FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

PP Attachment Problem

LIAD MAGEN

Remember our 2nd lesson?

Let’s recap

Types of Classification Problems

- Binary: $y \in \{-1,\ 1\}$
- Multi-Class: $y \in \{1, 2, \dots, k\}$
- Multi-Label: $y \in 2^{\{1,2,\dots,k\}}$
- (Regression...?)

Types of classifiers

- Generative vs Discriminative
- Probabilistic vs Non-Probabilistic
- Linear vs non-Linear

$$P(x, y)$$

$$P(y \mid x)$$

$$score(x, y)$$

$$f(x) = y$$

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

Types of classifiers

- > Generative vs Discriminative
- > Probabilistic vs Non-Probabilistic
- > Linear vs non-Linear

$P(x,y)$	Generative
$P(y \mid x)$	Discriminative
$score(x,y)$	Discriminative
$f(x) = y$	Discriminative

Types of classifiers

- Generative vs Discriminative
- Probabilistic vs Non-Probabilistic
- Linear vs non-Linear

prob	$P(x, y)$	Generative
prob	$P(y \mid x)$	Discriminative
Non-prob	$score(x, y)$	Discriminative
Non-prob	$f(x) = y$	Discriminative

Popular Classifiers

- kNN (k-Nearest Neighbors)
- Decision Trees
  - Decision Forests
  - Gradient-boosted Forests
- Logistic Regression
- Naïve Bayes
- SVM
- Neural Networks

Scikit-learn (sklearn):
a popular and good package for those activities

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

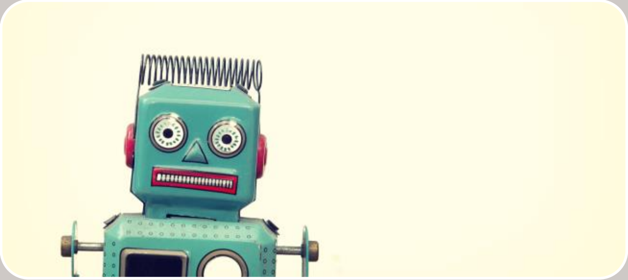
The Big Picture

Supervised

- Decision Tree
- Random Forest
- Logistic Regression
- Naïve Bayes
- K-Nearest Neighbor
- Support Vector Machine

Unsupervised

- Latent Dirichlet Allocation
- K-Means
- PCA



Reinforcement Learning

Generic NLP Solution

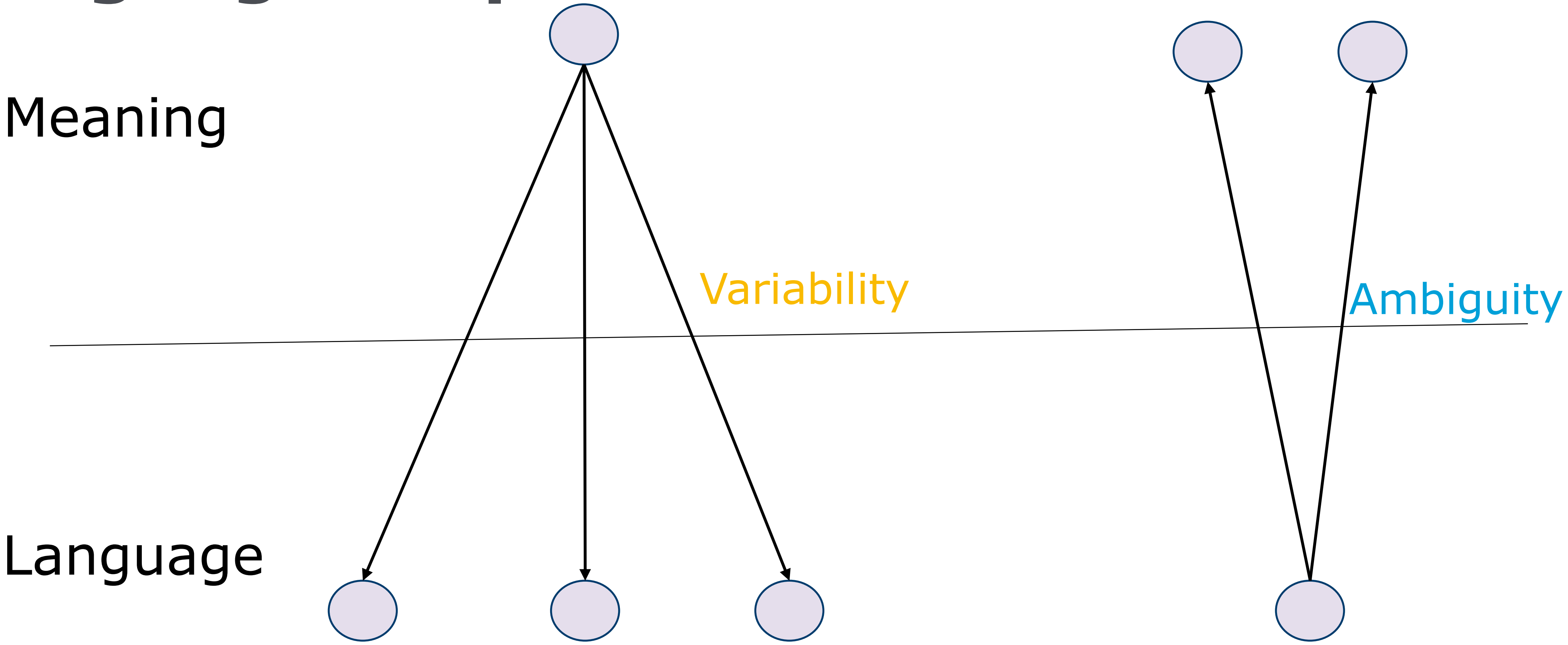
- Find an annotated corpus
- Split it into train/dev & test parts
- Convert it to a vector representation
- Decide on the output type
- Decide on the features
- Convert each training example to a feature vector
- Train a machine learning model on the training set
- Apply your model on the test-set
- Measure the accuracy

Generic NLP Solution

> **Find an annotated corpus**

- Difficult to create your own corpus (expensive)
- *Decide* what are you classifying?
  What should the output classes be?
- *Consider*: is the problem even solvable?
  Can humans do that?
  At what level of accuracy can humans do it?

Language Properties



Ambiguity

> I saw the dog with the blue hat
> He talked to the girl in a harsh voice
> Graucho shot an elephant in his pajamas
> John found a sack of money
> He thought about filling the garden with flowers
> Collect the young children after school
> I saw a mouse on the hill with a telescope

Ambiguity

- I saw the dog with the blue hat
- He talked to the girl in a harsh voice
- Graucho shot an elephant in his pajamas
- John found a sack of money
- He thought about filling the garden with flowers
- Collect the young children after school
- I saw a mouse on the hill with a telescope

Ambiguity

Verb NP(1) preposition NP(2)

> I saw the dog with the blue hat
> He talked to the girl in a harsh voice
> Graucho shot an elephant in his pajamas
> John found a sack of money
> He thought about filling the garden with flowers
> Collect the young children after school
> I saw a mouse on the hill with a telescope

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

Ambiguity

Verb NP(1) preposition NP(2)

> I saw the dog with the blue hat

> He talked to the girl in a harsh voice

> Graucho shot an elephant in his pajamas

> John found a sack of money

> He **thought about** filling the garden with flowers

> Collect the young children after school

> I saw a mouse on the hill **with a telescope**

Ambiguity

verb NP(1) preposition NP(2)

verb NP(1) preposition NP(2)

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

Ambiguity

verb NP(1) preposition NP(2)

I ate pizza with olives

verb NP(1) preposition NP(2)

I ate pizza with friends

Ambiguity

verb NP(1) preposition NP(2)

I ate pizza with olives

verb NP(1) preposition NP(2)

I ate pizza with friends

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

The N-V PP attachment problem

- Given a 4-tuple: (verb, NP1, prep, NP2)
  - talked to the girl in a harsh voice
  - shot an elephant in his pajamas
  - found a sack of money
  - filling the garden with flowers

- Predict: V or N , where
  - **V** means a **V-PREP** relation (ate pizza with friends)
  - **N** means a **N-PREP** relation (ate pizza with olives)
- A binary classification task

Ambiguity

I saw the dog with the blue hat

He talked to the girl in a harsh voice

Graucho shot an elephant in his pajamas

John found a sack of money

He thought about filling the garden with flowers

Collect the young children after school

I saw a mouse on the hill with a telescope

UNIVERSITY OF APPLIED SCIENCES

Ambiguity

Leaving only the head ("main") words of each phrase.

- Should we do it?
- Why yes? Why not?

I saw the dog with the blue hat

He talked to the girl in a harsh voice

Graucho shot an elephant in his pajamas

John found a sack of money

He thought about filling the garden with flowers

Collect the young children after school

I saw a mouse on the hill with a telescope

The N-V PP attachment problem

- Given a 4-tuple: (verb, Noun1, prep, Noun2)
  - talked girl in voice
  - shot elephant in pajamas
  - found sack of money
  - filling garden with flowers

- Predict: V or N , where
  - **V** means a **V-PREP** relation (ate pizza with friends)
  - **N** means a **N-PREP** relation (ate pizza with olives)
- A binary classification task

How do we solve it?

Supervised classification:

- Given a dataset - X annotated samples + correct answers (Y):
    - talked girl in voice --> V
    - shot elephant in pajamas --> V
    - found sack of money --> N
    - filling garden with flowers --> V
- Prediction of a new tuple based on previous observation

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

Steps to solve

1. (Always!) Look at the data
2. (Always!) Define your measurement metric

   acc = correct / (correct + incorrect)

Conditional Probability

if $P(V \mid verb, noun1, prep, noun2) > 0.5$:

  return V

else

  return N

e.g., P(V | saw, boy, with, hat)

Maximum Likelihood Estimation (MLE)

> $P(V\ |verb, noun1, prep, noun2) = \frac{count(V,verb,noun1,prep\ noun2)}{count(*,verb,noun1,prep,noun2)}$

> Is this reasonable to do?
>> **Data Sparsity**
>> **Overfitting**

Count: How many times did we see this combination in the past?

UNIVERSITY OF APPLIED SCIENCES

Next Try: Majority baseline

Ignore the conditional – return only P(V):

$$P(V \,|verb, noun1, prep, noun2) \approx P(V)$$

Is this reasonable? Would it work?
What score would you expect?

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

Option #3 – noun1 based

$$P(V\ |verb, noun1, prep, noun2)\ \approx P(V\ |noun1)$$

Is this reasonable? Would it work?
What score would you expect?

Option #4 – prep based

$$P(V \mid verb, noun1, prep, noun2) \approx P(V \mid prep)$$

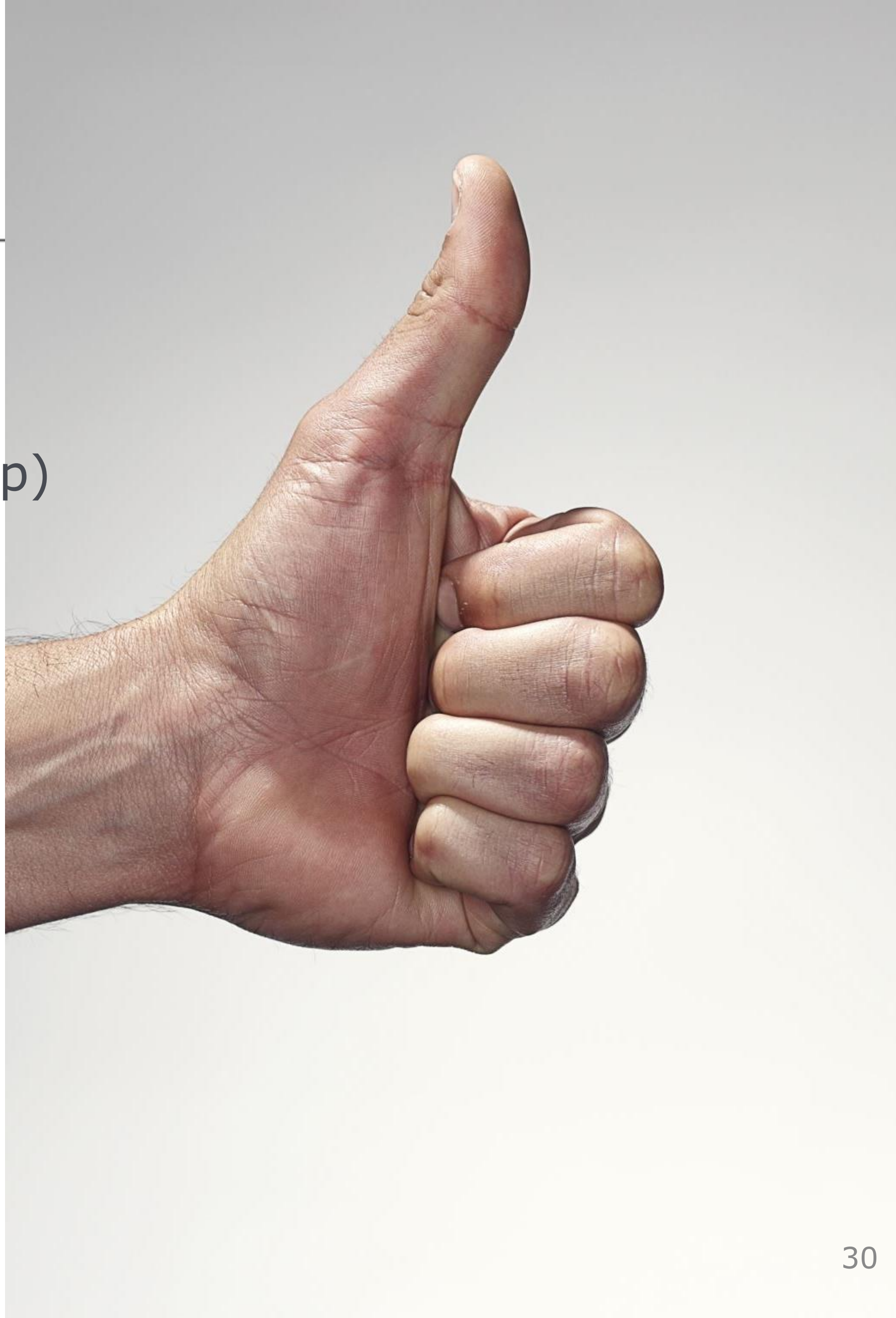
Is this reasonable? Would it work?
What score would you expect?

Option #4 – prep based

$$P(V \mid \text{verb}, \text{noun1}, \text{prep}, \text{noun2}) \approx P(V \mid \text{prep})$$

Works quite well. (Can you think why?)

But can we do better?



How about...

P(V| verb, prep) ?

P(V| noun1, prep) ?

P(V| noun1, noun2) ?

P(V| verb, noun1, noun2) ?

P(V| verb, noun1, prep) ?

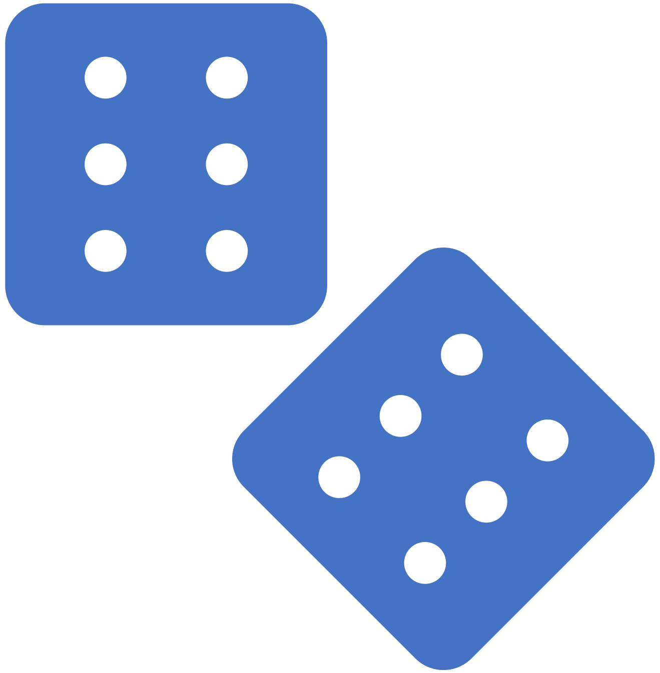
Or maybe a combination of all?

How can we combine the different probabilities?

Probability – a review

MLE (counting) leads to different fractions.
But:

> A probability function must:
  > Always be positive
  > Sum to one



Combining different probabilities

Obtain a probability through **linear interpolation**:

$$P_{interpolate} = \lambda_1 P_1 + \lambda_2 P_2 + \lambda_3 P_3 + \cdots + \lambda_k P_k$$

$$\lambda_1 + \lambda_2 + \lambda_3 + \cdots + \lambda_k = 1$$

Collins and Brooks' estimation

Interpolate triplets:

$$P_{triplet} \rightarrow P(V\,|v, n1, p), \qquad P(V\,|v, p, n2), \qquad P(V\,|n1, p, n2)$$

Notice we always include p (the preposition).

We do not have P(V|n1,n2) for example.

Why?

Interpolate pairs:

$$P_{pair} \rightarrow P(V\,|v, p), \qquad P(V\,|n1, p), \qquad P(V\,|\,p, n2)$$

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

Collins and Brooks' estimation

Interpolate triplets:

$$P_{triplet} \rightarrow \boldsymbol{P(V\,|v,n1,p)}, \qquad P(V\,|v,p,n2), \qquad P(V\,|n1,p,n2)$$

$$P(V|v,n1,p) = \frac{\#(V,v,n1,p,*)}{\#(*,v,n1,p)}$$

Interpolate pairs:

$$P_{pair} \rightarrow \boldsymbol{P(V\,|v,p)}, \qquad P(V\,|n1,p), \qquad P(V\,|\,p,n2)$$

$$P(V|v,p) = \frac{\#(V,v,*,p,\ *)}{\#(*,v,*,p,*)}$$

Combining the pair & triplet probabilities

Obtain a probability through **linear interpolation**:

How do we get this λ ?

$$P_{interpolate} = \lambda_1 P_1 + \lambda_2 P_2 + \lambda_3 P_3 + \cdots + \lambda_k P_k$$

$$\lambda_1 + \lambda_2 + \lambda_3 + \cdots + \lambda_k = 1$$

Collins and Brooks' interpolation: Gives more weight to frequent training samples.

$$\lambda_{v,n1,p} = \frac{count(v,n1,p)}{count(v,n1,p)+count(v,p,n2)+count(n1,p,n2)}$$

$$\lambda_{v,p,n2} = \frac{count(v,p,n2)}{count(v,n1,p)+count(v,p,n2)+count(n1,p,n2)}$$

$$\lambda_{n1,p,n2} = \frac{count(n1,p,n2)}{count(v,n1,p)+count(v,p,n2)+count(n1,p,n2)}$$

Collins and Brooks' estimation

> $P_3(V|v,n1,p,n2) = \frac{count(V,v,n1,p)+count(V,v,p,n2)+count(V,n1,p,n2)}{count(*,v,n1,p)+count(*,v,p,n2)+count(*,n1,p,n2)}$

This follows from:

$$P_3(V|v,n1,p,n2) = \lambda_{v,n1,p} P(V|v,n1,p) + \lambda_{n1,p,n2} P(V|n1,p,n2) + \lambda_{v,p,n2} P(V|v,p,n2)$$

$$P_{mle}(V|v,n1,p) = \frac{count(V,v,n1,p)}{count(*,v,n1,p)}$$

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

Collins and Brooks' estimation

> $P_3(V|v, n1, p, n2) = \frac{count(V,v,n1,p)+count(V,v,p,n2)+count(V,n1,p,n2)}{count(*,v,n1,p)+count(*,v,p,n2)+count(*,n1,p,n2)}$

> $P_2(V|v, n1, p, n2) = \frac{count(V,v,p)+count(V,n1,p)+count(V,p,n2)}{count(*,v,p)+count(*,n1,p)+count(*,p,n2)}$

> $P_1(V|v, n1, p, n2) = \frac{count(V,p)}{count(*,p)}$

Collins and Brooks' Back-off Algorithm

```
P(V|v,n1,p,n2) =
  if count(v, n1, p, n2) > 0
     return P4
  else if count(v,n1,p) + count(v,p,n2)+ count(n1,p,n2) > 0
     return P3
  else if count(v, p) + count(n1, p)+ count(p, n2) > 0
     return P2
  else if count(p) > 0
     return P1
  else:
     return P0 = count(V) / count(V+N)
```

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

Collins and Brooks' Back-off Algorithm

- Combination of probabilistic model and a heuristic
- Returns a well-behaved probability score –
  but not quite well motivated by probability theory
- Works well (84.1% accuracy) :

5 Results

The figure below shows the results for the method on the 3097 test sentences, also giving the total count and accuracy at each of the backed-off stages.

Stage	Total Number	Number Correct	Percent Correct
Quadruples	148	134	90.5
Triples	764	688	90.1
Doubles	1965	1625	82.7
Singles	216	155	71.8
Defaults	4	4	100.0
Totals	3097	2606	84.1

[3] At stages 1 and 2 backing off was also continued if $\hat{p}(1|v, n1, p, n2) = 0.5$. ie. the counts were 'neutral' with respect to attachment at this stage.

The End

The End?

Is that the best we can do?

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

PP-attachment revisited

We calculated:

$P(V \mid v = \text{saw},\ n1 = \text{mouse},\ p = \text{with},\ n2 = \text{telescope})$

Problems:

> Was not trivial to produce a formula.

> Hard to add more sources of information.

New solution:

> Encode as a binary or multiclass classification.

> Decide on the *features*.

> Apply a learning algorithm.

PP –attachment as a multiclass classification

Previously, it was defined as a binary classification problem:

Given $X = (v, n1, p, n2)$

Find a $y \in \{V, N\}$

Let's reframe it as a multiclass problem:

$y \in \{V, N, Other\}$

Our Features:

Single items

- Identity of v
- Identity of p
- Identity of n1
- Identity of n2

Pairs:

- Identity of (v, p)
- Identity of (n1, p)
- Identity of (p, n2)

Triplets:

- Identity of (v, n1, p)
- Identity of (v, p, n2)
- Identity of (n1, p, n2)

Quadruple:

- Identity of (v, n1, p, n2)

Additional Features

Corpus Level:

> Have we seen the (v, p) pair in a 5-word window in a big corpus?
> Have we seen the (n1, p) pair in a 5-word window in a big corpus?
> Have we seen the (n1, p, n2) triplet in a 5-word window in a big corpus?
    > Also: we can use counts, or binned counts.

Distance:

> Distance (in words) between v and p
> Distance (in words) between n1 and p

Exercise #4

> Can you correctly classify the ambiguity?