

# Empirical Approaches to Multilingual Lexical Acquisition

Lecturer: Timothy Baldwin



THE UNIVERSITY OF  
MELBOURNE

# Lecture 3

## Data Discovery: Language Identification

# What is Language Identification?

- Gold (1967): Given a document and a list of possible languages, in what language was the document written? (e.g. English, German, Japanese, Uyghur, ...)
- Orthography?
- A solved problem? (Muthusamy and Spitz 1996)

## An Example

What is the language of the following document:

*Tabel periodik unsur-unsur kimia adalah tampilan unsur-unsur kimia dalam bentuk tabel. Unsur-unsur tersebut diatur berdasarkan struktur elektronnya sehingga sifat kimia unsur-unsur tersebut berubah-ubah secara teratur sepanjang tabel. Setiap unsur didaftarkan berdasarkan nomor atom dan lambang unsurnya.*

## An Example

What is the language of the following document:

*Tabel periodik unsur-unsur kimia adalah tampilan unsur-unsur kimia dalam bentuk tabel. Unsur-unsur tersebut diatur berdasarkan struktur elektronnya sehingga sifat kimia unsur-unsur tersebut berubah-ubah secara teratur sepanjang tabel. Setiap unsur didaftarkan berdasarkan nomor atom dan lambang unsurnya.*

Indonesian

## Another Example

What is the language of the following document:

*Kiingereza ni lugha ya Kigermanik cha Magharibi iliyokua nchini  
Uingereza.*

## Another Example

What is the language of the following document:

*Kiingereza ni lugha ya Kigermanik cha Magharibi iliyokua nchini  
Uingereza.*

Swahili

## Yet Another Example

What is the language of the following document:

*úterý*

## Yet Another Example

What is the language of the following document:

*úterý*

Czech

## A Harder Example

What is the language of the following document:

```
11100011100000011010011011100011100000011001100111100  
0111000000110101000
```

## A Harder Example

What is the language of the following document:

```
11100011100000011010011011100011100000011001100111100  
0111000000110101000
```

A clue: ???

## A Harder Example

What is the language of the following document:

```
11100011100000011010011011100011100000011001100111100  
0111000000110101000
```

A clue: ???

Japanese UTF-8 (te-su-to)

# Why Language Identification?

- Language identification provides us with the means to automatically “discover” web data to convert into a corpus to perform lexical acquisition over
- Also research on:
  - ★ mining interlinear text (e.g. ODIN)
  - ★ cleaning web text (e.g. CLEAN EVAL)

# Basic Approaches

- Linguistically-grounded methods
- Similarity-based categorisation and classification
- Feature-based and kernel-based methods

# But, don't Websites Declare the Language and Encoding in Metadata/headers?

- These are frequently:
  - ★ not there
  - ★ wrong (e.g. S-JIS, EUC-JP)
- Remember: users are competent “scrollers”, but “above the fold” real estate still a premium

# Linguistically-grounded Methods

# Early Attempts: Diacritics

- Intuition: a language has a certain set of “special characters”
- Example: French vs. English:
  - ★ Once we see one of *à, é, ô*... we know the document is in French
  - ★ ... unless we're talking about a *résumé*, or a *prêt-à-porter* fashion show, or...
- Choose a set of “special characters” for each language, and search the document for them

- Advantages:
  - ★ cheap analysis: characters appear, or not
- Disadvantages:
  - ★ overlapping diacritic sets
  - ★ short documents may not contain diacritics
  - ★ only sensible for European languages
  - ★ assumes we know the document encoding

# Early Attempts: Discriminating Character

## *N*-grams

- Intuition: certain languages have certain strings which only/frequently occur in that language
  - ★ English: “ery ”
  - ★ French: “eux ”
  - ★ Italian: “cchi”
  - ★ Serbo-Croat: “lj”
- Notably, *zucchini*, *killjoy*...

- Advantages:
  - ★ cheap analysis: sequence appears, or not
- Disadvantages:
  - ★ sequences may occur in multiple languages
  - ★ short documents may not contain given sequence
  - ★ only sensible for European languages (?)

# Early Attempts: Stop Word Lists

- Intuition: common words in one language do not occur in another language
- Johnson (1993)
  - ★ List stop words, e.g.
    - \* English: *the, a, of, in, by, for...*
    - \* French: *le, la, les, de, un, une, à, en...*
    - \* German: *ein, das, der, die, in, im...*
  - ★ Document has stop words from one language
- Requires (commonly available?) stop word lists

- Advantages:
  - ★ cheap analysis: words in document  $\times$  words in list
  - ★ more generous than simple discrimination
  
- Disadvantages:
  - ★ overlap of stop word sets
  - ★ short documents may not contain stop words
  - ★ only sensible for European languages (?)

# Similarity-based Categorisation and Classification

# Modelling Document Similarity: Cosine Similarity

- Given two documents  $x$  and  $y$ , and their corresponding feature vectors  $\vec{x}$  and  $\vec{y}$ , respectively, we can calculate their similarity via their **vector cosine**:

$$\text{sim}(x, y) = \frac{\vec{x} \cdot \vec{y}}{|\vec{x}| |\vec{y}|} = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2} \sqrt{\sum_i y_i^2}}$$

## Cosine Similarity Example

- Calculate the cosine similarity of the following documents:

A = aardvark back abandon  
abandon abandon

B = aardvark abandonment  
back back back

$$\vec{A} = \langle 1, 3, 0, 1 \rangle$$

$$\equiv \left\langle \frac{1}{\sqrt{11}}, \frac{3}{\sqrt{11}}, 0, \frac{1}{\sqrt{11}} \right\rangle$$

$$\vec{B} = \langle 1, 0, 1, 3 \rangle$$

$$\equiv \left\langle \frac{1}{\sqrt{11}}, 0, \frac{1}{\sqrt{11}}, \frac{3}{\sqrt{11}} \right\rangle$$

$$\vec{A} \cdot \vec{B} = \frac{\frac{1}{\sqrt{11}} \times \frac{1}{\sqrt{11}} + \frac{3}{\sqrt{11}} \times 0 + 0 \times \frac{1}{\sqrt{11}} + \frac{1}{\sqrt{11}} \times \frac{3}{\sqrt{11}}}{1 \times 1} = \frac{4}{11}$$

# Modelling Document Distance: Relative Entropy

- Given two documents  $x$  and  $y$ , and their corresponding feature **unit-length** vectors  $\vec{x}$  and  $\vec{y}$ , respectively, we can interpret the feature vector as a probability distribution and calculate the **relative entropy** (or KL divergence):

$$D(x \parallel y) = \sum_i x_i (\log_2 x_i - \log_2 y_i)$$

or alternatively **skew divergence**:

$$s_\alpha(x, y) = D(x \parallel \alpha y + (1 - \alpha)x)$$

- This causes considerable grief for our MLE-based probabilities: why?
- A simplistic way of getting around this is via **Laplacian smoothing**:

$$\hat{P}(c_j) = \frac{\text{freq}(c_j) + 1}{k + \sum_k \text{freq}(c_k)}$$
$$\hat{P}(x_i|c_j) = \frac{\text{freq}(x_i, c_j) + 1}{\text{freq}(c_j) + l}$$

## Relative Entropy Example

- Calculate the relative entropy and skew divergence of the following documents:

aardvark back abandon  
abandon abandon

aardvark abandonment  
back back back

$$\mathbf{A} = \left\langle \frac{2}{9}, \frac{4}{9}, \frac{1}{9}, \frac{2}{9} \right\rangle$$

$$\mathbf{B} = \left\langle \frac{2}{9}, \frac{1}{9}, \frac{2}{9}, \frac{4}{9} \right\rangle$$

$$D(\mathbf{A}||\mathbf{B}) = \sum_i a_i (\log_2 a_i - \log_2 b_i)$$

$$\begin{aligned} &= \frac{2}{9}(\log \frac{2}{9} - \log \frac{2}{9}) + \frac{4}{9}(\log \frac{4}{9} - \log \frac{1}{9}) + \\ &\quad \frac{1}{9}(\log \frac{1}{9} - \log \frac{2}{9}) + \frac{2}{9}(\log \frac{2}{9} - \log \frac{4}{9}) \\ &\approx 0.56 \end{aligned}$$

## Skew Divergence Example

- Calculate the relative entropy and skew divergence of the following documents:

aardvark	back	abandon
abandon	abandon	

aardvark	abandonment
back	back back

$$\mathbf{A} = \langle 0.2, 0.6, 0.0, 0.2 \rangle$$

$$\mathbf{B} = \langle 0.2, 0.0, 0.2, 0.6 \rangle$$

$$\begin{aligned} s_{0.99}(\mathbf{A}, \mathbf{B}) &= D(\mathbf{A} \parallel 0.99\mathbf{B} + 0.01\mathbf{A}) \\ &= \sum_i a_i (\log a_i - \log(0.99b_i + 0.01a_i)) \end{aligned}$$

$$\begin{aligned} &= 0.2(\log 0.2 - \log(0.99 \times 0.2 + 0.01 \times 0.2)) + \\ &\quad 0.6(\log 0.6 - \log(0.99 \times 0.0 + 0.01 \times 0.6)) + \\ &\quad 0.0(\log 0.0 - \log(0.99 \times 0.2 + 0.01 \times 0.0)) + \\ &\quad 0.2(\log 0.2 - \log(0.99 \times 0.6 + 0.01 \times 0.2)) \\ &\approx 3.67 \end{aligned}$$

# Nearest Neighbour Classifiers

- There are various ways to combine these document–document scores to form an overall categorisation function, e.g.:
- **Method 1:** index all training documents, and query the training document set with each test document; classify the test document according to the class of the top-ranked training document [**1-NN**]
- **Method 2:** index all training documents, and query the training document set with each test document; classify the test document according to the **majority class** within the  $k$  top-ranked training documents [**k-NN**]

- **Method 3:** index all training documents, and query the training document set with each test document; classify the test document according to the class with the best accumulative score [**weighted k-NN**]
- **Method 4:** index all training documents, and query the training document set with each test document; classify the test document according to the class with the best accumulative score based on scores, factoring in an offset to indicate the prior expectation of a test document being classified as being a member of that class [**offset weighted k-NN**]

- Overall advantages of the nearest neighbour approach:
  - ★ simple
  
- Overall disadvantages of the nearest neighbour approach:
  - ★ expensive (in terms of index accesses)
  - ★ everything is done at run time (**lazy learner**)
  - ★ prone to bias
  - ★ arbitrary  $k$  value

# Feature-based and Kernel-based Methods

# Bayesian Methods

- Learning and classification methods based on probability theory
- Build a **generative model** that approximates how data is produced
- Categorisation produces a posterior probability distribution over the possible categories given a description of an instance

# Bayes' Rule

$$P(C, X) = P(C|X)P(X) = P(X|C)P(C)$$

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}$$

# Naive Bayes (NB) Classifiers

- Task: classify an instance  $D = \langle x_1, x_2, \dots, x_n \rangle$  according to one of the classes  $c_j \in C$

$$\begin{aligned}c &= \arg \max_{c_j \in C} P(c_j | x_1, x_2, \dots, x_n) \\ &= \arg \max_{c_j \in C} \frac{P(x_1, x_2, \dots, x_n | c_j) P(c_j)}{P(x_1, x_2, \dots, x_n)} \\ &= \arg \max_{c_j \in C} P(x_1, x_2, \dots, x_n | c_j) P(c_j)\end{aligned}$$

# Simplifying Assumption

- $P(c_j)$ 
  - ★ can be estimated from the frequency of classes in the training examples [**maximum likelihood estimate**]
- $P(x_1, x_2, \dots, x_n | c_j)$ 
  - ★  $O(|X|^n |C|)$  parameters (cannot be estimated in practice)
- Naive Bayes Conditional Independence Assumption:
  - ★ assume that the probability of observing the conjunction of attributes is equal to the product of the individual probabilities  $P(x_i | c_j)$  [**hence “naive”**]

# The Final NB Formulation

- Applying the conditional independence assumption:

$$\begin{aligned}c &= \arg \max_{c_j \in C} P(x_1, x_2, \dots, x_n | c_j) P(c_j) \\ &= \arg \max_{c_j \in C} P(c_j) \prod_i P(x_i | c_j)\end{aligned}$$

# Estimating the Probabilities (1)

- The most obvious way of generating the probabilities is via **maximum likelihood** estimation, using the frequency counts in the training data:

$$\hat{P}(c_j) = \frac{\text{freq}(c_j)}{\sum_k \text{freq}(c_k)}$$
$$\hat{P}(x_i|c_j) = \frac{\text{freq}(x_i, c_j)}{\text{freq}(c_j)}$$

- This is a very bad idea: why?

## Estimating the Probabilities (2)

- As before (c.f. relative entropy), a simplistic way of getting around this is via **Laplacian smoothing**:

$$\hat{P}(c_j) = \frac{\text{freq}(c_j) + 1}{\sum_k \text{freq}(c_k) + k}$$
$$\hat{P}(x_i|c_j) = \frac{\text{freq}(x_i, c_j) + 1}{\text{freq}(c_j) + l}$$

# Naive Bayes in Action

Outlook	Temperature	Humidity	Windy	Play
TRAINING DATA				
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
TEST DATA				
sunny	mild	normal	TRUE	(yes)

- Priors:

$$P(\text{yes}) = \frac{7}{12}$$

$$P(\text{no}) = \frac{5}{12}$$

- Conditional probs:

$$P(\text{Outlook} = \text{sunny}|\text{yes}) = \frac{2}{9}$$

$$P(\text{Outlook} = \text{o'cast}|\text{yes}) = \frac{3}{9}$$

$$P(\text{Outlook} = \text{rainy}|\text{yes}) = \frac{4}{9}$$

$$P(\text{Outlook} = \text{sunny}|\text{no}) = \frac{4}{7}$$

$$P(\text{Outlook} = \text{o'cast}|\text{no}) = \frac{1}{7}$$

$$P(\text{Outlook} = \text{rainy}|\text{no}) = \frac{2}{7}$$

- Classification of test instance  $T$ :

$$P(\text{yes}|T) = \frac{7}{12} \times \left( \frac{2}{9} \times \frac{3}{9} \times \frac{5}{8} \times \frac{2}{8} \right) \approx 0.0068$$

$$P(\text{no}|T) = \frac{5}{12} \times \left( \frac{4}{7} \times \frac{2}{7} \times \frac{2}{6} \times \frac{3}{6} \right) \approx 0.011$$

# Multivariate Binomial NB

- Represent each word as a binary feature (= DF model)
- Represent a document according to the word *types* it contains
- No indication of how often a given word occurs in a given document
- “Bag of word types” document model

# Multivariate Binomial NB: Mechanics

$$P(D|c_i) = \prod_{j=1}^{|\mathcal{V}|} (B_j P(t_j|c_i) + (1 - B_j)(1 - P(t_j|c_i)))$$

where  $B_j \in \{0, 1\}$  indicates the presence or absence of the  $j$ th term in  $D$ ,  $\mathcal{V}$  is the set of all terms, and

$$P(t|c_i) = \frac{1 + \sum_{k=1}^{|\mathcal{D}|} B_k P(c_i|D_k)}{2 + \sum_{k=1}^{|\mathcal{D}|} P(c_i|D_k)}$$

# Multivariate Binomial NB: Example

- Test document:

we few, we happy few, we band of brothers

- Test document representation:

$\langle 0, 0, \dots, 1, \dots 0, \dots 1, \dots, 1, \dots, 1, \dots 0, \dots, 1, \dots, 0 \rangle$   
 aardvark aback band betwixt brothers few happy thee we zymogen

- Shakespeare training document set:

then happy i, that love and am beloved

$\langle 0, 0, \dots, 0, \dots 0, \dots 0, \dots, 0, \dots, 1, \dots, 0, \dots, 0, \dots, 0 \rangle$

aardvark aback band betwixt brothers few happy thee we zymogen

if we shadows have offended

$\langle 0, 0, \dots, 0, \dots 0, \dots 0, \dots, 0, \dots, 0, \dots, 0, \dots, 1, \dots, 0 \rangle$

aardvark aback band betwixt brothers few happy thee we zymogen

- Beatles training document set:

we can work it out

$\langle 0, 0, \dots, 0, \dots 0, \dots 0, \dots, 0, \dots, 0, \dots, 0, \dots, 1, \dots, 0 \rangle$

aardvark aback band betwixt brothers few happy thee we zymogen

sgt pepper's lonely hearts club band

$\langle 0, 0, \dots, 1, \dots 0, \dots 0, \dots, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 0 \rangle$

aardvark aback band betwixt brothers few happy thee we zymogen

$$\bullet P(\text{we}|\text{Shakespeare}) = \frac{1+(0 \times 1 + 1 \times 1 + 1 \times 0 + 0 \times 0)}{2+(1+1+0+0)} = \frac{1}{2}$$

$$P(\text{we}|\text{Beatles}) = \frac{1+(0 \times 0 + 1 \times 0 + 1 \times 1 + 0 \times 1)}{2+(0+0+1+1)} = \frac{1}{2}$$

$$P(\text{band}|\text{Shakespeare}) = \frac{1+(0 \times 1 + 0 \times 1 + 0 \times 0 + 1 \times 0)}{2+(1+1+0+0)} = \frac{1}{4}$$

$$P(\text{band}|\text{Beatles}) = \frac{1+(0 \times 0 + 0 \times 0 + 0 \times 1 + 1 \times 1)}{2+(0+0+1+1)} = \frac{1}{2}$$

$$P(\text{happy}|\text{Shakespeare}) = \frac{1+(1 \times 1 + 0 \times 1 + 0 \times 0 + 0 \times 0)}{2+(1+1+0+0)} = \frac{1}{2}$$

$$P(\text{happy}|\text{Beatles}) = \frac{1+(1 \times 0 + 0 \times 0 + 0 \times 0 + 0 \times 0)}{2+(0+0+1+1)} = \frac{1}{4}$$

$$\bullet P(D|\text{Shakespeare}) = ((0 \times \frac{1}{4} + (1 - 0) \times \frac{3}{4}) \times (0 \times \frac{1}{4} + (1 - 0) \times \frac{3}{4}) \times \dots \times (1 \times \frac{1}{4} + (1 - 1) \times \frac{3}{4}) \times \dots \times (0 \times \frac{1}{4} + (1 - 0) \times \frac{3}{4}) \times \dots \times (1 \times \frac{1}{4} + (1 - 1) \times \frac{3}{4}) \times \dots \times (1 \times \frac{1}{4} + (1 - 1) \times \frac{3}{4}) \times \dots \times (1 \times \frac{1}{2} + (1 - 1) \times \frac{1}{2}) \times \dots \times (0 \times \frac{1}{4} + (1 - 0) \times \frac{3}{4}) \times \dots \times (1 \times \frac{1}{2} + (1 - 1) \times \frac{1}{2}) \times \dots \times (0 \times \frac{1}{4} + (1 - 0) \times \frac{3}{4}))$$

# Multinomial NB

- Represent each word as an integer
- Represent a document according to the word *tokens* it contains
- Optionally include a term for  $P(L = l_D | c_i)$  (to normalise for document length)
- “Bag of word tokens” document model
- Assumes that (a) the position of a word in the document and (b) the context of a word are irrelevant in classification

## Multinomial NB: Mechanics

$$P(D|c_i) = \prod_{j=1}^{|\mathcal{V}|} \frac{P(t_j|c_i)^{N_{D,t_j}}}{N_{D,t_j}!}$$

where  $N_{D,t_j}$  is the frequency of the  $j$ th term in  $D$ ,  $\mathcal{V}$  is the set of all terms, and:

$$P(t|c_i) = \frac{1 + \sum_{k=1}^{|\mathcal{D}|} N_{k,t} P(c_i|D_k)}{|\mathcal{V}| + \sum_{j=1}^{|\mathcal{V}|} \sum_{k=1}^{|\mathcal{D}|} N_{k,t_j} P(c_i|D_k)}$$

# Multinomial NB: Example

- Test document:

we few, we happy few, we band of brothers

- Test document representation:

$\langle 0, 0, \dots, 1, \dots 0, \dots 1, \dots, 2, \dots, 1, \dots 0, \dots, 3, \dots, 0 \rangle$   
 aardvark aback band betwixt brothers few happy thee we zymogen

- Assume  $|\mathcal{V}| = 100$

- Shakespeare training document set:

then happy i, that love and am beloved

$\langle 0, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 1, \dots, 0, \dots, 0, \dots, 0 \rangle$

aardvark aback band betwixt brothers few happy thee we zymogen

if we shadows have offended

$\langle 0, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 1, \dots, 0 \rangle$

aardvark aback band betwixt brothers few happy thee we zymogen

- Beatles training document set:

we can work it out

$\langle 0, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 1, \dots, 0 \rangle$

aardvark aback band betwixt brothers few happy thee we zymogen

sgt pepper's lonely hearts club band

$\langle 0, 0, \dots, 1, \dots, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 0, \dots, 0 \rangle$

aardvark aback band betwixt brothers few happy thee we zymogen

$$\bullet P(\text{we}|\text{Shakespeare}) = \frac{1+(0 \times 1 + 1 \times 1 + 1 \times 0 + 0 \times 0)}{100+(8+5)} = \frac{2}{113}$$

$$P(\text{we}|\text{Beatles}) = \frac{1+(0 \times 0 + 1 \times 0 + 1 \times 1 + 0 \times 1)}{100+(5+6)} = \frac{2}{111}$$

$$P(\text{band}|\text{Shakespeare}) = \frac{1+(0 \times 1 + 0 \times 1 + 0 \times 0 + 1 \times 0)}{100+(8+5)} = \frac{1}{113}$$

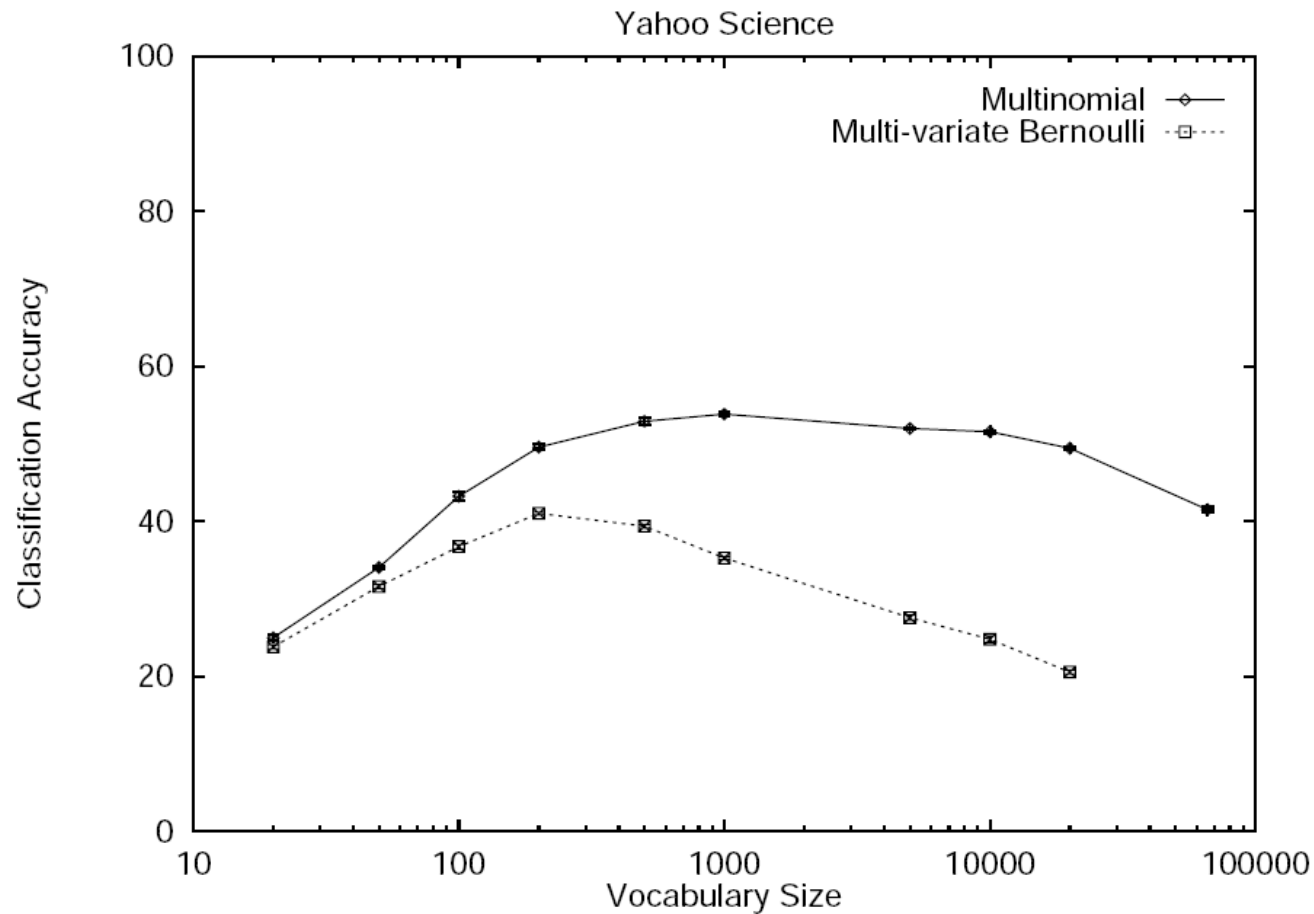
$$P(\text{band}|\text{Beatles}) = \frac{1+(0 \times 0 + 0 \times 0 + 0 \times 1 + 1 \times 1)}{100+(5+6)} = \frac{2}{111}$$

$$P(\text{happy}|\text{Shakespeare}) = \frac{1+(1 \times 1 + 0 \times 1 + 0 \times 0 + 0 \times 0)}{100+(8+5)} = \frac{2}{113}$$

$$P(\text{happy}|\text{Beatles}) = \frac{1+(1 \times 0 + 0 \times 0 + 0 \times 0 + 0 \times 0)}{100+(5+6)} = \frac{1}{111}$$

$$\bullet P(D|\text{Shakespeare}) = \frac{1^0}{113^{0!}} \times \frac{1^0}{113^{0!}} \times \dots \times \frac{1^1}{113^{1!}} \times \dots \times \frac{1^0}{113^{0!}} \times \dots \times \frac{1^1}{113^{1!}} \times \dots \times \frac{1^2}{113^{2!}} \times \dots \times \frac{2^1}{113^{1!}} \times \dots \times \frac{1^0}{113^{0!}} \times \dots \times \frac{1^3}{113^{3!}} \times \dots \times \frac{1^0}{113^{0!}}$$

# Results over the Yahoo Science Dataset



# Theoretical Properties of NB Models

- **Multiclass** classification method
- **Parametric**
  - only have to store attribute–value counts/probabilities for each class, not the actual instances
- **Incremental**
  - easy to add extra data to the classifier on the fly (implications for weakly supervised learning)
- Handles both **nominal** and **continuous** features
- Simple (→ fast)

# Practical Properties of NB Models

- Strong performer
- Highly robust over isolated irrelevant features (cf. decision trees)
- Very good at balancing up lots of “marginally relevant” features
- Unable to capture correlated attributes
- Actual posterior probability estimates tend to be awry, but as a classification task, we are only interested in the relative values
- Nice handling of missing values (simply ignore them!)

# Extra Features in Text Categorisation

- There's lots more to **web** text categorisation than words:
  - ★ metadata
  - ★ domain of source page
  - ★ page structure
  - ★ link structure
  - ★ diachronic stability of page
  - ★ balance of different content types
  - ★ relative use of different HTML attributes
  - ★ well-formedness of HTML
  - ⋮

# OPEN ISSUES AND SUMMARY

# Open Issues

- How well do existing techniques support language identification for languages which form the bulk of the more than 7000 languages identified in the Ethnologue?

- Can we treat LangID as an open-class classification problem?

$$\arg \max_{c \in C} lm(c, D) \text{ vs. } \arg \max_{c \in C \cup C'} lm(c, D)$$

- What is the performance of the variety of LangID systems in environments where the amount of gold standard data for training is small (e.g. 50/100/250 words or 50/100/250 characters)?

- Can we move away from a one-to-one view of LangID to a one-to-many view?
  - ★ finer granularity (e.g. sentence, paragraph, section)
  - ★ in quantitative terms (e.g. a document is 95% English, 3% French and 2% Italian)
  
- Can we move away from IR-style evaluation criteria to produce something more representative of reality?
  - ★ gradated judgements for source language
  - ★ gradated judgements for resource type
  - ★ possibly micro-level markup of the location of different languages in the document

# Summary

- What is language identification?
- Why is language identification important?
- What issues arise in language identification?
- What methods are used?

# References

- CHAKRABARTI, SOUMEN. 2003. *Mining the Web: Discovering Knowledge from Hypertext Data*. San Francisco, USA: Morgan Kaufmann.
- GOLD, E.M. 1967. Language identification in the limit. *Information and Control* 5.447–474.
- JACKSON, PETER, and ISABELLE MOULINIER. 2002. *Natural Language Processing for Online Applications: Text Retrieval, Extraction and Categorization*. Amsterdam, Netherlands: John Benjamins.
- JOHNSON, S. 1993. Solving the problem of language recognition. Technical report, School of Computer Studies, University of Leeds.
- MCCALLUM, ANDREW, and KAMAL NIGAM. 1998. A comparison of event models for Naive Bayes text classification. In *Proc. of the AAAI-98 Workshop on Learning for Text Categorization*, Madison, USA.
- MUTHUSAMY, YESHWANT K., and A. LAWRENCE SPITZ. 1996. Automatic language identification. In *Survey of the State of the Art in Human Language Technology*, ed. by Ronald A. Cole, Joseph Mariani, Hans Uszkoreit, Annie Zaenen, and Victor Zue, 273–85. Cambridge University Press.