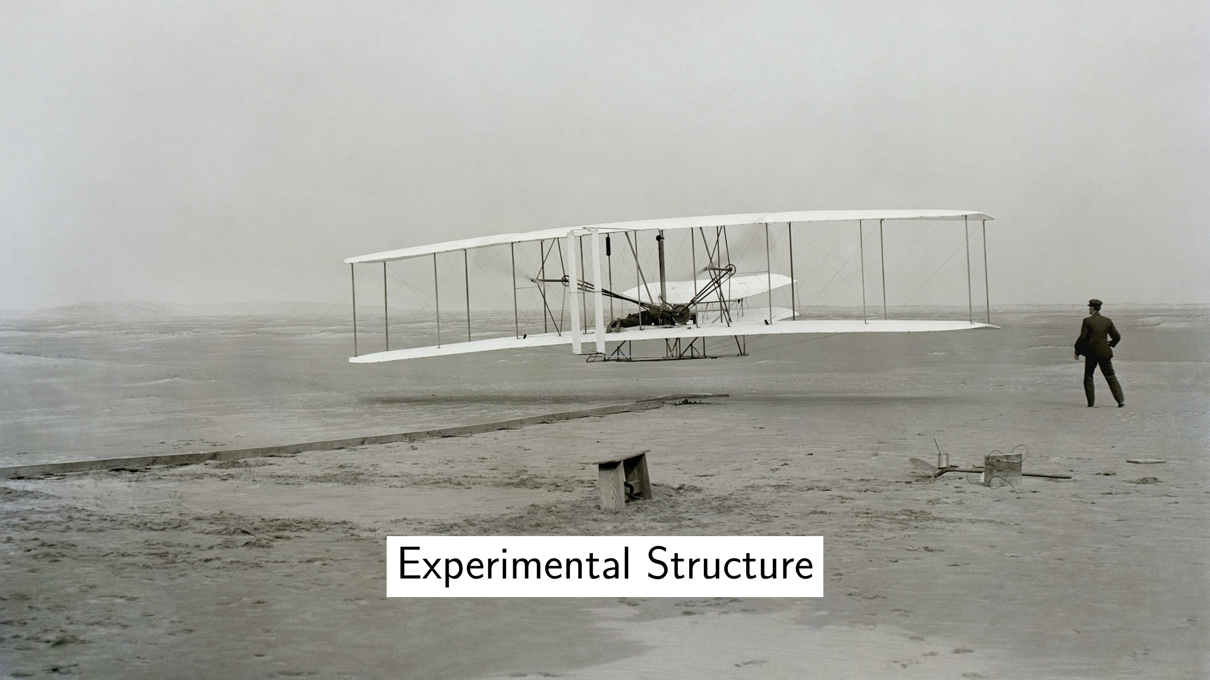
NLP-Experimente: Überblick und Workflow

HS Experimentelles Arbeiten in der Sprachverarbeitung

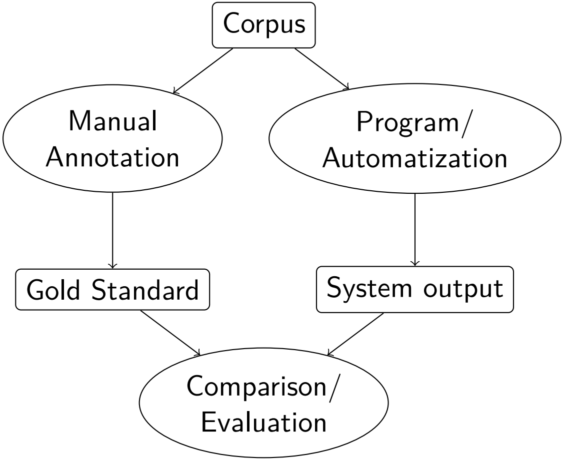
Nils Reiter
nils.reiter@uni-koeln.de

3. November 2022

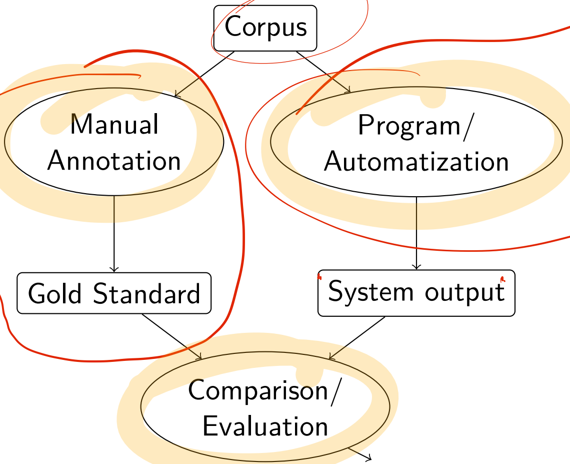


Experimental Structure

Experiments

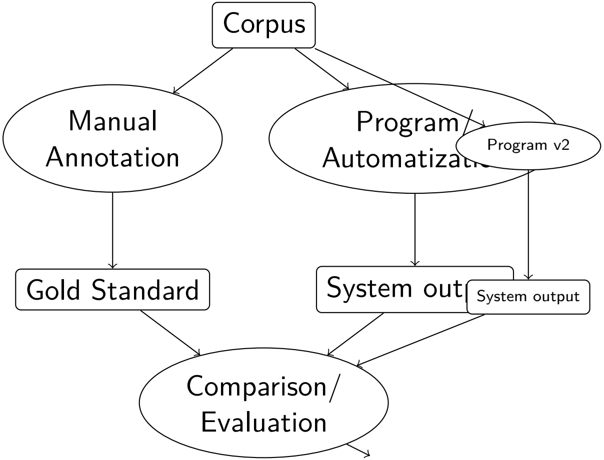


Experiments



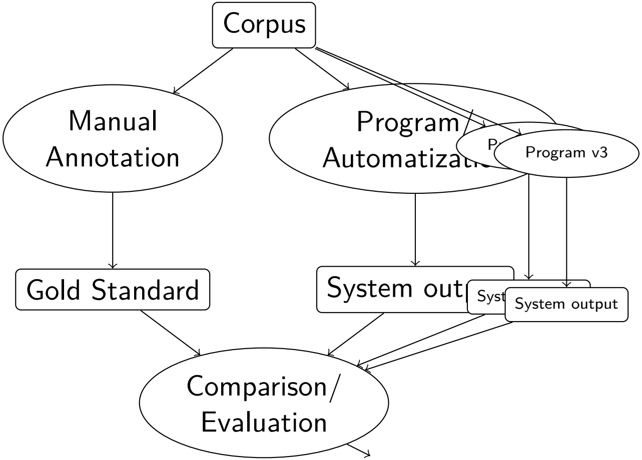
	P	R	F
Programm	5.7	9.2	...
v2	9.9	16.7	...
v3	15.3	21.8	...

Experiments



	P	R	F
Programm	5.7	9.2	...
v2	9.9	16.7	...
v3	15.3	21.8	...

Experiments



	P	R	F
Programm	5.7	9.2	…
v2	9.9	16.7	…
v3	15.3	21.8	…

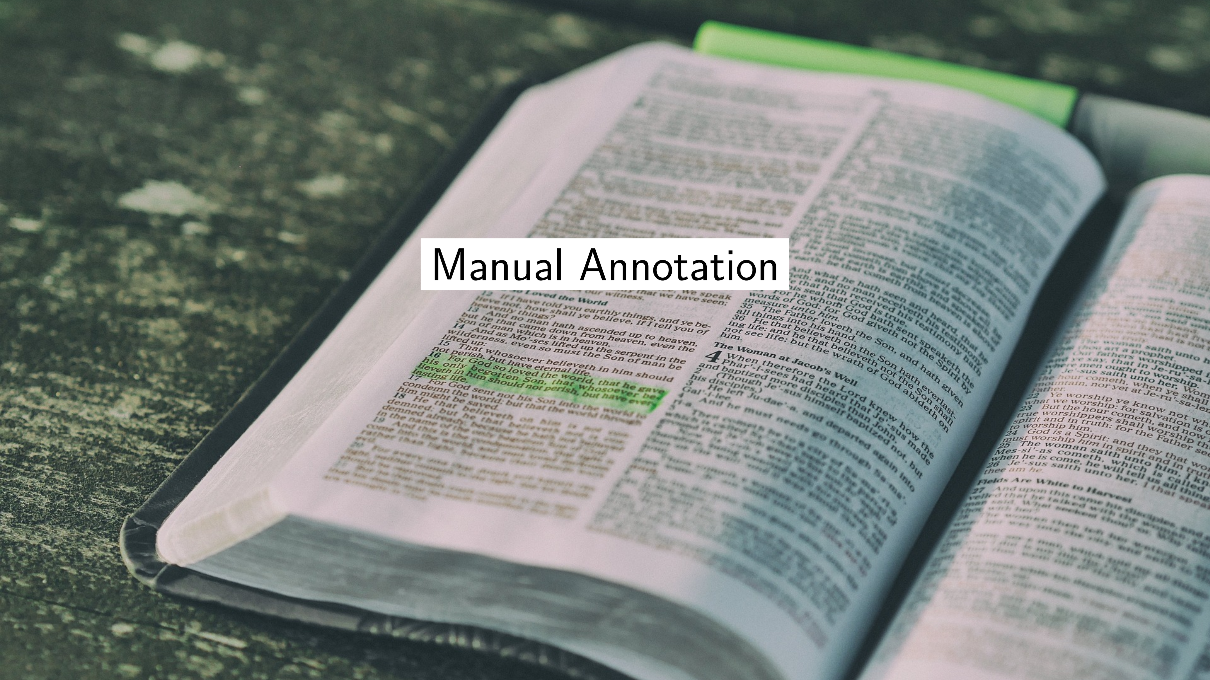
Experiments

- Reproducibility
- Hypotheses about the operationalisation of language/text phenomena

Example

- Position within a sentence is indicative for the part of speech
- Meaning of a word depends on its context
- The protagonist of a play is the character who talks the most

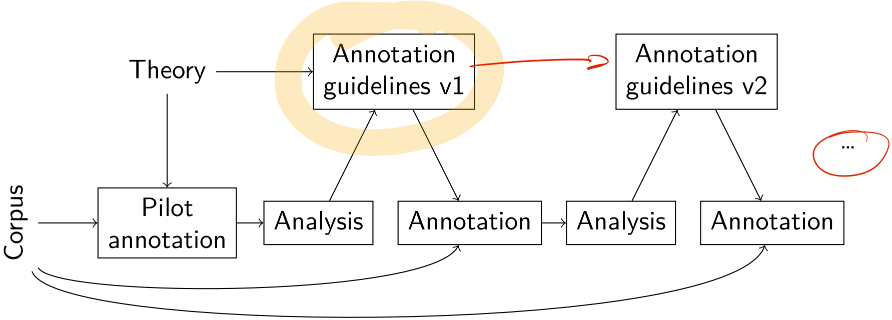
Manual Annotation



Annotation

- Interdisciplinary ‘false friend’
- Different meanings in different disciplines
  - Adding TEI/XML markup: DH community
  - Adding comments to page margins: Hermeneutic traditions
    - Literary studies, bible studies
  - Assigning categories to textual material: (computational) linguistics

Annotation Workflow



Hovy/Lavid (2010); Pagel et al. (2018)

Annotation guidelines

- Describe the way to create the machine-readable truth
- What is to be annotated (which words)
- Working definitions or tests for categories
- Living documents: Need to be iteratively improved
- Community-wide accepted standards are needed

Annotation Analysis

- Multiple annotators annotate the same text(s)
- Annotations are compared
- Disagreements can be quantified ('Inter-Annotator-Agreement', IAA)

  Cohen, 1960; Fleiss, 1971; Fournier, 2013; Mathet et al., 2015
- Inter- und Intra-AA
- … it's also a good idea to talk to the annotators

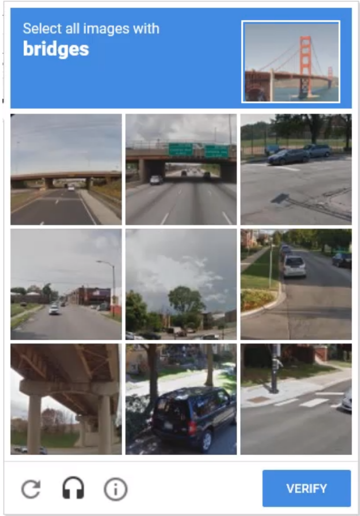
Indirect Annotations

- Annotations as a by-product of games
  - `https://www.artigo.org` Kohle (2010)
  - `https://anawiki.essex.ac.uk/phrasedetectives/` Chamberlain et al. (2008)

Indirect Annotations

- Annotations as a by-product of games
  - `https://www.artigo.org` Kohle (2010)
  - `https://anawiki.essex.ac.uk/phrasedetectives/` Chamberlain et al. (2008)
- Captchas for OCR correction

Indirect Annotations



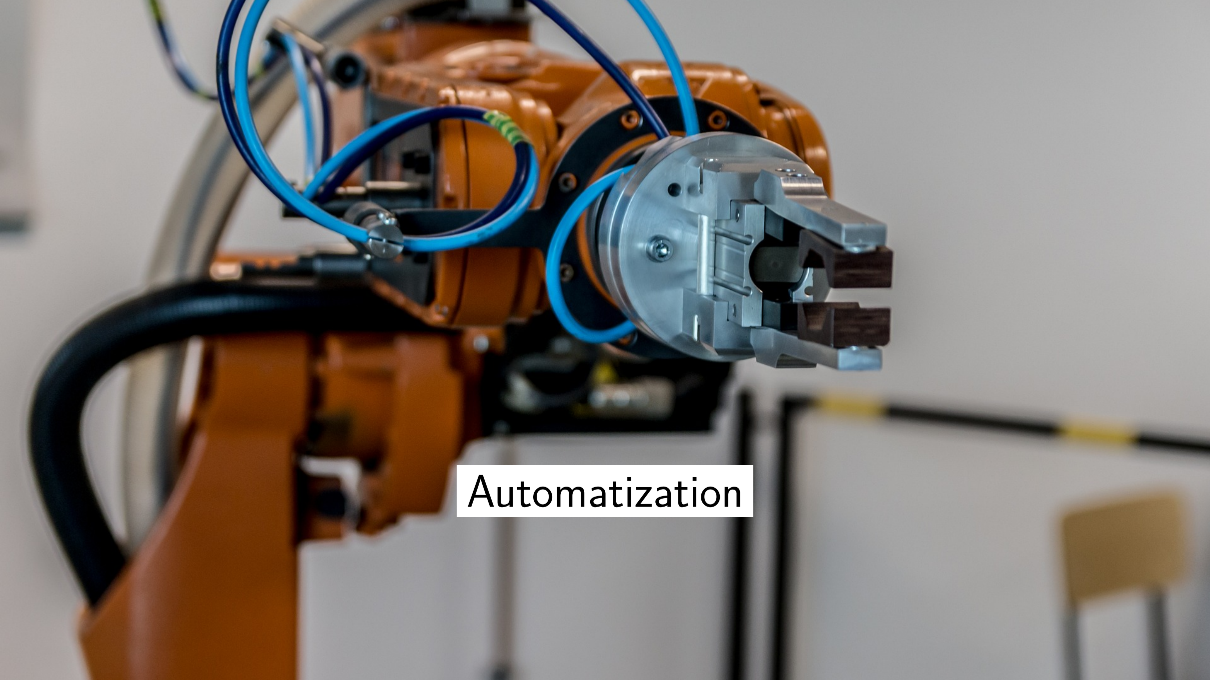
- Annotations as a by-product of games
  - https://www.artigo.org 2010)
  - https://anawiki.essex.ac.uk/phrasedetectives 08)
- Captchas for OCR correction

Learning from Raw Data

- Train on things that are already there
- word2vec: Is ‘dog’ a context word of ‘lazy’? Mikolov et al. (2013)
- BERT Devlin et al. (2019)
  - Can you fill in this blanked word? (“masked language modeling”, MLM)
  - Are these two sentences natural neighbours? (“next sentence prediction”, NSP)

Learning from Raw Data

- Train on things that are already there
- word2vec: Is 'dog' a context word of 'lazy'? Mikolov et al. (2013)
- BERT Devlin et al. (2019)
  - Can you fill in this blanked word? ("masked language modeling", MLM)
  - Are these two sentences natural neighbours? ("next sentence prediction", NSP)
- Training data available in abundance
  - As long as there is digital data for a language
  - ⚠ Difficult to control what exactly is in there
    - More obvious for text-image data sets Birhane et al. (2021) haveibeentrained.com


Automatization

Welche Methoden kennen Sie?

- Entscheidungsbäume / Decision Trees
- Naive Bayes
- Logistic Regression

	$\mathbb{1}_{cp}$	L	Annotation
T1	1	15	Complaint
T2	0	27	Kein Complaint

Systems

- Predict annotations
- Ideally: The same annotations as a human (the correct ones)
- Parameters
  - On what exactly does the program make predictions?
  - What information, criteria and features does it need?

Systems

- Predict annotations
- Ideally: The same annotations as a human (the correct ones)
- Parameters
  - On what exactly does the program make predictions?
  - What information, criteria and features does it need?

System types

- Rule-based (not so popular anymore)
- Supervised machine learning
  - Deep learning

Supervised Systems

- Classification: Assign items into previously known categories
  - Sequence labeling: Special case. Class for item n depends on item $n-1$
- Learn patterns from annotated data
- Relations between input (X) and output (Y)
  - Can be an n-to-m relation, but mostly n-to-1 (i.e., we predict a single target category)

Features

- The properties of a item that is to be classified
- Classical machine learning
  - Manual coding of explicit, scientifically validated features: Feature extraction
  - "Translation" of the corpus into feature vectors
  - Feature engineering
  - Design and implementation of feature extractors
  - Linguistic features need to be determined somehow
    $\rightarrow$ Dependencies, modularization

Features

- The properties of a item that is to be classified
- Classical machine learning
  - Manual coding of explicit, scientifically validated features: Feature extraction
  - “Translation” of the corpus into feature vectors
  - Feature engineering
  - Design and implementation of feature extractors
  - Linguistic features need to be determined somehow
    $\rightarrow$ Dependencies, modularization
- Deep learning
  - Embeddings used as features
    - A word is mapped onto an n-dimensional vector, which is then put into the ML system
    - Vector dimensions = features
    - But not interpretable anymore

Parameters and Hyper Parameters

Parameters

- What is learned by the algorithm during training
  - E.g., probability/frequency of feature F and class C ($=$ weights)
- Parameters are stored in the model

Parameters and Hyper Parameters

Parameters

- What is learned by the algorithm during training
  - E.g., probability/frequency of feature F and class C (= weights)
- Parameters are stored in the model

Hyper Parameters

- Set during the training process by us
  - E.g., number of training epochs in a neural network, data set size, ...
- Not automatically optimised, but important for performance

Parameters and Hyper Parameters

Parameters

- What is learned by the algorithm during training
  - E.g., probability/frequency of feature F and class C (= weights)
- Parameters are stored in the model

Hyper Parameters

- Set during the training process by us
  - E.g., number of training epochs in a neural network, data set size, ...
- Not automatically optimised, but important for performance
- Development set: Find optimal hyper parameters

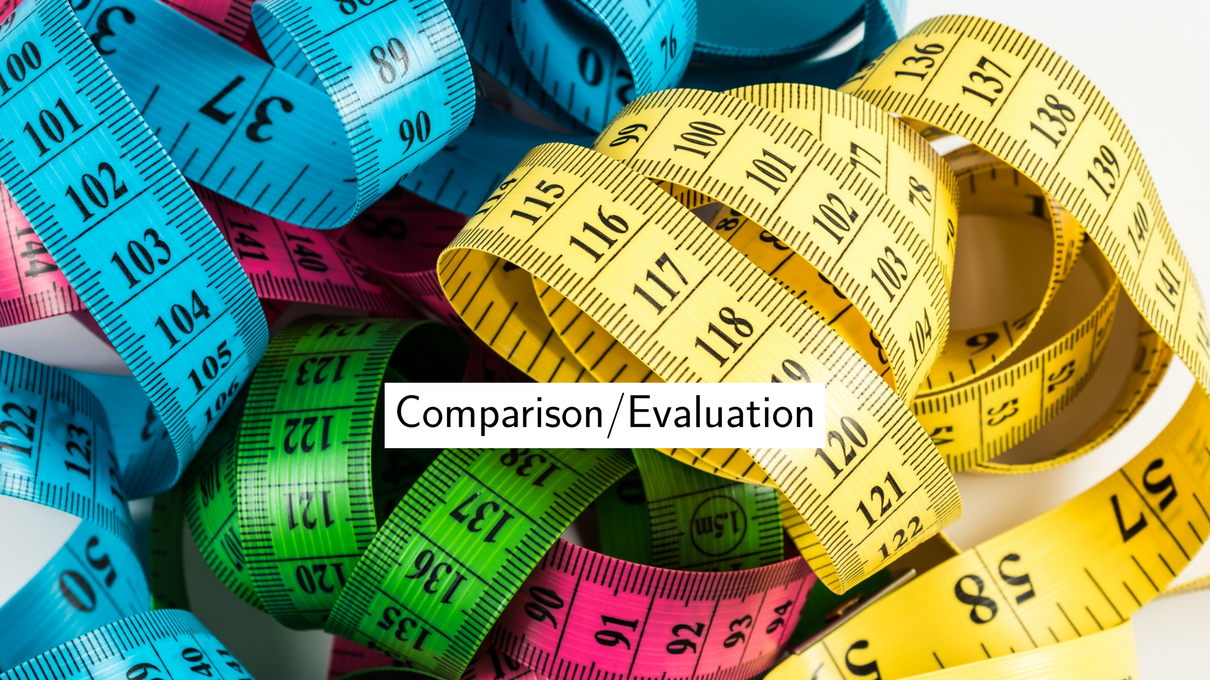
Example: Parts of Speech

Feature	Data type
Case	Binary
Length	> 0
Sentence initial	Binary

Table: Features

Token	Case	L.	S. initial
Der	u	3	Y
Hund	u	4	N
bellt	l	5	N
.	?	1	N
Die	u	3	Y
Katze	u	5	N
schnurrt	l	8	N
.	?	1	N

Table: Feature extraction



Comparison/Evaluation

Evaluation

Intrinsic

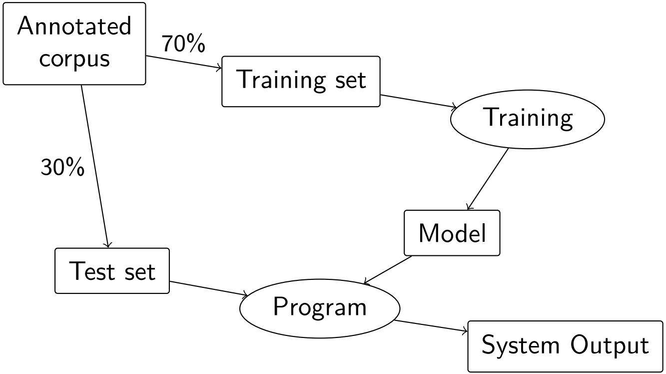
- Compare the automatically produced annotations with the gold standard
- Can be quantified (similar to IAA)
  - *precision*, *recall*, *f-score*
- System treated as a black box

Extrinsic

- Use of the program in another program that can be evaluated
  - *downstream tasks*
  - e.g., use of a PoS tagger in a machine translation system

Intrinsic Evaluation

- Goal: Predict the quality on new data
- The program cannot have seen the data, so that it's a realistic test



Classification Evaluation Metrics

(MS99, 267 ff.)

- Accuracy: How many items were correctly classified over all classes?
  (one value for everything)
- Precision: How many of the items classified as category C actually belong to category C?
  (one value per category)
- Recall: How many of the items in category C have been classified as C
  (one value per category)
- F-Score: Harmonic mean between precision and recall

Baseline

- What does an evaluation score tell us?
  - Nothing, if not compared to anything

Baseline

- What does an evaluation score tell us?
  - Nothing, if not compared to anything
- Artificial baselines
  - Majority baseline: Classify everything into the most frequent class
  - Random baseline: Classify everything at random

Baseline

- What does an evaluation score tell us?
  - Nothing, if not compared to anything
- Artificial baselines
  - Majority baseline: Classify everything into the most frequent class
  - Random baseline: Classify everything at random
- Self baselines
  - Take a single feature (classical machine learning)
  - Pre-trained embeddings
  - BERT without fine-tuning

Baseline

- What does an evaluation score tell us?
  - Nothing, if not compared to anything
- Artificial baselines
  - Majority baseline: Classify everything into the most frequent class
  - Random baseline: Classify everything at random
- Self baselines
  - Take a single feature (classical machine learning)
  - Pre-trained embeddings
  - BERT without fine-tuning
- Foreign baselines
  - Last year's system
  - Competition system
  - Shared task winner

If baseline has hyper parameters, they need to be optimized as well (for a fair comparison)

Results

	P	R	F
Baseline 1	...	...	...
Baseline 2	...	...	...
Variant 1	...	...	...
Variant 2	...	...	...
Variant 3	...	...	...

Table: A typical results table

Error Analysis

- Systems do not deliver perfect results (i.e., scores are below $100\,\%$)
- What can we say about the remaining errors?

Error Analysis

- Systems do not deliver perfect results (i.e., scores are below $100\,\%$)
- What can we say about the remaining errors?
- Workflow
  - Extract n errors, inspect them manually
  - Can we detect regularities/patterns in them? E.g., why they were misclassified?
  - Ideally, error analysis makes quantitative statements about error sources

Error Analysis

- Systems do not deliver perfect results (i.e., scores are below $100\,\%$)
- What can we say about the remaining errors?
- Workflow
  - Extract n errors, inspect them manually
  - Can we detect regularities/patterns in them? E.g., why they were misclassified?
  - Ideally, error analysis makes quantitative statements about error sources
- Directions for further improvements of the system

Analysis != Generation

- Analysis: Text as input, annotations as output
- Generation: Some data as input, text as output
  - Machine translation, digital assistants, summarization, …
- Different kinds of systems (not classification)
- Different evaluation metrics
  - Machine translation: BiLingual Evaluation Understudy (BLEU) Papineni et al. (2001)
    - Weighted overlap between reference and system