \quad .$$

- According to the n^{th} order Markov property:

$$P(w_1, \dots, w_m) \approx \prod_{t=1}^m P(w_t | w_{t-(n-1)}, \dots, w_{t-1}) \quad .$$

Statistical Language Model – n -gram

- The conditional probability can be calculated from n -gram model frequency counts:

$$P(w_t | w_{t-(n-1)}, \dots, w_{t-1}) = \frac{\text{count}(w_{t-(n-1)}, \dots, w_{t-1}, w_t)}{\text{count}(w_{t-(n-1)}, \dots, w_{t-1})} .$$

- The bigram and trigram language models denote n -gram models with $n = 2$ and $n = 3$, respectively.

- 1 Introduction
- 2 Statistical Language Modelling
- 3 Neural Networks in Language Modelling**
- 4 Conclusion

Neural Networks in Language Modelling – General Concept

- As language models are trained on larger and larger texts, the number of unique words (vocabulary) increases.
- The number of possible sequences of words increases exponentially with the size of the vocabulary. This brings the statistical language models to their limits.
- Neural networks avoid this problem by representing words in a distributed way, as non-linear combinations of weights in a neural net.

Neural Networks in Language Modelling – General Concept

- Typically, neural net language models are trained as probabilistic classifiers that learn to predict the probability distribution

$$P(w_t | \text{context})$$

- I.e., the network is trained to predict a probability distribution over the vocabulary, given some linguistic context.
- The context might be a fixed-size window of previous words, so that the network predicts

$$P(w_t | w_{t-k}, \dots, w_{t-1})$$

from a feature vector representing the previous k words.

Neural Networks in Language Modelling – General Concept

- Another option is to use “future” as well as “past” words as features:

$$P(w_t | w_{t-k}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+k}) \quad .$$

- This is called a bag-of-words model.

Neural Networks in Language Modelling – Bag-of-words

Example text documents:

d_1 : "John likes to watch movies. Mary likes movies too."

d_2 : "Mary also likes to watch football games."

Lists constructed from these documents:

I_1 : "John", "likes", "to", "watch", "movies", "Mary", "likes", "movies", "too"

I_2 : "Mary", "also", "likes", "to", "watch", "football", "games"

A corresponding bag-of-words representation:

BoW1 = {"John":1, "likes":2, "to":1, "watch":1, "movies":2, "Mary":1, "too":1}

BoW2 = {"Mary":1, "also":1, "likes":1, "to":1, "watch":1, "football":1, "games":1}

- ① Introduction
- ② Statistical Language Modelling
- ③ Neural Networks in Language Modelling
- ④ Conclusion**

Final Statements

- A critical step of the text-based information retrieval is an appropriate modelling of the language leading to text document representations.
- With growing text collections, the problem of sparsity decreases the usefulness of the automatically generated document representations.