SCRIGNO

Estrarre la conoscenza nascosta nei dati

Progetto di applicazione informatica, multipiattaforma, multilingua, Open Source

SCRIGNO - Introduzione

I dati che ci circondano aumentano a ritmo esponenziale.
Diventa sempre più complicato trovare le poche informazioni utili nell'oceano dei dati disponibili: le poche informazioni utili trovate sono gioielli da conservare, ad esempio in uno SCRIGNO.
Molto spesso si ha la netta sensazione che la risposta alle nostre domande è certamente presente nei dati in nostro possesso, ma non si sa come cercarla.
Sovente è anche difficile formulare delle domande avendo solo una vaga idea di quello che si vuol ricercare.
Estrarre la conoscenza nascosta nei dati richiede anche di essere aiutati nel formulare le domande che portino a informazioni a priori impreviste: dovrebbe anche aiutarci a chiedere ciò che non immaginiamo possa esistere.
Le tecniche che aiutano ad estrarre la conoscenza nascosta nei dati sono comprese fra quelle delle reti neurali e degli alberi decisionali.

SCRIGNO - Ambiti di utilizzo

Gli ambiti di utilizzo sono possibili in tutti i settori, comunità, imprese, istituzioni, enti, università, centri di ricerca ed organizzazioni in genere dove si raccolgono, si conservano e si consultano i dati per trarre da essi informazioni utili per conoscere e per meglio operare.

Il progetto SCRIGNO ha l'obiettivo di rendere disponibile un'applicazione informatica di tipo *globale*, intendendo con tale termine, che sono interessati tutti i possibili utilizzatori senza distinzione di specializzazione o di dimensione: è un'applicazione sia *orizzontale* sia *verticale*.

SCRIGNO – Problema (1/2)

Al crescere dei dati disponibili, aumentano le difficoltà di ricerca fra di essi e la successiva loro interpretazione al fine di ricavarne informazioni utili agli scopi prefissati.

Le normali tecniche di ricerca, definendo degli ipercubi OLAP (On Line Analytical Processing) di interrogazione dei data base, non sempre sono risolutive, poiché presuppongono che l'utilizzatore già sappia che l'informazione cercata è nell'incrocio delle diverse dimensioni dell'ipercubo OLAP.

Il Data Mining in genere richiede che la struttura dei data base e le diverse possibili configurazioni degli ipercubi OLAP sovrastanti siano progettati avendo ben presente la natura, la qualità e la pertinenza dei dati e degli incroci previsti dalle varie viste logiche sui dati.

SCRIGNO – Problema (2/2)

Il Data Mining è convenientemente applicabile in ambiti di certezza dei dati e delle loro relazioni: le informazioni che si estraggono, anche se impreviste erano a priori prevedibili poiché il percorso della loro ricerca era predeterminato da come l'ipercubo OLAP era stato configurato.

Quando invece non si conoscono compiutamente la natura dei dati, la qualità e le relazioni fra di loro esistenti, è impossibile costruire degli ipercubi OLAP e i dati sono e resteranno semplicemente dati senza mai trasformarsi in informazioni.

Ponendoci di fronte alla massa dei dati sconosciuti vorremmo poter dire

"***Svelatevi senza che io vi ponga alcuna domanda***"

oppure

"***Non so nulla di voi, presentatevi e ditemi in cosa potete essermi utili***".

SCRIGNO – Tecnologie potenziali

Il Data Mining SCRIGNO potrebbe essere realizzato utilizzando il linguaggio C oppure il linguaggio RUBY.

Il linguaggio C assicura alte velocità di elaborazione, buona portabilità, richiedendo però delle compilazioni dedicate al singolo sistema operativo ospitante.

Ruby è un linguaggio Open Source che consente di realizzare applicazioni in grado di **funzionare con qualsiasi sistema operativo**: DOS, Microsoft Windows, Mac OS X, BeOS, Amiga, Acorn, OS/2, Syllable, UNIX, Linux.

SCRIGNO – Ruby (1/3)

Ruby, un linguaggio completamente ad oggetti di estrema espressività e potenza, che riesce a fondere in una sintassi semplice e chiara funzionalità ereditate da Perl, Python, Lisp e Smalltalk. Ruby supporta modelli di programmazione leggeri che consentono di abbinare liberamente altri sistemi. Ruby è il linguaggio che più degli altri soddisfa i requisiti richiesti da WEB2.0: infatti consente di erogare servizi su piattaforma web, di realizzare sistemi basati sulla comunicazione bidirezionale dei dati e, facendo parte dell'infrastruttura del web in compagnia di Linux, Apache, MySQL, Perl, PHP e Python, si affida ai metodi di peer-production dell'open source, utilizzando l'intelligenza collettiva, creata dalla rete attraverso il passa parola del marketing virale dei singoli che comunicano e che collaborano.

Le applicazioni scritte in Ruby possono funzionare su computer indipendente, oppure collegato in rete locale, oppure in rete web.

SCRIGNO – Ruby (2/3)

Ruby è un potente linguaggio di scripting completamente ad oggetti. Nato nel 1993 come progetto personale del giapponese Yukihiro Matsumoto (spesso chiamato semplicemente *Matz*), Ruby è stato il primo linguaggio di programmazione sviluppato in oriente a guadagnare abbastanza popolarità da superare la barriera linguistica che separa l'informatica nipponica da quella internazionale e ad essere usato anche in occidente in progetti di rilievo. Il linguaggio che ha maggiormente ispirato l'autore è sicuramente lo Smalltalk, da quale Ruby ha tratto la maggior parte delle sue caratteristiche. A seguire ci sono il Lisp (ed in generale i linguaggi funzionali), da cui provengono le *chiusure* (blocchi o *proc*, in Ruby), e il Perl, per la sintassi e l'espressività. Nell'implementazione corrente, Ruby è un linguaggio interpretato. L'interprete, scritto in C, è rilasciato con una licenza stile BSD, e si trova attualmente alla versione 1.8.5. Negli ultimi mesi la popolarità di Ruby ha subito una forte impennata, dovuta alla comparsa di framework di successo per lo sviluppo di applicazioni web, come Nitro e Ruby On Rails.

SCRIGNO – Ruby (3/3)

Ruby è corredato da librerie di programmi ed algoritmi matematici e statistici. Ruby è compatibile con i gestori di data base più diffusi: MySQL, PostgreSQL, SQLite, SQL Server, IBM DB2, Informix, Oracle, Firebird, LDAP, SybaseASA. Altri data base stanno per diventare compatibili.

Purtroppo la documentazione per l'utilizzo del linguaggio al fine di realizzare applicazioni di tipo matematico-statistico è attualmente **carente**: io ho rimediato con una accurata ricerca in rete di documenti e di esempi, effettuando poi dettagliate e meticolose sperimentazioni delle funzioni e dei comandi disponibili.

Ho raggiunto un risultato molto concreto: la **riscrittura totalmente e solamente in Ruby** di un'applicazione di acquisizione della conoscenza funzionante in ambiente Fortran / Windows con lettura e scrittura di archivi, acquisizione di informazioni da tastiera e calcoli su matrici multidimensionali; l'applicazione è stata aggiornata ed eseguita sia con Windows XP, sia con Linux Ubuntu senza alcuna modifica del codice.

SCRIGNO – Algoritmi di riferimento (1/4)

Per la ricerca della conoscenza nascosta nei dati, solitamente ci si riferisce agli algoritmi delle rete neurali e degli alberi di decisione.

Gli algoritmi delle reti neurali possono essere più o meno complessi ed essere di tipo supervisionato o non supervisionato. Nelle reti neurali di tipo supervisionato sono presenti delle variabili obiettivo.

L'apprendimento nelle reti neurali di tipo supervisionato cercherà di ottenere un modello di tipo predittivo che applicato a nuovi eventi riesca a classificarli associandoli alle variabili obiettivo usate nel precedente apprendimento.

Per esemplificare se nella elaborazione di un campione di individui dei quali siano stati raccolti i principali dati anagrafici, socio-economici, di istruzione, di tendenza politica, ecc. con definita, come variabile obiettivo, la dichiarazione voto, allora sarebbe possibile sottoporre al modello predittivo calcolato, gli stessi dati di ingresso per altre persone ottenendo la stima della loro probabile dichiarazione di voto.

SCRIGNO – Algoritmi di riferimento (2/4)

Le reti neurali di tipo non supervisionato non richiedono che sia dichiarata una variabile obiettivo; occorre definire invece il numero dei gruppi di classificazione che si vogliono ottenere.

In questo caso l'algoritmo suddivide al meglio le registrazioni in un numero di gruppi non superiore a quello desiderato.

L'esame dei record inclusi in ogni gruppo può essere motivo di scoperte inattese di particolarità importanti e di caratteristiche degne di ulteriori analisi.

Si pensi scoprire che un certo farmaco ha effetti nettamente migliori se associato ad uno stato febbrile del paziente: chi l'avrebbe detto!

Gli algoritmi delle reti neurali, supervisionate e non supervisionate, hanno però alcuni difetti importanti, alcuni facilmente superabili e altri meno.

SCRIGNO – Algoritmi di riferimento (3/4)

I difetti più *fastidiosi* sono:

- elaborano solo dati numerici, mentre molte informazioni sono di tipo testuale.
- i dati numerici devono essere normalizzati con intervallo da -1 a + 1.
- elaborano dati strutturati, mentre molti dati sono di tipo non strutturato (testi).
- il modello appreso è una *scatola nera* e dal suo esame è quasi impossibile rispondere alla semplice domanda: quali sono le variabili che maggiormente determinano l'attribuzione delle registrazioni ad ogni singolo gruppo (e per quali valori di ogni variabile) ?
- i programmi disponibili sono quasi tutti di tipo proprietario, presentano quasi tutti i difetti sopra descritti, sono imprecisi nei risultati, lenti in esecuzione e molto costosi.

SCRIGNO – Algoritmi di riferimento (4/4)

Le Gli algoritmi riferibili agli alberi di decisione sono più maturi, più diffusi e con prestazioni normalmente soddisfacenti.
I software degli alberi di decisione hanno l'obiettivo di estrarre l'albero delle condizioni logiche con rami il più possibile corti, rami che, al loro apice, indicano il valore della variabile obiettivo.
Gli algoritmi riferibili agli alberi di decisione rispondono in parte all'esigenza di rendere meno *black box* la matrice di apprendimento ottenuta al termine dell'elaborazione della rete neurale.
Il difetto intrinseco degli alberi di decisione è però la loro caratteristica di esaminare una variabile per volta ripartendo, ad ogni iterazione, le registrazioni ancora da classificare in due sottoinsiemi perdendo di vista le variabili nel loro insieme, attenzione invece tipica delle reti neurali.

SCRIGNO – Contenuti del progetto

Il progetto SCRIGNO ha l'obiettivo di realizzare un'applicazione multi piattaforma, multilingua e Open Source, fruibile sia in rete locale sia su web in modalità ASP (Application Service Provider), idonea ad estrarre la conoscenza nascosta negli archivi di dati strutturati e non strutturati, dati rappresentati sia in forma numerica sia in forma testuale. L'applicazione dovrà essere in grado di spiegare le variabili di input che maggiormente contribuiscono a definire le particolarità di ogni gruppo di classificazione allo scopo di rendere comprensibili e praticamente utilizzabili i risultati del calcolo con i contributi dei matematici e degli statistici che parteciperanno al progetto.
Per la valutazione dei risultati intermedi del progetto, è necessario poter disporre di significativi archivi di prova, provenienti dai diversi possibili settori di utilizzo.

Gli appartenenti ai diversi settori di utilizzo daranno il loro contributo per valutare i risultati dei calcoli e la loro utilizzabilità a fini pratici.

SCRIGNO – Competenze nel progetto

- controllo dell'avanzamento del progetto ed amministrazione
- analisi e progettazione tecnologica
- analisi e progettazione matematica – statistica
- analisi e progettazione dei programmi di conversione dei formati
- analisi e progettazione delle interfacce utente
- analisi e programmazione dei programmi di conversione dei formati
- analisi e programmazione delle interfacce utente
- analisi e programmazione dei prg di acquisizione della conoscenza
- analisi e programmazione dei programmi degli alberi delle decisioni
- analisi e programmazione delle rappresentazioni grafiche dei risultati
- redazione della documentazione di installazione
- redazione della documentazione dell'utilizzatore
- preparazione e redazione degli esempi di utilizzo
- aspetti legali della licenza

SCRIGNO – Prototipo funzionante

Del progetto SCRIGNO esiste un prototipo funzionante, da me realizzato in ambiente Windows / DOS e Ruby (tutti i Sistemi Operativi); è attualmente composto da:

- programmi scritti in Fortran / Ruby per la normalizzazione dei dati (numerici e testuali)
- programmi scritti in Fortran / Ruby con algoritmi modificati riferibili alle reti neurali
- programmi scritti in Clipper DOS per l'algoritmo relativo agli alberi di decisione
- molti esempi di risultati di calcolo con input provenienti da ambiti diversi (competizioni sportive, analisi mediche, dati dei prodotti, dati dei clienti, prove dei materiali, sondaggi di opinione, dati socio economici delle province italiane, ricerche di mercato, caratteristiche chimiche degli alimenti, ecc.)
- documentazione varia sul tema e descrizione degli algoritmi inseriti nei programmi