Introduction to Natural Language Processing

Part IV: NLP using Lexicons

Henning Wachsmuth

https://ai.uni-hannover.de

Learning Objectives

Concepts

- Different types of lexicons
- The use of lexicons in NLP
- Benefits and limitations of lexicons

Methods

- Lexicon acquisition using statistical cooccurrences
- Information extraction from text using confidence lexicons

Covered tasks

- Attribute lexicon acquisition
- Abusiveness lexicon acquisition
- Attribute extraction

Outline of the Course

I. Overview

II. Basics of Linguistics

III. NLP using Rules

IV. NLP using Lexicons

- Introduction
- Lexicon Acquisition
- Lexicon Matching

V. Basics of Empirical Methods

VI. NLP using Regular Expressions

VII. NLP using Context-Free Grammars

VIII. NLP using Language Models

IX. Practical Issues

Introduction

Lexicons

Lexicon

- A repository of terms (in terms of words or phrases) that represents a language, a vocabulary, or similar

https://pxhere.com

Observations

- Lexicons often store additional information along with a term.
- Lexicons often have an explicit ordering, for example, alphabetically.

Lexicons

Selected Types of Lexicons

Types of lexicons

- Terms only. Term lists, language lexicons, vocabularies
- Terms with definitions. Dictionaries, glossaries, thesauri
- Terms with information. Gazetteers, frequency lists, confidence lexicons

  More on these on the next slides

Why ordering?

- For humans. To enable comfortable searching and browsing
- For computers. To enable efficient search

Representation of lexicons

- Ordered lists. For binary search over ordering
- Hashsets or hashmaps. For direct access to entries
- Regular expressions. For use as part of string patterns (see Part VI)

Lexicons

Lexicons of Terms Only

Term list

- A simple list of terms
- Used e.g. to cover all possible instances of a specific concept

Words			
a	Aachen	aba	...
AA	aardvark	abaca	...
AAA	aardwolf	aback	...

Language lexicon

- Words along with their stems, affixes, and inflections
- Used e.g. for morphological analysis

Word	Stem	Affixes	...
derive	deriv	-ing, -d, -s,	...
people	people	-s	...
quick	quick	-er, -st, -ly,	...
...	...	...	...

Vocabulary

- A list of terms that is known or used in a particular context
- Use e.g. to cover linguistic styles

Formal words		
admittedly	essentially	indeed
consequently	furthermore	likewise
conversely	hence	meanwhile
considerably	incidentally	...

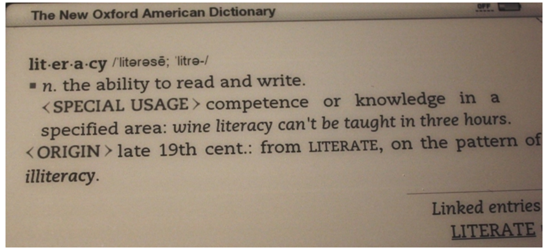
Informal words		
bastard	crap	dude
booze	cuz	hell
bummer	damn	iffy
cop	dope	...

Lexicons

Lexicons of Terms with Definitions

Dictionary

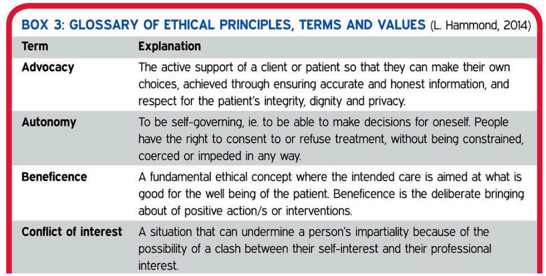
- A list of terms along with their definitions, grammatical information, and more
- Could be used to compare term meaning



http://flickr.com

Glossary

- A vocabulary with term definitions
- Could be used to compare term meaning



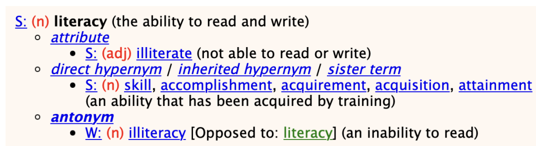
BOX 3: GLOSSARY OF ETHICAL PRINCIPLES, TERMS AND VALUES (L. Hammond, 2014)

Term	Explanation
Advocacy	The active support of a client or patient so that they can make their own choices, achieved through ensuring accurate and honest information, and respect for the patient's integrity, dignity and privacy.
Autonomy	To be self-governing, ie. to be able to make decisions for oneself. People have the right to consent to or refuse treatment, without being constrained, coerced or impeded in any way.
Beneficence	A fundamental ethical concept where the intended care is aimed at what is good for the well being of the patient. Beneficence is the deliberate bringing about of positive action/s or interventions.
Conflict of interest	A situation that can undermine a person's impartiality because of the possibility of a clash between their self-interest and their professional interest.

http://flickr.com

Thesaurus

- A dictionary of synonyms, with (possibly hierarchical) information on related terms
- Used e.g. to find similar terms



S: (n) **literacy** (the ability to read and write)
- *attribute*
  - S: (adj) illiterate (not able to read or write)
- *direct hypernym* / *inherited hypernym* / *sister term*
  - S: (n) skill, accomplishment, acquirement, acquisition, attainment (an ability that has been acquired by training)
- ***antonym***
  - W: (n) illiteracy [Opposed to: literacy] (an inability to read)

http://princeton.edu

Lexicons

Lexicons of Terms with (Structured) Information

Gazetteers

- Location names along with metadata (potentially also other entity names)
- Used e.g. as part of entity recognizers

Location	Latitude	Longitude
Bielefeld	52.0302	8.5325
Hannover	52.3759	9.7320
Paderborn	51.7189	8.7575
Weimar	50.9795	11.3235
...	....	...

Frequency list

- Terms along with their absolute or relative frequency in some text collection
- Used e.g. to decide what terms to use as machine learning features

Word	Count	Word	Count
the	23243	a	12780
i	22225	you	12163
and	18618	my	10839
to	16339	in	10005
of	15687	...	...

Confidence lexicons

- Terms along with confidence values (or probabilities) to represent some concept
- Used e.g. for attribute extraction

Word	Confidence
price	0.59
location	0.95
service	0.61
...	...

Lexicons

Lexicons in NLP

Selected analysis tasks

- Dismbiguation of punctuation, as in abbreviations (see Part III)
- Morphological analysis of words (see Part III)
- Attribute extraction, e.g., product aspects (see below)
- Entity recognition, e.g., time information (see Part VI)
- Style analysis, e.g., formal vs. informal language
- Sentiment analysis of texts, e.g., positive vs. negative words
- Social bias detection based on social group terms and bias terms

Selected generation tasks

- Templated-based generation of texts (see Part III)
- Spelling correction of words
- Language modeling to predict next words (see Part VIII)

Lexicon Acquisition

Lexicon Acquisition

Lexicon acquisition

- The creation of lexicons with (semi-)automatic methods
- This means to define a set of terms, possibly with meta-information.
- The goal is to obtain vocabularies, frequency lists, confidence lexicons, or similar for some concept(s) of interest.

Basis of lexicon acquisition

- Human expert knowledge of a concept, domain, or task
- A text corpus, from which terms can be derived

How many lexicons?

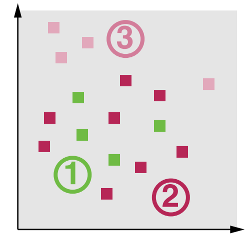
- In many cases, lexicons for multiple concepts are aimed for, such as *formal words* and *informal words*.
- The contrast between these lexicons may affect how they are acquired.

  Below, we look at individual lexicons for simplicity.

Lexicon Acquisition

Process

Typical steps in lexicon acquisition



1. Getting seed terms
2. Expanding the lexicon (possibly incrementally)
3. Finalizing the lexicon

Getting seed terms

- The first step is often to come up with a (small) set of initial terms.
- These terms usually closely relate to the core idea of a given concept.

Expanding the lexicon

- In many cases, seed terms do not sufficiently cover a given concept.
- Lexicons may then be expanded by terms related to the seeds.

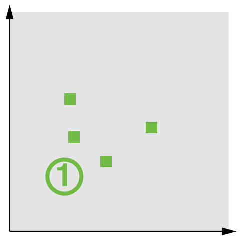
Finalizing the lexicon

- Not all terms found during expansion will reliably represent the concept.
- Given some measure, a threshold may be used to prune the lexicon.

Lexicon Acquisition

Getting Seed Terms

Techniques to get seed terms



- Experts may handcraft an initial list of seed terms.
- Seed terms may be obtained from an annotation study.
- Predefined term lists may exist already somewhere.

How many seed terms?

- The number depends on the concept of interest and on the feasible amount of manual labor.
- In practice, typical numbers range from a handful to a few hundreds.

Example: Hotel aspects (Wachsmuth et al., 2014)

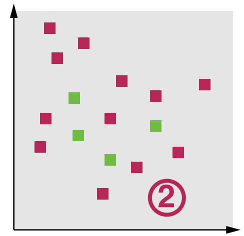
https://pixabay.com

- We annotated hotel aspects in 2100 TripAdvisor reviews.
- In total, 24,596 aspect mentions were annotated.
- In the training set (900 reviews), 625 *different* aspects were covered.

Lexicon Acquisition

Expanding the Lexicon

Techniques to expand a lexicon



- Find terms cooccurring with the seeds in a given corpus.
- Compute similarities between seeds and other terms.
- Train a term classifier on texts with the seeds and apply it.

  We focus on *cooccurrence analysis* below.

How to use these for expansion?

- Many techniques create some numeric score for each candidate term.
- The terms can thus be ranked by their suitability to be in the lexicon.
- A classifier may also just do one binary decision per term.

Incremental lexicon expansion

- After adding new terms to a lexicon, the expansion may be repeated.
- A stop condition is then needed to terminate the incremental process.
- In NLP, this process is called *bootstrapping*.

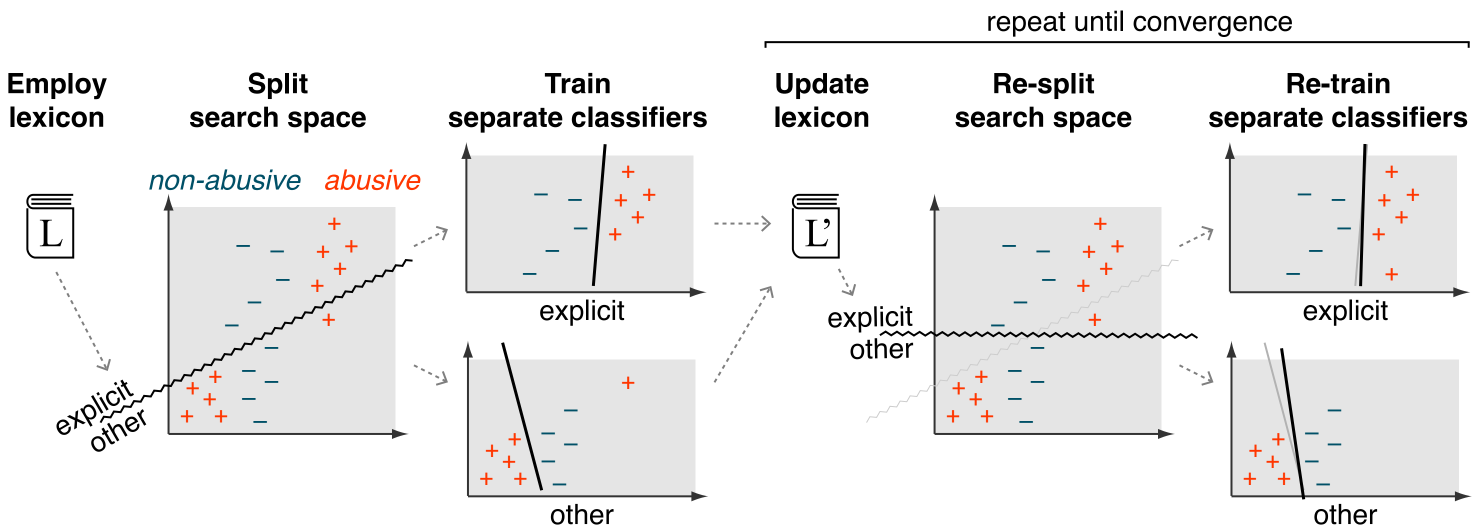
Lexicon Acquisition

Expanding the Lexicon: Bootstrapping

General bootstrapping process

1. Initialize the lexicon with a set of seed terms.
2. Use the seed terms to find new terms in some corpus.
3. Score the new terms, and add the best ones to lexicon.
4. Go back to Step 2, unless the stop condition is met.

Example: Abusiveness lexicon bootstrapping (Chen et al., 2019)



Cooccurrence Analysis

Cooccurrence analysis

- A statistical technique used to find relationships between two concepts A and B in a corpus
- In NLP, concepts may be terms only, terms and documents, or similar.
- Used for word associations, embedding representation, and much more
- The result is usually a score for each pair of concept instances.

Cooccurrence matrix

- Lists the cooccurrences of the concepts of interest
- Defines the basis for any cooccurrence analysis

	b_1	b_2	b_3	$\ldots$
a_1				
a_2				
a_3				
$\vdots$				

Selected analysis methods

- Latent Dirichlet Allocation. Soft clustering of discriminative words
- Latent Semantic Analysis. Singular value decomposition of word pairs
- Pointwise Mutual Information. Detection of associated words

We restrict our view to the latter here.

Cooccurrence Analysis

Pointwise Mutual Information

Pointwise mutual information (PMI)

- A measure that quantifies how much two words w_i and w_j cooccur in a corpus more than if they were independent.
- Used in NLP wherever strongly associated words are of interest

- Let $P(w_i)$ and $P(w_j)$ be the relative frequencies of w_i, w_j, and $P(w_i, w_j)$ their relative coccurrence frequency. Then:

$$PMI(w_i, w_j) \;:=\; log_2 \frac{P(w_i, w_j)}{P(w_i) \cdot P(w_j)}$$

Positive PMI (PPMI)

- Negative PMI values tend to be unreliable, unless huge data is given.
- Since the focus is often on associated rather than unassociated words, a common variation is PPMI:

$$PPMI(w_i, w_j) \;:=\; \text{max}(log_2 \frac{P(w_i, w_j)}{P(w_i) \cdot P(w_j)}, 0)$$

Cooccurrence Analysis

Example: PPMI of Hotel Aspects

Counting cooccurrences

- Two terms cooccur whenever they appear within the same window of consecutive terms of some size (say, 20) in a given corpus.
- Example. Cooccurrence matrix of selected seed words and other words

	front desk	towels	people	minibar	parking
room	1	6	0	4	0
location	0	0	1	0	1
service	2	1	0	1	0
trip	0	0	1	0	1

Computing PPMI

- Example. PPMI of “room” and “towels” according to the matrix

P(“room”) $= \frac{11}{19} = 0.58$ P(“towels”) $= \frac{7}{19} = 0.37$ P(“room“, “towels”) $= \frac{6}{19} = 0.32$

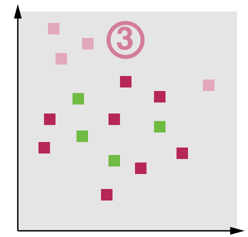
$\rightarrow$ PPMI(“room”, “towels”) $=$ max$(log_2 \frac{0.32}{0.37 \cdot 0.58}, 0) = 0.58$

- The score of a candidate term can, for example, be defined as the aggregated PPMI over all k seed terms: $\sum_{i=1}^{k} PPMI(w_i,$ “towels”$)$

Lexicon Acquisition

Finalizing the Lexicon

Techniques to finalize a lexicon



- Either, keep all terms from lexicon expansion (and seeds).
- Or, prune the lexicon based on some threshold τ of the *confidence values* of the terms.

Confidence values of expanded-lexicon terms

- The scores from lexicon expansion serve as confidence values.
- As shown, a candidate's value may be aggregated from multiple scores.
- The aggregate score may have to be normalized to a defined range.

Confidence value of seed terms?

- Assume we are given a training set where all seed terms $w_1, \ldots, w_k$ have been marked.
- Then the confidence value of w_i may be defined as the fraction of marked mentions of w_i under all occurrences of w_i.

Lexicon Acquisition

Example: A Lexicon of Hotel Aspects

Hotel aspect confidence lexicon

- We derived seeds from 900 training TripAdvisor reviews.
- The confidence values are computed as defined above.
- Below, 30 selected example terms are shown.

https://pixabay.com

High confidence

Hotel Aspect	Confidence
balcony	1.00
blankets	1.00
check-out	1.00
mini-bar	1.00
minibar	1.00
towels	0.97
location	0.95
a/c	0.92
lobby	0.83
wi-fi	0.83

Medium confidence

Hotel Aspect	Confidence
website	0.78
checkin	0.75
front desk	0.74
internet	0.73
reception desk	0.71
room	0.69
shuttle	0.65
parking	0.65
check-in	0.63
service	0.61

Low confidence

Hotel Aspect	Confidence
alcohol	0.50
beer	0.42
waiter	0.40
computer	0.36
ice	0.33
bike	0.25
buffet	0.21
atmosphere	0.17
king	0.10
people	0.01

Lexicon Acquisition

Benefits and Limitations

Benefits

- A lexicon is an intuitive representation of simple linguistic knowledge.
- Big lexicons can be acquired with largely unsupervised methods.
- Well-approved techniques exists for acquisition, such as PMI.

Limitations

- Coming up with adequate seed terms may be non-straightforward.
- Increasing the size of a lexicon usually leads to a decrease in quality.
- Lexicons manifest the limitation of focusing on the terms used.

Implications

- Most effort in lexicon acquisition goes into a careful filtering of terms.
- Predefined lexicons have many use cases until today (e.g., blocklists).
- They play a smaller role in state-of-the-art NLP methods, though.

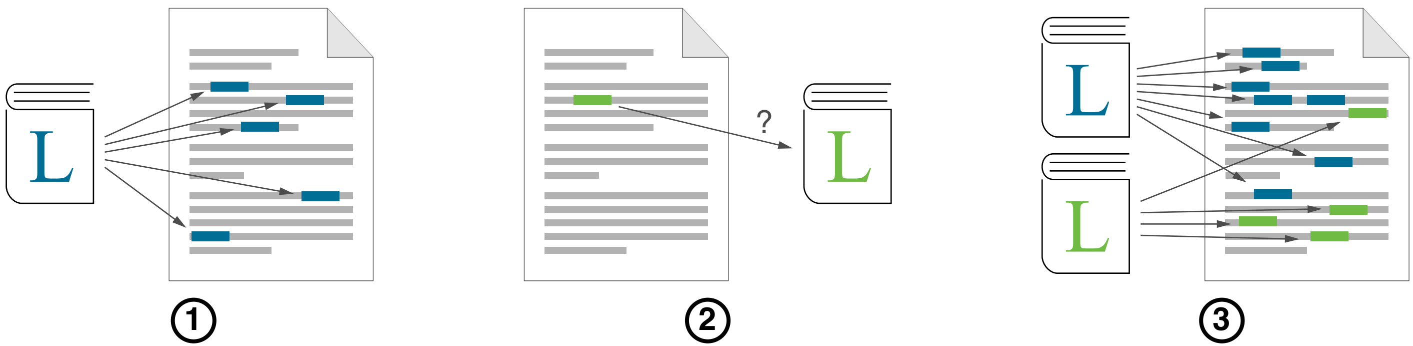
  Aside from the vocabularies of the models employed

Lexicon Matching

Lexicon Matching

Lexicon matching

- The identification of concepts in natural language texts, each being represented by a lexicon
- This requires to decide when a matching term refers to a concept.
- Main goals include to extract concept instances or to assess texts.



When to use lexicon matching?

1. A given lexicon can be used to find all term occurrences in a text.
2. The existence of a given term in a lexicon can be checked.
3. The density or distribution of vocabularies in a text can be measured.

Lexicon Matching

Attribute Extraction

Attribute extraction

- The text analysis that extracts attributes of some entity from text
- Input. A text, at least split into tokens
- Output. The list of all extracted attributes (including their text positions)

Role in NLP

- Used for tasks such as aspect-based sentiment analysis or the extraction of complex events

Example here: Extraction of hotel aspects

- Given a confidence lexicon of hotel aspects, use it to extract aspects in new hotel reviews.
- The approach we see below generalizes across lexicons.

https://pixabay.com

> "We spent one night at that hotel. The service at the front desk was perfect and our room looked clean and cozy... but this alone never justifies the price!"

Attribute Extraction with Lexicon Matching

Why is lexicon matching not trivial?

- Some terms may represent an attribute but not always.
- Some terms are nested in other terms.

"The food in the hotel was great."	vs.	"We left the hotel to go for food."
"The service was great."	vs.	"In-room service was amazing."

https://pixabay.com

Approach in a nutshell

1. Acquire confidence lexicon based on a collection of reviews. (as seen)
2. Choose a threshold $\tau \in [0, 1]$.
3. Extract each lexicon term from a text that has a confidence value $\geq \tau$.
4. Prefer longer terms over shorter terms (and ignore capitalization).

Confidence lexicon (as seen)

- A lexicon of attributes where each term is assigned a value $\in [0, 1]$.
- The value represents the confidence that a term really is an attribute.

Attribute Extraction with Lexicon Matching

Pseudocode

Signature

- Input. A tokenized `text`, a confidence `lexicon`, and a threshold τ

  For simpliclity, assume text and lexicon terms to be all lower-case.
- Output. A list of extracted attributes

extractAttributes(String text, Map lexicon, double τ)

```
List<Term> attribs ← ()
List<Token> tokens ← text.toTokens()
int maxTokens ← lexicon.getLongestAttribute().length
for int i ← 0 to tokens.length-1 do
    int j ← min{i+maxTokens-1, tokens.length-1}
    while j ≥ i do
        String term ← text[tokens[i].begin, tokens[j].end]
        if lexicon.contains(term) and lexicon.get(term)≥τ then
            attribs.add(new Attribute(term.begin, term.end))
            i ← j
            break // leave while loop
        j ← j - 1
return attribs
```

Attribute Extraction with Lexicon Matching

Evaluation of the Approach

What does the threshold τ do?

- The higher τ, the more likely an extracted term really is an attribute, but the fewer attributes will be extracted.
- τ trades *precision* (i.e., the proportion of correctly extracted attributes) against *recall* (i.e., the proportion of found attributes).

  The harmonic mean of precision and recall is the so-called F_1-score.

Evaluation of the approach (on 600 test TripAdvisor reviews)

τ	**Precision**	**Recall**	**F_1-score**
0.1	0.739	**0.460**	0.566
0.2	0.768	**0.460**	0.575
0.3	0.785	0.457	0.578
0.4	0.794	0.456	**0.580**
0.5	0.808	0.448	0.576
0.6	0.820	0.429	0.563
0.7	0.846	0.354	0.499
0.8	0.864	0.284	0.427
0.9	**0.893**	0.144	0.265

Attribute Extraction with Lexicon Matching

Insights from Analyzing Hotel Aspects


https://pixabay.com

Some the most often named aspects (in 2100 TripAdvisor reviews)

1. Room. Mentioned in 80% of all reviews
3. Location. Seen positive in 85% of all reviews
8. Service. If seen negative, highest overall score in 0% of all reviews
20. Towels. Seen negative in 67% of all reviews
24. Parking. If seen negative, highest overall score in 12% of all reviews; but if seen positive, lowest score in 0% of all reviews

Specific tokens (in 44,220 user comments on HRS)

- Most frequent.
  “the”, “and”, “to”, “was”, “a”, “in”, “very”, “is”
- Most clearly positive.
  “close”, “easy”, “friendly”, “modern”, “nice”
- Most clearly negative.
  “been”, “because”, “booked”, “cold”, “dirty”, “or”, “hot”, “so”, “them”

Lexicon Matching

Benefits and Limitations

Benefits

- Lexicon matching is particularly reliable for unambiguous terms.
  For entity types such as location names, huge gazetteer lists exist.
- Lexicons with confidence values allow for trading precision for recall.
- The idea of matching a lexicon is well-explainable.

Limitations

- Information that is not in the employed lexicons can never be found.
- Ambiguous terms require other methods for disambiguation.
- Composition of related information is hard to model with lexicons.

Implications

- Lexicon matching is most suitable for (more or less) closed-class terms.
- Such a matching is part of various techniques across NLP.
- It often bridges between text and embeddings, as in bias detection.

Conclusion

General Observations about NLP

Correctness vs. effectiveness

- NLP algorithms are rarely correct, i.e., their output contains errors from time to time.
- Rather, they have a certain effectiveness in terms of precision, recall, ...

Types of errors

- There are two general kinds of errors, often with a trade-off.
- False positives. Wrong information that was inferred from a text.
- False negatives. Correct information that was not inferred from a text.

Need for data

- Training data is needed to develop certain NLP methods.
- Test data is needed to evaluate the effectiveness of methods.
- The available data is a (if not *the*) decisive factor in NLP.

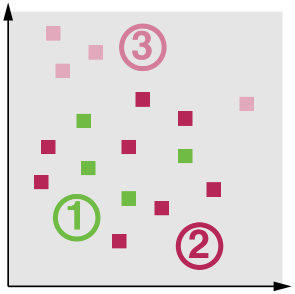
Conclusion

NLP using lexicons

- Lexicon: Repository of terms with meta-information
- Several types from term lists to confidence lexicons
- Used in NLP for tasks until today

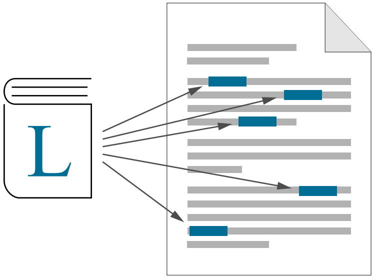
Lexicon acquisition

- Manual and/or automatic creation of lexicons
- Seed terms are often expanded by related terms
- Cooccurrences and similarities may be exploited



Lexicon matching

- Checking of lexicon term mentions in given texts
- May be used for extraction, style analysis, ...
- Confidence values help adjusting effectiveness



References

Some content and examples taken from

- **Jurafsky and Martin (2021).** Daniel Jurafsky and James H. Martin. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics. Draft or 3rd edition, December 29, 2021. https://web.stanford.edu/ jurafsky/slp3/

- **Brooke and Hirst (2013).** Julian Brooke and Graeme Hirst. Hybrid Models for Lexical Acquisition of Correlated Styles. In Proceedings of the Sixth International Joint Conference on Natural Language Processing, pages 82–90, 2013.

Other references

- **Chen et al. (2019).** Wei-Fan Chen, Khalid Al Khatib, Matthias Hagen, Henning Wachsmuth, and Benno Stein. Unraveling the Search Space of Abusive Language in Wikipedia with Dynamic Lexicon Acquisition. In Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda, pages 76–82, 2019.

- **Wachsmuth et al. (2014).** Henning Wachsmuth, Martin Trenkmann, Benno Stein, Gregor Engels, and Tsvetomira Palarkarska. A Review Corpus for Argumentation Analysis. In Proceedings of the of the 15th International Conference on Intelligent Text Processing and Computational Linguistics, pages 115–127, 2014.