Computational Argumentation – Part II

# Basics of Natural Language Processing


Henning Wachsmuth

https://ai.uni-hannover.de

# Learning goals

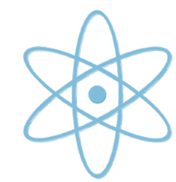
- **Concepts**
  - Basics from linguistics, statistics, and machine learning

https://commons.wikimedia.org

- **Methods**
  - How to develop and evaluate data-driven methods
  - Tasks tackled in natural language processing (NLP)
  - Common analysis techniques used in NLP

https://pixabay.com

- **Associated research fields**
  - Natural language processing



https://pixabay.com

- **Within this course**
  - Concepts and methods this course builds upon

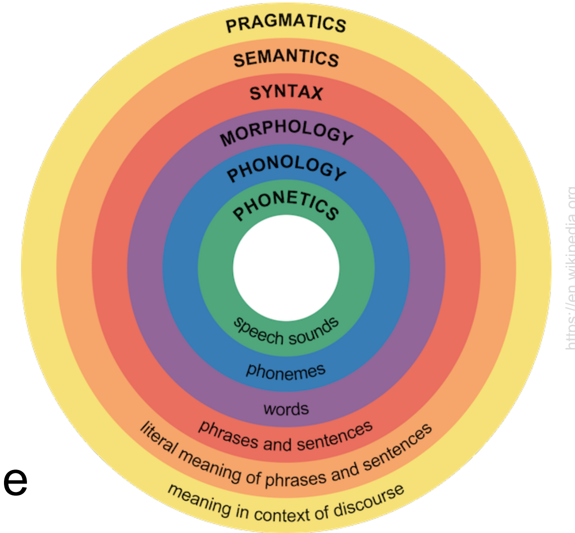
- **Disclaimer**
  - The basics selected here are all but complete and only revisited high-level

    For more details, see e.g. my other master course: https://www.ai.uni-hannover.de/en/teaching/courses/snlp

# Outline

# Linguistics

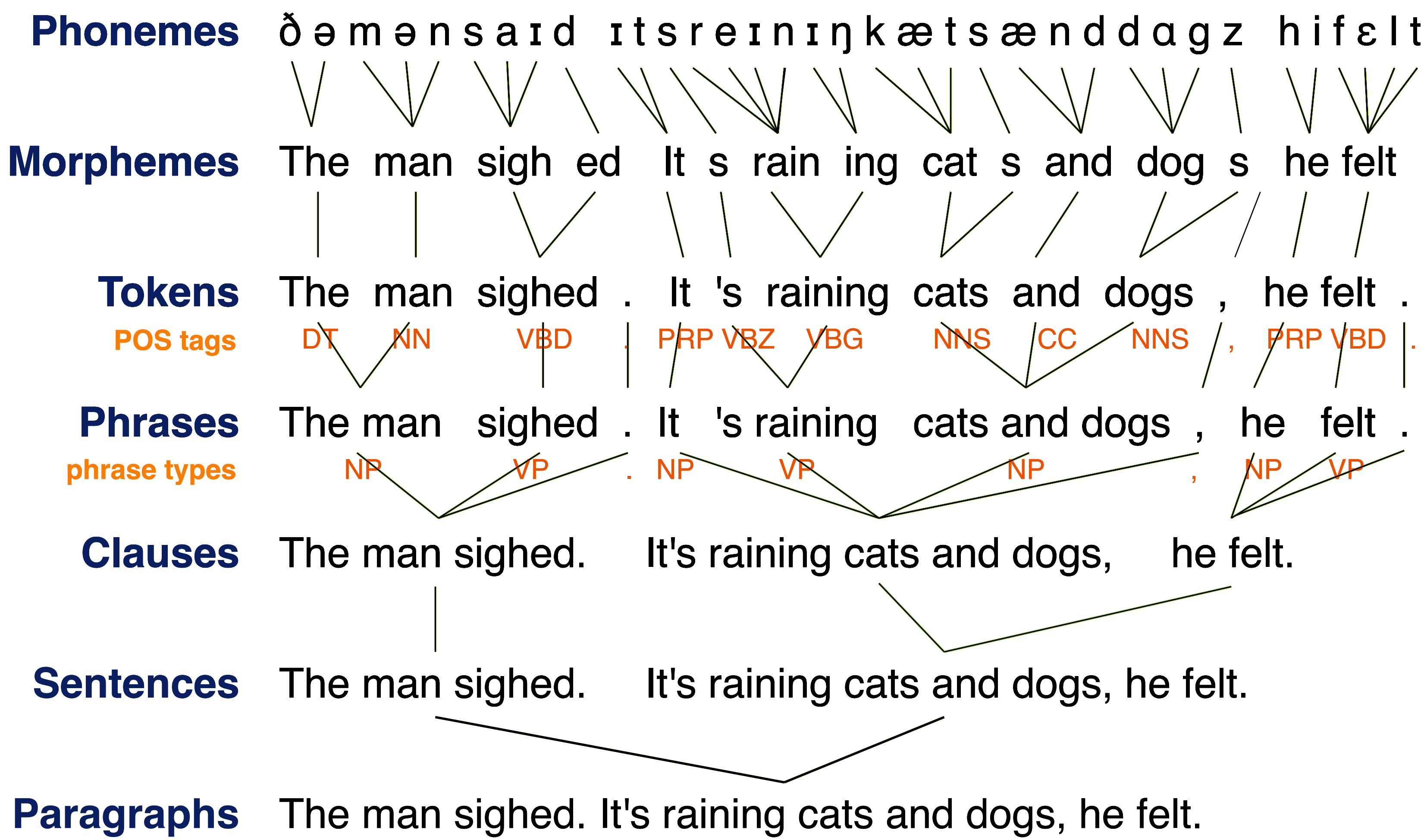
- **Linguistics**
  - The study of spoken and written natural language in terms of the analysis of form, meaning, and context




- **Levels of spoken language only**
  - Phonetics. Physical aspects of speech sounds
  - Phonology. Linguistic sounds of a particular language

- **Levels of spoken and written language**
  - Morphology. Senseful components of words and wordforms
  - Syntax. Structural relationships between words, usually in a sentence
  - Semantics. Meaning of single words and compositions of words
  - Discourse. Linguistic units larger than a single sentence
  - Pragmatics. How language is used to accomplish goals

# Linguistic text units



# Main morphological concepts

- **Word**
  - The smallest unit of language that is to be uttered in isolation
    Example: ”cats“ and ”ran“ in ”cats ran.“

- **Lemma**
  - The dictionary form of a word
    Example: ”cat“ for ”cats“, ”run“ for ”ran“

- **Stem**
  - The part of a word(form) that never changes
    Example: ”cat“ for ”cats“, ”ran“ for ”ran“

- **Token**
  - The smallest text unit in NLP: A wordform, number, symbol, or similar
    Example: ”cats“, ”ran“, and ”.“ in ”cats ran.“ (whitespaces are not considered as tokens)

# Main syntactic concepts

- **Part-of-speech (POS)**
  - The lexical category (or word class) of a word
  - Abstract classes. Nouns, verbs, adjectives, adverbs, prepositions, ...
  - POS tags. NN (single nouns), NNS (plural nouns), NNP (proper nouns), ...
- **Phrases**
  - A contiguous sequence of related words, functioning as a single meaning unit
  - Phrases often contain nested phrases.
  - Types. Noun phrase (NP), verb phrase (VP), prepositional phrase (PP)
    Sometimes also adjectival phrase (AP) and adverbial phrase (AdvP)
- **Clause**
  - The smallest grammatical unit that can express a complete proposition
  - Types. Main clause, subordinate clause
- **Sentence**
  - A grammatically independent linguistic unit consisting multiple tokens

# Main semantic concepts

- **Lexical semantics**
  - The meaning of words and multi-word expressions
    Different senses of a word, the roles of predicate arguments, ...
- **Compositional semantics**
  - The meaning of the composition of words in phrases, sentences, and similar
    Relations, scopes of operators, and much more
- **Entities**
  - An object from the real world
  - Named entities. Persons, locations, organizations, products, ...
    For example, ”Prof. Dr. Henning Wachsmuth”, “Hannover”, “Leibniz University Hannover”
  - Numeric entities. Values, quantities, ranges, periods, dates, ...
    For example, “in this year”, “2024-04-16”, “$ 100 000”, “762-123 77”
- **Relations**
  - Semantic. Relations between entities, e.g., organization *founded in* period
  - Temporal. Relations describing courses of events, e.g., as in news reports

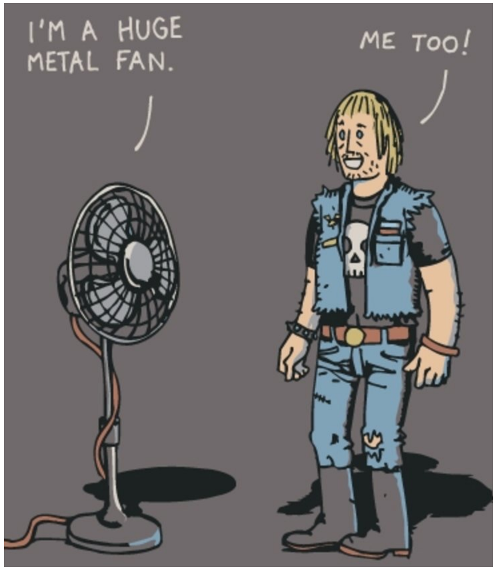
# Main discourse and pragmatics concepts

more in part on **basics of argumentation**

- **Discourse**
  - Linguistic utterances larger than a sentence, e.g., paragraphs or entire texts
    Usually monological; dialogical discourse is rather referred to as *dialogue*.
- **Discourse structure**
  - Discourse segments. Building block of a discourse in terms of linguistic units
  - Coherence relations. Semantic or pragmatic relations between segments
    Examples: A causes B, B elaborates A, A contrasts to B, …
- **Coreference**
  - Two or more expressions in a text that refer to the same thing
  - Types. Pronouns in anaphora and cataphora, coreferring noun phrases, ...
    Examples: "**Apple** is based in Cupertino. **The company** is actually called **Apple Inc.**, and **they** make hardware."
- **Speech acts**
  - Linguistic utterances with a performative function
  - Speakers may commit to something, make listeners do something, …
    Example: "Smoking is bad for your health" aims to make you not smoke.

# What makes language understanding hard?

- **Ambiguity**
  - The fundamental challenge of NLP is that language is ambiguous
- **Ambiguity is pervasive**
  - Phonetic. ”wreck a nice beach”
  - Word sense. ”I went to the bank.”
  - Part of speech. ”I made her duck.”
  - Attachment. ”I saw a kid with a telescope.”
  - Coordination. ”If you love money problems show up.“
  - Scope of quantifiers. ”I didn’t buy a car.”
  - Speech act. “Have you emptied the dishwasher?”




- **Other challenges**
  - World knowledge. ”Putin must rethink his view of Ukraine”
  - Domain dependency. ”Read the book!”

  ... and many more

# Next section: Empirical methods

I. Introduction to computational argumentation

**II. Basics of natural language processing**

- a) Linguistics
- **b) Empirical methods**
- c) NLP techniques
- d) Machine learning
- e) Conclusion

III. Basics of argumentation

IV. Argument mining

V. Argument assessment

VI. Argument generation

VII. Applications of computational argumentation

VIII. Conclusion

# Development and evaluation in NLP

- **Development and evaluation**
  - NLP methods tackle language-related analysis and synthesis tasks.
  - Development. They are built or trained based on *text corpora*.
  - Evaluation. Since their output is rarely free of errors, they are evaluated empirically againt ground-truth annotations.

- **Main evaluation criteria**
  - Effectiveness. The extent to which the output of a method is correct
  - Efficiency. The consumption of time (or space) of a method on an input
  - Usually, effectiveness in the focus of NLP

- **Evaluation measures**
  - Quantify the quality of a method on a specific task and text corpus
  - Methods can be ranked with respect to an evaluation measure.
  - Different measures are useful depending on the task.

# Annotated text corpora

- **Text corpus**
  - A collection of real-world texts with known properties, compiled to study a language problem
  - The texts are often *annotated* with meta-information.
  - Corpora are split into datasets for developing and/or evaluating (testing) an algorithm.

https://pixabay.com

- **Annotations**
  - Marks a text or span of text as representing meta-information of a specific type
  - Labels or numeric values
  - Also used to specify relations

Time entity — Organization entity

*" 2014 ad revenues of Google are going to reach*

Reference — Time entity

*$20B. The search company was founded in '98.*

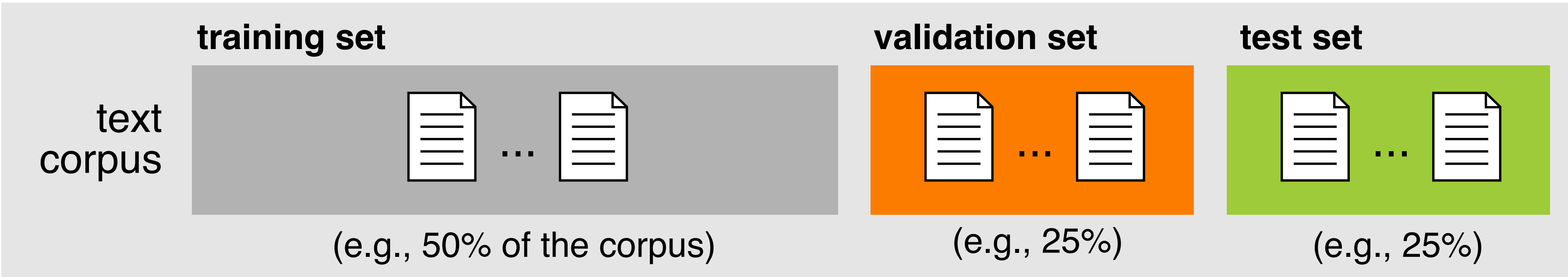
Reference — Time entity — Founded relation

*Its IPO followed in 2004. [...] "*

Topic: "Google revenues" Genre: "News article"

- **Types of annotations**
  - Ground-truth. Manual annotations, often created by experts
  - Automatic. NLP methods add annotations to texts.

# Training, validation, and test set



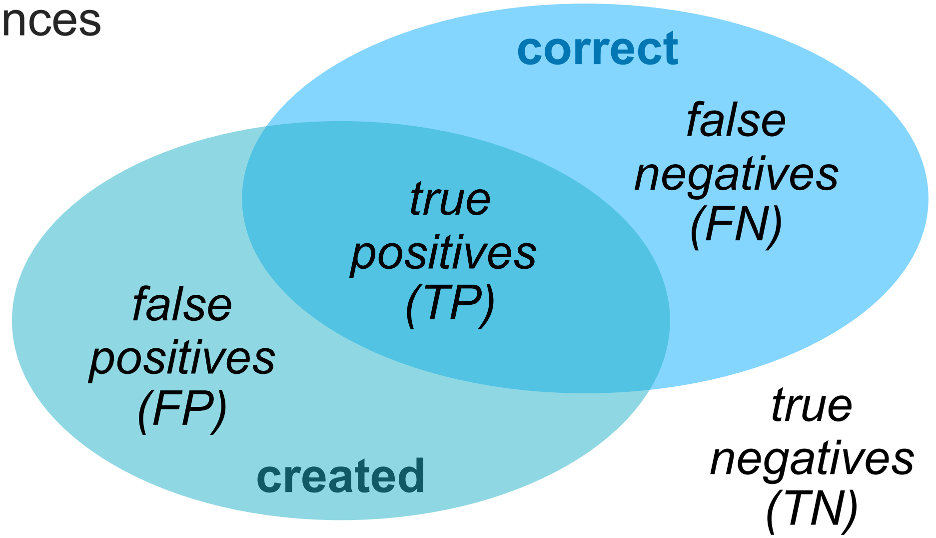

- **Training set**
  - Known instances used to develop or statistically learn a method
  - The training set may be analyzed manually and automatically.
- **Validation set** (aka development set)
  - Unknown test instances used to iteratively evaluate a method
  - A method is optimized on and adapts to the validation set.
- **Test set** (aka held-out set)
  - Unknown test instances used for the final evaluation of a method
  - The test set represents unseen data.

# Cross-validation


- **(Stratified) *n*-fold cross-validation**
  - Randomly split a corpus into *n* datasets of equal size, usually *n = 10.*
  - Development and evaluation have *n* runs; results are averaged over all runs.
  - In the *i*-th run, the *i*-th fold is used for evaluation (testing). All other folds are used for development (training).
- **Pros and cons of cross-validation**
  - Often preferred when data is small, as more data is given for training
  - Cross-validation avoids potential bias in a corpus split.
  - Random splitting often makes the task easier, due to corpus bias.

# Evaluation of effectiveness for nominal labels

- **Nominal instances (labels)**
  - Positives. The output instances (annotations) a method has created
  - Negatives. All other possible instances




- **Accuracy**
  - Used if positives and negatives are similarly important

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precision, recall, and $F_1$-score**
  - Used if positives are in the focus

$$Precision\ (P) = \frac{TP}{TP + FP} \qquad Recall\ (R) = \frac{TP}{TP + FN} \qquad F_1\text{-score} = \frac{2 \cdot P \cdot R}{P + R}$$

  - In multi-class tasks, *micro-* and *macro-averaged* values can be computed.

# Evaluation of effectiveness for numeric values

- **Numeric instances**
  - NLP methods may predict values $y_i$ from a real-valued scale.
  - The numeric difference to the ground-truth values $y_i^*$ is usually in the focus.

- **Mean absolute error (MAE)**
  - Used if outliers require no special treatment

$$MAE = \frac{1}{n} \cdot \sum_{i=1}^{n} |y_i - y_i^*|$$

- **Mean squared error (MSE)**
  - Used if outliers are considered particularly problematic

$$MSE = \frac{1}{n} \cdot \sum_{i=1}^{n} (y_i - y_i^*)^2$$

- **Root mean squared error (RMSE)**
  - Just a different way of quantifying the squared error, $RMSE = \sqrt{MSE}$

# Comparison of effectiveness

- **Need for comparison**
  - It is unclear how good a measured effectiveness result in a given task is.
  - Comparison against baselines is needed

- **Baseline** (aka lower bound)
  - An alternative approach to a task that exists already or is easy to develop
  - A new approach aims to be better than all relevant baselines.

| | | |
|---|---|---|
| **Approach.** Fine-tuned LLM for argument mining | **Baseline 1.** Zero-shot LLM for argument mining | **Baseline 2.** Bag-of-words classifier |

- **Types of baselines**
  - Trivial. An approach that can easily be derived from a given task or dataset
  - Standard. An approach that is often used for related tasks
  - Sub-approach. A sub-part of a new approach
  - State of the art. The best published approach for the addressed task

# Empirical research and variables

- **Empirical methods**
  - Quantitative methods based on numbers and statistics
  - Study questions on behaviors and phenomena by analyzing data
  - Ask about the relationships between *variables*

- **Variable**
  - An entity that can take on different nominal or numeric values
  - Independent. A variable *X* expected to affect another variable
  - Dependent. A variable *Y* expected to be effected by others
  - Other. Confounders, mediators, moderators, ...

*X.* Training method of LLM

*Y.* Accuracy of LLM

- **Scales of variables**
  - Nominal. Values that represent discrete, separate categories (labels)
  - Ordinal. Values that can be ordered/ranked by what is better
  - Interval. Values whose difference can be measured
  - Ratio. Interval values that have an absolute zero

# Statistics

- **Descriptive statistics**
  - Measures for summarizing and comprehending distributions of values
  - Used to describe phenomena based on a given sample
  - Often based on measures of central tendency and dispersion
- **Selected measures of central tendency**
  - Mean. The arithmetic average of a sample from a distribution of values
  - Median. The middle value of the ordered values in a sample
- **Selected measures of dispersion**
  - Variance. The mean squared difference between each value and the mean
  - Standard deviation. The square root of the variance
- **Inferential statistics**
  - Procedures that study hypotheses based on values
  - Used to make inferences about a distribution beyond a given sample
  - Most central to NLP is the *hypothesis test*

# Hypothesis tests

- **Two competing hypothesis**
  - Research hypothesis (*H*). Prediction about how some inpedendent variables affect a dependent variable
  - Null hypothesis ($H_0$). Antithesis to *H*

H. *"The accuracy of our mining approach is higher with fine-tuning than without."*

- **Hypothesis test** (aka statistical significance test)
  - A statistical procedure which determines the probability (*p*-value) that results supporting *H* are due to chance (or sampling error)
  - Significance given, if *p* is ≤ some significance level $\alpha$ (usually 0.05 or 0.01)

- **Steps in a hypothesis test**
  - State *H* and $H_0$, choose $\alpha$.
  - Compute *p*-value with an adequate test. Decide whether $H_0$ can be rejected.

- **Selected hypothesis tests**
  - Parametric. Independent, dependent, and one-sample student's *t*-test
  - Non-parametric. Mann-Whitney test, Wilcoxon Signed-Rank Test

# Next section: NLP techniques

I. Introduction to computational argumentation

**II. Basics of natural language processing**

a) Linguistics
b) Empirical methods
**c) NLP techniques**
d) Machine learning
e) Conclusion

III. Basics of argumentation

IV. Argument mining

V. Argument assessment

VI. Argument generation

VII. Applications of computational argumentation

VIII. Conclusion

# NLP tasks

- **Types of tasks**
  - NLP tackles analysis and synthesis tasks by making certain inferences
  - Analysis (decoding). Inference of new information from given text
    Also referred to as *natural language understanding (NLU)*
  - Synthesis (encoding). Inference of new text from given information and/or text
    Also referred to as *natural language generation (NLG)*

- **Selected analysis tasks**
  - Token and sentence splitting
  - Syntactic parsing
  - Entity recognition
  - Reference resolution
  - Relation extraction
  - Topic detection
  - Sentiment analysis

  ... among many other tasks

- **Selected synthesis tasks**
  - Grammatical error correction
  - Sentence generation
  - Discourse composition
  - Summarization
  - Text style transfer
  - Cluster labeling
  - Lexicon creation

  ... among many other tasks

# NLP techniques

- **Types of techniques**
  - Rule-based. Inference based on manually encoded expert knowledge
    Knowledge includes decision rules, lexicons, regular expressions, grammars, ...
  - Statistical. Inference based on statistical patterns in defined text features
    Features are encoded manually or semi-automatically.
  - Neural. Inference based on statistical patterns in self-learned functions
    Functions of arbitrary complexity may be approximated.

- **Selected techniques**
  - Rule-based. Lexicons, regular expressions, probabilistic parsing, …
    Appear sometimes in this course, but are not in the focus
  - Statistical. Comparison, clustering, classification, regression, seq. labeling, …
    Appear often in this course; covered on the following slides
  - Neural. Recurrent networks, transformers, language models, …
    Covered explicitly in this course

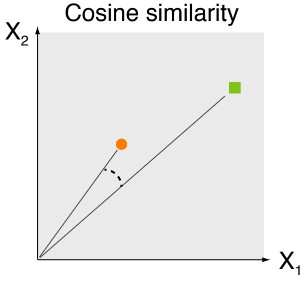
- **Notice**
  - *Synthesis tasks* and *neural techniques* are not considered prerequisites here.
  - They will be treated later in medium detail in the course.

# Comparison

- **Similarity measures**
  - A real-valued function that quantifies how similar two texts or spans of text are (between 0 and 1)
  - Distance measures can be used as (inverse) similarity measures.
  - For text comparison, similarity/distance measures are used mostly

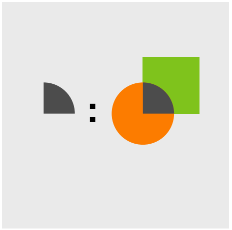
- **Types of similarity measures**
  - Vector-based. For feature and embedding vector representations
  - String-based. For character sequences
  - Concept-based. For taxonomic relatedness of concepts
  - Set-based. For sets of words or embeddings

Cosine similarity



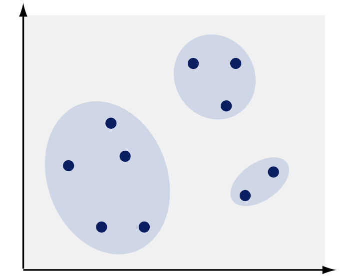

- **Selected measures**
  - Cosine similarity. Angle between vectors
  - Jaccard similarity. Ratio of intersection to union of sets
  - Word mover distance. Best alignment of embedding sequences

Jaccard similarity

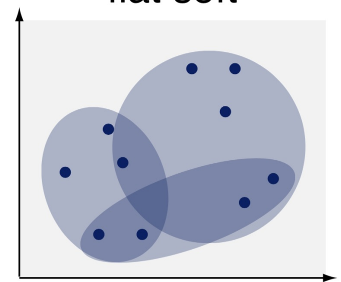


# Clustering

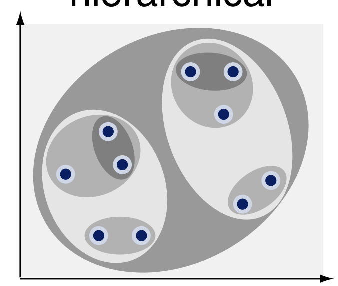
flat hard



flat soft



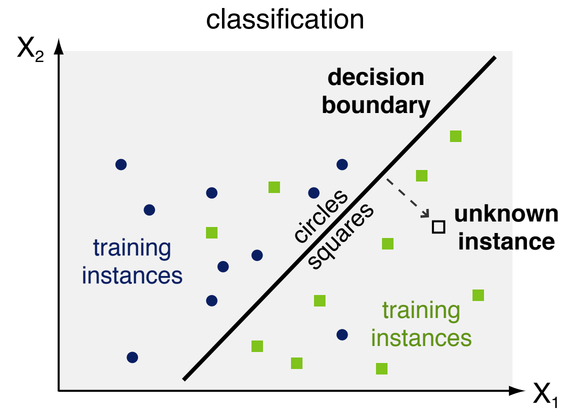
hierarchical



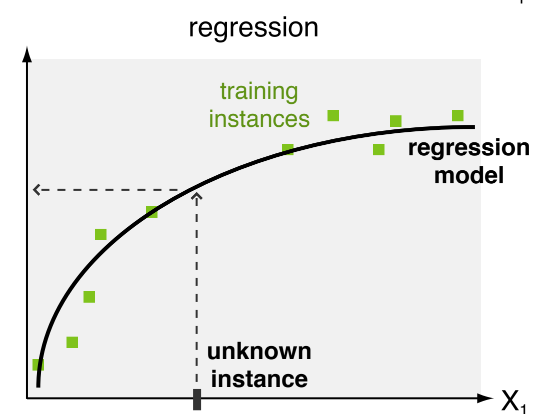
- **Clustering**
  - Group a set of texts based on similarity into a possibly but not necessarily predefined number of classes
  - The meaning of a class is usually unknown in advance.
- **Hard vs. soft clusters**
  - Hard. Each text belongs to a single cluster.
  - Soft. Spans belong to each cluster with a certain weight.
- **Flat vs. hierarchical clustering**
  - Flat. Group texts into a set of independent clusters.
  - Hierarchical. Create a binary clustering tree over all texts.
- **Selected clustering methods**
  - k-means. Create *k* hard clusters based on centroid distance.
  - LDA (topic modeling). Create overlapping clusters based on shared features.
  - Agglomerative. Incrementally merge closest clusters hierarchically.

# Classification and regression

- **Classification**
  - Assign a text to the most likely class of a set of predefined classes
  - A decision boundary *y* is learned that decides the class of unknown texts.



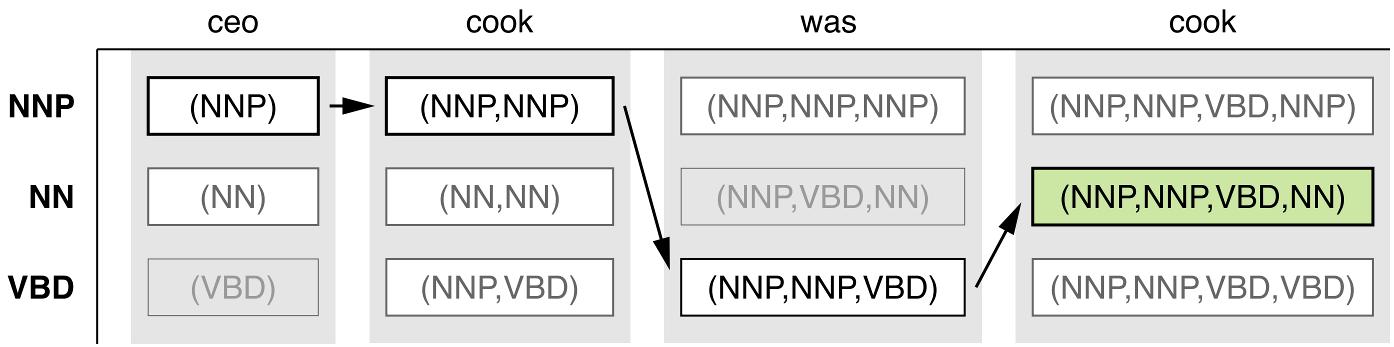

- **Regression**
  - Assign a text to the most likely value of a continuous target variable
  - A regression function *y* is learned that decides the value of unknown texts.




- **Selected classification and regression methods**
  - Support vector machine. Maximize the margin between classes
  - Linear regression. Predict output values using a learned linear function
  - Feedforward neural network. Details later in the lecture…

# Sequence labeling

- **Sequence labeling**
  - Assign the most likely sequence of labels to a sequence of text spans
  - Variation of classification that can exploit dependencies between spans/labels




- **Main elements**
  - Probalistic sequence model. Describes conditional dependencies
  - Decoding algorithm. Computes the most likely sequence

- **Selected sequence labeling methods**
  - Hidden Markov model. Learn joint probability distribution of spans and labels
  - Conditional random field. Learn conditional probability of labels under spans
  - Recurrent neural nework. Details later in the lecture…

# Next section: Machine learning

# NLP and machine learning

- **Machine learning**
  - A method learns from experience wrt. a prediction task and a performance measure, if its performance on the task increases with the experience.
  - The goal is to learn a model *y* that approximates an unknown target function $\gamma$

$\gamma$. A human expert on arguments

*y*. An NLP method for argument mining

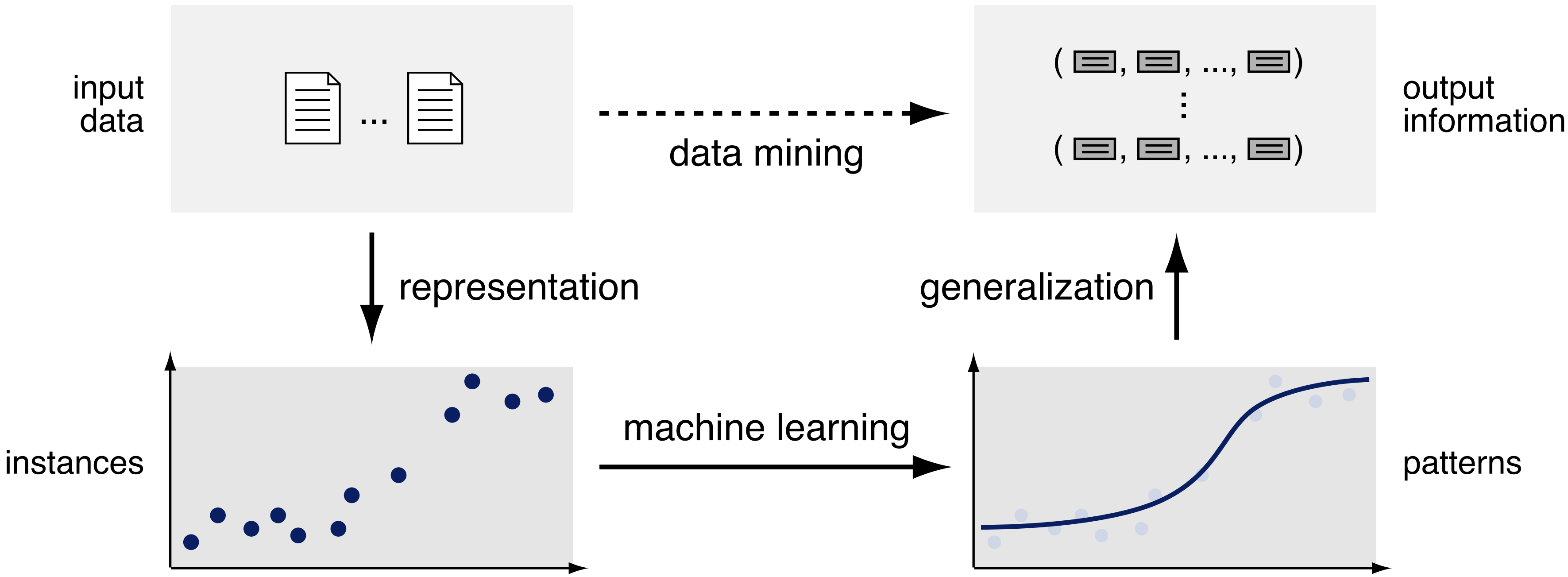
- **Main types of machine learning**
  - Supervised. Derive a model *y* from training input data with correct outputs; *y* can then predict output information for other inputs.
  - Unsupervised. Derive a model *y* from input data only; y describes the organization and association of data.

Training data. Input texts with annotated arguments

- **Machine learning in NLP**
  - Statistical and neural NLP methods use machine learning to infer outputs
  - Vice versa, the output of NLP may serve as input to machine learning

# NLP and data mining

- **NLP as data mining**
  - Machine learning enables NLP methods to perform a kind of data mining
  - Input data. A text corpus, i.e., a collection of texts to be processed
  - Output information. Annotations of the texts, or new texts

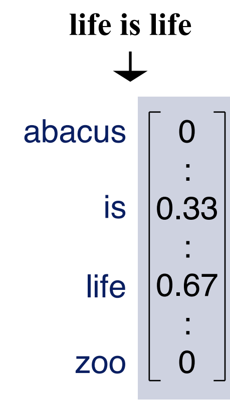


# Representation

- **Instance representation in NLP**
  - Most representations map an input to a vector of real values.

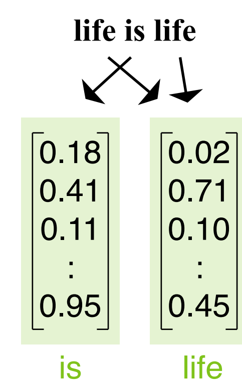
- **Feature representations**
  - Map input to a (sparse) vector of feature values **x** = ($x_1$, …, $x_m$).
  - Each $x_j$ captures a measurable property of the input.
  - Feature *types* are defined manually, features are learned.
  - Features are the basis of statistical NLP methods.

    Neural methods may also include features as part of their input.



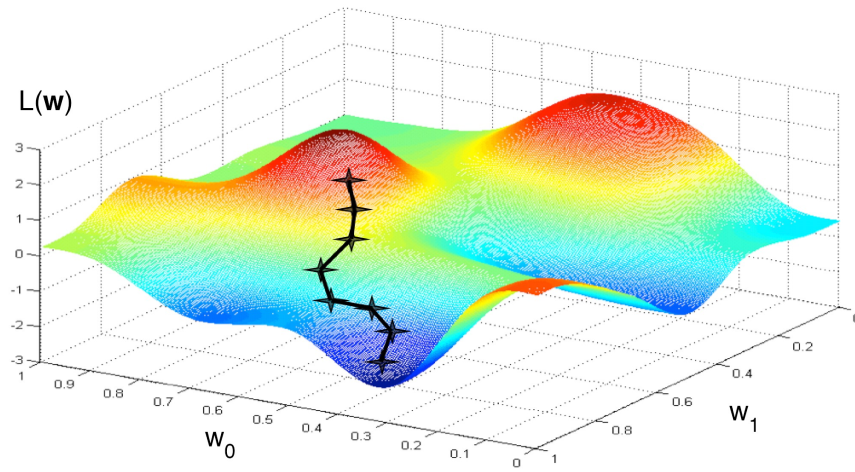

- **Embedding representations**
  - Map input to one or more (dense) vectors X = **($\mathbf{x}_1$, …, $\mathbf{x}_n$)**.
  - Each $\mathbf{x}_i$ *embeds* the *distributional semantics* of (parts of) an input.
  - An embedding model can be learned fully unsupervised.
  - Embeddings are the basis of neural NLP methods.

    Embeddings may also be part of a feature vector.



# Machine learning

- **Machine learning process**
  - A learning algorithm explores several candidate models *y*.
  - Each *y* assigns one weight $w_j$ to each feature $x_j$.
  - *y* is evaluated on training data against a cost function *L*.




Ng (2018)

  - Based on the result, the weights are adapted to obtain the next model.
  - The adaptation relies on an optimization procedure.

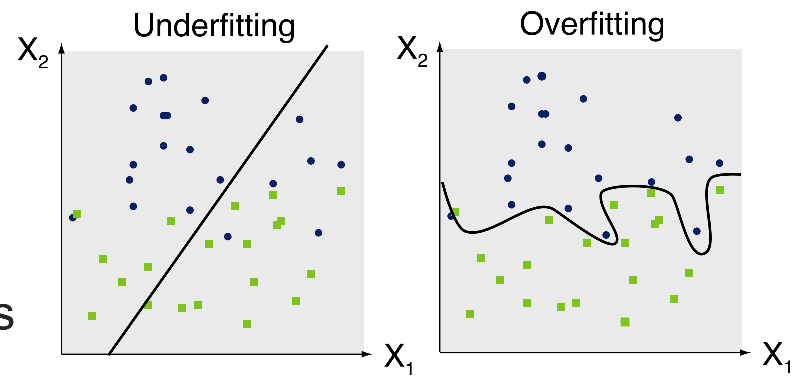
- **Common optimization procedures**
  - Batch gradient descent. In each step, *y* is adapted to all training instances
  - Stochastic gradient descent. Adapts *y* iteratively to each single instance

- **Hyperparameters**
  - Hyperparameters of learning algorithms cannot be optimized in training.
  - They need to be optimized against a validation set.

# Generalization

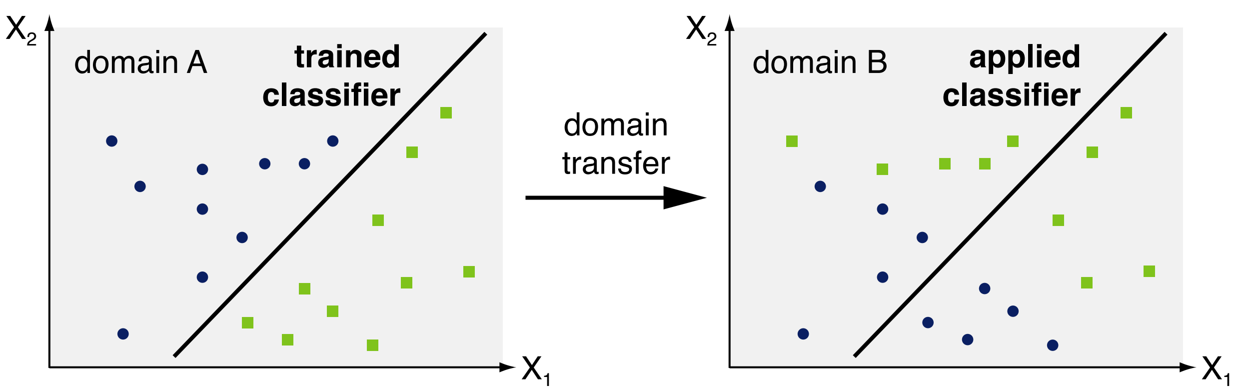
- **Underfitting and overfitting**
  - $y$ aims to approximate the complexity of $\gamma$ well based on the training data.
  - Underfitting. $y$ misses relevant properties
  - Overfitting. $y$ captures irrelevant properties

    A common technique to avoid overfitting is *regularization.*




- **Domain transfer**
  - Domain. A set of texts that share certain properties

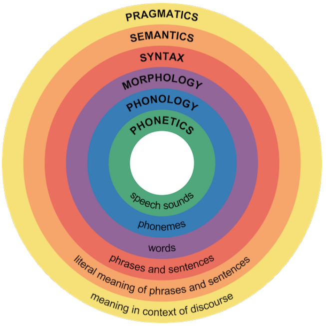
    May refer to a topic, genre, style, or similar — or combinations
  - Many algorithms work better in the training domain than in others.



# Next section: Conclusion

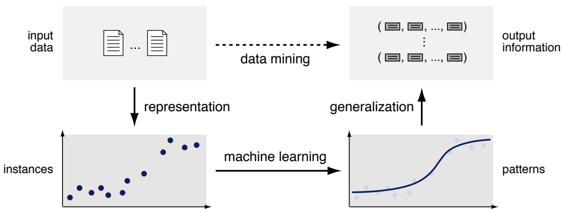
# Conclusion

- **Basics of NLP**
  - Linguistic knowledge, from phonetics to pragmatics
  - Empirical methods for development and evaluation
  - Stastistical (and rule-based) techniques for analysis tasks

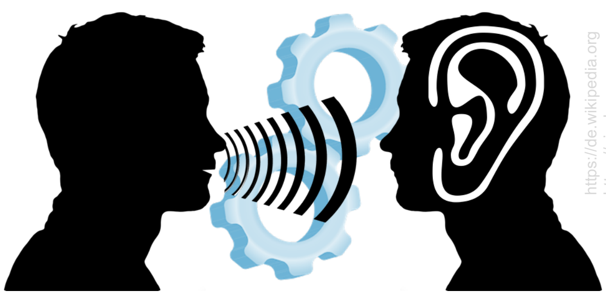



https://en.wikipedia.org

- **How to approach NLP tasks?**
  - Start from annotated text corpora
  - Develop approach that uses rules or learns patterns
  - Evaluate quality of its output empirically




- **Goals of NLP**
  - Technology that can process natural language
  - Empirical explanations of linguistic phenomena
  - Solutions to problems from the real world



https://de.wikipedia.org
https://pixabay.com

# References


- **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
- **Ng (2018).** Andrew Ng. Machine Learning. Lecture slides from the Stanford Coursera course. 2018. https://www.coursera.org/learn/machine-learning
- **Wachsmuth (2020).** Henning Wachsmuth. Introduction to Text Mining. Lecture slides. Winter term, 2020. https://cs.upb.de/css/teaching/courses/text-mining-w20/