- Seminario di ELN – A.A. 2002/2003

SONIA: A Service for Organizing Networked Information Autonomously

Stud. Davide D'Alessandro
Prof. Amedeo Cappelli

Introduzione

- Crescita esponenziale dell'informazione on-line e inadeguatezza degli attuali sistemi di recupero dell'informazione (motori di ricerca, gerarchie…)
- SONIA: sistema per la navigazione in informazioni organizzate per argomento che combina l'approccio query-based e quello tassonomico
- Uso di varie tecniche di machine learning (feature selection, clustering, classificazione…) per una categorizzazione dinamica dei documenti

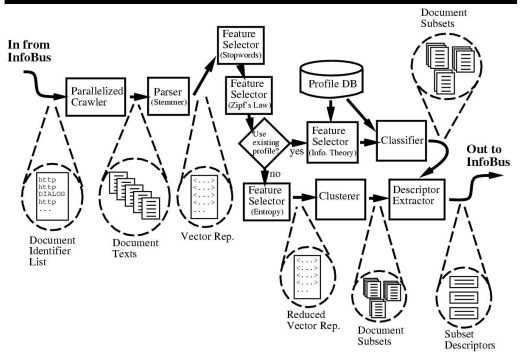
Breve descrizione del sistema

- Prende i risultati delle queries e automaticamente estrae, analizza e organizza i documenti in categorie
- Può salvare una data organizzazione in profili utente utilizzabili per classificazioni future
- Estrae i termini rilevanti dai documenti con metodi statistici di clustering e determina i potenziali argomenti di una collezione di documenti (Cluster Hypothesis)
- Usa classificatori Bayesiani per catalogare nuovi documenti in uno schema di categorizzazione esistente

Un sistema modulare

- Sistema formato da vari moduli
  - Recupero di documenti (Parallelized Crawler)
  - Parser e stemmer
  - Feature selection
    - Stopwords
    - Zipf's law
    - Entropia (se non esiste un profilo utente)
    - Teoria dell'Informazione (se esiste un profilo utente)
  - Classificazione
  - Clustering
  - Descriptor extraction

Schema del sistema



Rappresentazione dei documenti

- Ogni documento è un vettore numerico o booleano
- Ogni dimensione del vettore è un termine distinto
- Ogni componente numerica rappresenta il peso del termine corrispondente all'interno del documento

Computing is not about computers any more. It is about living.

Sample document 1.

To live is to compute!

Sample document 2.

Term	Vector for document 1	Vector for document 2
about	2	0
any	1	0
compute	0	1
computers	1	0
computing	1	0
is	2	1
it	1	0
live	0	1
living	1	0
more	1	0
not	1	0
to	0	2

Word stemming

- Per ridurre ogni termine ad una forma base
- I vettori sono modificati di conseguenza
- In alcuni casi risulta utile (es. "comput"); in altri risulta addirittura controintuitivo e inutile (es. "i")

Comput i not about comput ani more. It i about live.

Stemmed version of sample document 1.

To live i to comput!

Stemmed version of sample document 2.

Stem	Vector for document 1	Vector for document 2
about	2	0
any	1	0
comput	2	1
i	2	1
it	1	0
live	1	1
more	1	0
not	1	0
to	0	2

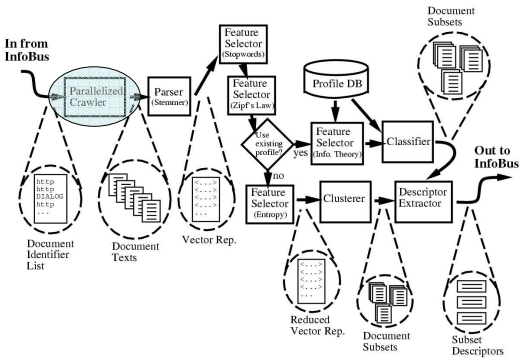
Multi – words

- Es. "President Clinton" o "personal computer"
- Si trovano guardando la frequenza con cui appaiono determinate sequenze di parole
- Non sempre è utile prenderle in considerazione
  - Se il nostro modello già riesce a catturare le dipendenze probabilistiche tra le parole

Vettori basati sulla frequenza

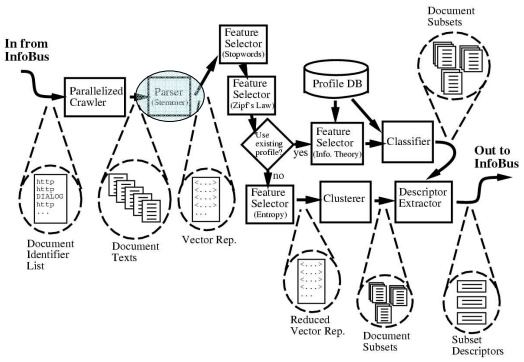
- Vettori “pesati” con una funzione di frequenza
- Funzioni più usate (α = # occorrenze di un termine)
  - $f(\alpha) = \log(\alpha+1)$ (usata per il recupero di documenti)
  - $f(\alpha) = \sqrt{\alpha}$ (usata per il clustering di documenti)
  - $TFIDF(\alpha) = \alpha \cdot IDF(t)$
    - $IDF(t) = \log (N/n_t)$ (N = # documenti, n_t = # documenti in cui appare il termine t)
  - Vettori booleani: $f(\alpha) = 1$ se $\alpha \geq 1$

$$\begin{cases} 0 & \text{altrimenti} \end{cases}$$



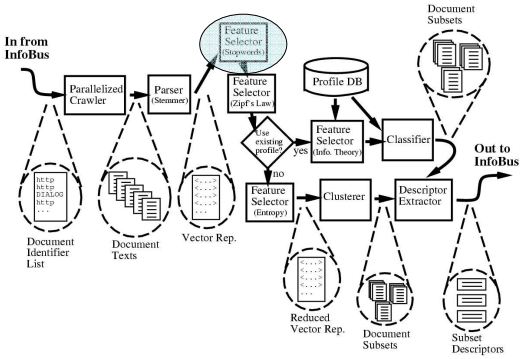
Recupero di documenti

- L’utente immette una query (“risolta” dalle varie sorgenti di informazione collegate a SONIA) e il sistema ritorna una lista dei documenti richiesti
- SONIA usa un parallelizzed crawler per recuperare il testo dai documenti presenti nella lista
- Si possono processare fino a 250 documenti in parallelo
- Si utilizza una condizione di time out (30 secondi) per evitare attese inutili



Parser e stemmer

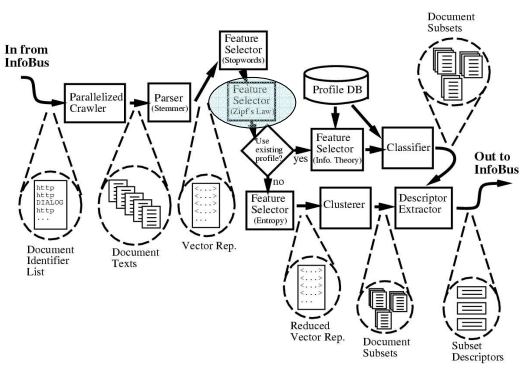
- I documenti recuperati vengono “parsati” in una serie di termini alfanumerici
- Opzionalmente questi termini possono essere ridotti alla loro forma radice (stemming)
  - Con lo stemming non si ottengono sostanziali miglioramenti



Feature selection: Stopwords

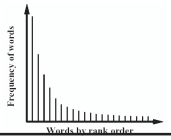
- Parole dal contenuto poco significativo (pronomi, preposizioni, congiunzioni…)
- Vengono eliminate per ridurre la dimensione dello spazio vettoriale su cui si lavora
- SONIA ha una lista di 570 stopwords più 100 parole di uso comune (“click”, “page”, “html”…)

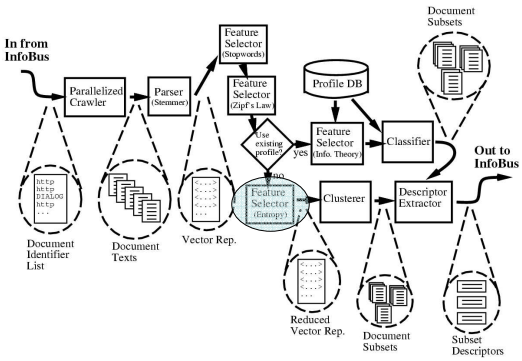
a	been	do
able	before	does
about	below	during
after	best	each
again	but	else
all	by	enough
almost	came	ever
also	can	except
am	cannot	few
and	clearly	for
are	come	former
as	consider	from
at	could	get
be	despite	goes
because	did	going



Feature selection: Zipf's law

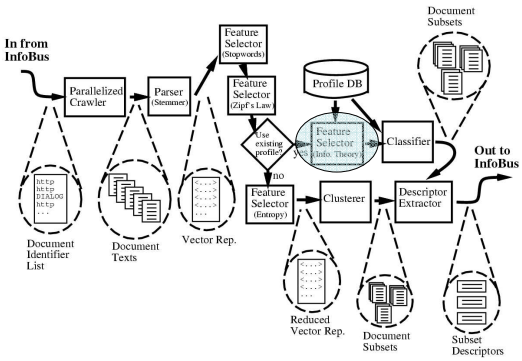
- Le parole che appaiono poco (o troppo) frequentemente non sono utili per scoprire similitudini tra i documenti
- $r_t \cdot \zeta_t \approx K$
  - $\zeta_t = \Sigma_{d \in D} \xi(t,d)$ è la frequenza totale di t nella collezione D
  - r_t è il rank di ogni t, ottenuto ordinando tutti i termini in senso discendente secondo ζ
  - $K \approx N/10$ dove N è il # totale di parole nella collezione
- SONIA elimina i termini che appaiono meno di 3 o più di 1000 volte





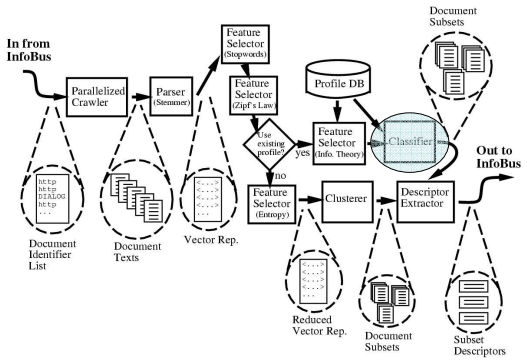
Feature selection: Entropia

- $P(t_i) = |D_{t_i}|/|D|$ è la probabilità che un termine t_i occorre in un documento scelto a caso
- $H(t_i) = -P(t_i) \cdot \log_2 P(t_i)$ è l'entropia del termine t_i
- Eliminiamo i termini con minore entropia poiché siamo interessati solo a quelli che hanno distribuzione molto varia
- Si elimina circa il 15% dei termini rimasti dopo le prime due fasi
- La feature selection non è molto brusca, poiché il clustering è computazionalmente poco pesante



Feature selection: Teoria dell'Informazione

- Dobbiamo cercare i termini con un maggior potere discriminante per i gruppi predefiniti
- Per una classificazione accurata dei documenti sono necessari pochi termini
- SONIA riduce brutalmente il numero di termini ai 50 con potere discriminante più alto



Classificazione

- Assegnare documenti a una delle categorie predefinite
  - Le categorie possono essere il risultato di un clustering oppure possono essere definite dall'utente
- Training set formato da dati con etichette assegnate; vogliamo classificare i nuovi dati (testing set) in una delle categorie già esistenti
- Usiamo reti Bayesiane di classificatori

Classificatore Bayesiano Naive

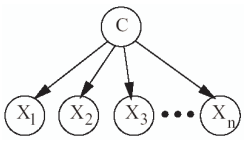
- Cerca di predire per ogni documento *d* la categoria c_j di appartenenza con probabilità più alta

$$P(c_j|d) = (P(d|c_j)\cdot P(c_j)) / P(d)$$

- Si assume che, data una categoria C, ogni occorrenza di un termine è independente dalle altre

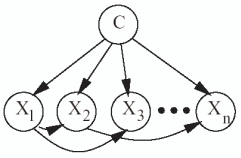
$$P(X_1,\ldots,X_n|c_j) = \Pi_{i=1 \text{ to } n} P(X_i|c_j)$$

- Questa assunzione è poco realistica, ma fornisce risultati empirici molto buoni



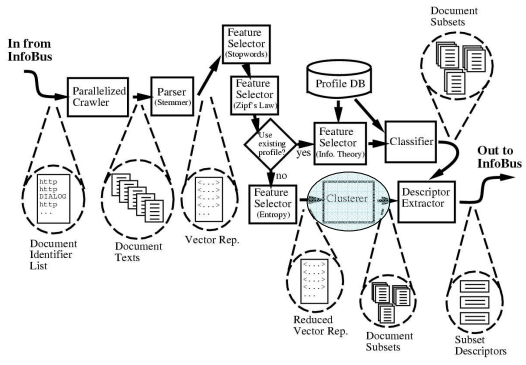
Classificatori con dipendenza limitata

- Un classificatore Bayesiano k-dipendente permette che ogni feature X_i abbia al massimo k features genitori
- Esempi di classificatori
  - Un classificatore 0-dipendente è quello Naive
  - Un classificatore (n-1)-dipendente è quello non ristretto
  - Un buon compromesso sono i classificatori 1-dipendenti



Classificazione gerarchica

- Si usa una gerarchia strutturata di topic
  - Topic vicini nella gerarchia hanno molto in comune
- È difficile cercare un topic per un documento (es. “color printers”) ma è facile decidere se esso parla di “agricoltura” o “computer”
- Abbiamo un classificatore specifico per ogni topic, che lavora solo su un set ristretto di features
  - Possiamo utilizzare algoritmi di classificazione computazionalmente più complessi → maggior precisione
- Importanza di una buona feature selection



Clustering

- Cerca di scoprire un insieme di categorie a cui assegnare i documenti, usando dati non etichettati
- In quanti cluster devo partizionare i dati?
  - Scelto dall'utente (da 2 a 10)
- Come assegnare ogni documento ad un cluster?
  - Nozione di distanza tra documenti e tra cluster
- Cluster hypothesis: documenti simili tendono ad essere rilevanti alle stesse richieste
- Se il clustering è fatto a priori, le queries sono confrontate solo con un rappresentante di ogni cluster e non con tutti i documenti

Concetto di similarità

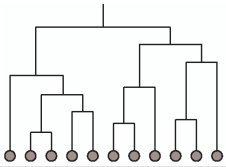
- Vogliamo cercare gruppi di dati che abbiano un alto livello di similarità
- Similarità tra documenti: per stabilire il grado di sovrapposizione tra una coppia di documenti
  - Cosine coefficient (si calcola il coseno dell'angolo tra i vettori normalizzati)
  - Expected overlap

$$EO(d_i,d_j,D) = \Sigma_{w \in d_i \cap d_j} P(Y_i = w|d_i) \cdot P(Y_j = w|d_j)$$

(Intuitivamente la sovrapposizione tra d_i e d_j può essere calcolata stimando la probabilità che ogni parola appaia in ogni documento e moltiplicando i risultati)

Algoritmi di Clustering: HAC

- Clustering agglomerativo gerarchico (HAC)
  - Inizialmente ogni documento è un cluster distinto
  - Si calcolano le similarità tra coppie di cluster e i due più vicini vengono uniti formando un nuovo cluster
- Questo procedimento genera un dendogramma
  - Noi scegliamo un appropriato livello di granularità
  - Un cluster deve contenere un # minimo di documenti (10)



HAC: group average

- Tre tipi di HAC, a seconda di come definiamo la similarità: single link, complete link, group average
- Similarità documento – cluster

$$Sim(doc,C) = \Sigma_{doc' \in C} 1/|C| \cdot Sim(doc,doc')$$

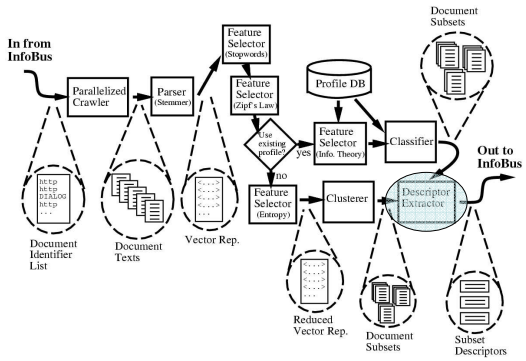
- Similarità cluster – cluster

$$Sim(C,C') = \Sigma_{doc \in C, doc' \in C'} 1/(|C| \cdot |C'|) \cdot Sim(doc,doc') = \Sigma_{doc \in C} 1/|C| \cdot Sim(doc,C')$$

- Intuitivamente tutti i documenti di un dato cluster sono ugualmente rappresentativi per quel cluster

Clustering iterativo

- Serve per ottimizzare l'algoritmo di clustering utilizzato precedentemente (HAC)
- Algoritmo:
  1. Initialize the *K* clusters
  2. For each document *doc* in the corpus
     Compute the similarity of *doc* to each cluster
  3. For each document *doc* in the corpus
     Assign *doc* to the cluster most similar to it
  4. Goto 2, unless some convergence criterion is satisfied
- La convergenza dell'algoritmo dipende da *K*
- SONIA esegue al max 10 iterazioni



Descriptor extraction

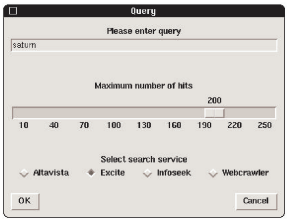
- Il clustering permette anche di estrarre automaticamente dei descrittori per i documenti
- Parole che sono presentate all'utente come etichette iniziali per ogni cluster
- Approccio centroid-based
  - Si calcola il centroide euclideo (vettore) di ogni cluster
  - Come descrittore del gruppo prendiamo i primi *k* termini corrispondenti alle dimensioni con il valore più alto
  - *K*=12 è un valore che dà descrittori brevi ma significativi
  - È molto efficace grazie all'eliminazione delle stop word
- Usato anche per suggerire all'utente i termini maggiormente pertinenti (50) a un insieme di doc

Un sistema completo

- L'utente può salvare il clustering dei documenti come una schema di classificazione gerarchica e riusarlo per categorizzare automaticamente i risultati di altre queries
- La combinazione di tecniche di clustering e di classificazione permette di navigare in una collezione di documenti e di costruire strutture gerarchiche per grandi quantità di dati
- Sistema altamente flessibile e modulare
  - Recupero di informazioni da fonti diverse
  - Facile interazione con l'interfaccia utente
  - Possibilità di personalizzare i vari moduli
  - Possibilità di modificare manualmente i risultati

Esempio d'uso 1: parallel crawler e parser

- SONIA invia la query "Saturn" a *Excite* e riceve in risposta 200 URLs
- Il modulo parallel crawler viene usato per recuperare simultaneamente le 200 pagine web
  - Il crawler recupera solo 150 documenti validi
- I documenti vengono "parsati" in vettori, contenenti inizialmente circa 4000 features



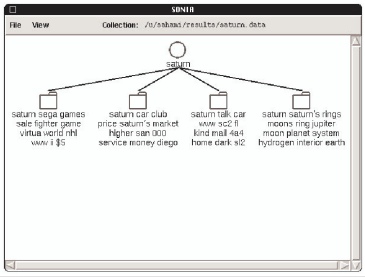
Esempio d'uso 1: feature selection preliminare e clustering

- Dapprima si elminano le stop word e poi si applica la Zipf's Law
  - In circa 1 minuto i vettori vengono ridotti a 1872 features
- L'utente sceglie di dividere i documenti in 4 cluster
  - Per il clustering e il descriptor extraction occorre 1 minuto

Extracted Descriptors	Sample Document Titles	Plausible Topics	No. Docs
saturn sega games sale fighter game virtua world nhl www ii $5	Sega Online: Strategy Guides Sega Saturn with 6 games for sale Sega Saturn Links	Sega Saturn Video Game	23
saturn car club price saturn's market higher san 000 service money diego	Saturn: A Case Study of How to Grow Saturn Falling On Hard Times Saturn of Honolulu	Saturn Car Businesses	19
saturn talk car www sc2 fl kind mail 4a4 home dark sl2	Saturn, Let's Talk Saturn New Car Sales Saturn is different! New cars and used	Saturn Car Talk and Info	70
saturn saturn's rings moons ring jupiter moon planet system hydrogen interior earth	Saturn's Small Moons The 1995-6 Saturn Ring Plane Crossings Science Tip - Saturn	Planet Saturn	38

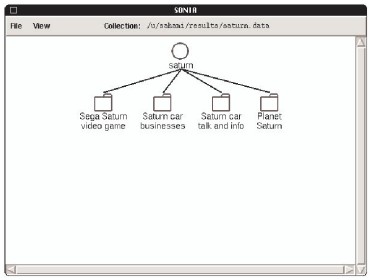
Esempio d'uso 1: descriptor extraction

- Dopo il clustering, l'interfaccia di SONIA presenta il primo livello della gerarchia dei topic



Esempio d'uso 1: ridenominazione dei descrittori

- L'utente può ridenominare l'etichetta di ogni cluster con un nome più significativo



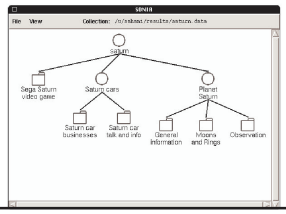
Esempio d'uso 1: clustering gerarchico

- L'utente vuole clusterizzare la sottocollezione dei 38 documenti riguardanti il pianeta Saturno
- I descrittori estratti automaticamente hanno molti termini in comune
  - Difficoltà per l'utente ad assegnare un topic al gruppo
  - Occorre guardare i titoli dei singoli documenti

Extracted Descriptors	Sample Document Titles	Plausible Topics	No. Docs
saturn hydrogen planet interior saturn's jupiter ice layer rings composition system core	Saturn Facts Composition of Saturn's Interior Saturn - What We Know	General information	17
saturn saturn's moons rings ring moon plane jupiter system voyager image planet	7 (Saturn) Moons, compare Saturn's Small Moons The 1995-6 Saturn Ring Plane Crossings	Moons and Rings	16
saturn jupiter moon venus mars mercury day rings side earth picture visible	Best times to observe Hourly Cycle of Solar System Objects APOD: July 5 - Night Side of Saturn	Observation	5

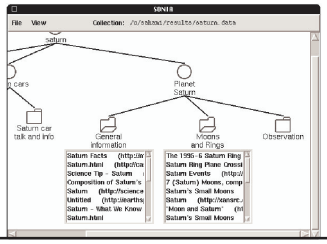
Esempio d'uso 1: interazione con l'utente

- Se l'utente pensa che alcuni documenti siano stati categorizzati erroneamente, può spostarli manualmente in un'altra categoria
- L'utente ridenomina i descrittori e può aggiungere sottocategorie nello schema gerarchico



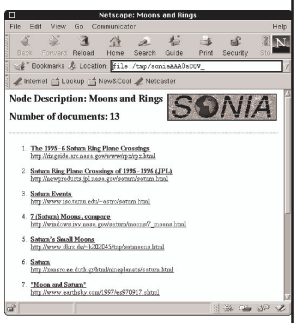
Esempio d'uso 1: browsing dei titoli dei documenti

- L'utente può vedere i titoli dei documenti e i rispettivi URLs
  - Quando i descrittori sono ambigui si possono consultare i titoli dei documenti



Esempio d'uso 1: browsing dei link ai documenti

- SONIA genera dinamicamente una pagina web che contiene i link a tutti i documenti contenuti in un nodo della gerarchia
- Parallelamente viene eseguito un browser per visualizzare la pagina web



Esempio d'uso 1: suggest query terms

- L'utente vuole cercare altri articoli su "Moons and Rings of Saturn"
  - Seleziona il relativo nodo nella gerarchia
  - Il descriptor extractor suggerisce i 50 termini più significativi relativi alla query dell'utente

saturn	system	solar
saturn's	voyager	tethys
moons	image	cassini
rings	planet	km
ring	pan	atlas
moon	earth	titan
plane	dione	telescope
jupiter	satellites	gif

Esempio d'uso 2: organizzazione dei files

- L'utente vuole organizzare 66 files
- Sui testi viene eseguito il parsing e le features selection iniziali
- L'utente decide di dividerli in 2 gruppi

Extracted Descriptors	Sample Document Titles	Plausible Topics	No. Docs
stanford computer science university teaching programming research department ca learning program interests	New_Resume cover-letter-education Andy-Reference	Job related	49
user error minor system • program users time problem command major model	CS147-paper1 psych251-week3 GradingCriteria2	Class related	17

Esempio d'uso 2: clustering gerarchico

- L'utente vuole suddividere ulteriormente le categorie "Job related" e "Class related"
  - SONIA ha il 97% di accuratezza, poiché sbaglia a classificare 2 documenti

Extracted Descriptors	Sample Document Titles	Plausible Topics	No. Docs
computer stanford science programming university software ca teaching learning resident responsibilities dormitory	NewResume Resume-newest CurriculumVitae	Resumes	10
stanford computer research university science interests department information teaching consideration learning machine	letter-MIT coverletter-Brown cover-_U.Michigan	Cover Letters	21
stanford computer program science class programming university teaching student section students eric	Ref-Kleper Phil.rec reference2en	References for others	20

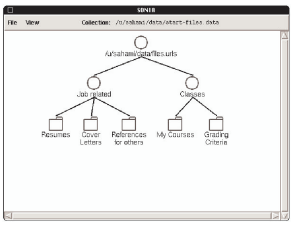
Job related

Class related

Extracted Descriptors	Sample Document Titles	Plausible Topics	No. Docs
user system • users command problem model information printer task provided knowledge	CS147-paper1 psych251-week3 GroupPaper	My Courses	9
error minor major errors properly program time array grading face smiley bar	GradingCriteria2 GC1 error-criteria	Grading Criteria	6

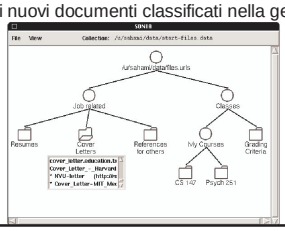
Esempio d'uso 2: classificazione

- Successivamente l'utente scrive 9 nuovi documenti che vuole classificare nella gerarchia preesistente
  - SONIA li classifica con un'accuratezza del 100%



Esempio d'uso 2: interazione e flessibilità

- L'utente vuole ridefinire la gerarchia dopo la classificazione dei nuovi documenti
  - Sceglie di clusterizzare "My Courses" in 2 gruppi
- Espandiamo il gruppo "Cover Letters"
  - * indica i nuovi documenti classificati nella gerarchia



Conclusioni e sviluppi futuri

- Utilizzi di SONIA
  - Organizzare collezioni di documenti in schemi organizzativi gerarchici (bookmarks, pagine web…)
  - Organizzare insiemi di files in un computer
- È un'alternativa più efficiente ed efficace alla classificazione e clusterizzazione manuale di dati
- Sviluppo di nuovi moduli
  - Algoritmi migliori per l'estrazione dei descrittori
  - Clustering e classificazione di documenti in topic multipli
    - Documenti inclusi nei topic la cui probabilità è maggiormente vicina alla categoria più probabile
  - Trattamento di dati non testuali