LET’S TALK ABOUT PALM LEAVES
FROM MINIMAL DATA
TO TEXT UNDERSTANDING

MAGNUS BENDER¹, MARCEL GEHRKE¹, TANYA BRAUN²

¹Institute of Information Systems, University of Lübeck
²Computer Science Department, University of Münster
AGENDA

1. Introduction to Semantic Systems [Tanya]
   - Components and context of semantic systems
   - Learning & inference tasks
   - Existing formalisms

2. Supervised Learning [Marcel]

3. Unsupervised and Relational Learning [Magnus]

4. Summary [Tanya]
HELPING HUMANS: TEXT UNDERSTANDING

- Picture by Eva Wilden, in: Tamil Satellite Stanzas: Genres and Distribution
HELPING HUMANS: TEXT UNDERSTANDING

- Tamil poems
  - Original poem + Transcript + Translation
    - Credit: Eva Wilden

- Interestingly, Tamil poems sometimes consist of
  - Poem itself
  - Comments (annotations) for specific words in the poem added inline, possibly centuries later

If you do not know the original poem, poem and inline annotation are not easily distinguishable.
HELPING HUMANS: TEXT UNDERSTANDING

- Setting:
  - Set of documents (corpus)
  - Each document contains main text (content) and inline comments (annotation) for preceding words

- Goal: Text understanding
  - Help human to identify which parts of original text are annotation

- Task: Classification
  - Classify which words are content and which are annotation

- Problem: Minimal data
  - Set of manually annotated poems very limited → 91 poems
COMPONENTS AND CONTEXT

SEMANTIC SYSTEMS
THE SETTING: A CORPUS OF DOCUMENTS AND ANNOTATIONS

- Corpus = set of documents \( \mathcal{D} \)
- Each document \( d \) has a set of annotations \( g(d) \)
  - Annotation \( \triangleq \) subjective content description (SCD)
  - Reflect the context of the purpose of the corpus
- Types of SCDs can be manifold
  - Figures, notes, references, …
- Each SCD associated with words at specific locations throughout the corpus
  - Assumption: Words closer to location \( \rightarrow \) influence higher
THE LARGER CONTEXT

DATA PROCESSING SYSTEM

purpose

Queries
Percepts

Goal
Answers
Actions

Comprehensible behaviour*

*requires in-time answers/actions
Agent: Something that perceives its environment through sensors and acts through actuators
- E.g., a document retrieval agent
  - Sensors: User interface to receive query documents
  - Actuators: User interface for returning documents
- E.g., a decision support system
  - Sensors: e.g., interfaces for GPS data
  - Actuators: User interface for presenting suggested decisions / actions

Figure based on: S. Russel and P. Norvig: Artificial Intelligence – A Modern Approach, 1995/2020.
HUMAN-AWARE ARTIFICIAL INTELLIGENCE

- Agent acting in collaboration with or on behalf of a human
  - Also considers representation of
    - The human’s view of the world
    - The human’s belief of the agent’s view of the world
  - Why?
    - Anticipate human behaviour
    - Conform to expectations or explain differing behaviour

Modelling humans in the loop makes one thing very clear: Ethics and AI are intricately linked!
TASKS
SEMANTIC SYSTEMS
TASKS: USER PERSPECTIVE

- Information retrieval
  - Depending on the system and its purpose, e.g.,
    - Identify inline annotations of a given document
    - Find fitting documents → document retrieval
      - To a given document or search string
    - And possibly points of interests in such documents
    - Get an overview (→ explore) in terms of, e.g.,
      - Summary
      - Topics
      - Actors, objects, connections among them
  - From system perspective: External task

FROM MINIMAL DATA TO TEXT UNDERSTANDING
Information retrieval can often be formulated as some form of *classification*

- Part of text annotation or not?
- Document relevant or not to a given search?
- Which parts of a document are relevant?
  - To a given search string
  - For a summary
- Exploration can include classification tasks but may also require different techniques

How to realise a task depends, among other things, on which information is used from the documents
TASKS: SYSTEM PERSPECTIVE

- E.g., for document retrieval given a document:
  - **Topics**: Provide documents with similar topics
  - **Named entities and relations between them**: Provide documents with matching entities
  - **Embedding spaces**: Provide documents that map to a similar position in an embedding space
- E.g., for exploration of a corpus:
  - **Topics**: Provide topic and word distributions of a corpus
  - **Named entities and relations between them**: Provide a knowledge graph
  - **Language models**: Provide a summary of texts
Another important aspect, in small-scale corpora:

Well-rounded corpus needed for high-quality information retrieval

→ Corpus enrichment to extend corpus with documents that provide added value in task context

- From system perspective: Internal task
- Again, a classification problem
  - Input: new document $d$, corpus $D$
  - Possible classes?
    - Quasi-copy, revision, extension, unrelated, complementary?
THE PROBLEM OF MINIMAL DATA & TASK-SPECIFIC CORPORA

- Number of documents in the low hundreds
  - Not enough data for training / adapting LLMs
  - Less support in NLP tools or no pre-trained tools for less common languages
- Annotations of various kinds → can help connect documents, supplement content with information (added value)
  - Citations, entities; (Inline) text, translations, transcriptions
  - Figures, pictures, sensor data
  - Possibly, only manual → expensive
  - Possibly, no annotations at all → no added value
- User-supplied corpora need to be handled on demand in a reasonable amount of time
Documents inhabit a vector space

**tf.idf** (term frequency x inverse document frequency)

- \(tf\): how often occurs a word in a document
- \(df\): in how many documents does the word occur
- \(idf: \log\left(\frac{n}{df}\right)\), \(n\) number of documents in corpus

Document: Vector of tf.idf weights over the vocabulary
Corpus: Matrix of document vectors

**LSI** (dimension reduction using singular value decomposition)

- Reduce matrix to \(m\) dimensions with largest Eigen values
- Example with \(m = 2\) and corpus \(C = \{d_1, d_2, d_3\}\)

Example taken from Grossmann & Frieder (2004)

\(d_1\): “Shipment of gold damaged in a fire”
\(d_2\): “Delivery of silver arrived in a truck”
\(d_3\): “Shipment of gold arrived in a truck”
IR given a search document / string $d'$:

- Return top-$k$ documents closest to $d'$
- Compute a (reduced) vector for $d'$ and
- Find the top-$k$ closest vectors using cosine similarity:
  \[
  \text{sim}(d, d') = \frac{\vec{d} \cdot \vec{d}'}{|\vec{d}| \cdot |\vec{d}'|}
  \]
- Dot product

Example with $m = 2$ and corpus $C = \{d_1, d_2, d_3\}$

- Query: “gold silver truck”
- Corpus enrichment?
TOPIC MODELS

- Assumption: Topics “cause” the words in a document
- Latent Dirichlet Allocation: Generative topic model
  - Each topic has a word distribution $\varphi_k$
    - Drawn from a Dirichlet prior, parameterised by $\beta$
  - Each document $d_i$ has a topic distribution $\theta_i$
    - Drawn from a Dirichlet prior, parameterised by $\alpha$
  - Each word $w_{i,j}$ has a topic $z_{i,j}$
    - Drawn from $\theta_i$
  - Dirichlet distribution: distribution over distributions
    - Larger $\beta, \alpha \rightarrow$ more uniform distributions

- Learning algorithm to fit parameters
- Document retrieval:
  - Estimate topic distribution for new document
  - Provide documents from corpus with similar topic distribution (cosine similarity)
- Corpus enrichment?

Blei et al. (2003)
Extensions, a selection

- Hierarchical Dirichlet process to model topic hierarchy (Teh et al., 2006)
- Dynamic topic modelling to model evolution over time (Blei & Lafferty, 2006)
- Relational topic model (Chang & Blei, 2009)
- Extension to entities (Kuhr et al., 2021)

Applications, a selection

- Social networks (Cha & Cho 2012)
- Tweets (Negara et al., 2019)
- Digital humanities (Redzuan et al., 2023)
NAMED ENTITIES AND KNOWLEDGE GRAPHS

- Documents are about identifiable items, i.e., named entities
- Named entity recognition: Automatically extract named entities from text
  - E.g., OpenIE
    - https://stanfordnlp.github.io/CoreNLP/openie.html
- Problem of named entity matching, entity linking
- SPO triples (subject, predicate, object)
  - Entities form relations
    - Arranged in a graph → Knowledge graph
    - Ontologies as schema layer → Logical inference

- E.g., RDF graph
  - Query language: SPARQL
    - https://www.w3.org/RDF/
      - https://www.w3.org/TR/rdf-sparql-query/
NAMED ENTITIES AND KNOWLEDGE GRAPHS

- Possible to set up entity types in a type hierarchy
- Link entities / SPO triples to points in document
- Information retrieval
  - Query graph for relations
  - Walk graph for exploration
  - Find points of interest through links to text
- Corpus enrichment?
  - Using hierarchy?

- <Olympics ‘20, in, Tokyo>
- <UEFA euro ‘20, in, Europe>
SUBJECTIVE CONTENT DESCRIPTIONS (SCDs)

- Assume annotations “cause” words in a document
- Annotations describe content
  - Subjective to a user / (implicit or explicit) task, at specific points in document
- Form a vector representation of annotations
  - SCD: Associated with words at specific locations
  - SCD-word matrix
    - For each SCD: Probability distribution over vocabulary
      - Which words occur around an SCD
      - Compare: Document-word matrices in LSI
      - Compare: Topic-word distributions in LDA

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Dolor sed viverra ipsum nunc aliquet bibendum enim. In massa tempor nec feugiat. Nunc aliquet bibendum

Kuhr et al. (2019)
SUBJECTIVE CONTENT DESCRIPTIONS (SCDs)

- Information retrieval:
  - Estimate SCD-word distribution for new document
  - Find similar documents through cosine similarity of SCD-word distributions
  - Return points of interest by locating similar SCDs
- Discussion: Corpus enrichment?

Kuhr et al. (2019)
Predictive probabilistic modelling of language: Predict the next word / sentence → Imitate

Long history of models
- Example systems: ELMo, BERT, GPT, ChatGPT, … (Peters et al., 2018; Devlin et al., 2019; Radford et al., 2018; OpenAI, 2022)

Transformer-based models (Vaswani et al., 2017)
- Encoder-decoder architecture
- Attention mechanism
- Figure: Transformer architecture, taken from an article by Vaswani et al. (2017)
(LARGE) LANGUAGE MODELS

- Information retrieval
  - Cue: Prompt engineering
- Question answering
- Summarisation
- Fine-tune a model: adapt to a specific context

Figure taken from a talk by Malte Schilling
https://www.dropbox.com/s/nsenp948uc93l5w/schilling_2023_06_LLM_Mechanisms.pdf?dl=0
INTERIM SUMMARY

- Setting: Corpus of possibly annotated documents
- Tasks:
  - User-driven: Information retrieval
  - Internal: Corpus enrichment
- Formalisms
  - Vector space representation: tf.idf and LSI
  - Topic modelling: LDA
  - Named entities and knowledge graphs
  - SCD-word matrix
  - (Large) language models

**The Problem of Minimal Data**
- Few documents
- Various types of annotations
AGENDA

1. Introduction to Semantic Systems [Tanya]
2. Supervised Learning [Marcel]
   - Subjective content descriptions
   - Corpus enrichment
   - Inline annotations (🌴)
3. Unsupervised and Relational Learning [Magnus]
4. Summary [Tanya]
BIBLIOGRAPHY

- **Blei et al. (2003)**

- **Blei & Laffert (2006)**

- **Cha & Cho (2012)**

- **Chang & Blei (2009)**
BIBLIOGRAPHY

- Deerwester et al. (1990)

- Devlin et al. (2019)

- Kuhr et al. (2019)

- Kuhr et al. (2021)
Negara et al. (2019)

OpenAI (2022)

Peters et al. (2018)

Radford et al. (2018)
BIBLIOGRAPHY

- Redzuan et al. (2023)

- Spärck Jones (1972)

- Teh et al. (2006)

- Vasvani et al. (2017)