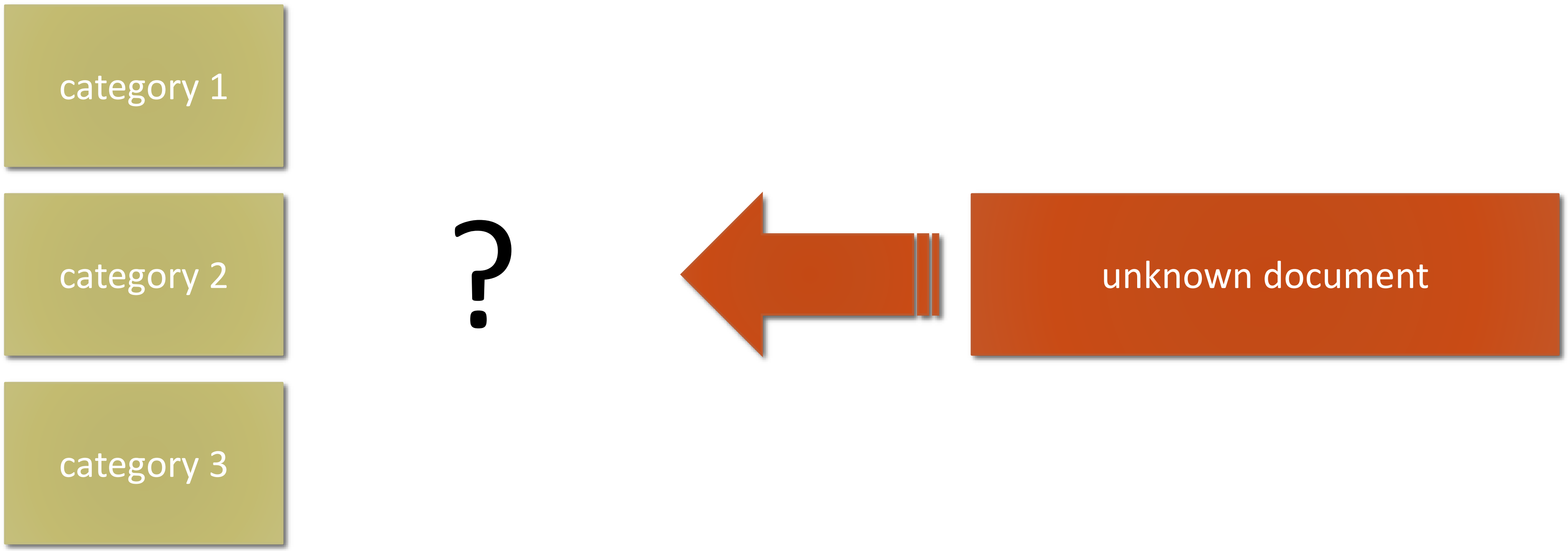
Augmenting Naive Bayes Classifiers with Statistical Language Models

PENG, SCHUURMANS, WANG

Naive Bayes classifier



category 1

category 2

category 3

?

unknown document

Get max P(c|d)

$$P(c|d) = P(c) \cdot \frac{P(d|c)}{P(d)}$$

$$d = (v1, v2, v3 \ldots)$$

$$c^* = \arg\max_{c \in C} \left\{ P(c) \cdot \prod P\left(v_j|c\right) \right\}$$

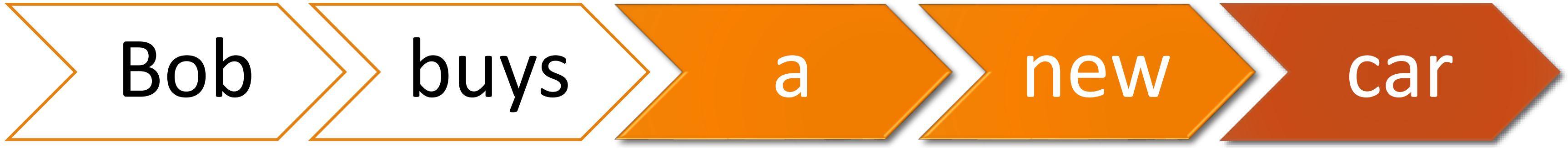
given:

-v_j are independant

-P(d) is constant

Statistical n-grams

- measures probability of a token given the tokens before
- Markov independance: only the last n-1 tokens are relevant



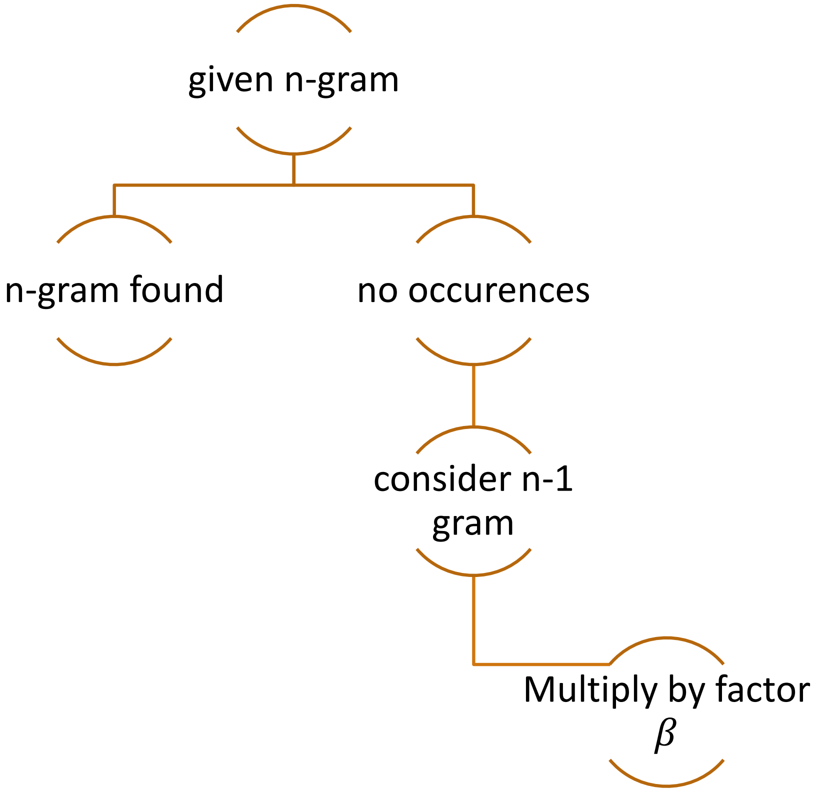
n = 3

Statistical n-grams

$$P(w_i|w_{i-n+1} \dots w_{i-1}) = \frac{\#(w_{i-n+1} \dots w_i)}{\#(w_{i-n+1} \dots w_{i-1})}$$

What to do if w_i never occurs?

Back off model:



How to choose β?

$$\beta(w_{i-n+1} \dots w_{i-1}) = \frac{1 - \sum_{x:\#(w_{i-n+1} \dots w_{i-1} x)>0} P\,(x|w_{i-n+1} \dots w_{i-1})}{1 - \sum_{x:\#(w_{i-n+1} \dots w_{i-1} x)>0} P\,(x|w_{i-n+2} \dots w_{i-1})}$$

Smoothing

Absolute:

$$P(w_i|w_{i-n+1} \dots w_{i-1}) = \frac{\#(w_{i-n+1} \dots w_i) - b}{\#(w_{i-n+1} \dots w_{i-1})}$$

Linear:

$$P(w_i|w_{i-n+1} \dots w_{i-1}) = \left(1 - \frac{n_1}{T}\right) \frac{\#(w_{i-n+1} \dots w_i)}{\#(w_{i-n+1} \dots w_{i-1})}$$

Conclusion

$$c^* = \arg\max_{c \in C} \left\{ P(c) \cdot \prod_i P_c(w_i | w_{i-n+1} \dots w_{i-1}) \right\}$$

Consider all n-grams in test data and multiply the probs

Greek authorship attribution

data set

Training data

- 10 authors
- 10 texts each

Test data

- Same 10 authors
- 10 texts each

profile based approach

Results (greek authorship attribution)

char level models:

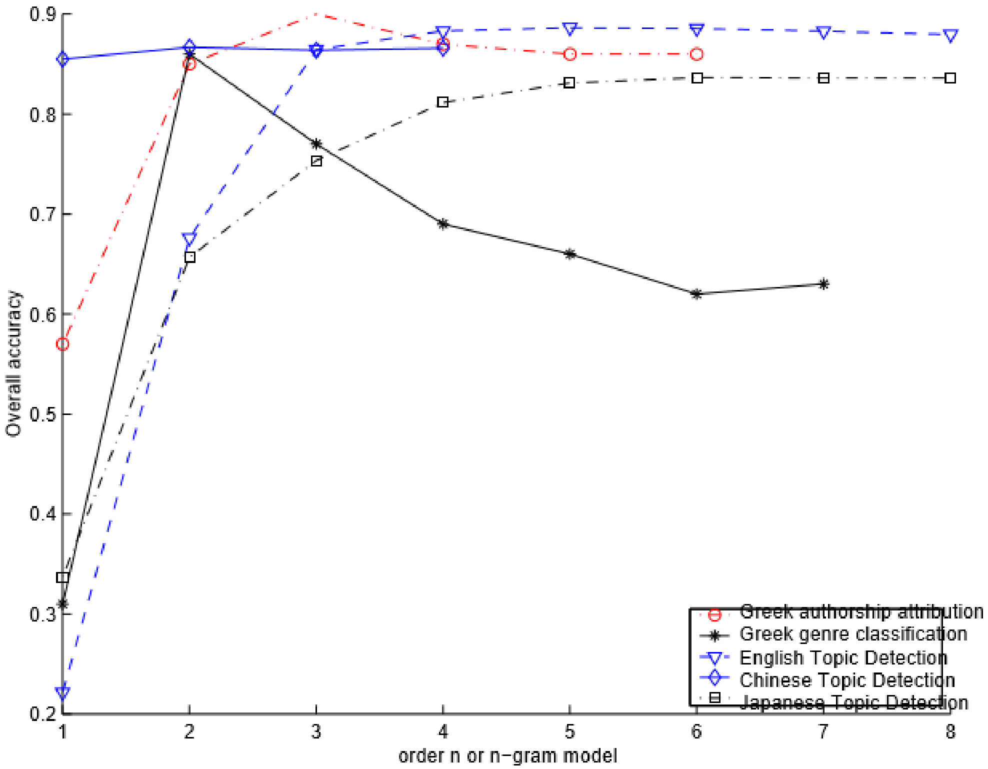
n=1, absolute smoothing: 47% (paper: 57%)

world level models:

n=1, absolute smoothing: 67% (paper: 96%)
n=2, absolute smoothing: **92%** (paper: 96%)
n=1, linear smoothing: 66% (paper: 96%)

good results for long texts

n-gram size


Overall accuracy
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
1
2
3
4
5
6
7
8
order n or n−gram model
Greek authorship attribution
Greek genre classification
English Topic Detection
Chinese Topic Detection
Japanese Topic Detection

Points of interest

- Which probability do we assign, if a unigram does not exist in training data?
  - Add one to count
- Options to reduce computation time
  - Multicore
- The algorithm seems to work with just taking word lenght into consideration
  - approx. 50% accuracy