SENTIMENT ANALYSIS

90 MIN WORKSHOP

Robin Rojowiec

https://www.linkedin.com/in/rojowiec/

robin.rojowiec@de.ibm.com

https://twitter.com/Robin_it_is

https://www.instagram.com/robinlphood/

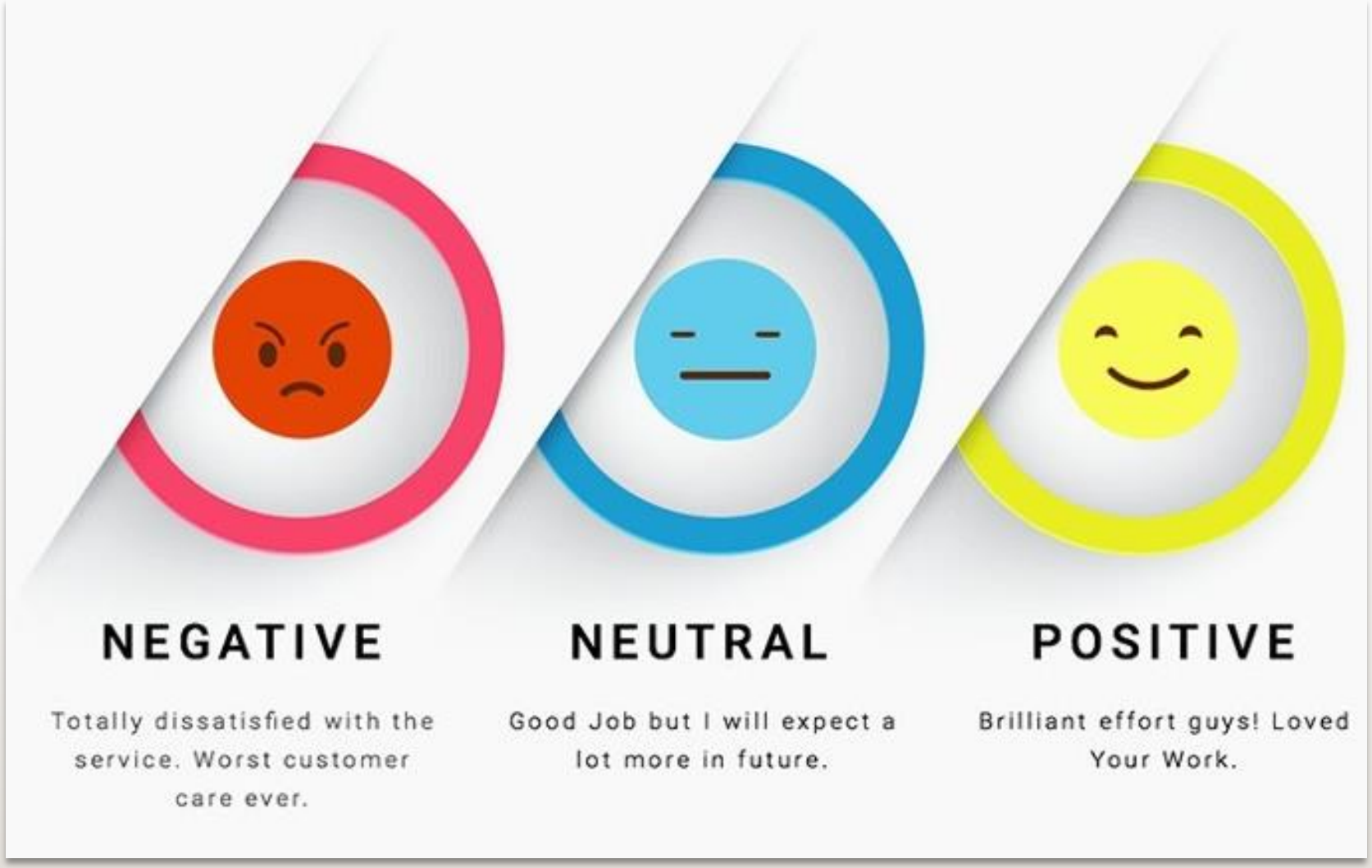
Cognitive Engineer,
IBM Watson and Cloud Platform

Computational Linguistics,
Master of Science (ongoing)

#nlp #ml #cognitive #Watson #linguistics #java
#climbing #drums #moredrums #coding #travel

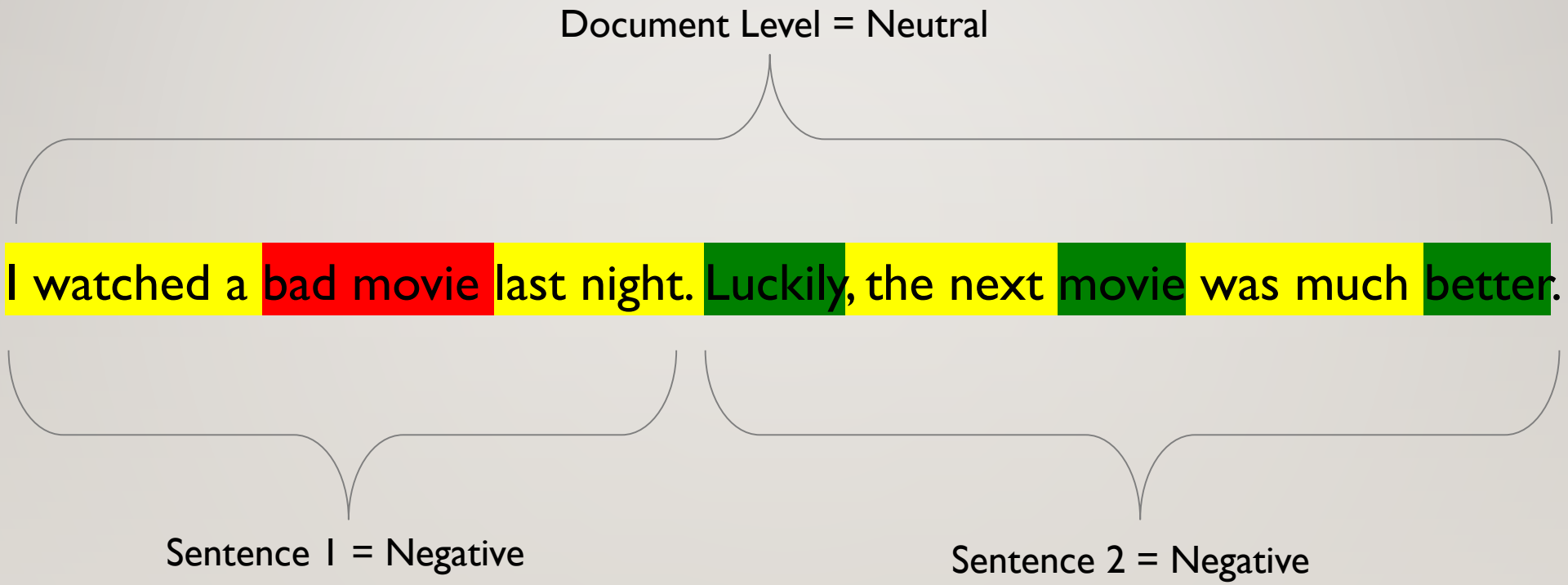
AGENDA

- Introduction 10 min
- Excercise 70 min
  - Task
  - Implementation
  - Testing and Improvements
- Wrap up & Outlook 10 min



Quelle: https://www.kdnuggets.com/2018/03/5-things-sentiment-analysis-classification.html

SENTIMENT BASED ON WORDS



SENTIMENT BASED ON WORDS (ASPECTS)

I watched a bad movie last night. Luckily, the next movie was much better.

Movie: Negative

Movie: Positive

EXERCISE

Task

WORD BASED BINARY SENTIMENT CLASSIFICTION

- Predict sentiment class of movie reviews
  - 25 000 reviews for training, equal split positive/negative
- Calculate class probability, feature probability and the probability of each word belonging to each class
- Sentiment Class Probability will be:
  - class_probability * feature_probabilities = class_prediction_probability
  - Normalize by all probabilites: P_pos = P_pos / (P_pos + P_neg)
- Use logarithmus naturalis (ln) to avoid underflow and discard unknown tokens!

WORD BASED BINARY SENTIMENT CLASSIFICTION STEPS

1) Load text files
2) Tokenize and remove stopwords
3) Calculate for each token:
    1) Number of occurences in the documents
    2) Number of occurences per class (positive/negative)
4) Sum up counts and calculate probabilites
5) Write prediction function which normalizes the probabilites
6) Run test script and modify your code to improve results

(You should end up with around 70% accuracy)

WORD BASED BINARY SENTIMENT CLASSIFICTION BONUS

Bouns 1)

Calculate integrate the TF/IDF value (document frequency ratio to word frequency) to improve accuracy

Bonus 2)

Change granularity to sentence level or aspect-level (rule-based)

Bonus 3)

Try to improve performance with word bi- and trigrams as well as character bi- and trigrams

EXERCISE

Implementation

WRAP UP & QUESTIONS

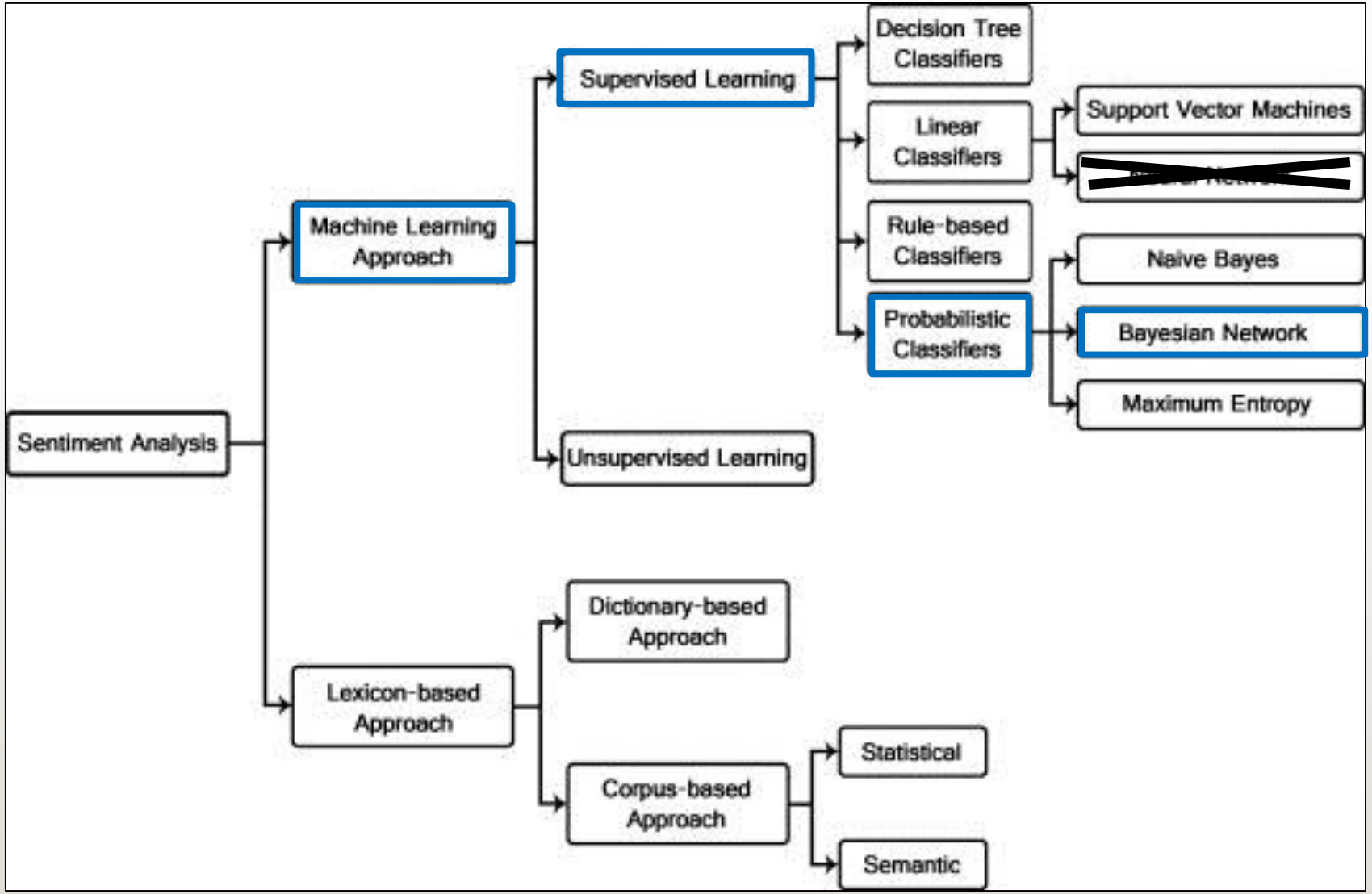
WRAP UP

- ~ 70 % Accuracy => 7/10 Classifications are correct
- Simple Algorithm which uses Probabilistic Properties
- Nice Excercise, in practice you would use on of these:

And many more!

FURTHER QUESTIONS ?

BACKUP


Sentiment Analysis
Machine Learning Approach
Supervised Learning
Decision Tree Classifiers
Linear Classifiers
Support Vector Machines
[redacted]
Rule-based Classifiers
Probabilistic Classifiers
Naive Bayes
Bayesian Network
Maximum Entropy
Unsupervised Learning
Lexicon-based Approach
Dictionary-based Approach
Corpus-based Approach
Statistical
Semantic

PROBAIBILISTIC SENTIMENT CLASSIFICATION

- Word-based (Unigram, 1 Word = 1 Feature)
- Probability -> Frequency for xi / Frequency x
- Granularity:
  - Word-Level (Aspects)
  - Sentence-Level
  - Document-Level
- Unseen words:
  - Backoff / Laplace Smoothing
  - Discard words