

# Intelligenza Artificiale

Anno accademico  
2008-2009

*Information Retrieval: Text Categorization*

# Una definizione formale

Sia  $\mathbf{D}$  il dominio dei documenti

Sia  $\mathbf{C} = \{c_1, \dots, c_{|\mathbf{C}|}\}$  un insieme di categorie predefinite

Il task della classificazione di testi e' di approssimare la  
funzione target sconosciuta

$$\Phi : \mathbf{D} \times \mathbf{C} \rightarrow \{T, F\}$$

Con una funzione  $\Phi : \mathbf{D} \times \mathbf{C} \rightarrow \{T, F\}$  chiamata **classificatore**  
tale che  $\Phi$  coincida il piu' possibile con  $\Phi^l$

# Single Label, Multi Label e Binary

- Al Task della Classificazione si possono aggiungere dei vincoli: per un dato  $k$ , esattamente  $k$ , ( $0 \leq k, 0 \geq k$ ) elementi di  $\mathbf{C}$  vengano assegnati ad un documento  $d_j$
- Single Label: soltanto una categoria puo' essere assegnata a un documento ( $k=1$ )
- Multi Label: 0 o + categorie possono essere assegnate a un documento
- Binary: un documento o appartiene a  $c_i$  o appartiene a  $\neg c_i$

# Single Label, Multi Label e Binary

- Un algoritmo per binary puo' essere usato anche per multilabel: si trasforma il problema di classificare sotto le categorie  $\{c_1, \dots, c_{|C|}\}$  in  $|C|$
- classificazione binaria sotto le categorie  $\{c_i, \neg c_i\}$
- Un **classificatore** per una categoria  $c_i$  e' una funzione  $\phi_i : D \rightarrow \{T, F\}$  che approssima la **funzione target sconosciuta**

$$\phi_i : D \rightarrow \{T, F\}$$

# Approccio Knowledge Engineering

- Molto popolare negli anni '80
- La creazione di un classificatore di testi automatico consiste nella creazione di un **sistema esperto** capace di prendere decisioni di classificazione.
- Un tale sistema esperto e' un insieme di regole (definite manualmente) del tipo

***if***<DNF Formula> ***then*** < $c_i$ > ***else*** < $\tau c_i$ >

(DNF=Forma Normale Disgiuntiva)

# Approccio Knowledge Engineering

- Le regole venivano definite da un ingegnere della conoscenza con l'aiuto di un **esperto del dominio**
- L'esempio piu' famoso e' il sistema **CONSTRUE** progettato dal Carnegie Group per l'agenzia di stampa *Reuters*
- SVANTAGGI:
  1. Se si deve modificare l'insieme di categorie e' di nuovo necessario l'aiuto dell'esperto di dominio
  2. Se si vuole cambiare dominio del classificatore si deve chiamare un nuovo esperto di dominio e riniziare il lavoro da capo

# Approccio Machine Learning

- Si sviluppa a partire dai primi anni '90
- Un processo induttivo costruisce **automaticamente** un classificatore per una categoria  $C_i$
- Dalle caratteristiche osservate il processo induttivo decide quale caratteristiche deve avere un nuovo documento per essere classificato sotto  $C_i$

# Approccio Machine Learning

- Non si costruisce un classificatore, ma un costruttore di classificatori che va bene per ogni dominio
- **RISORSA CHIAVE:** Documenti classificati manualmente (spesso sono già disponibili ma anche se non sono disponibili...)
- E' piu' facile classificare documenti manualmente piuttosto che stabilire delle regole per la classificazione dei documenti perché è spesso più facile caratterizzare un concetto estensionalmente piuttosto che intensionalmente



# Approccio Machine Learning

- Il corpo iniziale dei documenti già classificati viene diviso in tre insiemi:
  1. **Training Set:** Insieme dei documenti che vengono usati per costruire il classificatore
  2. **Validation Set:** Una volta costruito il classificatore potrebbe essere necessario aggiustare dei parametri. Per valutare il giusto valore da assegnare ai parametri si fanno test su questo insieme
  3. **Test Set:** Usato per testare l'**efficacia** del classificatore
- I tre insiemi devono essere assolutamente disgiunti

# Approccio Machine Learning

La costruzione di un classificatore si articola in tre fasi:

1. Indicizzazione dei documenti e riduzione dimensionale
2. Induzione del classificatore
3. Valutazione dell'efficacia del classificatore

# INDICIZZAZIONE DEI DOCUMENTI

- I documenti non possono essere interpretati direttamente da un classificatore
- Per questo si applica una procedura di indicizzazione che mappa un documento in una **rappresentazione compatta del suo contenuto**
- Un documento  $d_j$  viene rappresentato come un vettore di pesi

$$d_j = \langle w_{1j}, \dots, w_{|T|j} \rangle$$

dove:

$T$  e' l'insieme dei **termini**

$0 < w_{kj} < 1$  rappresenta quanto il termine  $t_k$  contribuisce alla semantica di  $d_j$

# COS'E' UN TERMINE?

- **Bag of word:** Un termine è una **parola**.  $T$  è l'insieme di tutte le parole che occorrono in tutti i documenti del Training Set ( $Tr$ )
- Molti autori hanno provato a usare **frasi** (piuttosto che parole) come termini, ma i risultati non sono stati soddisfacenti. Nuovi approcci all'indicizzazione con frase sembrano essere efficaci. L'ultima parola deve ancora essere detta.

# Prima di indicizzare...

- **Rimozione delle stop word** (articoli, preposizioni, congiunzioni ecc...): viene quasi sempre effettuata
- **Stemming** (raggruppare le parole per la loro radice morfologica): ci sono un po' di controversie. Per adesso la tendenza e' quella di adottarlo in quanto riduce:
  1. Lo spazio dei termini
  2. Il livello di dipendenza stocastica tra i termini

# CLASSIFICATORI BASATI SU ESEMPI

- IDEA: Non si costruisce una rappresentazione della categoria, ma si confida sui documenti del training set che sono piu' vicini al documento che vogliamo classificare.
- Il primo metodo Example-Based introdotto per la Classificazione di Testi e' del 92 [Creecy et al. 1992; Masand et al. 1992]
- Noi mostreremo un algoritmo implementato da Yang [1994]:  $k$ -NN

# CLASSIFICATORI BASATI SU

## ESEMPI: $k$ -NN (k nearest neighbours)

- IDEA: Per decidere se classificare  $d_j$  sotto  $c_i$   $k$ -NN guarda se i  $k$  documenti del training set piu' simili a  $d_j$  sono stati classificati sotto  $c_i$ . Se una parte grande abbastanza e' stata classificata sotto  $c_i$  allora il documento viene classificato in  $c_i$  altrimenti no .
- La similarita' tra documenti e' calcolata con una qualche funzione  $RSV(d_i, d_z)$ . Nell'implementazione di Yang si cerca di dare piu' peso ai documenti piu' vicini.

$$CSV_i(d_j) = \sum_{d_z \in \text{Trk}(d_j)} RSV(d_i, d_z) [\Phi^l(d_z, c_i)]$$

$\text{Trk}(d_j)$  = i  $k$  documenti del training set piu' simili a  $d_j$  secondo la funzione  $RSV$

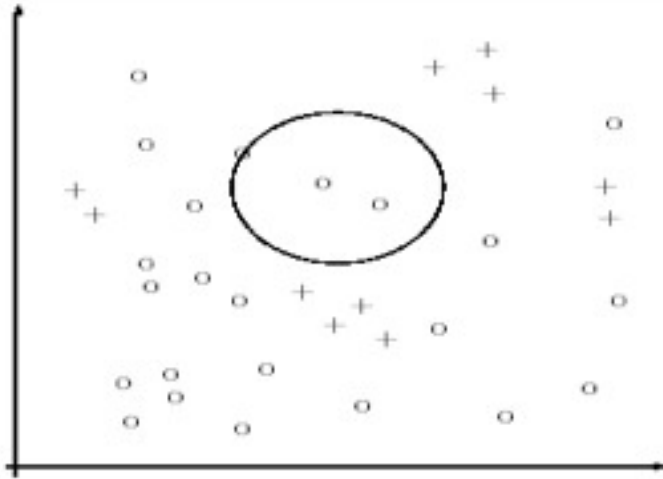
$[x]$  = 1 se  $x$  e' TRUE, 0 se  $x$  e' FALSE

# CLASSIFICATORI BASATI SU ESEMPI: $k$ -NN (k nearest neighbours)

- La soglia  $k$  viene determinata empiricamente attraverso test sul validation set. E' stato dimostrato che aumentando di molto  $k$  non diminuiscono significativamente le performance
- VANTAGGI:  $k$ -NN, diversamente dai classificatori lineari non suddivide lo spazio dei documenti linearmente. Quindi risulta essere piu' "locale" [\[11\]](#)
- SVANTAGGI: Inefficienza a tempo di classificazione:  $k$ -NN deve calcolare la similarita' di tutti i documenti del training set con il documento da classificare
- E' conveniente utilizzarlo per document-pivoted categorization: calcolare la somiglianza dei training document puo' essere fatto una volta per tutte le categorie.



# CLASSIFICATORI LINEARI vs BASATI SU ESEMPI

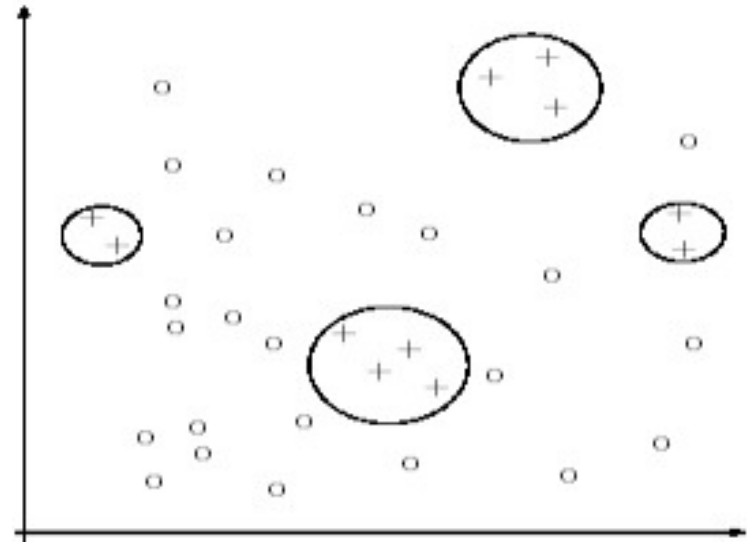


*Le croci e i cerchi rappresentano gli esempi positivi e negativi*

*I cerchi sono l'area di influenza dei classificatori*

*Per facilitare la comprensione, la similarità tra documenti è vista come **distanza Euclidea** piuttosto che come **dot product***

*Sulla sinistra è mostrato il comportamento di un classificatore costruito con Rocchio*  
*In basso un classificatore costruito con k-nn*



# VALUTARE UN CLASSIFICATORE DI TESTI

- La valutazione dei classificatori di testi e' effettuata **sperimentalmente** piuttosto che **analiticamente**.
- La Classificazione di Testi **non puo' essere formalizzata** (a causa della sua natura soggettiva) e quindi non puo' essere valutata analiticamente
- La valutazione sperimentale di un classificatore solitamente misura la sua **efficacia**: l'abilita' di prendere la giusta decisione di classificazione

# MISURE DELL'EFFICACIA DI CLASSIFICAZIONE DI TESTI

●  $\pi_i$  [*Precision wrt  $c_i$* ] =

$$P(\Phi^I(d_{x'}, c_i) = T | \Phi(d_{x'}, c_i) = T)$$

indica il grado di **Correttezza** del classificatore rispetto alla categoria  $c_i$

●  $\rho_i$  [*Recall wrt  $c_i$* ] =

$$P(\Phi(d_{x'}, c_i) = T | \Phi^I(d_{x'}, c_i) = T)$$

indica il grado di **Completezza** del classificatore rispetto alla categoria  $c_i$

CALCOLO DI  $\pi_i = P(\Phi^l(d_{x'}, c_i) = T \mid \Phi(d_{x'}, c_i) = T)$  E

$$\rho_i = P(\Phi(d_{x'}, c_i) = T \mid \Phi^l(d_{x'}, c_i) = T)$$

*Le probabilita' vengono stimate sui risultati del classificatore su un test set*

$$\pi_i = TP_i / (TP_i + FP_i)$$

$$\rho_i = TP_i / (TP_i + FN_i)$$

$TP_i$  = Veri Positivi = #documenti classificati correttamente sotto  $c_i$

$FP_i$  = Falsi Positivi = #documenti classificati incorrettamente sotto  $c_i$

$VN_i$  = Veri Negativi = #documenti non classificati correttamente sotto  $c_i$

$FN_i$  = Falsi Negativi = #documenti non classificati incorrettamente sotto  $c_i$

Category $c_i$		expert judgments	
		YES	NO
classifier judgments	YES	$TP_i$	$FP_i$
	NO	$FN_i$	$TN_i$

# MISURE DELL'EFFICACIA DI CLASSIFICAZIONE DI TESTI

- $\pi_i$  e  $\rho_i$  sono misure di **efficacia** relative alla categoria  $c_i$ . Vogliamo definire l'efficacia di un Classificatore Globalmente.

$\pi$  global precision e  $\rho$  global recall

- Si possono calcolare in due metodi distinti:

- Microaveraging

$$\pi^\mu = \sum_{i=1 \dots |C|} TP_i / \sum_{i=1 \dots |C|} (TP_i + FP_i)$$

$$\rho^\mu = \sum_{i=1 \dots |C|} TP_i / \sum_{i=1 \dots |C|} (TP_i + FN_i)$$

- Macroaveraging

$$\pi^M = \sum_{i=1 \dots |C|} \pi_i / |C|$$

$$\rho^M = \sum_{i=1 \dots |C|} \rho_i / |C|$$

# COMBINARE MISURE DI EFFICACIA

- Le misure  $\pi$  e  $\rho$  prese singolarmente non bastano per esprimere l'efficacia:
  - Il classificatore che classifica tutti i documenti sotto  $c_i$  ha  $\rho = 1$  (non ci sono falsi negativi)
  - Il classificatore che classifica tutti i documenti sotto  $\neg c_i$  ha  $\pi = 1$  (non ci sono falsi positivi)
- $\pi$  e  $\rho$  sono inversamente proporzionali, per valutare l'efficacia di un classificatore si deve trovare la giusta combinazione di queste due misure: anche per questo scopo sono state elaborate numerose funzioni di combinazione

# CONCLUSIONI

- Dai primi anni 90 ad oggi **l'efficacia** dei classificatori di testo **e' aumentata** notevolmente grazie all'impiego di metodi di Machine Learning
- La classificazione automatica di testi e' divenuta un'area molto interessante. A causa di molte ragioni:
  1. I suoi domini di applicazioni sono numerosi e importanti e dato l'aumento di documenti di testo in digitale sono destinati ad aumentare considerevolmente
  2. E' indispensabile in molte applicazioni in cui l'elevato numero di documenti e il breve tempo di risposta richiesto dall'applicazioni, rendono l'alternativa manuale impossibile
  3. Puo' aumentare la produttivita' di un classificatore umano
  4. Si sono raggiunti livelli di efficacia paragonabili a quelli di un classificatore umano