COMPUTATIONAL ANALYSIS OF STYLE: BASIC CONCEPTS

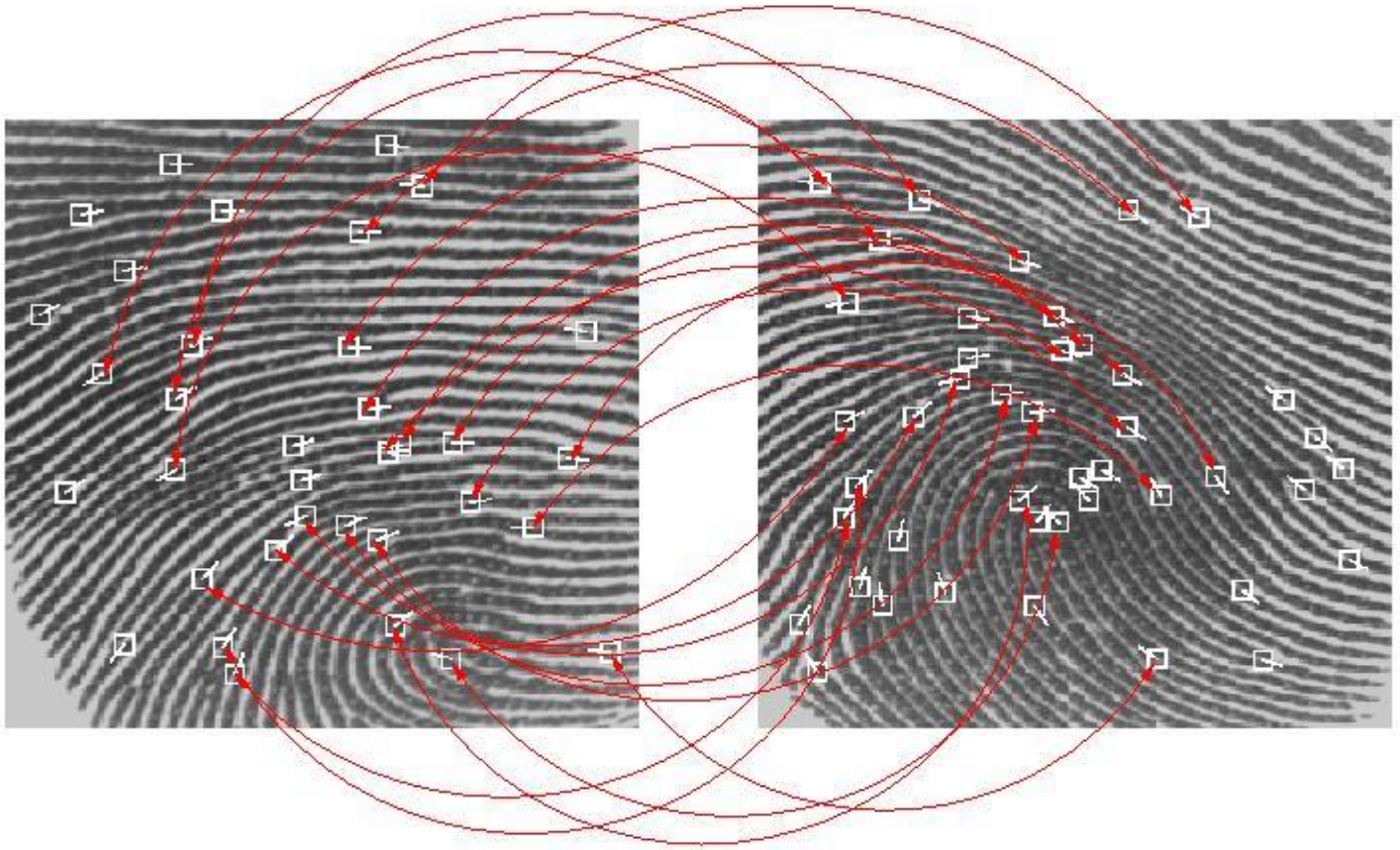
George K. Mikros

National and Kapodistrian University of Athens – University of Massachusetts, Boston

Lecture Overview

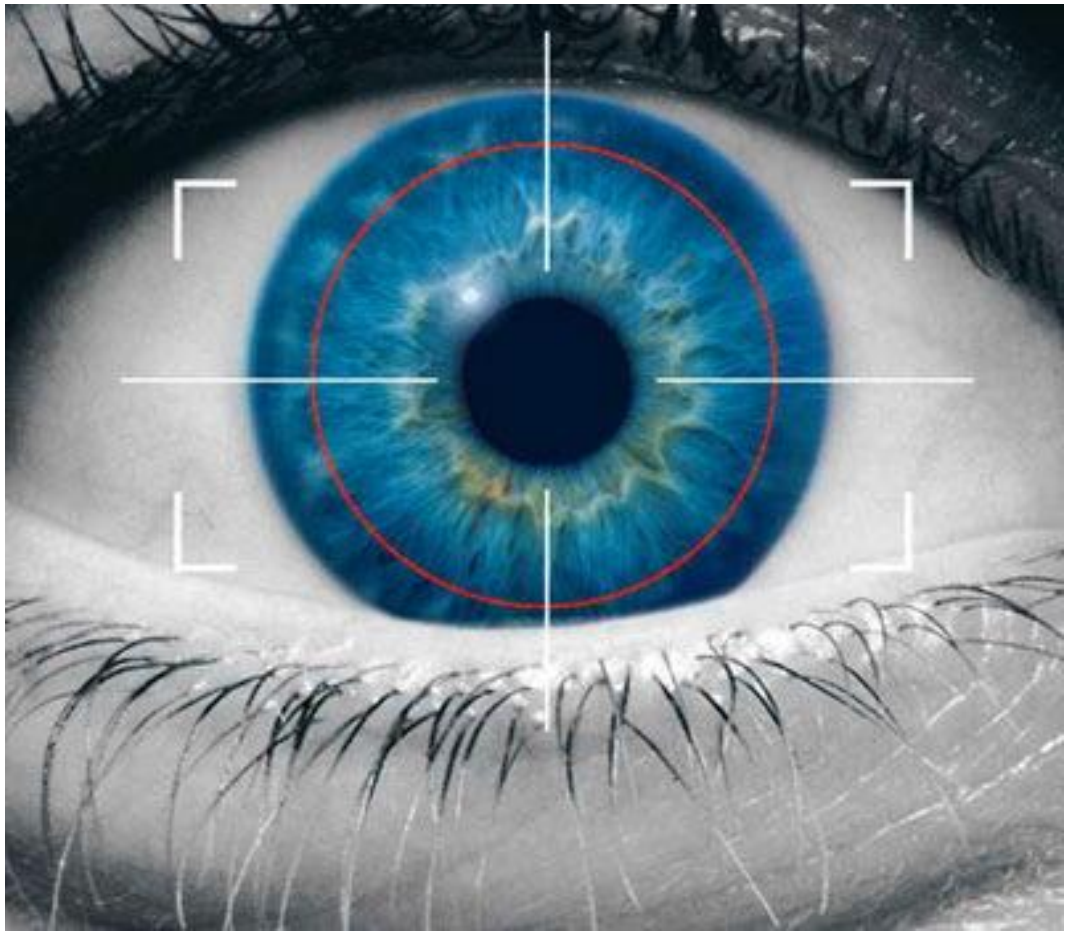
- Biometric Methods
- Stylometry:
  - Principles
  - Assumptions
  - Features
  - Applications
- Short historical
- Successes ☺ and failures ☹
- Principles of validation and reliability
- N-grams
- Machine Learning
  - Supervised learning
  - Classification evaluation
  - Classification measures
  - Classification algorithms
    - Random Forests
    - Support Vector Machines - SVM
- Conclusions

Biometric Methods: Fingerprints



Biometric Methods: DNA

Biometric Methods: Iris



Styometric Methods: ?

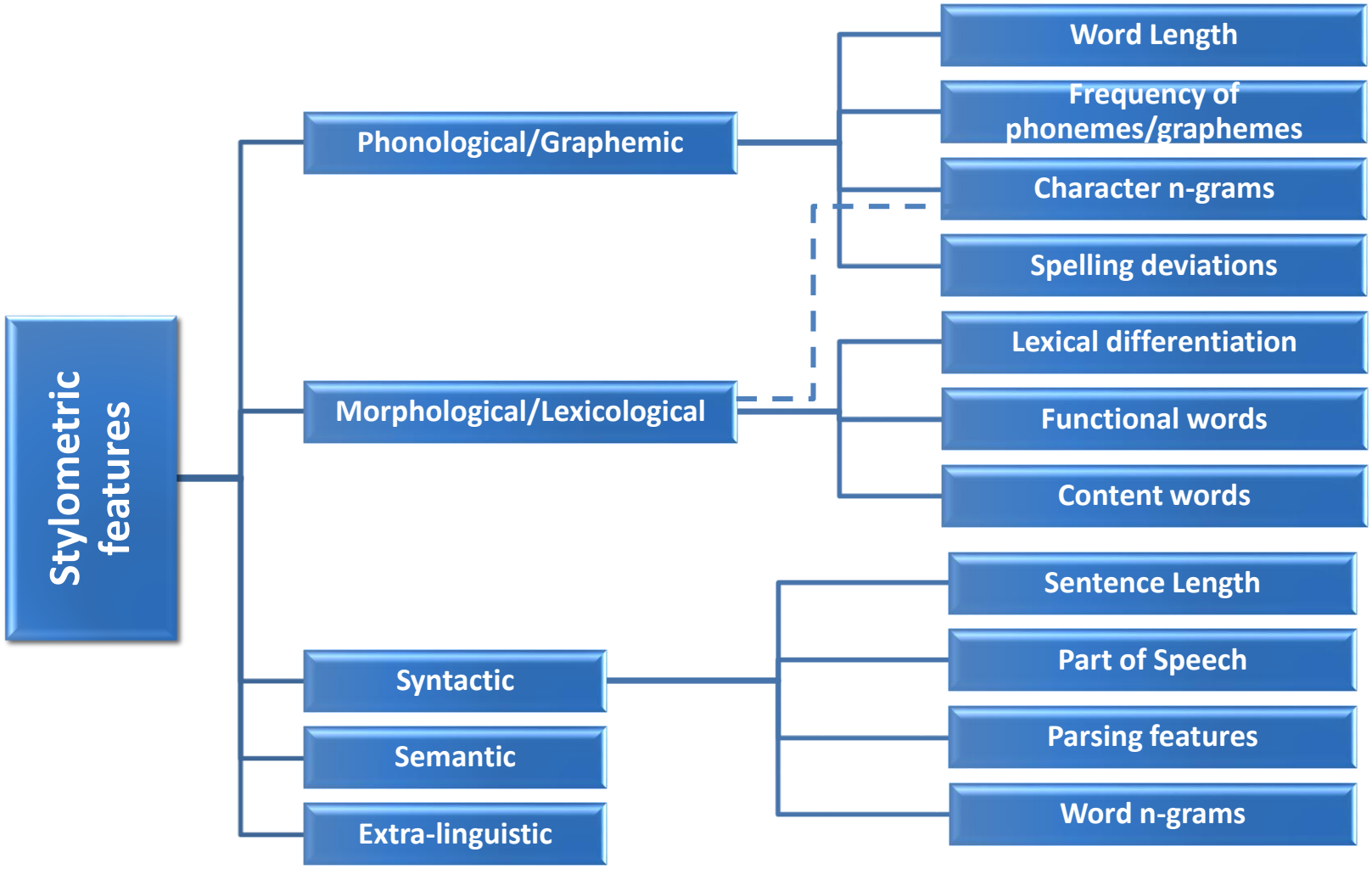
Stylometry: Principles

- Definition: Quantitative description of the textual style
- Textual linguistic features:
  - Conscious author's linguistic choices: words, syntactic patters etc.
  - Unconscious author's linguistic usage: word and sentence length, character frequencies, functional words usage etc.
- In stylometric authorship attribution we care about "**how**" writes an author and not "**what**" he/she writes.

Stylometry: Assumptions

- Each author has an idiosyncratic way to use language. Its language usage is unique. Author and language usage are biuniquely related.
- There are always aspects of language usage in an author that never change (i.e. quantitative stable). These represent the “stylomes” which quantitatively distinguish each author from all the others.
- Each author makes both conscious and unconscious linguistic choices. If style was based only on conscious selections each author could change it radically and we could never associate him with a characteristic style.

Stylometry: Feature typology



Stylometry: Applications

- **Literature**
  - The majority of literature works of disputed authorship has been written before the 20th century. Major international copyright agreements (e.g. Berne convention - 1896) were established at the beginning of the 20th century. Until then it was fairly easy to copy an already published literary work and put your name on it!
  - Famous cases of disputed literary authorship: … Shakespeare …
- **Forensics**
  - USA: Daubert standard for accepting scientific testimonies in the court:
    - Falsifiability
    - Peer review and publication
    - Known error rate
    - General admission in the scientific community
- **Text reuse and Plagiarism**
- **Information Retrieval**
  - Marketing
  - Custom Search Results
- **Social Sites Vandalism** (e.g. Wiki Vandalism)
- **Education (text readability)**

Automatic Authorship Attribution: Interdisciplinary Approach

Short Historical Overview
19th century: Mendenhall (1887)

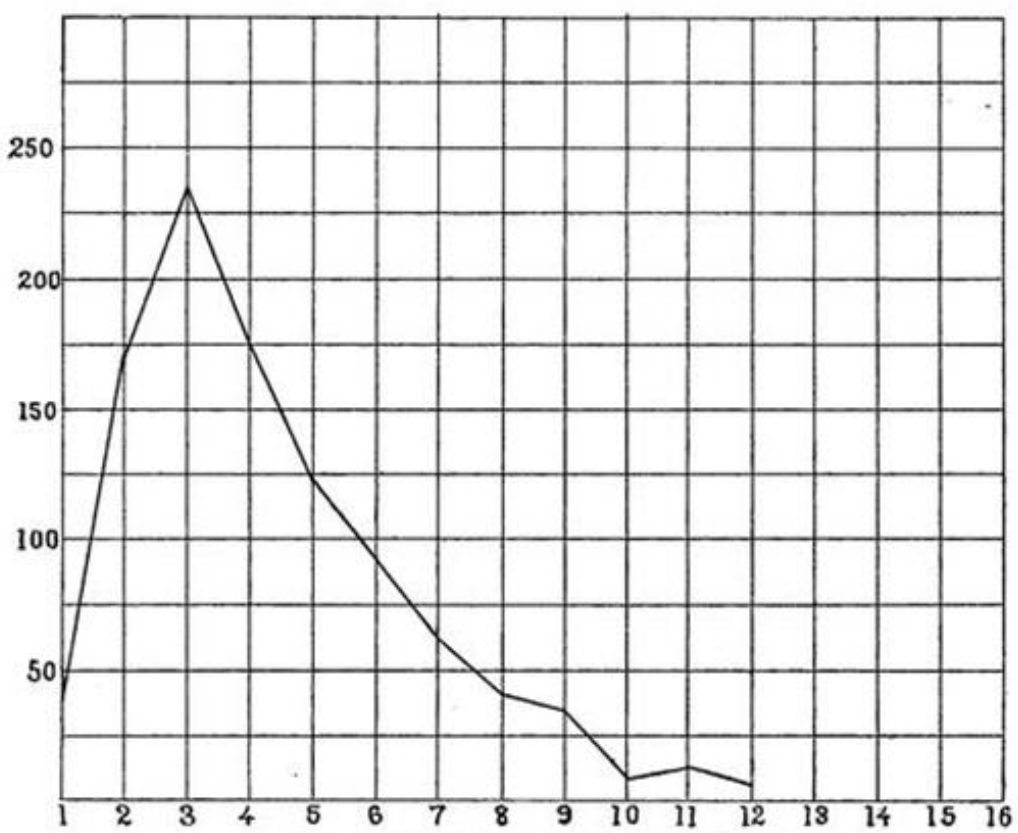
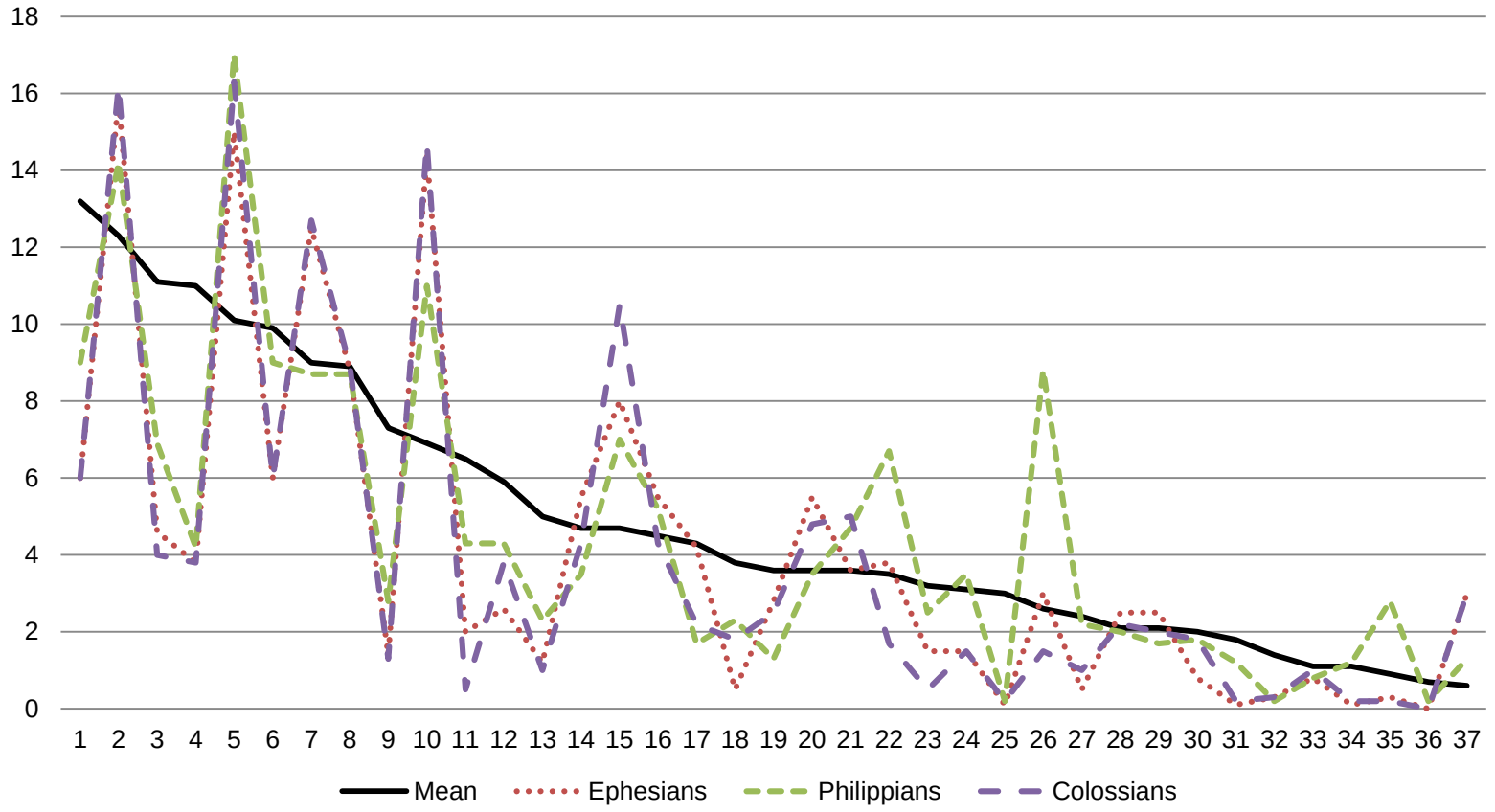


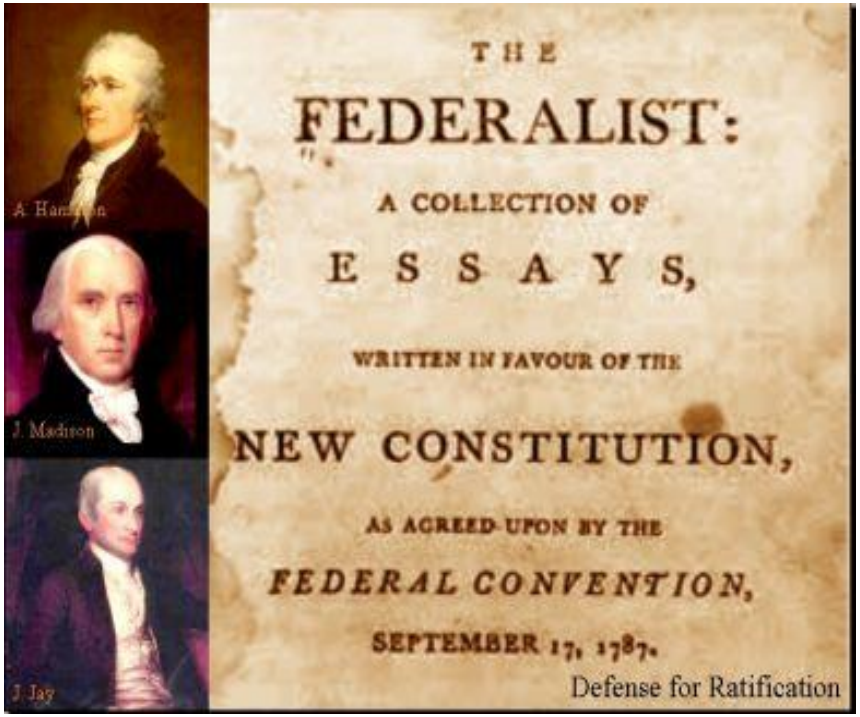
FIG. 1. — FIRST ONE THOUSAND WORDS IN 'OLIVER TWIST.'

Short Historical Overview
19th century: Mascol (1888)



Succeses ☺

- The Federalist Papers
  - 3 authors (A. Hamilton, J. Madison, J. Jay)
  - 85 articles and essays written in 1787-1788, under the pseydonym “Publius”. Their aim was to to influence the vote of New Yorkers in favor of ratifying the USA Constitution.
    - Hamilton= 51
    - Madison= 14
    - Jay= 5
    - Hamilton + Madison= 3
    - ; = 12



Successes ☺
Mosteller & Wallace (1984)

	enough	while	whilst	upon
Hamilton	0.59	0.26	0	2.93
Madison	0	0	0.47	0.16
Disputed texts	0	0	0.34	0.08
Co-authored texts	0.18	0	0.36	0.36

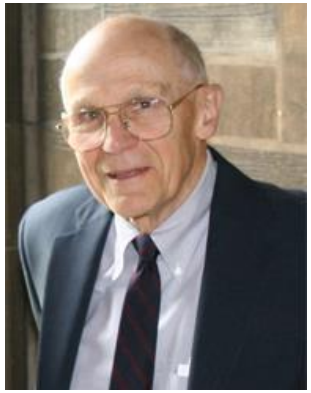
F. Mosteller
Harvard

Frequency of 165 words (mainly functional)

Bayes theorem

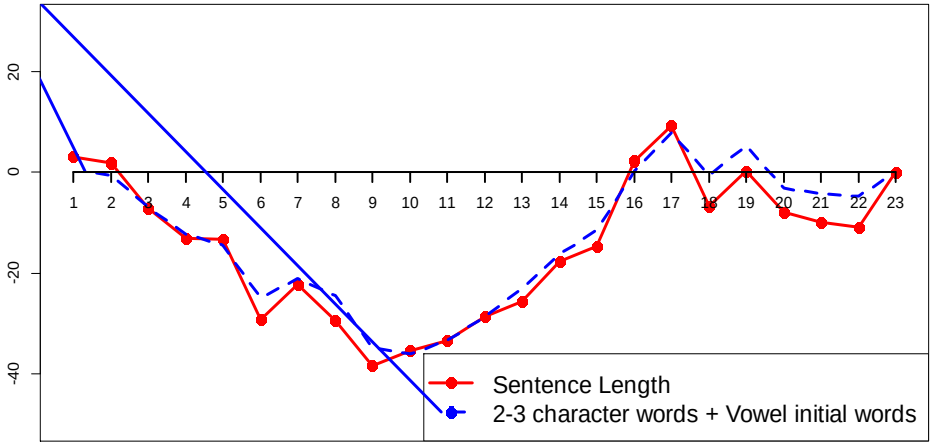
$$P(A|X) = \frac{P(X|A) \cdot P(A)}{P(X|A) \cdot P(A) + P(X|\sim A) \cdot P(\sim A)}$$

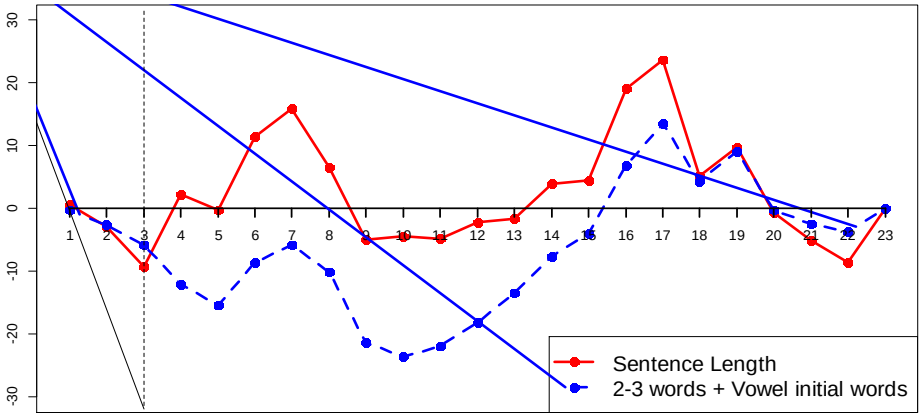
D. L. Wallace
University of Chicago



Failures ☹

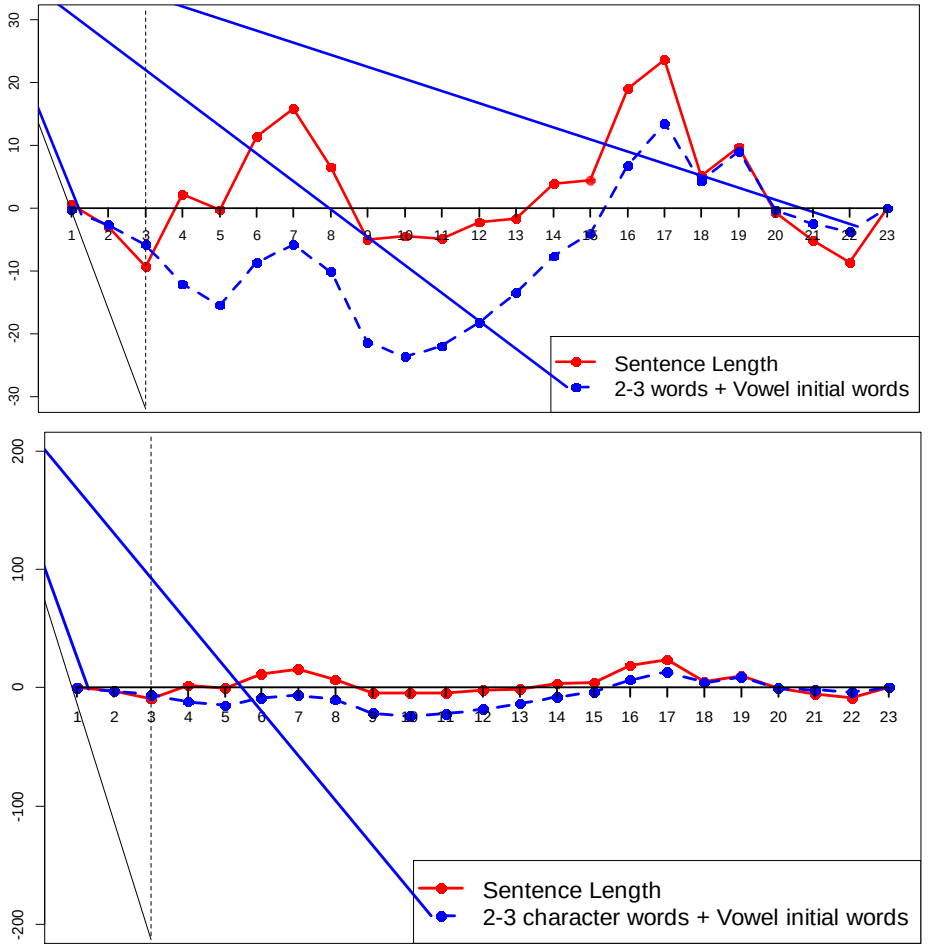
- CUSUM
  - Andrew Morton in the early '60 adapted **Cu**mulative **Sum** – CUSUM or QSUM (a method which originally was used in the industrial quality control) to be used in texts.
    - It measures the deviations of specific stylometric features from their respective mean in a specific text.
    - After some research, Morton selected empirically two stylometric features:
      - Sentence Length (in words)
      - Number of words with 2 or 3 characters + number of vowel initial words.
    - Morton claims that this method works best in texts of 25 – 50 sentences.





Failures ☹

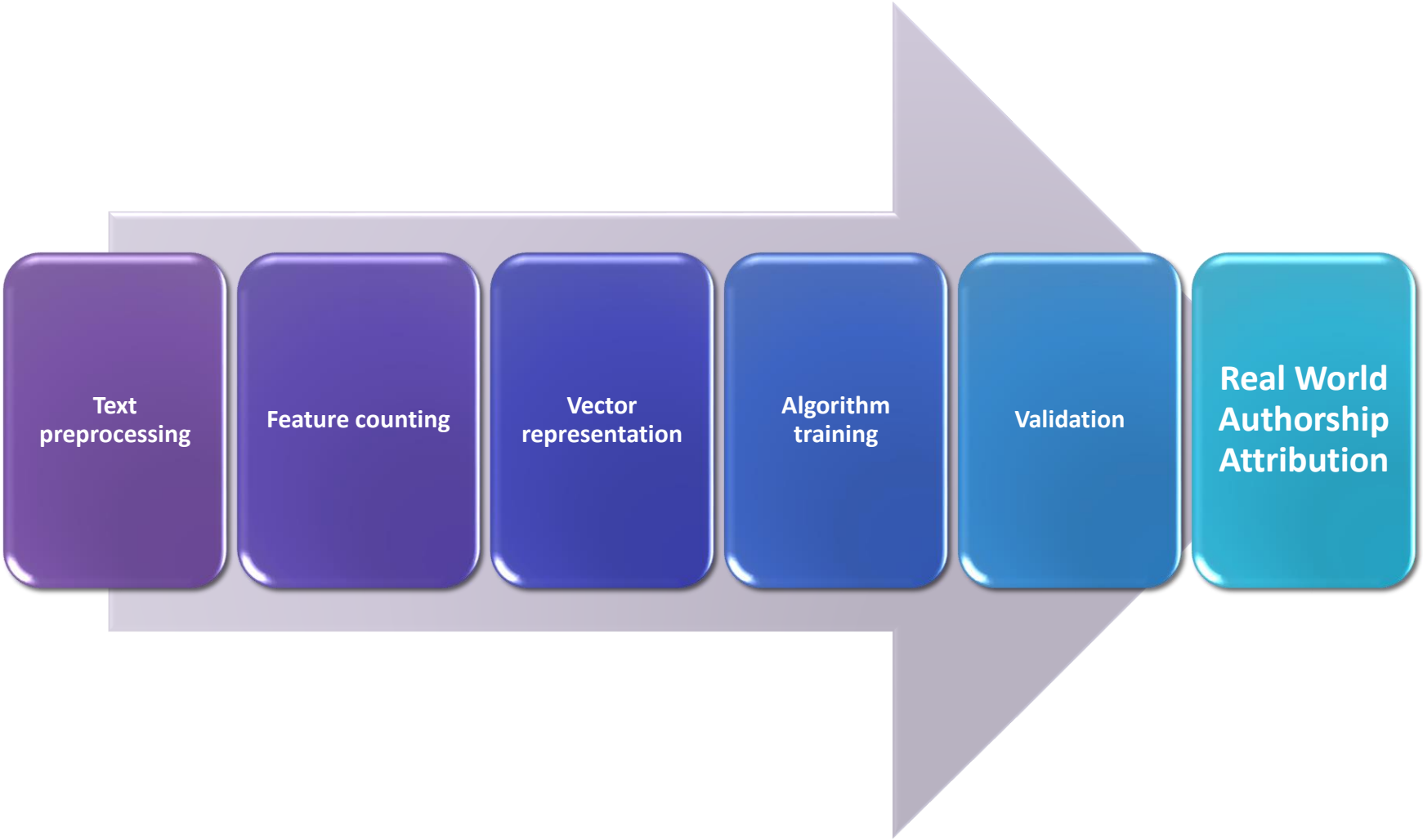
- Lack of adequate theoretical framework.
- Insufficient validity
- Lack of testing statistical significance
- Subjective criteria in diagram scaling.
- BBC live show (1993)
  - Documents of convicted criminals were attributed to … the Secretary of State for Justice!!!



Validity principles in Authorship Attribution (Smith 1990, 249-250)

- The onus of proof lies entirely with the person making the ascription.
- The argument for adding something to an author's canon has to be vastly more stringent than for keeping it there.
- If doubt persists, an anonymous work must remain anonymous.
- Avoidance of a false attribution is far more important than failing to recognize a correct one.
- Only works of known authorship are suitable as a basis for attributing a disputed work
- There are no short-cuts in attribution studies.

Methodology



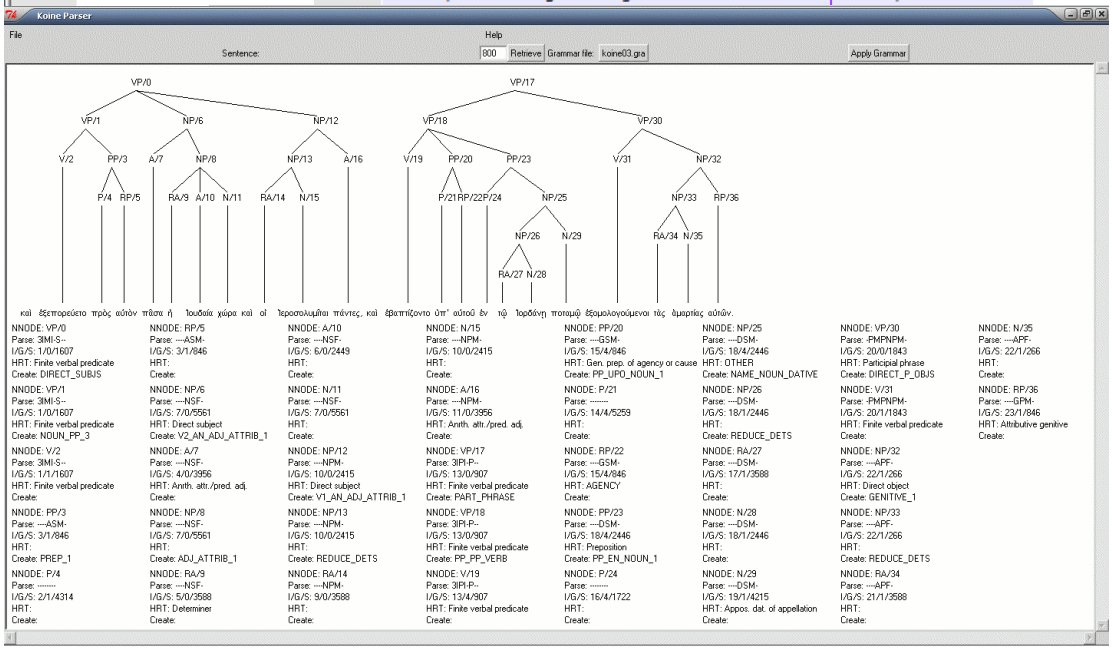
Text Preprocessing

- Tokenization
- Sentence splitting
- Part-of-Speech Tagging
- Parsing

```
1  Δεν
2  το
3  αποκλείω
4  καθόλου
5  όταν
6  θα
7  ξανακατέβω
8  στην
9  πατρίδα
10 μου
11 την
12 Καλαμάτα
13 και
14 κάνω
15 την
16 απαραίτητη
17 βόλτα
18 στην
19 παραλία
```

```
1  <s> Αυτό είναι ένα δοκιμαστικό κείμενο </s>
2  <s> Οι Η.</s>
3  <s> Οι Π.</s>
4  <s> Οι Α.</s>
5  <s> εξέλεξαν πρόσφατα τον πρόεδρο.</s>
```

```
1  Εγκρίνουμε   VbMnIdPr01PlXxIpAvXx   εγκρίνω
2  την          AtDfFeSgAc             ο
3  μετατροπή    NoCmFeSgAc             μετατροπή
4  της          AtDfFeSgGe             ο
5  αμιγούς      AjBaFeSgGe             αμιγούς
6  Δημοτικής    NoCmNeSgGe             Δημοτικής
7  Επιχείρησης  NoCmFeSgGe             επιχείρηση
8  με           AsPpSpNvNvNv           με
9  την          AtDfFeSgAc             ο
10 επωνυμία     AjBaFeSgAc             επωνυμία
```



N-grams: Definitions

- A contiguous sequence of n items from a given sequence of text or speech.
  - Phonemes
  - Syllables
  - Characters
  - Words
  - Application-based units
- Tokenization considerations
  - Punctuation
  - Digits
  - Space

- Character n-grams (n=2)
  - *This is a text*
  - [Th], [hi], [is], [s_], [_a], [a_], [_t], [te], [ex], [xt]
- Word n-grams (n=2)
  - *This is a bigger text*
  - [this is], [is a], [a bigger], [bigger text]

N-grams: History

- Andrey Markov (1913): study of the sequences of vowels and consonants in the first 20 K characters of the novel in verse *Eugene Onegin* written by Alexander Pushkin (1837).
- Benett (1976): Probably the first application to authorship attribution.
- Kjell (1994), Kjell et al. (1993): Character 2 & 3-gram frequencies used in the “classic” authorship attribution problem, i.e. The Federalist Papers.
- 1st and 2nd positions in most authorship attribution shared tasks ((Argamon & Juola, 2011, Juola, 2004, Juola, Sofko, & Brennan, 2006, Mikros & Perifanos 2011).

N-grams: Pros & Cons

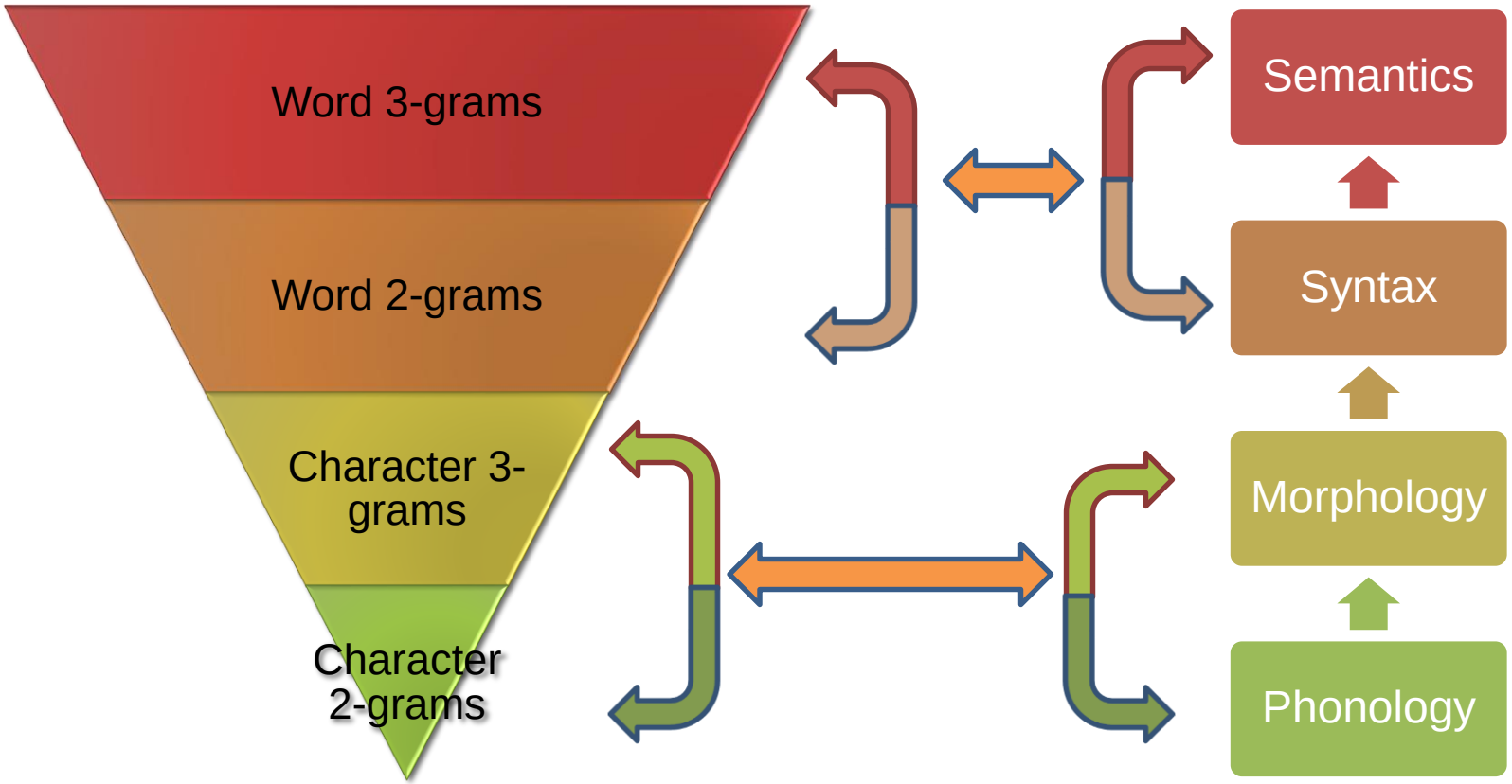
+

- Easy calculation
- Cover many aspects of text production (punctuation, use of capital letters etc.)
- Resistant to textual “noise”, i.e. various non-systematic deviations in spelling, punctuation etc.
- They can be applied to all scripts. Good choice for Eastern Asian languages where the word limits are unclear.

-

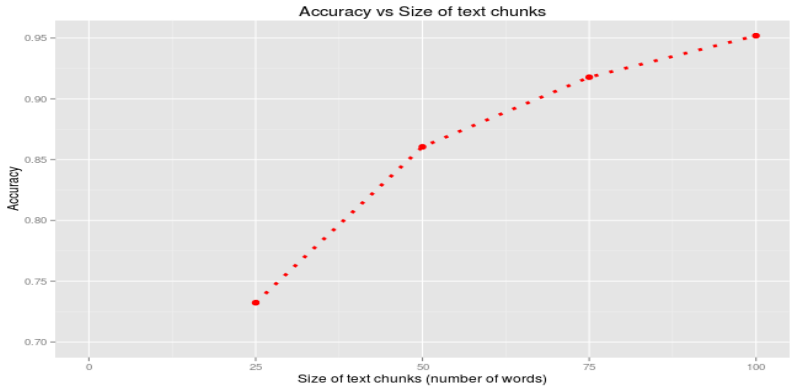
- Lack of any explicit representation of long range dependency (Chomsky’s critique).
- They are surface structures. They can’t capture deeper linguistic knowledge.

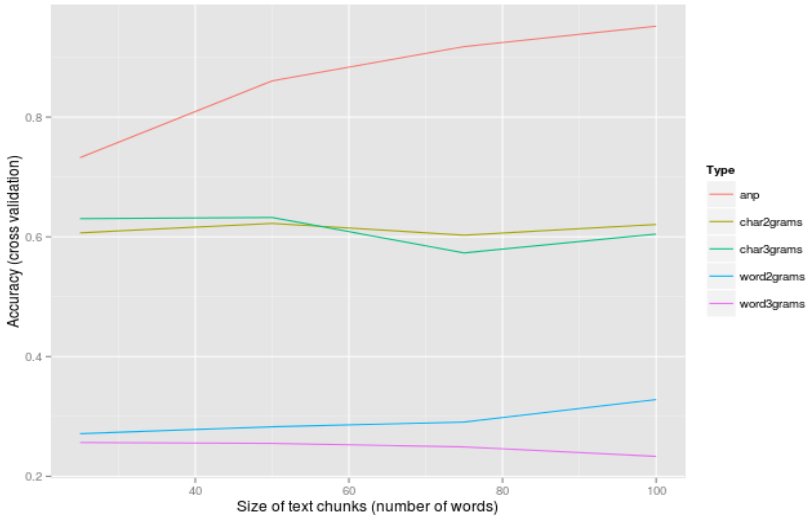
Author's Multilevel N-gram Profile (AMNP)



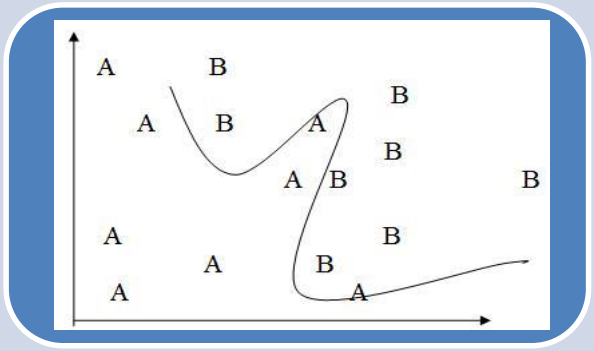
Authorhsip Attribution in Twitter

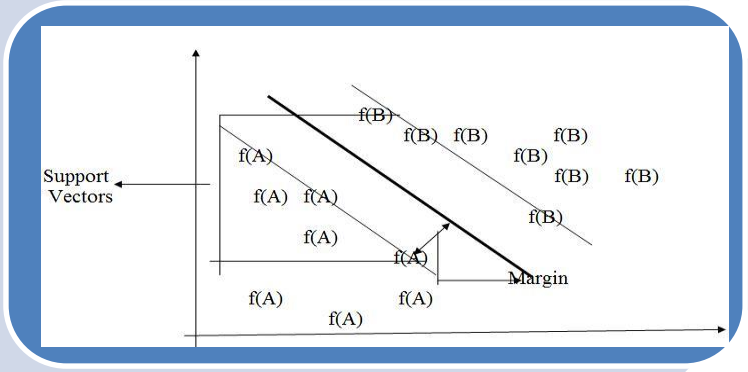
(Mikros & Perifanos 2013)





Machine Learning





Big questions

- How might we automatically generate rules that do well on observed data?
- What kind of confidence do we have that they will do well in the future?

Or, in reverse order,

- What to optimize? [sample complexity]
- How to optimize it? [algorithms]

Supervised Learning

- Like human learning from past experiences.
- A computer does not have “experiences”.
- A computer system learns from data, which represent some “past experiences” of an application domain.
- Our focus: learn a target function that can be used to predict the values of a discrete class attribute, e.g., approve or not-approved, and high-risk or low risk.
- The task is commonly called: Supervised learning, classification, or inductive learning.

The data and the goal

- **Data**: A set of data records (also called examples, instances or cases) described by
  - *k* attributes: $A_1, A_2, \ldots A_k$.
  - a class: Each example is labelled with a pre-defined class.
- **Goal**: To learn a classification model from the data that can be used to predict the classes of new (future, or test) cases/instances.

Vector representation

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Class	text file	Words	St_sTTR	St_AWL	St_sdAWL	St_ASL	St_sdASL	St_HL	St_DL	St_D_H	St_LD	St_K	St_Entrop	St_RelEnt	St_1LW	St_2LW	St_3LW
2	O	P_Dros_votanifragment.txt _ 0.txt	1001	63.22222137	5.517517567	3.09136796	17.24137878	10.04081154	49.05095	5.094905	0.10387	1.306452	66.5668	2.531807	84.38134	4.595405	8.591409	24.37562
3	O	P_Kark_dihghmata.txt _ 0.txt	1006	52.88888931	4.831831932	2.802043915	19.58823776	15.69353485	40.65606	5.964215	0.146699	1.091476	85.07602	2.44126	81.30493	2.385686	14.81113	27.7336
4	O	P_Kark_dihghmata.txt _ 1.txt	1005	55.22222137	5.046046257	2.746081591	22.72727585	14.95009232	41.69154	3.681592	0.088305	0.936416	93.83926	2.428381	80.88763	3.482587	10.94527	28.35821
5	O	P_Kark_dihghmata.txt _ 10.txt	1016	55.33333206	4.928928852	2.670246124	15.36923122	13.35988522	43.40551	3.740157	0.086168	0.996071	94.62769	2.416226	80.35621	6.003937	10.82677	25.7874
6	O	P_Kark_dihghmata.txt _ 11.txt	1002	57.11111069	5.034034252	2.674537182	22.22222137	17.00386238	43.81238	5.489022	0.125285	0.980237	75.1989	2.470612	82.32992	3.093812	11.57685	25.9481
7	O	P_Kark_dihghmata.txt _ 12.txt	1015	55.44444275	4.968968868	2.800307274	16.39344215	10.58029461	41.08374	5.221675	0.127098	1.084189	82.31212	2.438617	81.11241	5.812808	10.83744	26.79803
8	O	P_Kark_dihghmata.txt _ 13.txt	1019	51.77777863	4.859000206	2.597665548	17.87499809	14.515275	39.15604	5.103042	0.130326	0.978641	86.42466	2.410404	80.12846	6.771344	8.635918	27.77233
9	O	P_Kark_dihghmata.txt _ 14.txt	1007	54.77777863	4.911911964	2.686388254	16.66666412	11.8159132	40.21847	6.653426	0.165432	0.914449	81.92898	2.437487	81.16761	2.979146	13.30685	24.92552
10	O	P_Kark_dihghmata.txt _ 15.txt	1010	62.22222137	5.354645252	2.823668003	17.57894135	15.11615467	47.52475	5.148515	0.108333	1.27991	63.42515	2.533724	84.33597	3.564356	9.306931	24.75248
11	O	P_Kark_dihghmata.txt _ 16.txt	1017	57	4.965965748	2.793644905	15.62499905	11.91704178	43.46116	3.834808	0.088235	1.005917	80.2677	2.444266	81.27718	4.916421	10.81613	26.74533
12	O	P_Kark_dihghmata.txt _ 17.txt	1020	56.11111069	4.939939976	2.658294439	14.28571606	12.98733234	39.70588	7.156863	0.180247	1.077393	66.47443	2.478332	82.37491	6.568627	10.58824	22.94118
13	O	P_Kark_dihghmata.txt _ 18.txt	1016	56.66666794	5.016016006	2.684801579	13.15789223	10.50086117	42.12598	5.216535	0.123832	1.129979	63.29825	2.486406	82.69019	6.102362	11.12205	22.04724
14	O	P_Kark_dihghmata.txt _ 19.txt	1020	56.55555725	4.940940857	2.741632462	12.19512177	11.78557301	43.72549	3.823529	0.087444	1.056452	83.02576	2.44357	81.21949	6.470588	11.07843	25.78431
15	O	P_Kark_dihghmata.txt _ 2.txt	1005	56.55555725	5.041040897	2.82156682	22.22222519	14.50217533	43.38308	4.278607	0.098624	0.889098	83.54249	2.452509	81.69131	4.378109	11.14428	26.66667
16	O	P_Kark_dihghmata.txt _ 20.txt	1012	57.22222137	4.99699688	2.902039766	16.37704849	14.52717495	43.97233	6.225296	0.141573	1.112735	80.67225	2.462685	81.94798	4.841897	13.63636	25.79051
17	O	P_Kark_dihghmata.txt _ 21.txt	1008	58.88888931	4.974026203	2.809150219	18.90566063	14.49438095	45.83333	4.662698	0.101732	1.044625	81.07757	2.463316	82.01593	4.563492	11.40873	27.38095
18	O	P_Kark_dihghmata.txt _ 22.txt	1000	56.66666794	5.227227211	3.010572433	22.68181992	15.37556458	46.1	4.4	0.095445	0.934236	87.4	2.440316	81.34385	3.4	10.6	27
19	O	P_Kark_dihghmata.txt _ 23.txt	1013	59	5.42342329	3.096622467	16.66666794	14.00322819	47.38401	5.429418	0.114583	1.119247	83.22216	2.475398	82.35928	4.442251	12.14215	23.29714
20	O	P_Kark_dihghmata.txt _ 24.txt	1001	58.77777863	5.356356144	3.101673365	22.72727203	19.55952835	47.25275	4.295704	0.090909	1.05123	97.78433	2.449396	81.63471	4.095904	10.58941	25.57443
21	O	P_Kark_dihghmata.txt _ 25.txt	1016	58	5.271271229	2.938495398	15.38461208	11.42187023	46.65354	4.625984	0.099156	1.199134	74.9039	2.480689	82.50005	5.314961	11.22047	24.31102
22	O	P_Kark_dihghmata.txt _ 26.txt	1016	57.33333206	5.201201439	2.97089386	17.24138069	14.11644554	44.48819	5.019685	0.112832	1.090535	88.93143	2.445606	81.3333	4.527559	12.00787	26.47638
23	O	P_Kark_dihghmata.txt _ 27.txt	1011	59.88888931	5.319319248	2.987136364	16.64999962	11.72253323	46.19189	4.648863	0.100642	1.046559	84.02136	2.467657	82.12517	3.956479	9.198813	27.99209
24	O	P_Kark_dihghmata.txt _ 28.txt	1021	57.88888931	5.342342377	2.942487955	14.49275303	14.07503414	44.17238	4.89716	0.110865	1.190987	73.15522	2.483211	82.52543	5.288932	11.45935	23.21254
25	O	P_Kark_dihghmata.txt _ 29.txt	1016	57.88888931	5.013986111	2.796748877	13.18420887	11.22700214	41.14173	4.724409	0.114833	1.052525	64.3445	2.486405	82.69015	5.905512	11.51575	23.12992
26	O	P_Kark_dihghmata.txt _ 3.txt	1002	58	5.298298359	2.855067492	21.27659798	11.49234581	43.31337	5.888224	0.135945	1.024242	71.47382	2.48542	82.82338	2.59481	9.181637	26.84631
27	O	P_Kark_dihghmata.txt _ 30.txt	1016	58.55555725	5.146146297	2.821095467	14.70588017	12.99375057	44.19291	5.019685	0.113586	1.027944	73.47015	2.478951	82.44225	5.11811	11.12205	24.01575
28	O	P_Kark_dihghmata.txt _ 31.txt	1019	57.22222137	4.932932854	2.701685429	13.51351452	11.68800354	41.70756	5.593719	0.134118	1.02988	88.00408	2.447273	81.35411	6.084396	12.16879	25.51521
29	O	P_Kark_dihghmata.txt _ 32.txt	1006	54	4.846847057	2.712246418	18.51851845	14.42055702	41.94831	4.671968	0.111374	0.815884	105.1148	2.391751	79.65604	4.572565	13.8171	26.64016
30	O	P_Kark_dihghmata.txt _ 33.txt	1016	57.22222137	5.157156944	2.811429262	17.85714149	14.03557682	43.40551	5.610236	0.129252	1.052525	75.25265	2.472602	82.23111	3.937008	11.02362	25.59055
31	O	P_Kark_dihghmata.txt _ 34.txt	1019	56.77777863	5.059059143	2.774330378	14.2857151	11.74258041	41.41315	5.888126	0.14218	1.114108	81.14712	2.454894	81.60744	6.673209	9.126595	25.61335
32	O	P_Kark_dihghmata.txt _ 35.txt	1007	54	4.964964867	2.769677877	17.8571434	14.04464054	42.70109	4.468719	0.104651	0.94027	80.48921	2.439997	81.25118	5.163853	11.12214	26.01787
33	O	P_Kark_dihghmata.txt _ 36.txt	1017	57.33333206	4.942943096	2.709322453	15.38461494	9.822951317	41.98623	5.309735	0.126464	1.005917	80.67378	2.45002	81.46852	4.621436	12.48771	26.15536
34	O	P_Kark_dihghmata.txt _ 37.txt	1019	54.55555725	5.169168949	2.9277215	15.625	12.61077881	40.52993	5.299313	0.130751	1.136268	81.47455	2.442966	81.21091	5.495584	11.28557	26.10402
35	O	P_Kark_dihghmata.txt _ 38.txt	1003	56	5.087087154	2.825844049	20.40816307	10.04970551	42.97109	6.081755	0.141531	1.022177	97.69296	2.428257	80.90682	4.287139	10.9671	25.42373
36	O	P_Kark_dihghmata.txt _ 39.txt	1026	54.33333206	4.967967987	2.78648591	12.04819298	9.427391052	39.86355	6.042885	0.151589	1.056112	93.57105	2.401525	79.75447	6.725146	12.47563	26.02339
37	O	P_Kark_dihghmata.txt _ 4.txt	1004	55.22222137	5.015015125	2.757168531	18.51852608	12.24858952	39.14343	7.071713	0.180662	0.912381	79.78048	2.454089	81.75573	3.884462	11.15538	26.49402
38	O	P_Kark_dihghmata.txt _ 40.txt	1004	57.22222137	5.336336136	3.005837917	33.30000305	25.76839828	45.81673	4.880478	0.106522	1.020121	78.15352	2.474121	82.42306	2.788845	12.15139	25
39	O	P_Kark_dihghmata.txt _ 41.txt	1008	56.44444275	5.154690742	2.927192211	23.3255825	23.98086739	43.55159	5.257937	0.120729	0.92	76.01883	2.470745	82.26327	4.265873	10.51587	26.4881

Supervised vs. unsupervised Learning

- Supervised learning: classification is seen as supervised learning from examples.
  - Supervision: The data (observations, measurements, etc.) are labeled with pre-defined classes. It is like that a “teacher” gives the classes (supervision).
  - Test data are classified into these classes too.
- Unsupervised learning (clustering)
  - Class labels of the data are **unknown**
  - Given a set of data, the task is to establish the existence of classes or clusters in the data

Assumption of learning

- The distribution of training examples is identical to the distribution of test examples (including future unseen examples).
  - In practice, this assumption is often violated to certain degree.
  - Strong violations will clearly result in poor classification accuracy.
- To achieve good accuracy on the test data, training examples must be sufficiently representative of the test data.

Evaluating classification methods

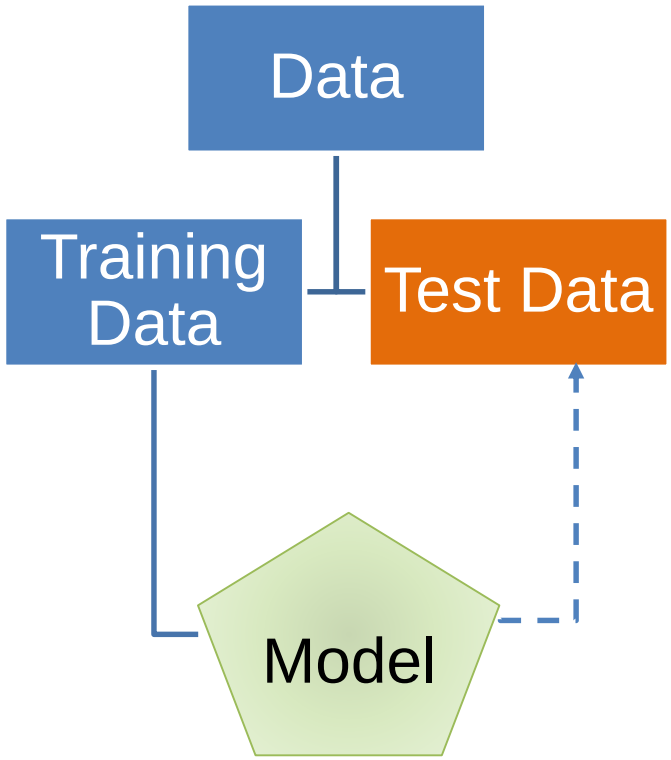
- Predictive accuracy

$$Accuracy = \frac{Number\ of\ correct\ classificiations}{Total\ number\ of\ test\ cases}$$

- Efficiency
  - time to construct the model
  - time to use the model
- Robustness: handling noise and missing values
- Scalability: efficiency in disk-resident databases
- Interpretability:
  - understandable and insight provided by the model
- Compactness of the model: size of the tree, or the number of rules.

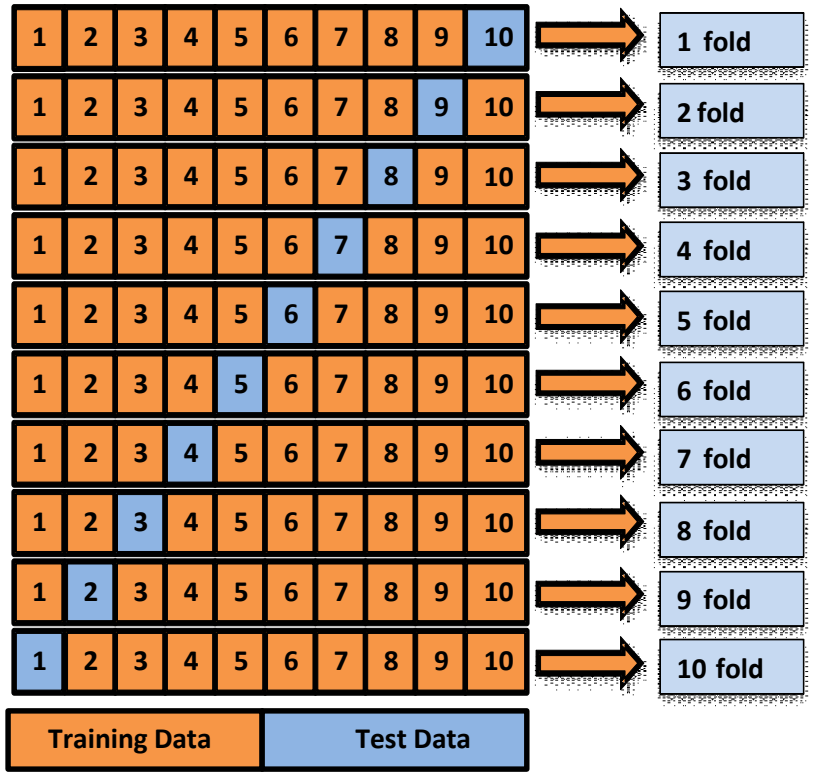
Evaluation methods: Holdout (test) Set

- **Holdout set**: The available data set D is divided into two disjoint subsets,
  - the training set D_{train} (for learning a model)
  - the test set D_{test} (for testing the model)
- **Important**: training set should not be used in testing and the test set should not be used in learning.
  - Unseen test set provides a unbiased estimate of accuracy.
- The test set is also called the holdout set. (the examples in the original data set D are all labeled with classes.)
- This method is mainly used when the data set D is large.



Evaluation methods: n-fold cross-validation

- **n-fold cross-validation**: The available data is partitioned into *n* equal-size disjoint subsets.
- Use each subset as the test set and combine the rest *n*-1 subsets as the training set to learn a classifier.
- The procedure is run *n* times, which give *n* accuracies.
- The final estimated accuracy of learning is the average of the *n* accuracies.
- 10-fold and 5-fold cross-validations are commonly used.
- This method is used when the available data is not large.

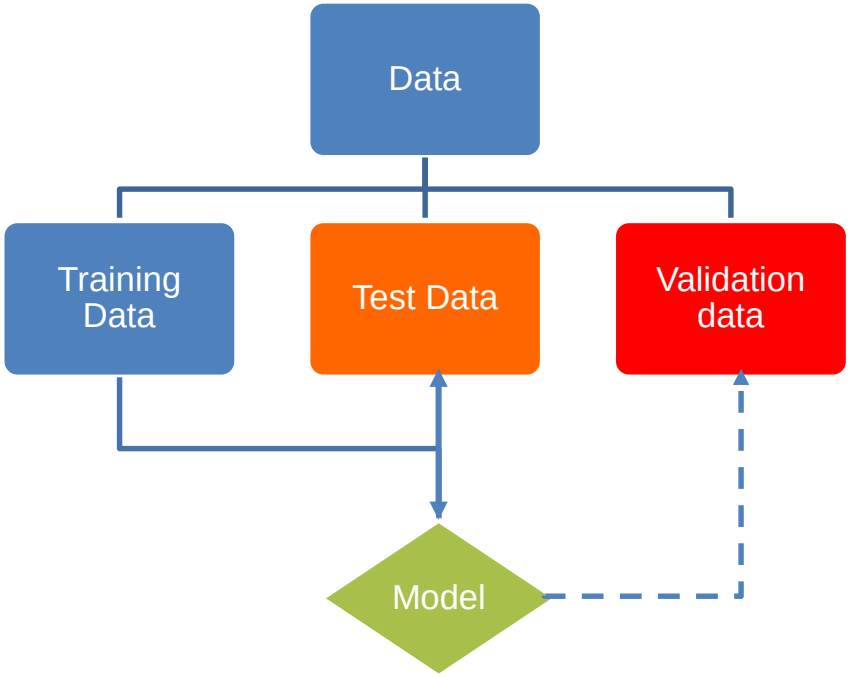


Evaluation methods: Leave-one-out cross-validation

- **Leave-one-out cross-validation**: This method is used when the data set is very small.
- It is a special case of cross-validation
- Each fold of the cross validation has only a single test example and all the rest of the data is used in training.
- If the original data has *m* examples, this is *m*-fold cross-validation

Evaluation methods: Validation Set

- **Validation set**: the available data is divided into three subsets,
  - a training set,
  - a validation set and
  - a test set.
- A validation set is used frequently for estimating parameters in learning algorithms.
- In such cases, the values that give the best accuracy on the validation set are used as the final parameter values.
- Cross-validation can be used for parameter estimating as well.



Classification measures

- Accuracy is only one measure (error = 1-accuracy).
- Accuracy is not suitable in some applications.
  - In text mining, we may only be interested in the documents of a particular topic, which are only a small portion of a big document collection.
  - In classification involving skewed or highly imbalanced data, e.g., network intrusion and financial fraud detections, we are interested only in the minority class.
    - High accuracy does not mean any intrusion is detected.
    - E.g., 1% intrusion. Achieve 99% accuracy by doing nothing.
- The class of interest is commonly called the positive class, and the rest negative classes.

Precision and recall measures [1]

- Used in information retrieval and text classification.
- We use a confusion matrix to introduce them.

	Classified Positive	Classified Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

Where

- *TP*: the number of correct classifications of the positive examples (true positives)
- *FN*: the number of incorrect classifications of the positive examples (false negatives)
- *FP*: the number of incorrect classifications of the negative examples (false positives)
- *TN*: the number of correct classifications of the negative examples (true negatives)

Precision and recall measures [2]

	Classified Positive	Classified Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

$$p = \frac{TP}{TP + FP} \qquad r = \frac{TP}{TP + FN}$$

- **Precision *p*** is the number of correctly classified positive examples divided by the total number of examples that are classified as positive or what percent of the positive predictions were correct.
- **Recall *r*** is the number of correctly classified positive examples divided by the total number of actual positive examples in the test set or what percent of the positive cases were caught.

F_1-value (also called F_1-score)

- It is hard to compare two classifiers using two measures. F_1-score combines precision and recall into one measure.

$$F_1 = 2 \cdot \frac{r \cdot p}{r + p}$$

- F_1-score is the harmonic mean of precision and recall.
  - The harmonic mean of two numbers tends to be closer to the smaller of the two.
  - For F_1-value to be large, both *p* and *r* much be large.

An authorship attribution example

Classified as =>	A	B	C	D	Total documents
A	113	29	1	2	**145**
B	15	157	0	2	**174**
C	0	0	57	0	**57**
D	3	1	0	91	**95**
					471

- Classifier's accuracy= **0.8875** or 88.75% (113+157+57+91)/471
  - Precision (A)= 0.863 => 111/(113+15+3)
  - Recall (A)= 0.779 => 113/(113+29+1+2)
  - F1-value (A)= 0.819 => 2*((0.863 * 0.779)/(0.863 + 0.779))

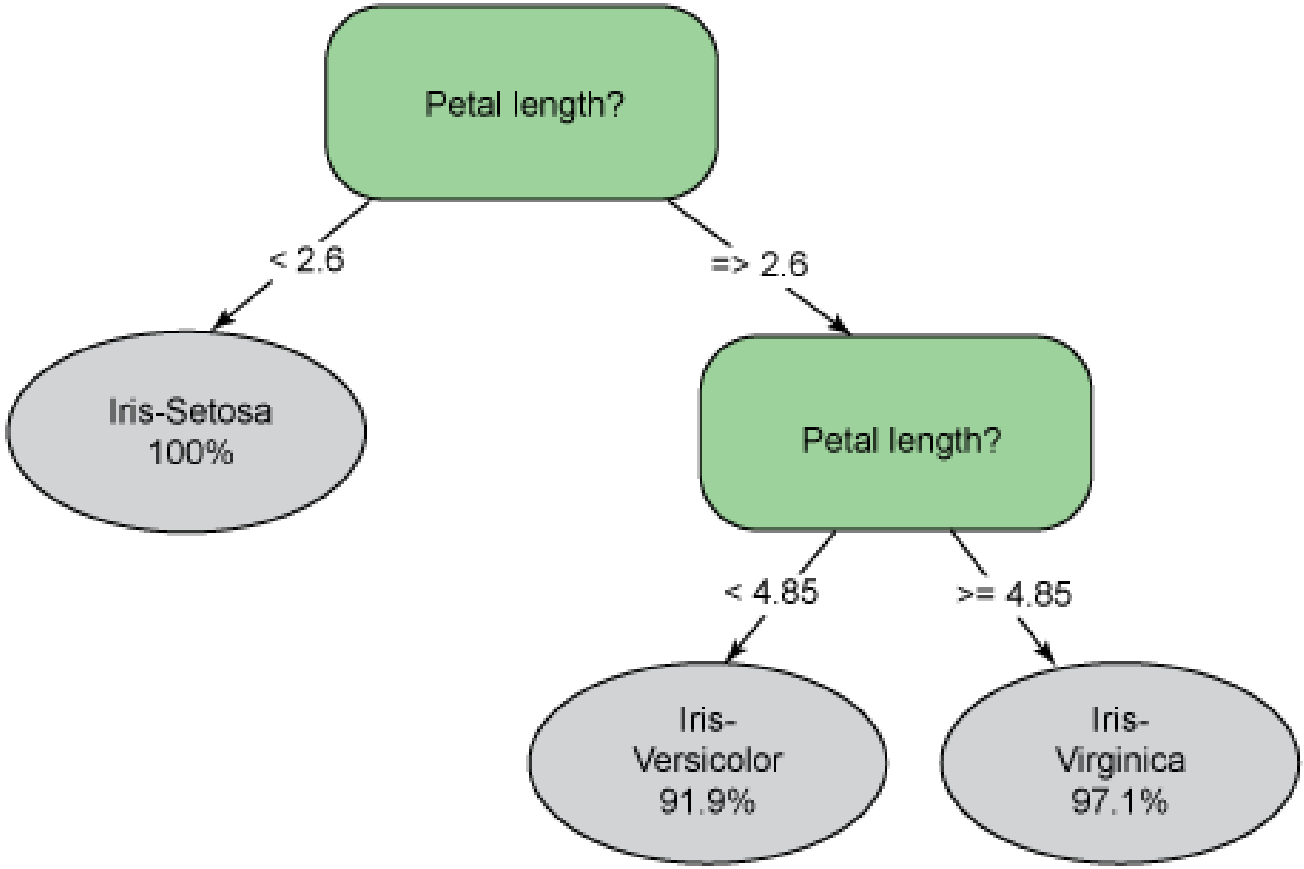
Random Forests

- A random forest is an ensemble (i.e., a collection) of unpruned decision trees (Breiman 2001).
- Random forests are often used when we have very large training datasets and a very large number of input variables (hundreds or even thousands of input variables). A random forest model is typically made up of tens or hundreds of decision trees.
- Can be used for classification or regression.
- Accuracy and variable importance information is provided with the results.
- For a really simple explanation check the Edwin Chen's Quora answer: http://www.quora.com/Machine-Learning/How-do-random-forests-work-in-laymans-terms

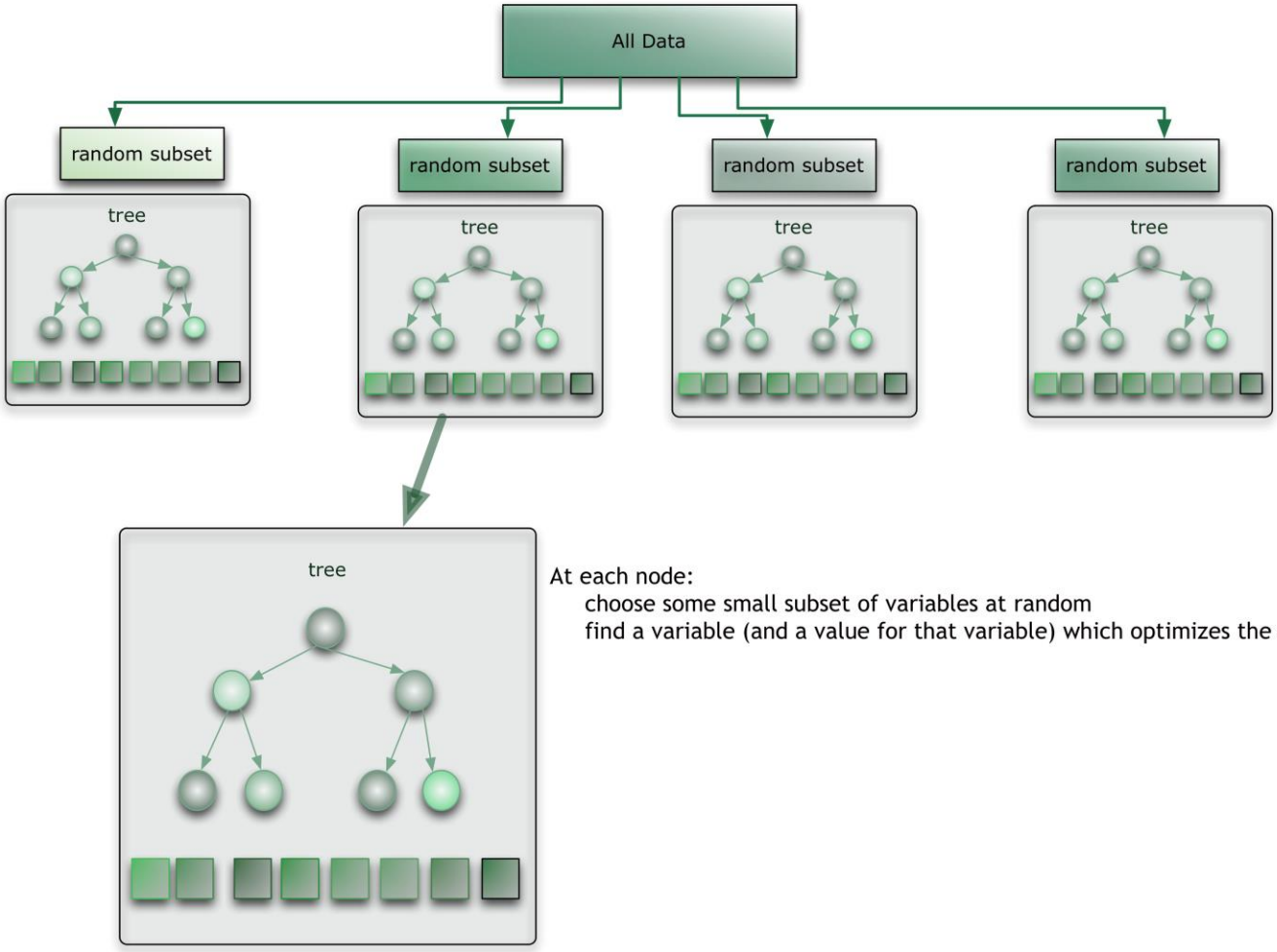
How it works?

- Each decision tree is built from a random subset of the training dataset, using what is called replacement sampling (thus it is doing what is known as bagging). That is, some entities will be included more than once in the sample, and others won't appear at all. Generally, about two thirds of the entities will be included in the subset of the training dataset, and one third will be left out.
- In building each decision tree model based on a different random subset of the training dataset a random subset of the available variables is used to choose how best to partition the dataset at each node.
- Each decision tree is built to its maximum size, with no pruning performed.
- Together, the resulting decision tree models of the forest represent the final ensemble model where each decision tree votes for the result, and the majority wins.

From Decision Trees …



… to Random Forests

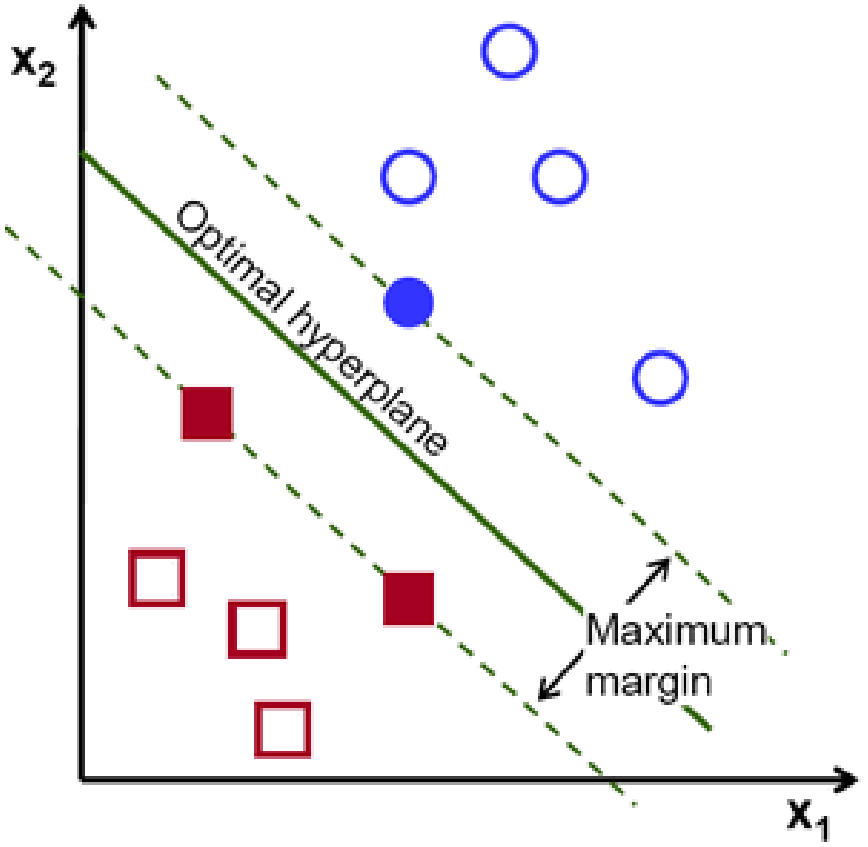


Advantages

- It produces a highly accurate classifier and learning is fast
- It runs efficiently on large data bases.
- Does not require data preprocessing (normalization, missing values imputations etc.) and is resilient to outliers.
- It can handle thousands of input variables without the need for executing variable selection procedures before.
- Because many trees are built and there are two levels of randomness and each tree is effectively an independent model, RF tends not to overfit to the training dataset.

Support Vector Machines - SVM

- A **support vector machine** (**SVM**) is a concept in statistics and computer science for a set of related supervised learning methods that analyze data and recognize patterns, used for classification and regression analysis (Vapnik 1995).
- It involves finding the hyperplane (line in 2D, plane in 3D and hyperplane in higher dimensions.
- More formally, a hyperplane is n-1 dimensional subspace of an n-dimensional space) that best separates two classes of points with the maximum margin.
- The data points that kind of "support" this hyperplane on either sides are called the "support vectors".
- For cases where the two classes of data are not linearly separable, the points are projected to an exploded (higher dimensional) space where linear separation may be possible.
- A problem involving multiple classes can be broken down into multiple one-versus-one or one-versus-rest binary classification problems



The kernel function

- If we are examining data in one dimension (one variable) we can plot them across a line.
- In figure 1 we can not linearly separate red from blue dots since red dots are in the middle of the blue dots.
- We can solve the problem by adding a higher dimension to the data by taking the power of 2.
- In figure 2 we are now having a two-dimensional plot (x vs. x^2) and the data now can be linearly separated.
- Kernel function is a trick which permits SVM to project data in a higher dimensional space. It can be proved that for every dataset there is a kernel function that separates them linearly.

Fig. 1

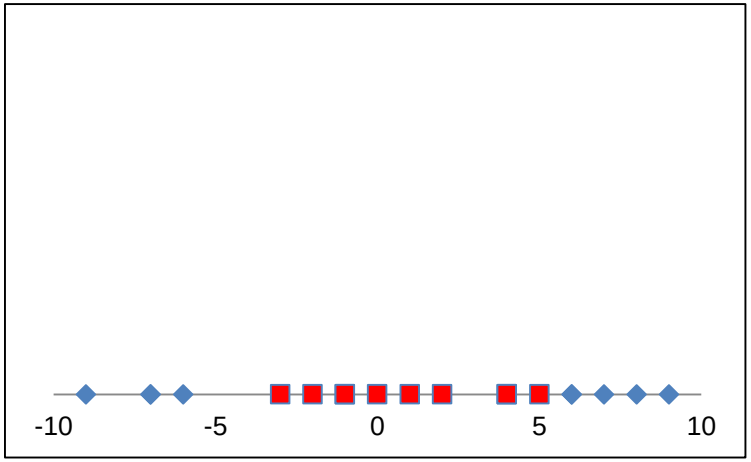
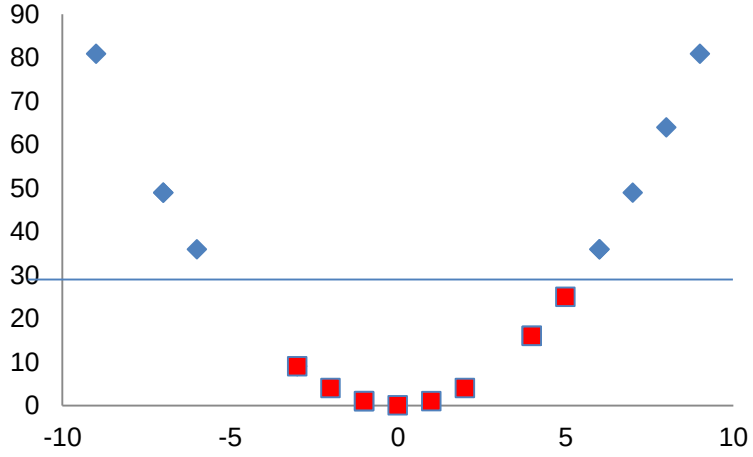


Fig. 2



Conclusions

- Authorship attribution has become the ultimate interdisciplinary field!
- Accuracies reach nearly 100% under the following conditions:
  - Closed set of authors
  - Set of candidate authors (<5)
  - Text size (>100 words)
  - Number of texts per author (>50)
- Open research issues
  - Theory!!!
  - Authorship attribution in “big data”.
  - Small texts and / or small number of texts per author.
  - Author verification (open set of candidate authors)

References [1]

- Bing Liu, 2012. Supervised Learning. Teaching Slides, http://www.cs.uic.edu/~liub/teach/cs583-spring-12/CS583-supervised-learning.ppt
- Argamon, Shlomo, & Juola, Patrick. (2011). Overview of the International Authorship Identification Competition at PAN-2011 *Proceedings of PAN 2011 Lab, Uncovering Plagiarism, Authorship, and Social Software Misuse held in conjunction with the CLEF 2011 Conference on Multilingual and Multimodal Information Access Evaluation, 19-22 September 2011, Amsterdam*.
- Bennett, William Ralph. (1976). *Scientific and engineering problem-solving with the computer*. Englewood Cliffs, N.J.: Prentice Hall.
- Breiman, Leo, 2001. Random Forests. *Machine Learning* 45(1): 5-32.
- Juola, Patrick. (2004). Ad-hoc authorship attribution competiion *Proceedings 2004 Joint International Conference of the Association for Literary and Linguistic Computing and the Association for Computers and the Humanities (ALLC/ACH 2004), 11-16 June 2004, Gothenbug, Sweden*.
- Juola, Patrick, Sofko, John, & Brennan, Patrick. (2006). A prototype for authorship attribution studies. *Literary and Linguistic Computing, 21*(2), 169-178.
- Kjell, Bradley, Woods, W. Addison, & Frieder, Ophir. (1993). Discrimination of authorship using visualization. *Information Processing & Management, 30*(1), 141-150. doi: 10.1016/0306-4573(94)90029-9
- Kjell, Bradley. (1994). Authorship determination using letter pair frequency features with neural network classifiers. *Literary and Linguistic Computing, 9*(2), 119-124. doi: 10.1093/llc/9.2.119

References [2]

- Markov, Andrey A. (1913). An Example of Statistical Analysis of the Text of “Evgenii Onegin" Illustrating the Linking of Events into a Chain. *Bulletin de l, Acadamie Imperiale des Sciences de St. Petersburg, 6*(7), 153-162.
- Mascol, Conrad. (1888). Curves of Pauline and Pseudo-Pauline Style I. *Unitarian Review, 30*, 452-460.
- Mendenhall, Thomas Corwin. (1887). The characteristic curves of composition. *Science, 11*, 237-249.
- Mikros, George K., & Perifanos, Kostas. (2011). Authorship identification in large email collections: Experiments using features that belong to different linguistic levels *Proceedings of PAN 2011 Lab, Uncovering Plagiarism, Authorship, and Social Software Misuse held in conjunction with the CLEF 2011 Conference on Multilingual and Multimodal Information Access Evaluation, 19-22 September 2011, Amsterdam*.
- Mikros, George K., & Perifanos, Kostas. (2013). Authorship attribution in Greek tweets using multilevel author’s n-gram profiles. In E. Hovy, V. Markman, C. H. Martell & D. Uthus (Eds.), *Papers from the 2013 AAAI Spring Symposium "Analyzing Microtext", 25-27 March 2013, Stanford, California* (pp. 17-23). Palo Alto, California: AAAI Press.
- Mosteller, Frederick, & Wallace, David L. (1984). *Applied bayesian and classical inference. The case of The Federalist Papers* (2nd ed.). New York: Springer-Verlag.
- Smith, M. W. A. (1990). Attribution by statistics: a critique of four recent studies. *Revue Informatique et Statistique dans les Sciences humaines, 26*, 233-251.
- Vapnik, Vladimir. (1995). *The nature of statistical learning theory*. New York: Springer-Verlag.

- **Links**
  - http://citizennet.com/blog/2012/11/10/random-forests-ensembles-and-performance-metrics/
  - http://www.quora.com/Machine-Learning/How-do-random-forests-work-in-laymans-terms

Thank you!!!