Κατηγοριοποίηση Κειμένου

Εισαγωγή

- Πολλές φορές οι χρήστες θέλουν να εκτελέσουν ένα ερώτημα επαναληπτικά (π.χ. κάθε μέρα) ώστε να παίρνουν ενημερωμένες πληροφορίες για ένα θέμα που τους ενδιαφέρει
- Αυτού του είδους τα ερωτήματα ονομάζονται standing queries
- Τα ερωτήματα αυτά είναι τα ίδια με τα υπόλοιπα, απλά εκτελούνται περιοδικά
- Για να επιτύχουμε καλό recall και να μην χάσουμε κάποια αποτελέσματα πρέπει να ενημερώνουμε το ερώτημα, οπότε θα γίνει αρκετά περίπλοκο με το πέρασμα του χρόνου

Εισαγωγή

- Με βάση τα standing queries ορίζουμε ένα πρόβλημα κατηγοριοποίησης (classification)
- Δοσμένου ενός συνόλου από κλάσεις / κατηγορίες, ψάχνουμε να καθορίσουμε σε ποιες από αυτές ανήκει ένα αντικείμενο
- Η κατηγοριοποίηση των standing queries ονομάζεται και routing ή filtering
- Τα αποτελέσματα της κατηγοριοποίησης πρέπει να είναι κλάσεις που να είναι κοντά στα ερωτήματα
- Αν οι κλάσεις είναι γενικές τότε μιλάμε για: text classification, text categorization, topic classification, topic spotting

Κατηγοριοποίηση Κειμένου

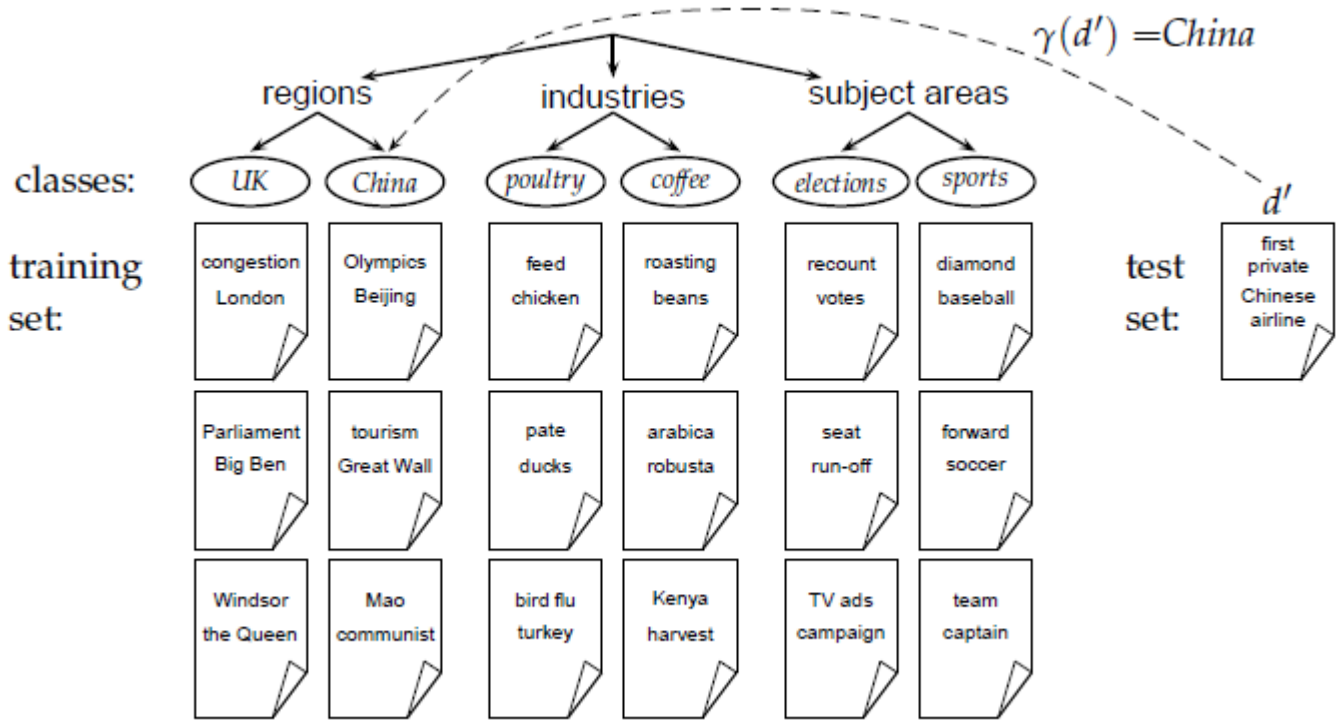
- Μας δίνεται η περιγραφή ενός εγγράφου $d \in X$
- Μας δίνεται επίσης ένα σταθερό σύνολο από κλάσεις / κατηγορίες $C=\{c_1, c_2, \ldots, c_J\}$
- Συνήθως οι κλάσεις ορίζονται από τους ανθρώπους
- Βασιζόμαστε σε ένα training set <d,c> όπου <d,c> ∈ X dace C
- Παράδειγμα:
  - <d,c>=<Ο Δήμαρχος Λαμιέων επισκέφθηκε τις πληγείσες περιοχές, Λαμία>

Κατηγοριοποίηση Κειμένου

- Γιοθετώντας ένα αλγόριθμο μηχανικής μάθησης μπορούμε να εκπαιδεύσουμε τον κατηγοριοποιητή ώστε να αντιστοιχεί τα έγγραφα στις κλάσεις γ: $X \rightarrow C$
- Ο τύπος του αλγορίθμου ανήκει στην κατηγορία supervised learning

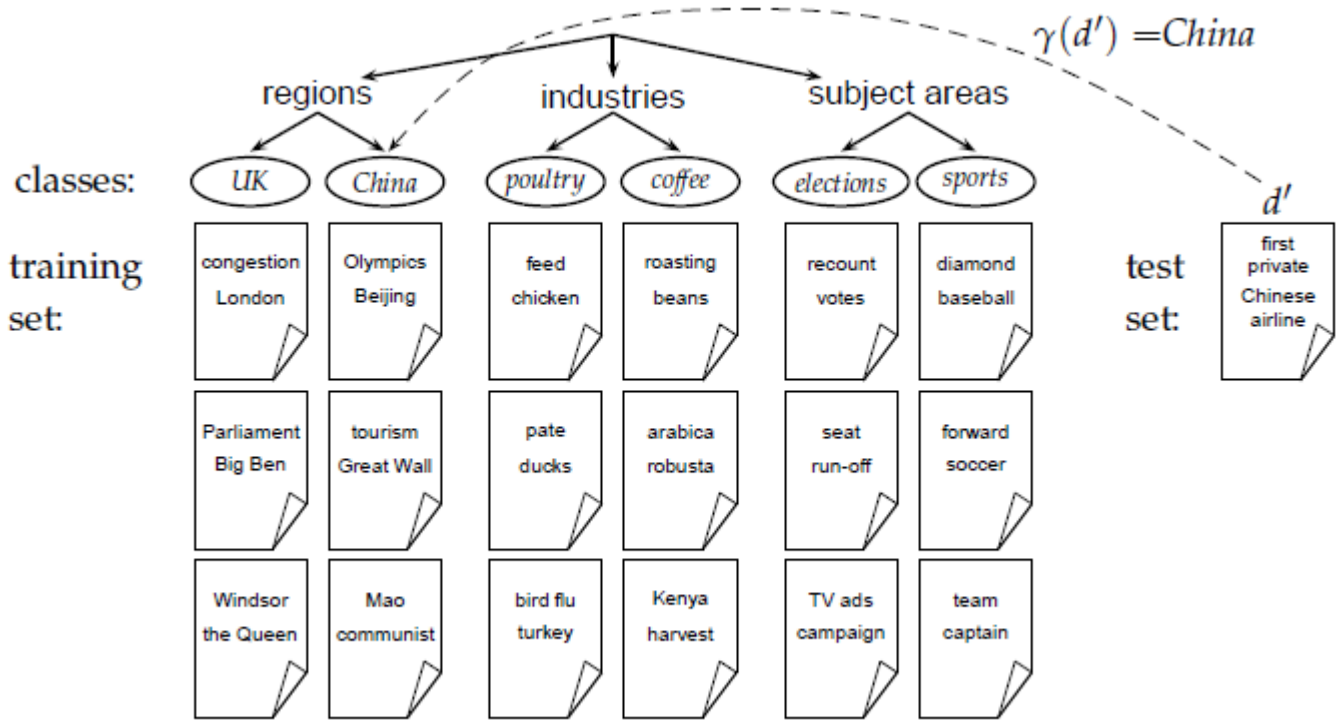
Κατηγοριοποίηση Κειμένου

- Στο ακόλουθο παράδειγμα έχουμε 6 κλάσεις με τρια έγγραφα το καθένα στο training set



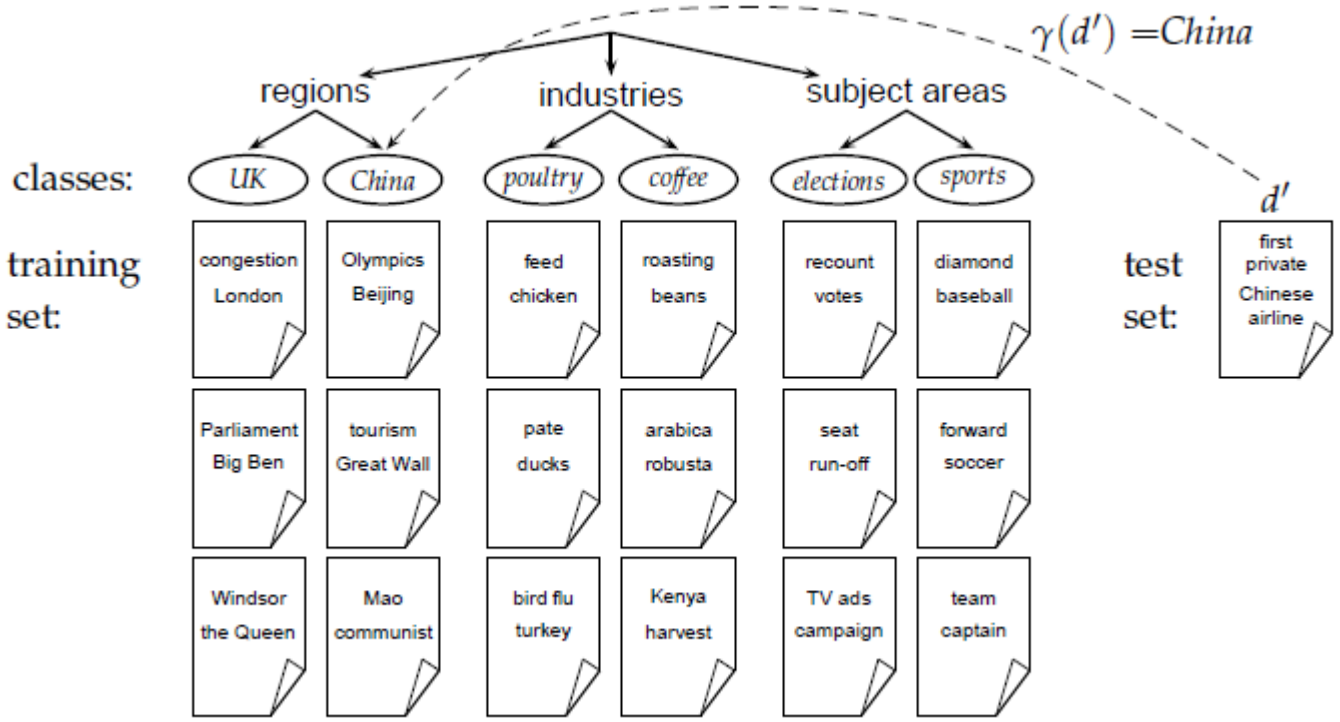
Κατηγοριοποίηση Κειμένου

- Μόλις εκπαιδεύσουμε το σύστημα το εφαρμόζουμε σε ένα νέο έγγραφο του οποίου η κλάση δεν είναι γνωστή



Κατηγοριοποίηση Κειμένου

- Ο αλγόριθμος αντιστοιχεί ένα έγγραφο σε μια κλάση
- Όμως αυτό μπορεί να είναι περιοριστικό
- Παράδειγμα: το έγγραφο με το 2008 Olympics μπορεί να ανήκει και στο China και στο sports



Κατηγοριοποίηση Κειμένου

- Προφανώς ο στόχος του συστήματος είναι να αυξήσει την ακρίβεια
- Είναι εύκολο να πετύχουμε μεγάλη ακρίβεια στο training set
- Αυτό δεν σημαίνει όμως πως θα το πετύχουμε και με τα πραγματικά δεδομένα
- Προφανώς, όταν εκπαιδευουμε το σύστημα, υποθέτουμε πως τα training data και τα πραγματικά δεδομένα ακολουθούν την ίδια κατανομή

Κατηγοριοποιητής Naïve Bayes

- Πρόκειται για μια πιθανοτικοθεωρητική τεχνική
- Η πιθανότητα ένα έγγραφο να ανήκει σε μια κλάση είναι:

$$P(c|d) \propto P(c) \prod_{1 \le k \le n_d} P(t_k|c)$$

- $P(t_k|c)$ είναι η πιθανότητα ότι ο όρος t_k ανήκει στην κλάση c
- Τη μεταφράζουμε ως την απόδειξη ότι ο όρος ανήκει στην κλάση
- P(c) είναι η prior πιθανότητα το έγγραφο να ανήκει στην κλάση c
- Τα $<t_1,t_2,\ldots,t_{nd}>$ είναι τα tokens στο d που αποτελούν μέρη του λεξικού
- n_d είναι το ολικό πλήθος των tokens

Κατηγοριοποιητής Naïve Bayes

- Ο στόχος μας είναι να βρούμε την κλάση που ταιριάζει περισσότερο
- Η καλύτερη κλάση είναι αυτή που μεγιστοποιεί την posterior πιθανότητα c_{map}

$$c_{\text{map}} = \arg\max_{c \in \mathbf{C}} \hat{P}(c|d) = \arg\max_{c \in \mathbf{C}} \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k|c)$$

- Τις τιμές των $P(t_k|c)$ & $P(c)$ τις εκτιμούμε από το training set

Κατηγοριοποιητής Naïve Bayes

- Στον προηγούμενο μαθηματικό τύπο πολλαπλασιάζουμε πιθανότητες
- Όμως μπορεί να έχουμε προβλήματα με τους δεκαδικούς αριθμούς ιδιαίτερα όταν εστιάζουμε σε μεγάλο πλήθος όρων
- Συνεπώς, είναι καλύτερα να υιοθετήσουμε τον λογάριθμο των ποσοτήτων
- Ο μαθηματικός τύπος γίνεται ως εξής:

$$c_{\text{map}} = \arg\max_{c \in \mathbf{C}} [\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c)]$$

- Κάθε όρος του γινομένου είναι η συνεισφορά του κάθε όρου στη συνολική πιθανότητα
- Είναι η ένδειξη του πόσο ταιριάζει ο όρος

Κατηγοριοποιητής Naïve Bayes

- Αντίστοιχα το log(P(c)) είναι η ένδειξη της συχνότητας της κλάσης c
- Οι περισσότερο συχνές κλάσεις είναι πιο πιθανό να αποτελούν τις σωστές κλάσεις

Κατηγοριοποιητής Naïve Bayes

- Η εκτίμηση της πιθανότητας των κλάσεων γίνεται ως εξής:

$$\hat{P}(c) = \frac{N_c}{N}$$

- N_c είναι το πλήθος των εγγράφων στην κλάση c και N είναι ο συνολικός αριθμός των εγγράφων
- Επίσης έχουμε (V είναι το λεξικό):

$$\hat{P}(t|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}$$

- T_{ct} είναι το πλήθος των εμφανίσεων του t στο σύνολο των εγγράφων που ανήκουν στη c
- Είναι το πλήθος των εμφανίσεων σε όλες τις θέσεις k στα έγγραφα του training set

Κατηγοριοποιητής Naïve Bayes

- Το πρόβλημα της μεθόδου είναι πως αν ένας συνδυασμός όρος – κλάση δεν υπάρχει στο training set τότε η πιθανότητα θα είναι 0
- Στη συνέχεια στο γινόμενο το τελικό αποτέλεσμα θα υπολογιστεί επίσης ίσο με το 0
- Το φυσιολογικό είναι πως το training set δεν θα είναι ποτέ τόσο μεγάλο που να καλύπτει όλες τις περιπτώσεις και συνδυασμούς
- Προχωρούμε στη μέθοδο add-one ή Laplace smoothing:

$$\hat{P}(t|c) = \frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)} = \frac{T_{ct}+1}{(\sum_{t' \in V} T_{ct'}) + B}$$

- B είναι το πλήθος των όρων στο λεξικό (|V|)

Κατηγοριοποιητής Naïve Bayes

- Παράδειγμα:

	docID	words in document	in $c =$ *China*?
training set	1	Chinese Beijing Chinese	yes
	2	Chinese Chinese Shanghai	yes
	3	Chinese Macao	yes
	4	Tokyo Japan Chinese	no
test set	5	Chinese Chinese Chinese Tokyo Japan	?

$$\begin{aligned}
\hat{P}(\text{Chinese}|c) &= (5+1)/(8+6) = 6/14 = 3/7 \\
\hat{P}(\text{Tokyo}|c) = \hat{P}(\text{Japan}|c) &= (0+1)/(8+6) = 1/14 \\
\hat{P}(\text{Chinese}|\overline{c}) &= (1+1)/(3+6) = 2/9 \\
\hat{P}(\text{Tokyo}|\overline{c}) = \hat{P}(\text{Japan}|\overline{c}) &= (1+1)/(3+6) = 2/9
\end{aligned}$$

$$\begin{aligned}
\hat{P}(c|d_5) &\propto 3/4 \cdot (3/7)^3 \cdot 1/14 \cdot 1/14 \approx 0.0003 \\
\hat{P}(\overline{c}|d_5) &\propto 1/4 \cdot (2/9)^3 \cdot 2/9 \cdot 2/9 \approx 0.0001
\end{aligned}$$

$$\hat{P}(t|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}} \qquad \hat{P}(c) = \frac{N_c}{N}$$

$$\hat{P}(t|c) = \frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)} = \frac{T_{ct}+1}{(\sum_{t' \in V} T_{ct'}) + B}$$

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \hat{P}(c|d) = \arg\max_{c \in \mathbb{C}} \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k|c)$$

Κατηγοριοποιητής Naïve Bayes

TRAINMULTINOMIALNB($\mathbb{C}, \mathbb{D}$)
1 $V \leftarrow$ EXTRACTVOCABULARY($\mathbb{D}$)
2 $N \leftarrow$ COUNTDOCS($\mathbb{D}$)
3 **for each** $c \in \mathbb{C}$
4 **do** $N_c \leftarrow$ COUNTDOCSINCLASS($\mathbb{D}, c$)
5 $prior[c] \leftarrow N_c/N$
6 $text_c \leftarrow$ CONCATENATETEXTOFALLDOCSINCLASS($\mathbb{D}, c$)
7 **for each** $t \in V$
8 **do** $T_{ct} \leftarrow$ COUNTTOKENSOFTERM($text_c, t$)
9 **for each** $t \in V$
10 **do** $condprob[t][c] \leftarrow \frac{T_{ct}+1}{\sum_{t'}(T_{ct'}+1)}$
11 **return** $V, prior, condprob$

APPLYMULTINOMIALNB($\mathbb{C}, V, prior, condprob, d$)
1 $W \leftarrow$ EXTRACTTOKENSFROMDOC(V, d)
2 **for each** $c \in \mathbb{C}$
3 **do** $score[c] \leftarrow \log prior[c]$
4 **for each** $t \in W$
5 **do** $score[c] \mathrel{+}= \log condprob[t][c]$
6 **return** $\arg\max_{c \in \mathbb{C}} score[c]$

Κατηγοριοποιητής Bernoulli

- Ένας εναλλακτικός τρόπος κατηγοριοποίησης είναι το μοντέλο Bernoulli (multivariate Bernoulli model ή Bernoulli model)
- Δημιουργείται ένας indicator που είναι 1 αν ο όρος υπάρχει ή 0 σε διαφορετική περίπτωση
- Το μοντέλο εκτιμά την πιθανότητα P(t|c) σαν το κλάσμα των εγγράφων της κλάσης c που περιέχουν τον όρο $(Nct + 1)/(Nc + 2)$
- Το μοντέλο λαμβάνει υπόψιν του μόνο την παρουσία και όχι τη συχνότητα των όρων
- Για παράδειγμα μπορεί να κατηγοριοποιήσει ένα ολόκληρο έγγραφο σε μια κλάση ακόμα και για μια παρουσία του όρου μέσα σε αυτό

Κατηγοριοποιητής Bernoulli

- Αλγόριθμος:

```
TRAINBERNOULLINB(C, D)
1  V ← EXTRACTVOCABULARY(D)
2  N ← COUNTDOCS(D)
3  for each c ∈ C
4  do N_c ← COUNTDOCSINCLASS(D, c)
5     prior[c] ← N_c / N
6     for each t ∈ V
7     do N_ct ← COUNTDOCSINCLASSCONTAININGTERM(D, c, t)
8        condprob[t][c] ← (N_ct + 1) / (N_c + 2)
9  return V, prior, condprob

APPLYBERNOULLINB(C, V, prior, condprob, d)
1  V_d ← EXTRACTTERMSFROMDOC(V, d)
2  for each c ∈ C
3  do score[c] ← log prior[c]
4     for each t ∈ V
5     do if t ∈ V_d
6           then score[c] += log condprob[t][c]
7           else score[c] += log(1 − condprob[t][c])
8  return arg max_{c∈C} score[c]
```

Κατηγοριοποιητής Bernoulli

- Παράδειγμα:
  - Ο κατηγοριοποιητής Bernoulli αποφασίζει πως το έγγραφο δεν ανήκει στην κλάση China (αντίθετα με τον Naïve bayes που είδαμε πιο πριν)

	docID	words in document	in $c = China$?
training set	1	Chinese Beijing Chinese	yes
	2	Chinese Chinese Shanghai	yes
	3	Chinese Macao	yes
	4	Tokyo Japan Chinese	no
test set	5	Chinese Chinese Chinese Tokyo Japan	?

$$\hat{P}(c) = 3/4,\ \hat{P}(\overline{c}) = 1/4$$

$$\hat{P}(\text{Chinese}|c) = (3+1)/(3+2) = 4/5$$

$$\hat{P}(\text{Japan}|c) = \hat{P}(\text{Tokyo}|c) = (0+1)/(3+2) = 1/5$$

$$\hat{P}(\text{Beijing}|c) = \hat{P}(\text{Macao}|c) = \hat{P}(\text{Shanghai}|c) = (1+1)/(3+2) = 2/5$$

$$\hat{P}(\text{Chinese}|\overline{c}) = (1+1)/(1+2) = 2/3$$

$$\hat{P}(\text{Japan}|\overline{c}) = \hat{P}(\text{Tokyo}|\overline{c}) = (1+1)/(1+2) = 2/3$$

$$\hat{P}(\text{Beijing}|\overline{c}) = \hat{P}(\text{Macao}|\overline{c}) = \hat{P}(\text{Shanghai}|\overline{c}) = (0+1)/(1+2) = 1/3$$

$$\begin{aligned}\hat{P}(c|d_5) &\propto \hat{P}(c)\cdot\hat{P}(\text{Chinese}|c)\cdot\hat{P}(\text{Japan}|c)\cdot\hat{P}(\text{Tokyo}|c)\\ &\quad\cdot(1-\hat{P}(\text{Beijing}|c))\cdot(1-\hat{P}(\text{Shanghai}|c))\cdot(1-\hat{P}(\text{Macao}|c))\\ &= 3/4\cdot 4/5\cdot 1/5\cdot 1/5\cdot(1-2/5)\cdot(1-2/5)\cdot(1-2/5)\\ &\approx 0.005\\ \hat{P}(\overline{c}|d_5) &\propto 1/4\cdot 2/3\cdot 2/3\cdot 2/3\cdot(1-1/3)\cdot(1-1/3)\cdot(1-1/3)\\ &\approx 0.022\end{aligned}$$

Feature Selection

- Η τεχνική feature selection στοχεύει στην επιλογή ενός υπο-συνόλου όρων από το training set και η υιοθέτησή του για την κατηγοριοποίηση
- Δύο είναι οι κύριοι στόχοι της τεχνικής:
  - Στη μείωση του χώρου των μεταβλητών οπότε και του υιοθετούμενου λεξικού
  - Αύξηση της ακρίβειας αφού εξαλείφουμε το θόρυβο
- Ένα στοιχείο θορύβου είναι αυτό που όταν προστεθεί στην αναπράσταση ενός εγγράφου αυξάνει το σφάλμα
- Παράδειγμα: ένας σπάνιος όρος που μπορεί να υπάρξει σε ένα έγγραφο

Feature Selection

- Μπορούμε να δούμε τη διαδικασία σαν την αντικατάσταση ενός περίπλοκου κατηγοριοποιητή με ένα απλούστερο που βασίζεται σε λιγότερους όρους
- Για μια κλάση c υπολογίζουμε το όφελος για κάθε όρο A(c,t)
- Επιλέγουμε τα k που έχουν τις καλύτερες τιμές
- Το κυρίαρχο πρόβλημα είναι το πως θα υπολογίσουμε το όφελος των όρων
- Κυρίαρχες μέθοδοι είναι: mutual information, χ^2 test, frequency

```
SELECTFEATURES(𝔻, c, k)
1  V ← EXTRACTVOCABULARY(𝔻)
2  L ← []
3  for each t ∈ V
4  do A(t, c) ← COMPUTEFEATUREUTILITY(𝔻, t, c)
5     APPEND(L, ⟨A(t, c), t⟩)
6  return FEATURESWITHLARGESTVALUES(L, k)
```

Mutual Information

- Η τεχνική μετράει το πόση πληροφορία είναι παρούσα/απούσα για ένα όρο κατά την απόφαση κατηγοριοποίησης

$$I(U;C) = \sum_{e_t \in \{1,0\}} \sum_{e_c \in \{1,0\}} P(U = e_t, C = e_c) \log_2 \frac{P(U = e_t, C = e_c)}{P(U = e_t)P(C = e_c)}$$

- e_t=1: το έγγραφο περιέχει τον όρο t
- e_t=0: το έγγραφο δεν περιέχει τον όρο t
- e_c=1: το έγγραφο ανήκει στην κλάση c
- e_c=0: το έγγραφο δεν ανήκει στην κλάση c

Mutual Information

- Εναλλακτικά:

$$I(U;C) = \frac{N_{11}}{N}\log_2\frac{NN_{11}}{N_{1.}N_{.1}} + \frac{N_{01}}{N}\log_2\frac{NN_{01}}{N_{0.}N_{.1}} + \frac{N_{10}}{N}\log_2\frac{NN_{10}}{N_{1.}N_{.0}} + \frac{N_{00}}{N}\log_2\frac{NN_{00}}{N_{0.}N_{.0}}$$

- Ο πρώτος δείκτης απεικονίζει το αν ο όρος υπάρχει στο έγγραφο
- Ο δεύτερος δείκτης απεικονίζει το αν το έγγραφο είναι στην κλάση c

Mutual Information

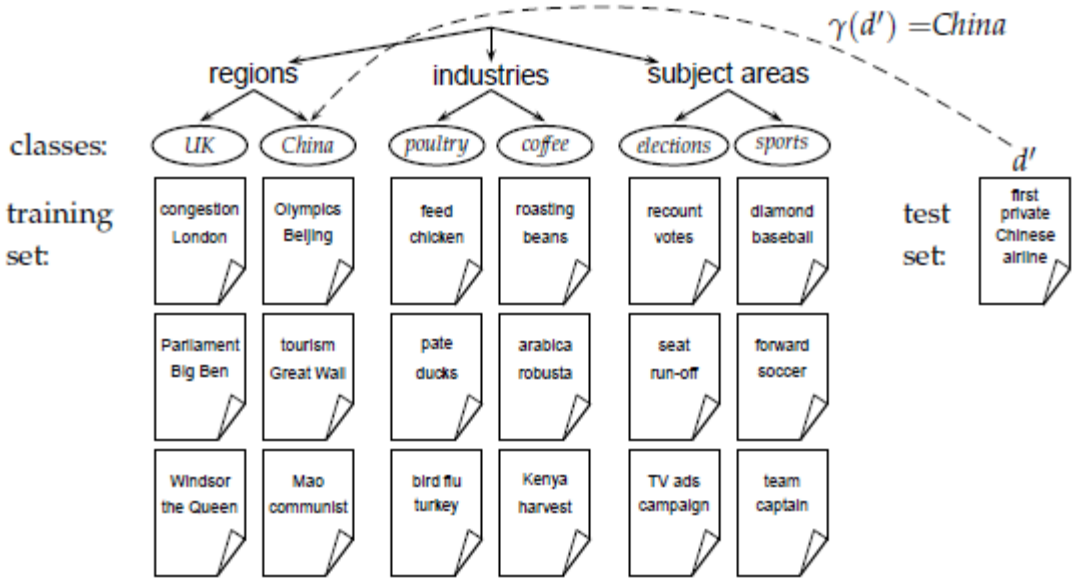
- Παράδειγμα:

$$I(U;C) = \frac{N_{11}}{N}\log_2\frac{NN_{11}}{N_{1.}N_{.1}} + \frac{N_{01}}{N}\log_2\frac{NN_{01}}{N_{0.}N_{.1}} + \frac{N_{10}}{N}\log_2\frac{NN_{10}}{N_{1.}N_{.0}} + \frac{N_{00}}{N}\log_2\frac{NN_{00}}{N_{0.}N_{.0}}$$

	$e_c = e_{poultry} = 1$	$e_c = e_{poultry} = 0$
$e_t = e_{export} = 1$	$N_{11} = 49$	$N_{10} = 27{,}652$
$e_t = e_{export} = 0$	$N_{01} = 141$	$N_{00} = 774{,}106$

$$\begin{aligned} I(U;C) &= \frac{49}{801{,}948}\log_2\frac{801{,}948\cdot 49}{(49+27{,}652)(49+141)} \\ &\quad + \frac{141}{801{,}948}\log_2\frac{801{,}948\cdot 141}{(141+774{,}106)(49+141)} \\ &\quad + \frac{27{,}652}{801{,}948}\log_2\frac{801{,}948\cdot 27{,}652}{(49+27{,}652)(27{,}652+774{,}106)} \\ &\quad + \frac{774{,}106}{801{,}948}\log_2\frac{801{,}948\cdot 774{,}106}{(141+774{,}106)(27{,}652+774{,}106)} \\ &\approx 0.0001105 \end{aligned}$$

Mutual Information



UK

london	0.1925
uk	0.0755
british	0.0596
stg	0.0555
britain	0.0469
plc	0.0357
england	0.0238
pence	0.0212
pounds	0.0149
english	0.0126

China

china	0.0997
chinese	0.0523
beijing	0.0444
yuan	0.0344
shanghai	0.0292
hong	0.0198
kong	0.0195
xinhua	0.0155
province	0.0117
taiwan	0.0108

poultry

poultry	0.0013
meat	0.0008
chicken	0.0006
agriculture	0.0005
avian	0.0004
broiler	0.0003
veterinary	0.0003
birds	0.0003
inspection	0.0003
pathogenic	0.0003

coffee

coffee	0.0111
bags	0.0042
growers	0.0025
kg	0.0019
colombia	0.0018
brazil	0.0016
export	0.0014
exporters	0.0013
exports	0.0013
crop	0.0012

elections

election	0.0519
elections	0.0342
polls	0.0339
voters	0.0315
party	0.0303
vote	0.0299
poll	0.0225
candidate	0.0202
campaign	0.0202
democratic	0.0198

sports

soccer	0.0681
cup	0.0515
match	0.0441
matches	0.0408
played	0.0388
league	0.0386
beat	0.0301
game	0.0299
games	0.0284
team	0.0264

χ^2 Feature Selection

- Γενικά το χ^2 τεστ χρησιμοποιείται για να ελέγξει την ανεξαρτησία δύο συμβάντων
- Στο feature selection τα δύο συμβάντα είναι η ύπαρξη του όρου και η ύπαρξη της κλάσης
- Ταξινομούμε τους όρους με βάση την ακόλουθη ποσότητα:

$$X^2(\mathbb{D}, t, c) = \sum_{e_t \in \{0,1\}} \sum_{e_c \in \{0,1\}} \frac{(N_{e_t e_c} - E_{e_t e_c})^2}{E_{e_t e_c}} \qquad X^2(\mathbb{D}, t, c) = \frac{(N_{11} + N_{10} + N_{01} + N_{00}) \times (N_{11}N_{00} - N_{10}N_{01})^2}{(N_{11} + N_{01}) \times (N_{11} + N_{10}) \times (N_{10} + N_{00}) \times (N_{01} + N_{00})}$$

- Το E είναι η αναμενόμενη συχνότητα για τους συνδυασμούς e_t, e_c
- Παράδειγμα: E_{11} είναι η αναμενόμενη συχνότητα όπου τα t και c απαντώνται ταυτόχρονα

χ^2 Feature Selection

	$e_c = e_{poultry} = 1$	$e_c = e_{poultry} = 0$
$e_t = e_{export} = 1$	$N_{11} = 49$	$N_{10} = 27{,}652$
$e_t = e_{export} = 0$	$N_{01} = 141$	$N_{00} = 774{,}106$

$$E_{11} = N \times P(t) \times P(c) = N \times \frac{N_{11} + N_{10}}{N} \times \frac{N_{11} + N_{01}}{N}$$

$$= N \times \frac{49 + 141}{N} \times \frac{49 + 27652}{N} \approx 6.6$$

	$e_{poultry} = 1$		$e_{poultry} = 0$	
$e_{export} = 1$	$N_{11} = 49$	$E_{11} \approx 6.6$	$N_{10} = 27{,}652$	$E_{10} \approx 27{,}694.4$
$e_{export} = 0$	$N_{01} = 141$	$E_{01} \approx 183.4$	$N_{00} = 774{,}106$	$E_{00} \approx 774{,}063.6$

$$X^2(\mathbb{D}, t, c) = \sum_{e_t \in \{0,1\}} \sum_{e_c \in \{0,1\}} \frac{(N_{e_t e_c} - E_{e_t e_c})^2}{E_{e_t e_c}} \approx 284$$

Frequency Based Feature Selection

- Η τρίτη τεχνική επιλέγει τους όρους που είναι πιο κοινοί σε κάθε κλάση
- Η συχνότητα μπορεί να είναι το df ή το cf

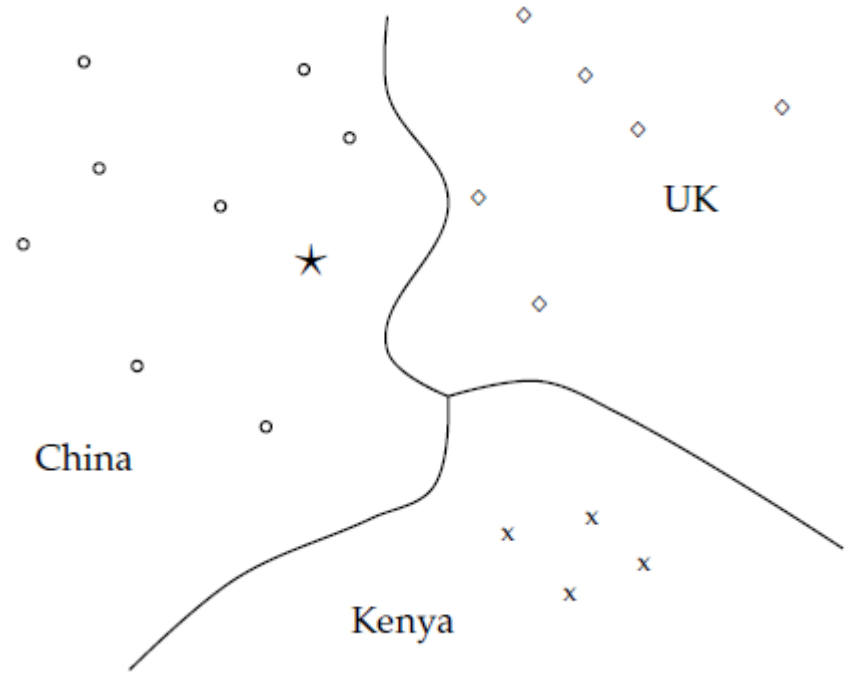
Vector Space Classification

Εισαγωγή

- Βασίζεται στο vector space μοντέλο που έχουμε ήδη δει
- Αναπαριστά τα έγγραφα σαν διανύσματα όπου σε κάθε θέση υπάρχει μια τιμή, συνήθως η tf-idf για κάθε όρο
- Θα συζητήσουμε δύο μεθόδους:
  - Rocchio
  - k-Nearest Neighbor

Rocchio Classification

- Η εικόνα μας δείχνει ένα παράδειγμα κατηγοριοποίησης σε τρεις κλάσεις
- Τα έγγραφα είναι οι κύκλοι, οι ρόμβοι και τα Χ
- Τα όρια στην εικόνα ονομάζονται decision boundaries
- Για την κατηγοριοποίηση ενός εγγράφου καθορίζουμε την περιοχή στην οποία το συναντούμε και το αναθέτουμε στην αντίστοιχη κλάση



Rocchio Classification

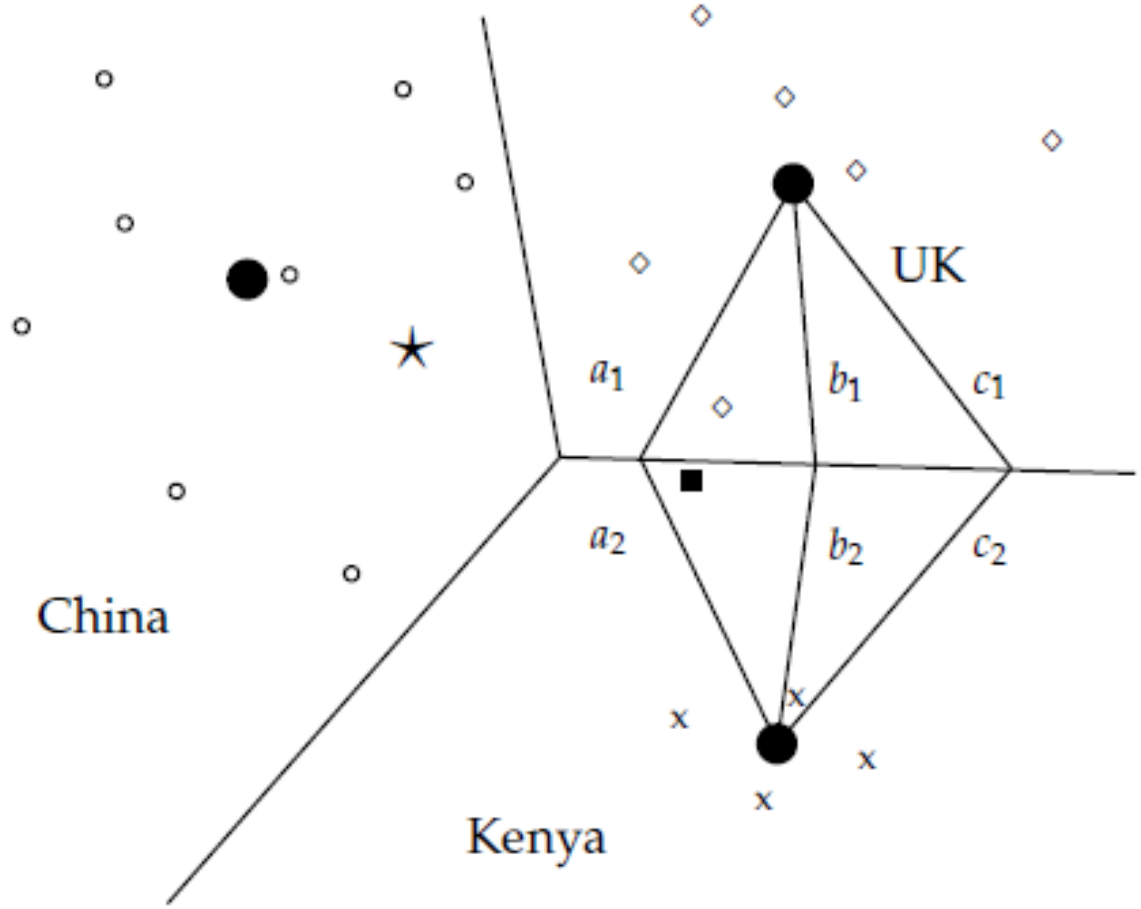
- Το πρόβλημά μας είναι να υιοθετήσουμε αλγόριθμο ο οποίος να παράξει καλά όρια
- Προφανώς, η έκφραση καλό όριο σημαίνει πως από τα αποτελέσματα του training παίρνουμε υψηλή ακρίβεια
- Ο αλγόριθμος Rocchio βρίσκει τα κεντροειδή ώστε να ορίσει τα όρια
- Το κεντροειδές υπολογίζεται ως εξής:

$$\vec{\mu}(c) = \frac{1}{|D_c|} \sum_{d \in D_c} \vec{v}(d)$$

- Το D_c είναι το σύνολο των εγγράφων των οποίων η κλάση είναι η c
- Επίσης, $\vec{v}(d\ ) = \vec{V}(d\ )/|\vec{V}(d\ )|$

Rocchio Classification

- Τα όρια είναι τα σύνολα σημείων που έχουν ίση απόσταση από τα κεντροειδή
- Παράδειγμα:
  - |a1|=|a2|, |b1|=|b2|, |c1|=|c2|



Rocchio Classification

- Ο κανόνας για την κατηγοριοποίηση είναι σε συμφωνία με την περιοχή στην οποία ‘πέφτει’ το έγγραφο

TRAINROCCHIO($\mathbb{C}, \mathbb{D}$)
1 **for each** $c_j \in \mathbb{C}$
2 **do** $D_j \leftarrow \{d : \langle d, c_j \rangle \in \mathbb{D}\}$
3 $\quad \vec{\mu}_j \leftarrow \frac{1}{|D_j|} \sum_{d \in D_j} \vec{v}(d)$
4 **return** $\{\vec{\mu}_1, \ldots, \vec{\mu}_J\}$

APPLYROCCHIO($\{\vec{\mu}_1, \ldots, \vec{\mu}_J\}, d$)
1 **return** $\arg\min_j |\vec{\mu}_j - \vec{v}(d)|$

Rocchio Classification

- Παράδειγμα (ανάθεση):

	term weights					
vector	Chinese	Japan	Tokyo	Macao	Beijing	Shanghai
$\vec{d}_1$	0	0	0	0	1.0	0
$\vec{d}_2$	0	0	0	0	0	1.0
$\vec{d}_3$	0	0	0	1.0	0	0
$\vec{d}_4$	0	0.71	0.71	0	0	0
$\vec{d}_5$	0	0.71	0.71	0	0	0
$\vec{\mu}_c$	0	0	0	0.33	0.33	0.33
$\vec{\mu}_{\overline{c}}$	0	0.71	0.71	0	0	0

$$\mu_c = 1/3 \cdot (\vec{d}_1 + \vec{d}_2 + \vec{d}_3)$$

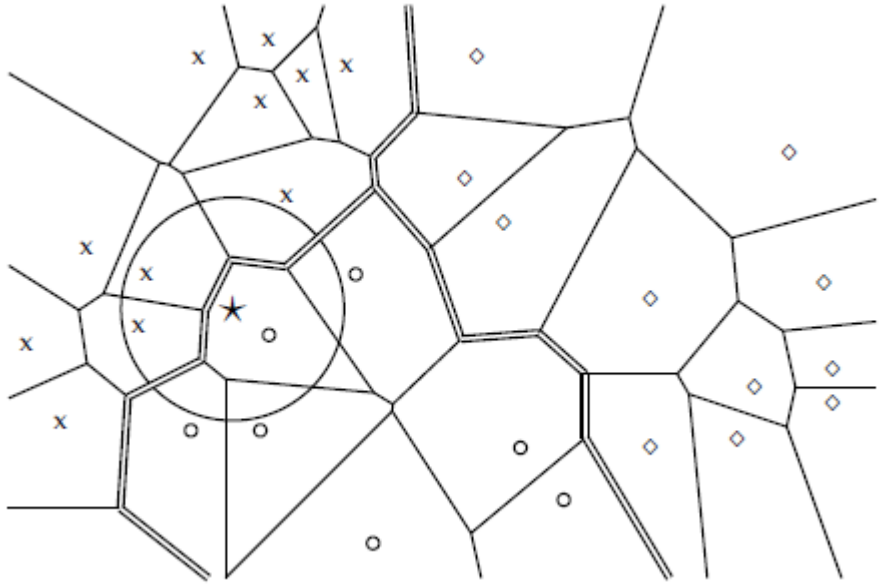
$$\mu_{\overline{c}} = 1/1 \cdot (\vec{d}_4)$$

$$|\mu_c - \vec{d}_5| \approx 1.15$$

$$|\mu_{\overline{c}} - \vec{d}_5| = 0.0$$

k Nearest Neighbor

- Όταν k=1 αναθέτουμε κάθε έγγραφο στην κλάση του πιο κοντινού γείτονα
- Γενικά, η τεχνική αναθέτει το έγγραφο στην πλειοψηφική κλάση από τους k κοντινότερους γείτονες
- Τα decision boundaries τώρα είναι τα τμήματα μιας Voronoi τμηματοποίησης



k Nearest Neighbor

- Κάθε κελί Voronoi περιλαμβάνει ένα αντικείμενο καθώς και τα πιο κοντινά του σημεία
- Στη περίπτωσή μας τα αντικείμενα είναι έγγραφα
- Χωρίζουμε την περιοχή σε |D| πολύγωνα που το καθένα περιλαμβάνει ένα έγγραφο

k Nearest Neighbor

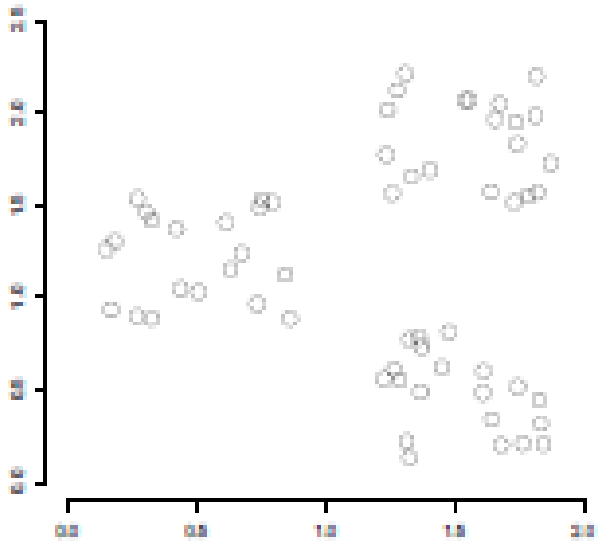
TRAIN-KNN($\mathbb{C}, \mathbb{D}$)
1 $\mathbb{D}' \leftarrow$ PREPROCESS($\mathbb{D}$)
2 $k \leftarrow$ SELECT-K($\mathbb{C}, \mathbb{D}'$)
3 **return** $\mathbb{D}', k$

APPLY-KNN($\mathbb{C}, \mathbb{D}', k, d$)
1 $S_k \leftarrow$ COMPUTENEARESTNEIGHBORS($\mathbb{D}', k, d$)
2 **for each** $c_j \in \mathbb{C}$
3 **do** $p_j \leftarrow |S_k \cap c_j|/k$
4 **return** $\arg\max_j p_j$

Clustering

Εισαγωγή

- Οι αλγόριθμοι συσταδοποίησης (clustering algorithms) ομαδοποιούν τα έγγραφα σε ένα σύνολο από συστάδες / ομάδες (clusters)
- Ο στόχος είναι οι συστάδες να είναι συνεκτικές εσωτερικά
- Επίσης, πρέπει να είναι διαφορετικές μεταξύ τους



Εισαγωγή

- Η συσταδιοποίηση ανήκει στην περιοχή του unsupervised learning
- Δεν χρειάζεται η ανθρώπινη παρέμβαση με οποιοδήποτε τρόπο
- Η τεχνική του flat clustering δημιουργεί ένα σύνολο συστάδων χωρίς ρητή δομή που θα συσχετίσει τις συστάδες μεταξύ τους
- Η τεχνική του hierarchical clustering δημιουργεί μια ιεραρχία συστάδων
- Επίσης μπορεί να γίνει ο διαχωρισμόςσε hard & soft clustering
- Hard clustering. Υπολογίζει μια ισχυρή ανάθεση – κάθε αντικείμενο είναι μέλος μια συστάδας
- Soft clustering. Υπολογίζει μια κατανομή του αντικειμένου σε διάφορες συστάδες

Συσταδοποίηση

- Βασιζόμαστε στην ακόλουθη υπόθεση:
  - Έγγραφα που ανήκουν στην ίδια κλάση έχουν παρόμοια συμπεριφορά
- Αυτό σημαίνει πως αν υπάρχει κάποιο έγγραφο που είναι σχετικό με ένα ερώτημα, πιθανόν να υπάρχουν και άλλα έγγραφα στη συστάδα που να είναι παρόμοια επίσης
- Ο λόγος είναι πως οι αλγόριθμοι συσταδοποίησης τοποθετούν στις ίδιες ομάδες έγγραφα με παρόμοια χαρακτηριστικά

Συσταδοποίηση

- Μας δίνεται ένα σύνολο εγγράφων $D=\{d_1,d_2,\ldots,d_N\}$
- Επίσης, μας δίνεται ένας επιθυμητός αριθμός συστάδων Κ
- Μια συνάρτηησ που αποτιμά το κάθε έγγραφο και το κατατάσσει σε μια συστάδα $\gamma: D \rightarrow \{1,2,\ldots,K\}$
- Στόχος της συνάρτησης είναι να ελαχιστοποιήσει τις αποστάσεις μέσα στις συστάδες και να μεγιστοποιήσει τις αποστάσεις ανάμεσα σε διαφορετικές συστάδες

Συσταδοποίηση

- Η συνάρτηση αρκετές φορές εκφράζεται μέσω της ομοιότητας ή την απόσταση των εγγράφων
- Συνήθως, για τα έγγραφα επιθυμούμε ομοιότητα στα topics ή σε ένα πλήθος διαστάσεων όπως και στο vector space μοντέλο
- Αν θελήσουμε να μετρήσουμε την ομοιότητα για κάποια άλλα χαρακτηριστικά το πιο πιθανό είναι ότι θα χρειαστούμε διαφορετική μοντελοποίηση
- Φυσικά, κατά τον υπολογισμό της ομοιότητας μπορούμε να αφαιρέσουμε τα stop words

Συσταδοποίηση

- Ένα πολύ δύσκολο ζήτημα είναι ο καθορισμός του K
- Πολλές φορές το K είναι μια εκτίμηση
- Άλλες φορές το K εξάγεται από κάποιες τεχνικές
- Παράδειγμα:
  - Για τον K-means αλγόριθμο υπάρχει τεχνική που εξάγει το K
- Αρκετές φορές μπορεί να τεθούν περιορισμοί για την τιμή του K από την ίδια την τεχνική

Συσταδοποίηση

- Αφού επιθυμούμε να βρούμε συστάδες και ομοιότητα ανάμεσα σε έγγραφα, η συσταδοποίηση είναι ένα πρόβλημα αναζήτησης (search problem)
- Όμως, μια brute force προσέγγιση (αναζητούμε και εξετάζουμε όλες τις πιθανές λύσεις) δεν είναι δυνατό να εφαρμοστεί
- Ο λόγος είναι πως αυξάνονται εκθετικά οι τμηματοποιήσεις όσο αυξάνει το πλήθος των εγγράφων
- Για αυτό το λόγο αρκετές τεχνικές παράγουν μια αρχική τμηματοποίηση και στη συνέχεια την ενημερώνουν
- Δυστυχώς, αν η πρώτη τμηματοποίηση δεν είναι καλή, τότε μπορεί να οδηγηθούμε σε τοπικά βέλτιστες λύσεις

Αποτίμηση της Συσταδοποίησης

- Το εσωτερικό κριτήριο (internal criterion) σχετίζεται με την intra-cluster ομοιότητα – πόσο όμοια είναι τα έγγραφα που ανήκουν στην ίδια συστάδα
- Εναλλακτικά, μπορούμε να αποτιμήσουμε την εφαρμογή απ' ευθείας αντί για την ποιότητα της κάθε συστάδας
- Επίσης, το εξωτερικό κριτήριο (external criterion) αποτυπώνει το πόσο καλά ταιριάζουν οι συστάδες που έχουν παραχθεί με ένα σύνολο κλάσεων που έχουν δημιουργήσει οι experts
- Εξωτερικά κριτήρια είναι: purity, normalized mutual information, rand index, F-measure

Αποτίμηση της Συσταδοποίησης

- Purity
  - Κάθε συστάδα ανατίθεται στην κλάση που είναι η πιο συχνή μέσα στη συστάδα
  - Η ακρίβεια της ανάθεσης μετράται ως το πλήθος των σωστών εγγράφων ως προς το N
  - Ο μαθηματικός τύπος είναι:

$$\text{purity}(\Omega, \mathbf{C}) = \frac{1}{N} \sum_{k} \max_{j} |\omega_k \cap c_j|$$

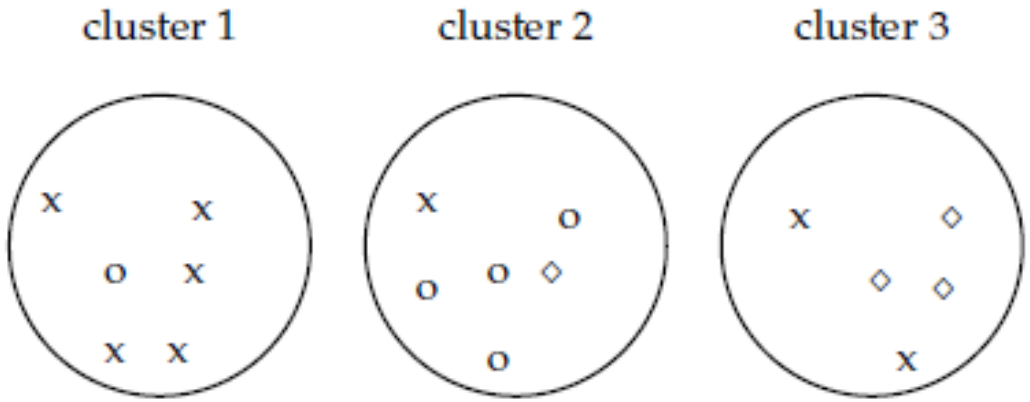
  - Ω είναι το σύνολο των συστάδων και C είναι το σύνολο των κλάσεων
  - Επιθυμούμε μια τιμή κοντά στο 1

Αποτίμηση της Συσταδοποίησης

- Purity

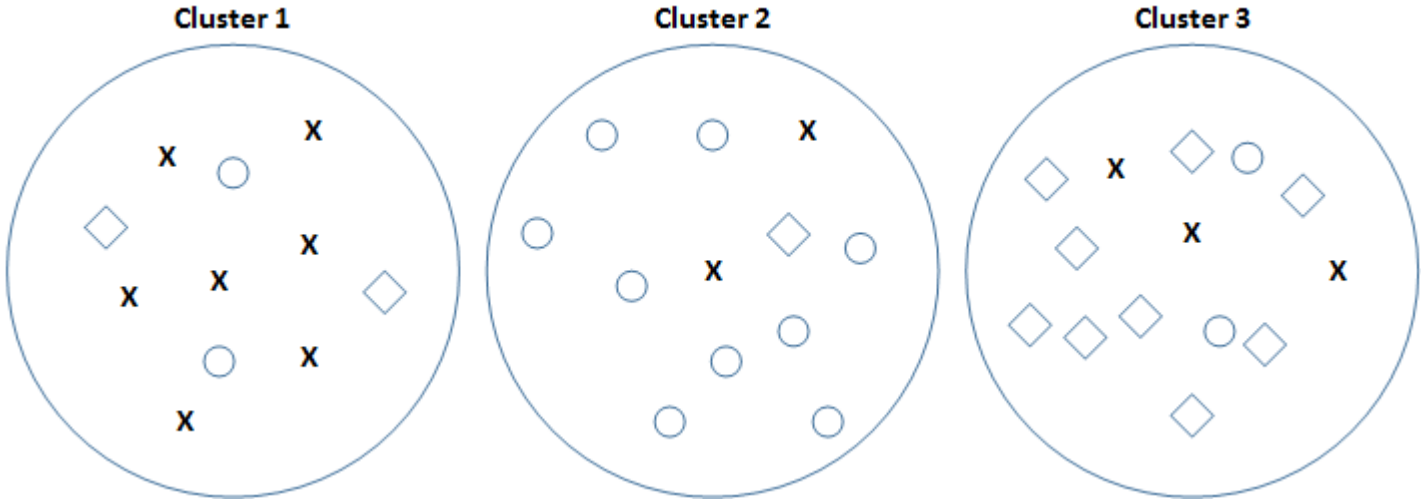
$$\text{purity}(\Omega, \mathbf{C}) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j|$$

  - Παράδειγμα
    - Στην 1η συστάδα πρέπει να μπουν τα X, στη 2η οι κύκλοι και στην 3η οι ρόμβοι
    - Στο σχήμα το purity είναι: 1/17 X (5+4+3) = 0.71



Αποτίμηση της Συσταδοποίησης

- Purity
  - Παράδειγμα
    - Ποιο είναι το purity για το ακόλουθο σχήμα;



    - Απάντηση: 1/37 X (7+9+9) = 0.68

Αποτίμηση της Συσταδοποίησης

- Normalized mutual information
  - Βασίζεται στη μετρική του mutual information που υιοθετείται για feature selection

$$\mathrm{NMI}(\Omega, \mathbf{C}) = \frac{I(\Omega;\mathbf{C})}{[H(\Omega)+H(\mathbf{C})]/2}$$

$$I(\Omega;\mathbf{C}) = \sum_k \sum_j P(\omega_k \cap c_j) \log \frac{P(\omega_k \cap c_j)}{P(\omega_k)P(c_j)}$$

$$= \sum_k \sum_j \frac{|\omega_k \cap c_j|}{N} \log \frac{N|\omega_k \cap c_j|}{|\omega_k||c_j|}$$

  - $P(\omega_k), P(c_j), P(\omega_k \cap c_j)$ είναι οι πιθανότητες έναέγγραφο να ανήκει στη συστάδα ω_k, στην κλάση c_j και στην τομή τους
  - Η είναι η εντροπία: $H(\Omega) = -\sum_k P(\omega_k) \log P(\omega_k)$

$$= -\sum_k \frac{|\omega_k|}{N} \log \frac{|\omega_k|}{N}$$

Αποτίμηση της Συσταδοποίησης

- Rand index
  - ‘Βλέπει’ τη συσταδοποίηση ως μια σειρά αποφάσεων για καθένα από τα $\frac{N(N-1)}{2}$ ζεύγη των εγγράφων
  - Θέλουμε να βάλουμε δύο έγγραφα στη ίδια συστάδα εφόσον είναι όμοια
  - Μια απόφαση TP αναθέτει τα έγγραφα στην ίδια συστάδα
  - Μια απόφαση TN βάζει δύο ανόμοια έγγραφα σε διαφορετικές συστάδες
  - Μια απόφαση FP αναθέτει δύο ανόμοια έγγραφα στην ίδια συστάδα
  - Μια απόφαση FN αναθέτει δύο όμοια έγγραφα σε διαφορετικές συστάδες

Αποτίμηση της Συσταδοποίησης

- Rand index
  - Η μετρική αποτυπώνει το ποσοστό των σωστών αναθέσεων

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}}$$

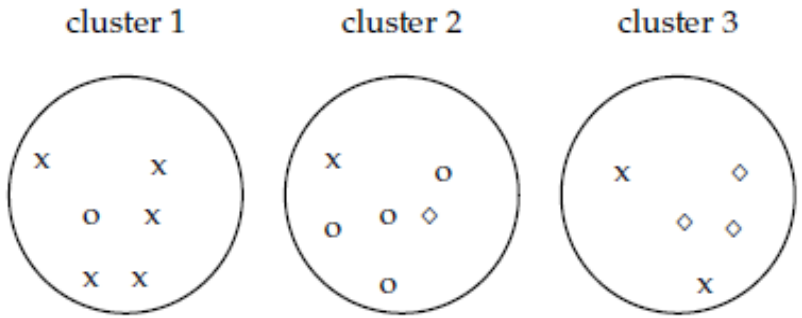
Αποτίμηση της Συσταδοποίησης

- Rand index
  - Οι συστάδες περιέχουν 6, 6, και 5 στοιχεία
  - Ο συνολικός αριθμός των 'θετικών' ζευγών είναι:

$$TP + FP = \binom{6}{2} + \binom{6}{2} + \binom{5}{2} = 40$$

  - Τα ζεύγη Χ, κύκλων και ρόμβων είναι τα TP

$$TP = \binom{5}{2} + \binom{4}{2} + \binom{3}{2} + \binom{2}{2} = 20$$



  - FP=40-20

	Same cluster	Different clusters
Same class	TP = 20	FN = 24
Different classes	FP = 20	TN = 72

  - Οπότε το τελικό αποτέλεσμα είναι: (20+72)/(20+20+24+72)

Αποτίμηση της Συσταδοποίησης

- F-measure
  - Βασίζεται στα precision & recall όπως έχουμε ήδη δει

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad R = \frac{\text{TP}}{\text{TP} + \text{FN}} \qquad F_\beta = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

  - Παράδειγμα:

	Same cluster	Different clusters
Same class	TP = 20	FN = 24
Different classes	FP = 20	TN = 72

  - P=20/40=0.5, R=20/44=0.455, F-measure=0.48 για β=1, 0.456 για β=5

K-means

- Ο αλγόριθμος k-Means είναι από τους πιο γνωστούς αλγορίθμους συσταδοποίησης
- Βασίζεται στην Euclidean απόσταση των εγγράφων από τα κέντρα των συστάδων
- Τα κεντροειδή ορίζονται ως εξής:

$$\vec{\mu}(\omega) = \frac{1}{|\omega|} \sum_{\vec{x} \in \omega} \vec{x}$$

- Ο ορισμός υποθέτει πως τα έγγραφα αποτυπώνονται ως κανονικοποιημένα διανύσματα

K-means

- Μια μετρική που αποτυπώνει πόσο καλά αναπαριστούν τα κεντροειδή τα μέλη μιας συστάδας είναι το residual sum of squares - RSS
- Πρόκειται για το τετράγωνο της απόστασης του κάθε διανύσματος από το κεντροειδές
- Αθροίζουμε αυτές τις αποστάσεις για όλα τα διανύσματα

$$\mathrm{RSS}_k = \sum_{\vec{x} \in \omega_k} |\vec{x} - \vec{\mu}(\omega_k)|^2$$

$$\mathrm{RSS} = \sum_{k=1}^{K} \mathrm{RSS}_k$$

- Στόχος είναι να ελαχιστοποιήσουμε αυτή την ποσότητα

K-means

- Το πρώτο βήμα είναι η επιλογή των αρχικών κεντροειδών
- Αυτά επιλέγονται τυχαία και ονομάζονται seeds
- Στη συνέχεια ο αλγόριθμος μετακινεί τα κεντροειδή ώστε να ελαχιστοποιήσει το RSS
- Αυτό γίνεται επαναληπτικά
- Επαναλαμβάνουμε την εκτέλεση μέχρι να ικανοποιηθεί ένα κριτήριο τερματισμού
- Στη συνέχεια αφού μετακινήσουμε τα κεντροειδή αναθέτουμε ξανά τα έγγραφα στα κεντροειδή και στις συστάδες

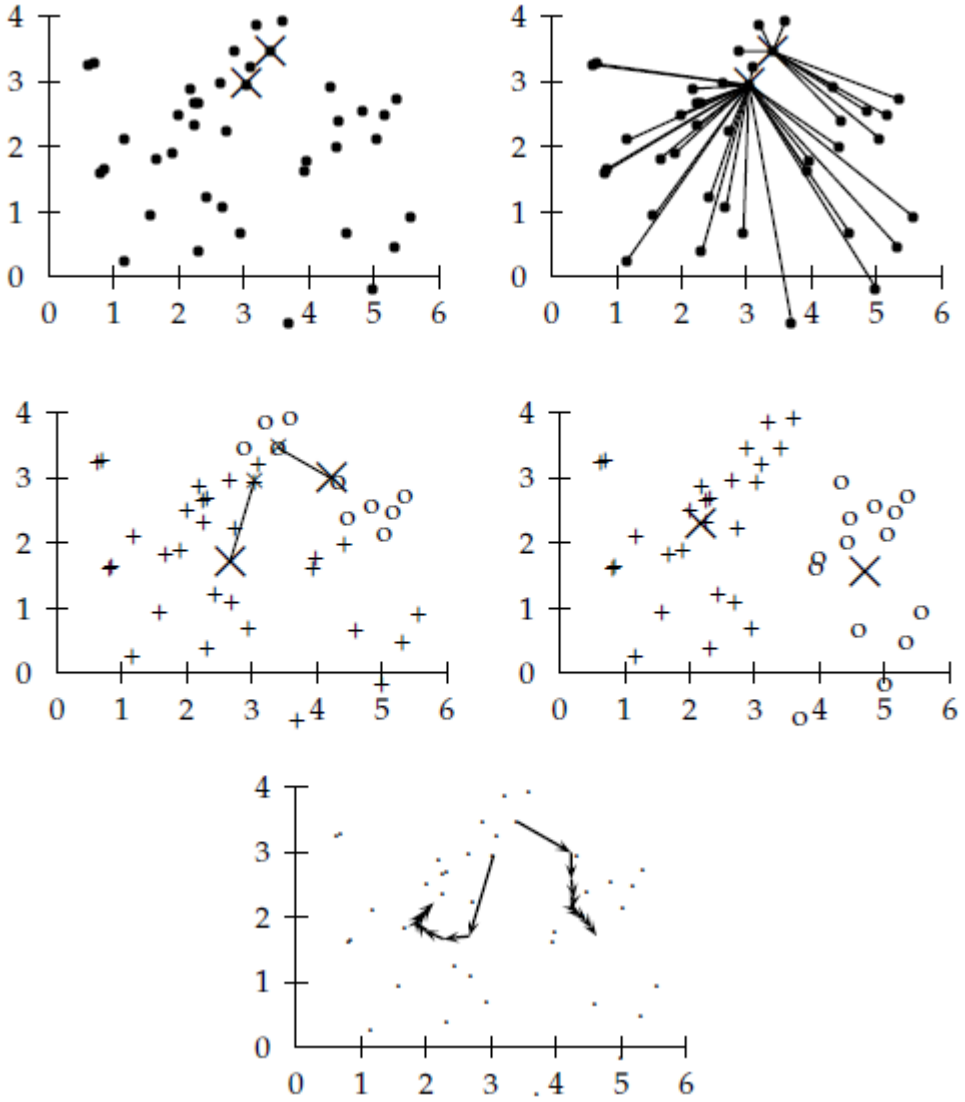
K-MEANS($\{\vec{x}_1, \ldots, \vec{x}_N\}, K$)
1 $(\vec{s}_1, \vec{s}_2, \ldots, \vec{s}_K) \leftarrow$ SELECTRANDOMSEEDS($\{\vec{x}_1, \ldots, \vec{x}_N\}, K$)
2 **for** $k \leftarrow 1$ **to** K
3 **do** $\vec{\mu}_k \leftarrow \vec{s}_k$
4 **while** stopping criterion has not been met
5 **do for** $k \leftarrow 1$ **to** K
6 **do** $\omega_k \leftarrow \{\}$
7 **for** $n \leftarrow 1$ **to** N
8 **do** $j \leftarrow \arg\min_{j'} |\vec{\mu}_{j'} - \vec{x}_n|$
9 $\omega_j \leftarrow \omega_j \cup \{\vec{x}_n\}$ *(reassignment of vectors)*
10 **for** $k \leftarrow 1$ **to** K
11 **do** $\vec{\mu}_k \leftarrow \frac{1}{|\omega_k|} \sum_{\vec{x} \in \omega_k} \vec{x}$ *(recomputation of centroids)*
12 **return** $\{\vec{\mu}_1, \ldots, \vec{\mu}_K\}$

K-means

- Τα κριτήρια τερματισμού μπορεί να είναι τα ακόλουθα:
  - Ένας προκαθορισμένος αριθμός επαναλήψεων
  - Δεν παρατηρείται αλλαγή στην ανάθεση των εγγράφων στις συστάδες
  - Τα κεντροειδή δεν αλλάζουν σε κάθε επανάληψη
  - Το RSS έχει πέσει κάτω από ένα προκαθορισμένο όριο
  - Η μείωση στο RSS έχει πέσει κάτω από ένα προκαθορισμένο όριο

K-MEANS($\{\vec{x}_1, \ldots, \vec{x}_N\}, K$)
1 $(\vec{s}_1, \vec{s}_2, \ldots, \vec{s}_K) \leftarrow$ SELECTRANDOMSEEDS($\{\vec{x}_1, \ldots, \vec{x}_N\}, K$)
2 **for** $k \leftarrow 1$ **to** K
3 **do** $\vec{\mu}_k \leftarrow \vec{s}_k$
4 **while** stopping criterion has not been met
5 **do for** $k \leftarrow 1$ **to** K
6 **do** $\omega_k \leftarrow \{\}$
7 **for** $n \leftarrow 1$ **to** N
8 **do** $j \leftarrow \arg\min_{j'} |\vec{\mu}_{j'} - \vec{x}_n|$
9 $\omega_j \leftarrow \omega_j \cup \{\vec{x}_n\}$ *(reassignment of vectors)*
10 **for** $k \leftarrow 1$ **to** K
11 **do** $\vec{\mu}_k \leftarrow \frac{1}{|\omega_k|} \sum_{\vec{x} \in \omega_k} \vec{x}$ *(recomputation of centroids)*
12 **return** $\{\vec{\mu}_1, \ldots, \vec{\mu}_K\}$

K-means



K-means

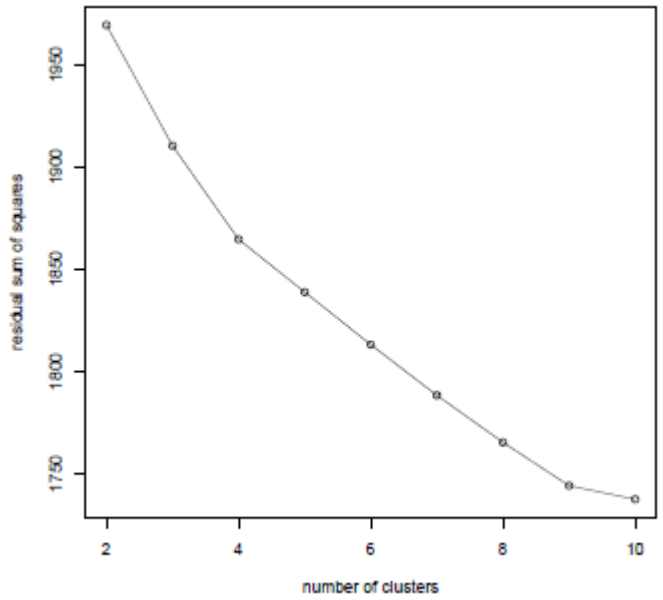
- Επιλογή του K
  - Η τιμή του K παίζει σημαντικό ρόλο στο αποτέλεσμα της διαδικασίας και στην ποιότητα της συσταδοποίησης
  - Μια προσέγγιση θα ήταν να επιλέξουμε το K που ελαχιστοποιεί το RSS
  - Αν συνδέσουμε το K με το RSS τότε το RSS είναι μια μονότονη συνάρτηση του K
  - Όσο το K αυξάνει φτάνοντας το N το RSS θα μειώνεται
  - Μπορούμε να καταλήξουμε στο να έχουμε N συστάδες με μηδενικό σφάλμα

K-means

- Επιλογή του K
  - Αρχικά εφαρμόζουμε i συσταδιοποιήσεις με K συστάδες και υπολογίζουμε το RSS
  - Στη συνέχεια παίρνουμε τις μικρότερες από τις i RSS τιμές
  - Καταγράφουμε την ελάχιστη τιμή
  - Έπειτα, μπορούμε να εξετάσουμε το RSS καθώς αυξάνουμε το K και να παρατηρούμε το σημείο όπου η γραφική παράσταση κάνει μια καμπύλη

K-means

- Επιλογή του K
  - Στο παράδειγμα υπάρχουν δύο τέτοια σημεία: K=4, K=9
  - Σπάνια θα υπάρξει ένα μόνο σημείο για το K
  - Εφαρμόζοντας εξωτερικά κριτήρια επιλέγουμε την τελική τιμή



K-means

- Visualization:
- http://tech.nitoyon.com/en/blog/2013/11/07/k-means/
- http://stanford.edu/class/ee103/visualizations/kmeans/kmeans.html
- https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

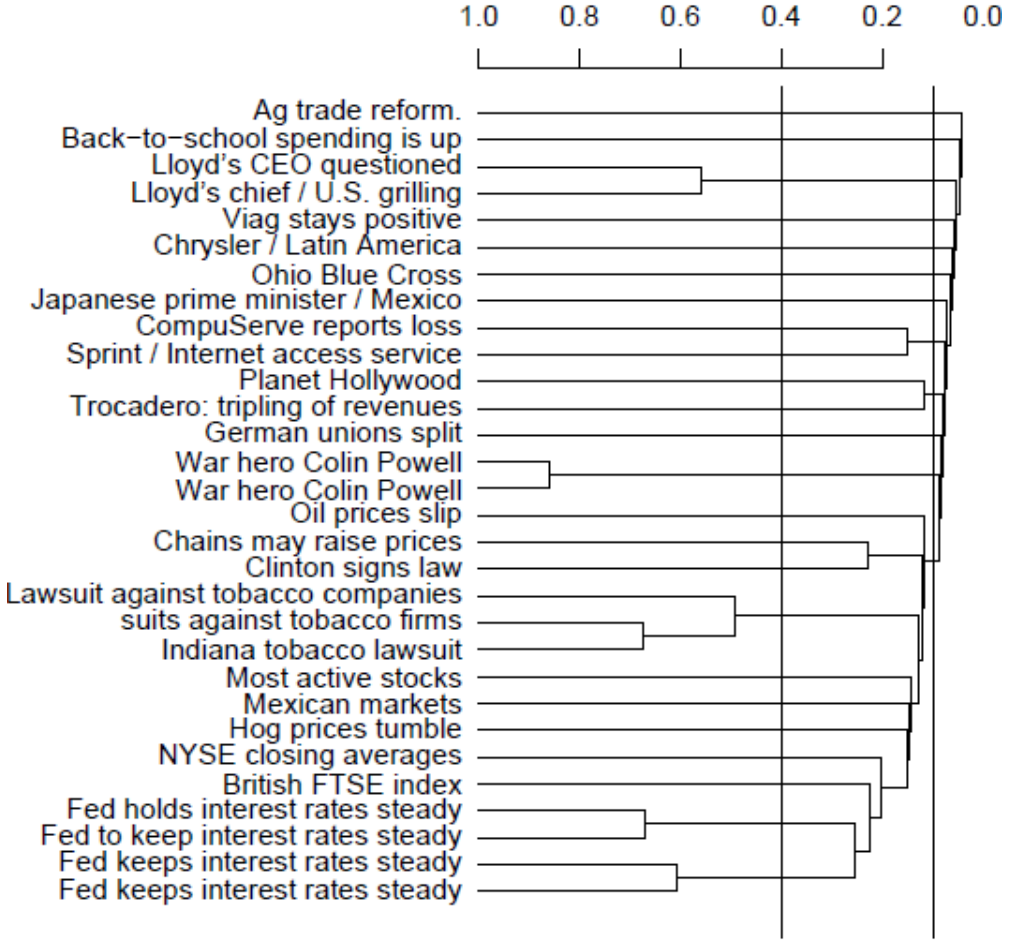
Ιεραρχική Συσταδοποίηση

Εισαγωγή

- Οι αλγόριθμοι ιεραρχικής συσταδοποίησης λειτουργούν είτε από πάνω προς τα κάτω ή από κάτω προς τα πάνω
- Οι bottom up αλγόριθμοι χειρίζονται αρχικά κάθε έγγραφο σαν μια συστάδα και στη συνέχεια προχωρούν σε συγχωνεύσεις
- Συγχωνεύουν ζεύγη συστάδων μέχρι όλα τα έγγραφα να σχηματίσουν μια μεγάλη συστάδα
- Καλούνται hierarchical agglomerative clustering
- Οι top down αλγόριθμοι υιοθετούν μια μέθοδο διαχωρισμού των συστάδων
- Διαχωρίζουν τις συστάδες επαναληπτικά μέχρι κάθε έγγραφο να αποτελεί μια ξεχωριστή συστάδα

Εισαγωγή

- Η ιεραρχική συσταδοποίηση μπορεί να απεικονιστεί με ένα δένδρο



Εισαγωγή

- Προφανώς η ιεραρχική διαδικασία θα πρέπει να σταματήσει σε κάποιο σημείο
- Μπορούν να υιοθετηθούν ένα πλήθος από κριτήρια:
  - Σταματάμε σε ένα προκαθορισμένο επίπεδο ομοιότητας (π.χ. μπορούμε να σταματήσουμε στο 0.5)
  - Σταματάμε όταν το κενό ανάμεσα σε δύο συνεχόμενες ομοιότητες είναι μεγάλο
  - Μπορούμε να ορίσουμε εκ των προτέρων το πλήθος των συστάδων και να σταματήσουμε όταν το φτάσουμε

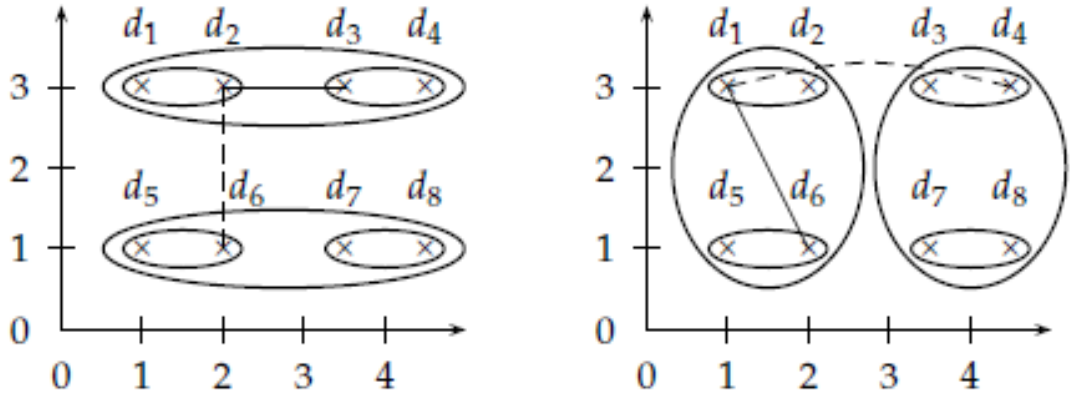
Εισαγωγή

- Ένας απλός αλγόριθμος είναι ο ακόλουθος:
  - Αρχικά υπολογίζουμε ένα NXN πίνακα ομοιότητας
  - Εκτελούμε N-1 βήματα συγχωνεύοντας τις πιο όμοιες συστάδες
  - Σε κάθε επανάληψη, συγχωνεύουμε τις δύο πιο όμοιες συστάδες και ενημερώνουμε τις στήλες

```
SIMPLEHAC(d_1, ..., d_N)
 1  for n ← 1 to N
 2  do for i ← 1 to N
 3       do C[n][i] ← SIM(d_n, d_i)
 4       I[n] ← 1  (keeps track of active clusters)
 5  A ← []  (assembles clustering as a sequence of merges)
 6  for k ← 1 to N − 1
 7  do ⟨i, m⟩ ← arg max_{⟨i,m⟩: i≠m ∧ I[i]=1 ∧ I[m]=1} C[i][m]
 8       A.APPEND(⟨i, m⟩)  (store merge)
 9       for j ← 1 to N
10       do C[i][j] ← SIM(i, m, j)
11            C[j][i] ← SIM(i, m, j)
12       I[m] ← 0  (deactivate cluster)
13  return A
```

Single link and Complete link Clustering

- Στην τεχνική single-link, η ομοιότητα δύο συστάδων είναι η ομοιότητα των δύο πιο όμοιων στοιχείων τους
  - Δίνουμε σημασία μόνο στην περιοχή όπου οι συστάδες μοιάζουν περισσότερο
- Στην τεχνική complete-link η ομοιότητα δύο συστάδων είναι η ομοιότητα των δύο πιο ανόμοιων στοιχείων τους
  - Ολόκληρη η δομή των συστάδων επηρεάζει τη συγχώνευση τους



Single link and Complete link Clustering

```
SINGLELINKCLUSTERING(d_1, ..., d_N)
 1  for n ← 1 to N
 2  do for i ← 1 to N
 3     do C[n][i].sim ← SIM(d_n, d_i)
 4        C[n][i].index ← i
 5     I[n] ← n
 6     NBM[n] ← arg max_{X∈{C[n][i]:n≠i}} X.sim
 7  A ← []
 8  for n ← 1 to N − 1
 9  do i_1 ← arg max_{{i:I[i]=i}} NBM[i].sim
10     i_2 ← I[NBM[i_1].index]
11     A.APPEND(⟨i_1, i_2⟩)
12     for i ← 1 to N
13     do if I[i] = i ∧ i ≠ i_1 ∧ i ≠ i_2
14           then C[i_1][i].sim ← C[i][i_1].sim ← max(C[i_1][i].sim, C[i_2][i].sim)
15        if I[i] = i_2
16           then I[i] ← i_1
17     NBM[i_1] ← arg max_{X∈{C[i_1][i]:I[i]=i∧i≠i_1}} X.sim
18  return A
```

Centroid Clustering

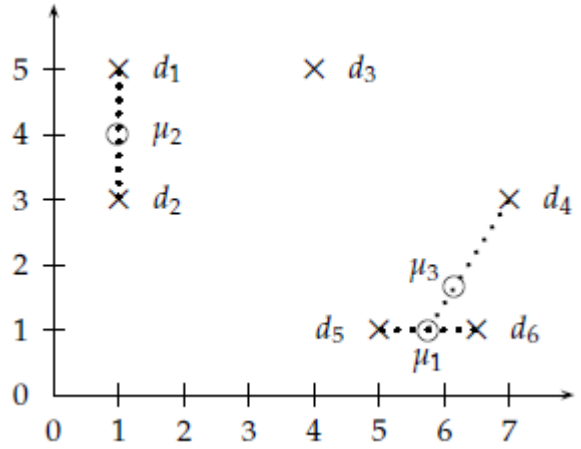
- Στην τεχνική αυτή η ομοιότητα δύο συστάδων ορίζεται ως η ομοιότητα των κεντροειδών τους

$$\begin{aligned} \text{SIM-CENT}(\omega_i, \omega_j) &= \vec{\mu}(\omega_i) \cdot \vec{\mu}(\omega_j) \\ &= \left(\frac{1}{N_i} \sum_{d_m \in \omega_i} \vec{d}_m\right) \cdot \left(\frac{1}{N_j} \sum_{d_n \in \omega_j} \vec{d}_n\right) \\ &= \frac{1}{N_i N_j} \sum_{d_m \in \omega_i} \sum_{d_n \in \omega_j} \vec{d}_m \cdot \vec{d}_n \end{aligned}$$

- Η ομοιότητα των συστάδων είναι η μέση ομοιότητα όλων των ζευγών εγγράφων από διαφορετικές συστάδες

Centroid Clustering

- Στην εικόνα φαίνονται τα 3 βήματα της τεχνικής
- Οι πρώτες δύο επαναλήψεις σχηματίζουν τις συστάδες {d5,d6} με κεντροειδές μ1 και {d1,d2} με κεντροειδές μ2
- Στην τρίτη επανάληψη η μεγαλύτερη ομοιότητα είναι ανάμεσα στο μ1 και στο d4
- Οπότε δημιουργείται η συστάδα {d5,d6,d4} με κεντροειδές το μ3



Web Search

Εισαγωγή

- Το Web βασίζεται στη λογική μιας client-server αρχιτεκτονικής
- Ο εξυπηρετητής επικοινωνεί με τον πελάτη μέσω ενός πρωτοκόλλου
- Το πρωτόκολλο αυτό είναι το http
- Πρόκειται για ένα lightweight πρωτόκολλο που δίνει τη δυνατότητα στα πακέτα να 'κουβαλήσουν' μια ποικιλία από περιεχόμενο
- Το περιεχόμενο είναι κωδικοποιημένο στη γλώσσα HTML
- Ο πελάτης (συνήθως ένας φυλλομετρητής – browser) μεταφράζει την εισαρχόμενη html ενώ μπορεί να αγνοήσει ότι δεν καταλαβαίνει

Εισαγωγή

- Η βασική λειτουργία είναι η ακόλουθη:
  - Ο πελάτης στέλνει ένα http αίτημα σε ένα εξυπηρετητή
  - Ο φυλλομετρητής καθορίζει ένα URL – Unified Resource Locator
  - Από το domain εντοπίζεται το περιεχόμενο ανακτάται και στέλνεται στον πελάτη
  - Ο πελάτης μεταφράζει την εισερχόμενη html
  - Η html περιλαμβάνει συνδέσμους και περιεχόμενο
  - Επίσης περιλαμβάνει κανόνες μορφοποίησης

Εισαγωγή

- Ο μεγάλος όγκος πληροφοριών στο Web μπορεί ανακαλυφθεί και να υιοθετηθεί από τους χρήστες
- Δύο είναι οι μέθοδοι:
  - Η χρήση των μηχανών αναζήτησης όπου εισάγουμε ερωτήματα στη μορφή κειμένων
    - Η αναζήτηση υποστηρίζεται από τα inverted indexes
    - Η αναζήτηση υποστηρίζεται από μηχανισμούς για το ranking
  - Ταξινομίες που έχουν εμπλουτίζονται με ιστοσελίδες κατηγοριοποιημένες (π.χ. Yahoo)
    - Οι χρήστες πλοηγούνται σε διάφορες κατηγορίες
    - Οι κατηγορίες τοποθετούνται ιεραρχικά

Εισαγωγή

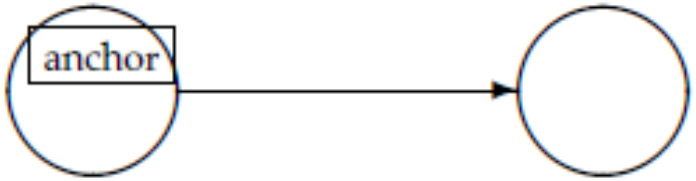
- Ο βασικός λόγος που επηρέασε την έκρηξη του όγκου πληροφορίας στο Web είναι η μη κεντρικοποιημένη παραγωγή της πληροφορίας
- Οι συγγραφείς των σελίδων παράγουν περιεχόμενο σε πολλές γλώσσες
- Η πληθώρα των πηγών πληροφορίας απαιτεί διαφορετικές τεχνικές stemming και γλωσσολογική επεξεργασία
- Οι ιστοσελίδες χαρακτηρίζονται από ετερογένεια

Εισαγωγή

- Στατικές σελίδες
  - Περιλαμβάνουν στατικό περιεχόμενο
- Δυναμικές σελίδες
  - Περιλαμβάνουν κώδικα ο οποίος παράγει περιεχόμενο

Web Graph

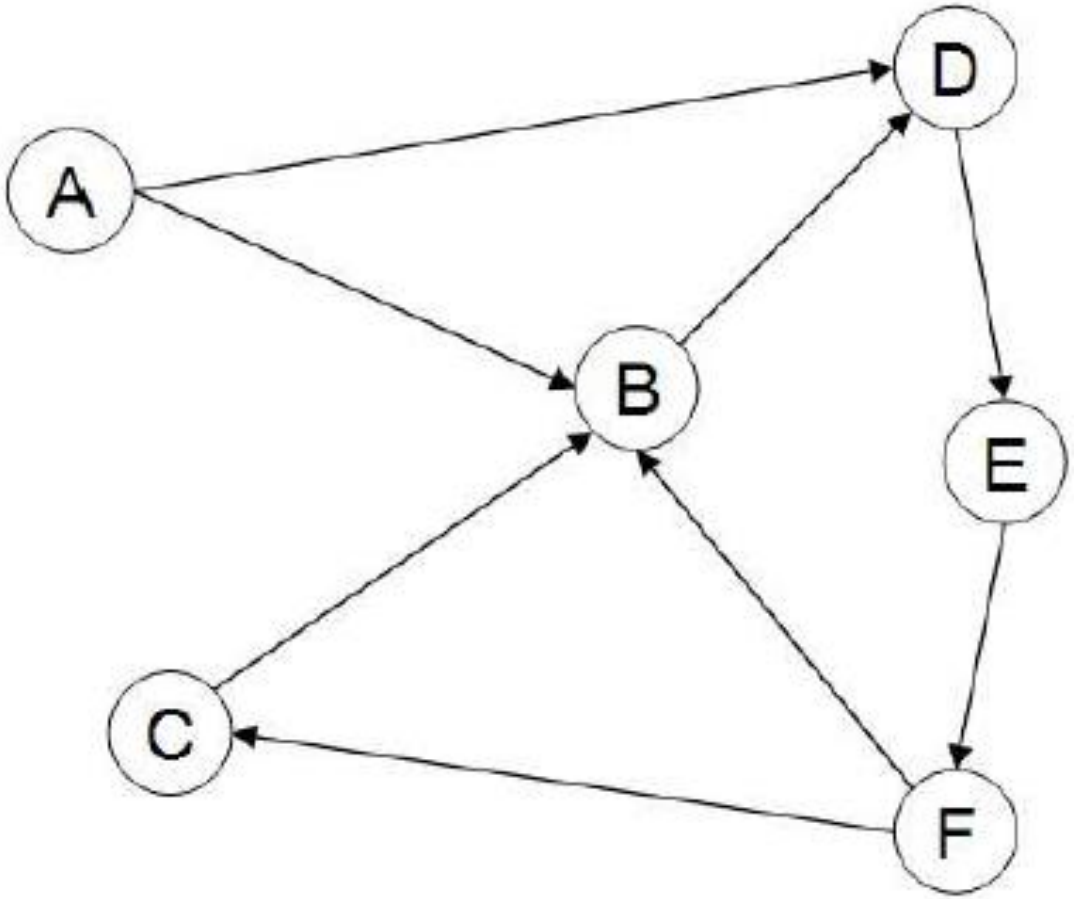
- Μπορούμε να δούμε τις στατικές σελίδες μαζί με τους υπεσυνδέσμους τους σαν ένα κατευθυνόμενο γράφο
- Κάθε σελίδα είναι ένας κόμβος και κάθε υπερσύνδεσμος είναι μια ακμή



Web Graph

- Συνήθως τα κείμενα στα οποία αποτυπώνονται οι σύνδεσμοι τοποθετούνται στο tag <a> και στην ιδιότητα href
- Το κείμενο αυτό αναφέρεται ως anchor text
- Ο γράφος δεν είναι strongly connected
  - Υπάρχουν ζεύγη που συνδέονται αλλά και ζεύγη στα οποία δεν μπορούμε να μετακινηθούμε σε μια σελίδα
- Τα in-links είναι σύνδεσμοι προς μια σελίδα
- Τα out-links είναι οι σύνδεσμοι που φεύγουν από μια σελίδα
- Σε σειρά μελετών τα in-links κινούνται ανάμεσα στο 8 με 15

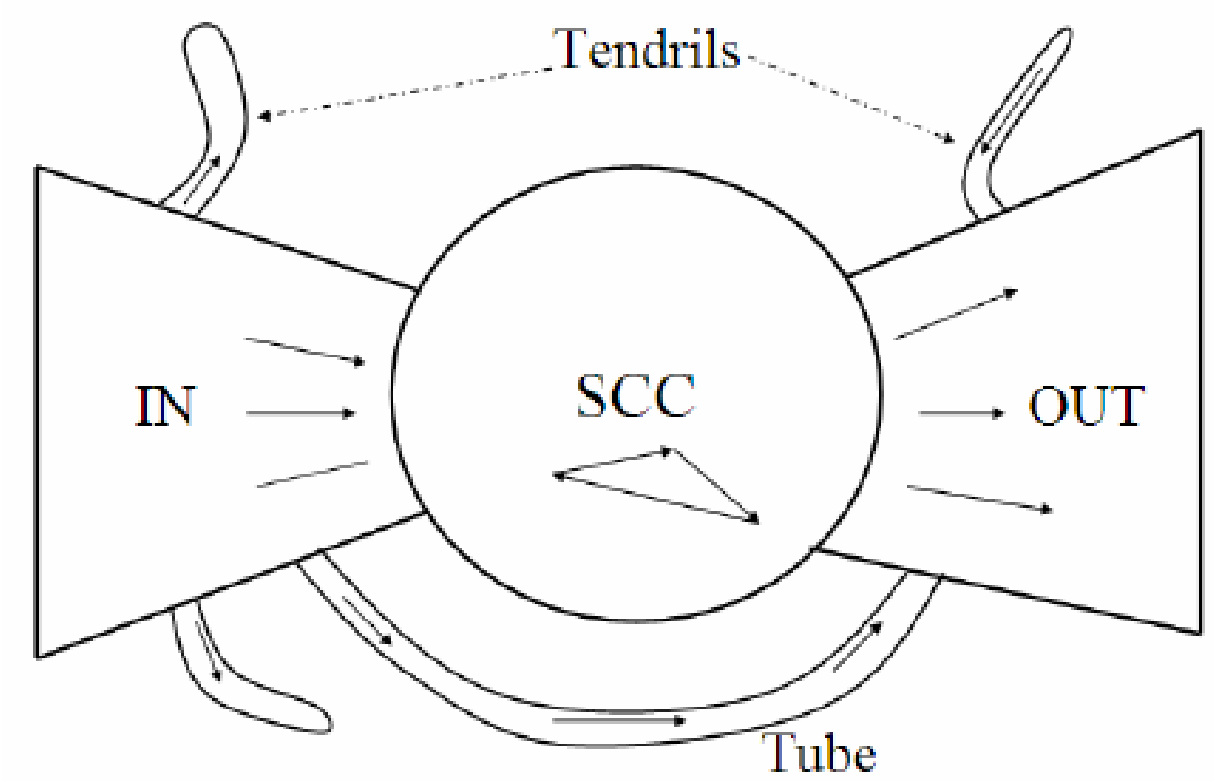
Web Graph



Web Graph

- Οι σύνδεσμοι δεν κατανέμονται τυχαία στις σελίδες
- Γενικά, η κατανομή των συνδέσμων ακολουθεί τον power law
- Ο συνολικός αριθμός των σελίδων με in-degree i είναι ανάλογος με το $1/i^{\alpha}$
- Το α ορίζεται και στο νόμο του Zipf (α=1)
- Υπάρχουν τρια (3) είδη σελίδων: IN, OUT, SCC (strongly connected component)
- Οι χρήστες μπορούν να περάσουν από μια IN σελίδα σε μια SCC μέσω των συνδέσμων
- Μπορούμε να περάσουμε από μια SCC σελίδα σε οποιαδήποτε OUT σελίδα
- Επίσης, οι χρήστες μπορούν να μεταβούν από μια SCC σε οποιαδήποτε SCC
- Όμως δεν μπορούμε να πάμε από μια SCC σε μια IN ή από μια OUT σε μια SCC

Web Graph



Web Graph

- Μελέτες δείχνουν ότι οι IN & OUT είναι περίπου ίσες σε μέγεθος ενώ οι SCCs είναι μεγαλύτερες
- Οι περισσότερες σελίδες ανήκουν σε αυτές τις τρεις κατηγορίες
- Οι υπόλοιπες ανήκουν στην κατηγορία tubes που είναι μικρά σύνολα έξω από τις SCCs και οδηγούν από τις IN απ' ευθείας στις OUT ή στις tendrils που δεν οδηγούν πουθενά από τις IN ή σε κάποια OUT