STUDIO PROGETTUALE

CAIMANS

Classificazione Automatica di Istanze Multiple di Archivi Non Strutturati

Marzo 2013
Versione 2.1.3433

Lista di distribuzione elettiva

Luca Marchese (PM)	Ruggero Ballardini
Roberto Brenna	Carlo Bergamini (DBA)

L'autore

.....

Razionale della documentazione

Questa documentazione ha una doppia valenza essendo diretta sia ai portatori di interesse (stakeholders) che alle risorse implicate nella progettualità
Il testo è quindi formalmete discriminato con contenuti di formato cosmeticamente funzionale al target cui sono destinati.
Essenziale nella lettura il riferimento pregiudiziale al numero di versione del documento e alla data di pubblicazione.
Molti dei capitoli in esso sono in continua evoluzione e aggiornamento pertanto l'autore mette a disposizione sul proprio sito una pagine dedicata alla documentazione progettuale intesa come aggiornata alla ultima versione di rilazio. In merito alla porzione della documentazione elettiivamente tecnica, l'accesso ad alcuni documenti potrebbere esse secretato, inoltre gli stessi documenti protetti in lettura/stampa per ovvie ragioni di diritto d'autore.

I riferimenti bibliografici sono stati organizzati in modo tematico per favorire l'accesso delle fonti primarie citate nello studio progettuale. Questo documento non prescinde dalla disponibilità nella stessa cartella del file di sinospi bibliografica anche se alcune risorse sono legate alla risorsa web disponibile via internet sul sito dell'autore.

Un importante sviluppo documentale degli allegati è mirato alla documentazione tecnica e alla specifiche funzionali di progetto che sono da considerarsi confidenzialmente condivise solo dall' autore e dallo staff tecnico accreditato nell'ambito del progetto.

Definizioni propedeutiche

Per una scorrevole lettura della porzione divulgativa della documentazione è essenziale non prescindere da alcune basilari definizioni. L'uso degli acronimi è vitale per poter indirizzare gli argomenti non confondendo i contesti in cui i vari ambiti si snodano. In alcune circostanza l'uso dei termini estesi sarà indifferentemente usato nelle accessioni inglese e italiana

CA	Classificazione Automatica (Automatica Classification)
IR	Information Retrieval
TC	Text Categorisation (Categorizzazione testuale)
BD	Big Data
SIG	Società della Informazione Globale.
TCA	Text Classification Application
DCA	Distributed Classifier Architecture
ADM	Automated Document Mining

Parole chiave

KNN	Nearest Neighbour classifier
NBT	Naïve Bayes
SVM	Support Vector Machine
CBC	Centroid based classification
ABC	Association Based Classification
DTI	Decision Tree Induction
TGM	Term Graph Model
NNC	Neural Network. Classification

APPROCCIO TEORICO

Introduzione

CAIMANS è una piattaforma modulare di strumenti per la Classificazione Automatica (CA) e la Categorizzazione Testuale o Text Categorization (TC). Questi due ambiti sono di enorme importanza per via della drammatica espansione del fenomeno del BIG DATA (wiki) che raccoglie l'enorme problematica della immensa disponibilità e incremento di contenuti digitali giornamente disponibili nella Società della Informazione Globale.

Questo scenario non contenibile né tantomeno reversibile pone immanente la necessità di strumenti evoluti di Data Mining per la navigazione la classificatione e la visualizzazione di questo oceano contenutistico. Le tecniche di classificazioni devono identificare Classi e Categorie di informazione tramite Text Classification Application e l'ambito di tutto lo sviluppo applicativo software che affronta questo dilemma va sotto il nome di Distributed Categorisation Architecture (DCA). Altrettanto diffuse definizioni in sovrapposizione sono : *Text Classification, Classification techniques e Text Mining / Contents Mining*.

Presupposti di fattibilità

L'autore ha prodotto una "scheda preliminare di progetto" la cui lettura è raccomandata, seppure non mandatoria, per la corretta interpretazione di questo documento.

Nell'ambito estremamente largo descritto nel paragrafo precedente, ci sono numerose e svariate applicazioni tecnologiche per lo sviluppo di software. Si tratta tanto di software gestionale che di nicchia che va dal Document ERP/CRM alla tutela del marchio nel DRM (1). Alla data di stesura della prima versione di questo Studio, il nostro progetto CAIMANS è originale e unico. Originale perché pensato per affrontare il work-flow operativo di una Rassegne Stampa automatizzata, ma soprattutto per rivoluzionare la classificazione dei contenuti (Es. : articoli, web crawling, scientific contents plagiary control ecc.) applicando algoritmi di calcolo e modelli evoluti di Pattern Matching.

Intraprendere il progetto CAIMANS è altamente fattibile in quanto la CA può già basarsi su un immenso repository proprietario di articoli classificati in modo euristico (rules based e SBE – Supervised by Expert); questo enorme valore aggiunto può essere ulteriormente evoluto con un approccio di automazione che certamente snellirà i tempi del business-flow, ma anche, e più importante, potrà riclassificare tutto lo scibile disponibile per creare una Associated Knowledge Base (AKB).

La AKB permetterà di rivoluzionare l'approccio alla ricerca/selezione di contenuti grazie a inferenze di un IR (Information Retrival) che possa determinare una più accurata/affidabile selezione a partire non solo da elementari parole chiave ma direttamente da contenuti testuali complessi quali : un intero articolo, alcune key-phrases, citation and sentences). Questo è l'ambito che definiamo TC ossia di Text Categorisation.

Una volta realizzata la AKB, la realizzazione di CAIMANS può aprire sviluppi applicativi di text-mining che potranno applicare gli algoritmi di processo testuale proprietari a nuove fonti di contenuti aprendo nuove Line of Business per il Automated Document Mining (ADM).

Scopi e applicazioni

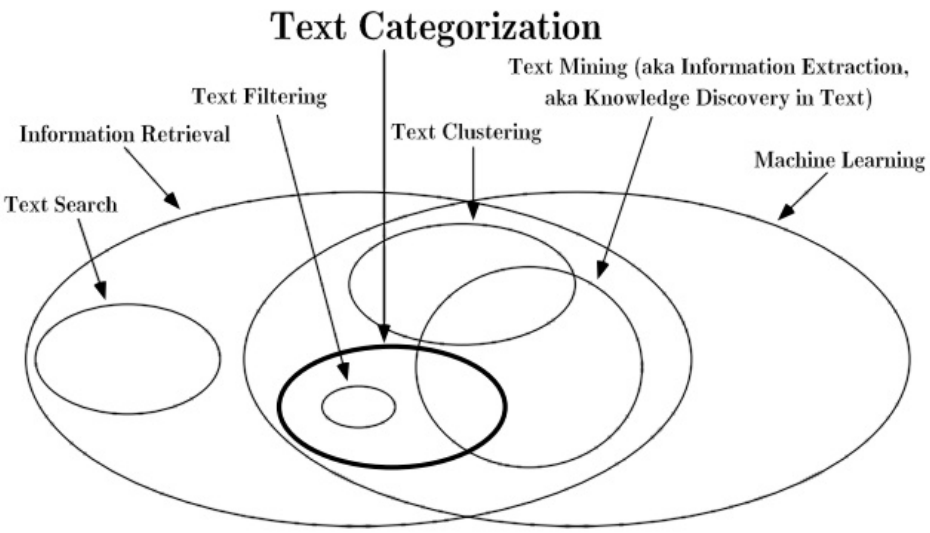
Rispetto all'approccio euristico e di tipo SBE, quello della CA/TC offre la possibilità di alimentare il bagaglio informativo della AKB in modo iterativo crescente grazie all' auto apprendimento. Le inferenze non saranno dirette in modo booleano ad una base di dati secondo logiche di algebra binaria (AND / OR), piuttosto ogni classe di categoria (Settore, sub-settore, argomento, evento, entità nominale ecc) viene tradotta in un vettore digitale (features vector) che traduce in modo numerico l'equivalente dello stampo digitale degli articoli (codice genetico delle stream testuale).

In definitiva sarà quindi usare tecniche di Pattern Matching che renderanno granulare e atomina la finitura delle varianti delle categorie (classi oggettive di Traning Sets), al punto che una volta a regime il sistema di CA deciderà autonomamente di classificare nuove categorie emergenti e/o nuove.

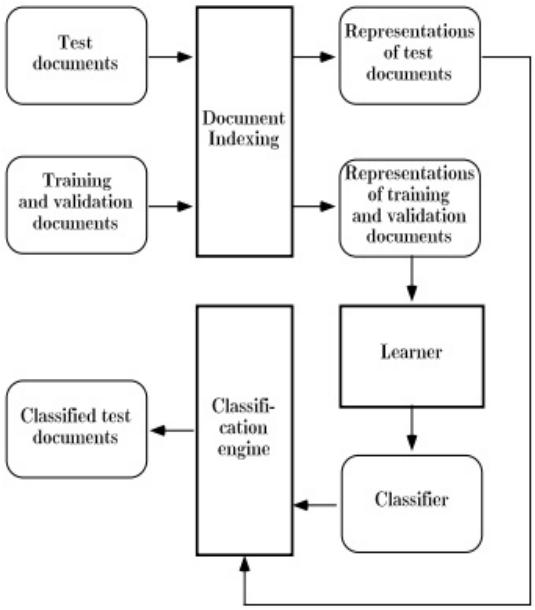
Da ultimo, ma non per ciò meno importante, la base di conoscenza anche definito META-DDB (Document Data Base) fornirà come risultato un listing di articoli candidati che saranno elencati in modo obiettivo secondo criteri di affinità e pertinenza. Il sorting SQL standard non è in grado di speculare su questi parametri perché ha i limiti della architettura Entità-Relazione con il quale è stato disegnato il database.

Che sia euristica o automatica lo scopo della classificazione è la creazione di una Knowledge base che possa essere esposto ad una inferenza o Query proveniente da un sistema **IR** (Information Retrieval).

Molto istruttiva per questo scenario è l'immagine sotto riprodotta tratta da un libro di Sebastiani (2009)



Dallo stesso testo recepiamo un secondo disegno che riassume con un diagramma a flusso il processo di Text Categorisation a partire da documenti crudi (*raw articles*).



Commentanto velocemente si parte da contenuti testuali non classificati che vengono Indicizzati per confronto con una collezione di documenti precedentemente classificati in un Training Set. La indicizzazione è una attività grazie alla quale tutti i documenti sono stati "pesati" secondo uno score. Lo score o classificatore può non essere unico, anzi è sempre meglio averne svariati perché diversi algoritmi hanno diverse capacità di classificazione a seconda dell'ambito dei contenuti che si considera- In CAIMAS si sono adottati tre algoritmi di calcolo degli Score :

- **TF-IDF (Term Frequency – Inverse Data Frequency)**
- **Likelihood Ratio con Laplace smoothing**
- **Naïve Bayesian similarity**

Seguendo il diagramma si comprende che la rappresentazione del testo da classificare, che nella nostra piattaforma CAIMAS abbiamo chiamato DATAGRAM, viene passato (si legga confrontato e correlato) con il Classificatore risultante dalla immagine che il Learner (per noi Istruttore CA) parallelizza nel Motore di Classificazione (Classification Engine nella figura accanto).

Il motore, finalmente confrontando i due Scores che sono numeri scalari attribuisce al documento in entrata una etichettatura o classificazione sulla base della "vicinanza" dei sue punteggio o pesi (o Scores per l'appunto).

In estrema semplificazione metaforica in Biologia, è come andare a fare il la ricerca di un filamento genetico di RNA (messaggero codogenetico) di una proteina contro tutta la sequenza di DNA di una cellula (il genoma), per capire dove questo viene espresso. Si tratta di sovrapporre le immagini della sequenza di acidi ribonucleici (fingerprint)

Concettualizzando ulteriormente, abbiamo una impronta digitale di un sospetto che viene rintracciata fra tutte quelle conosciute dalla polizia.

Nozioni teoriche sul Modello Applicato

Quanto sopra brevemente introdotto investe molti ambiti di conoscenza teorica e applicativi che non possono essere trattati in modo esteso e accademico, tuttavia per chi è coinvolto progettualmente è deprecabile un approccio semplicistico.

Anche solo se stakeholders, commerciali e/o amministrativi chiunque sia coinvolto a vario titolo nel progetto CAIMANS può contribuire meglio avendo apprezzato nozioni più intime della materia teorica. Ovviamente per un completo apprendimento lo studio dei riferimenti bibliografici rimane l'unico metodo plausibile.
La documentazione progettuale comunque è compendiata da allegati che riportano i richiami alla letteratura anche per motivi di riconoscimento delle paternità d'autore.

Quanto sopra premesso, abbiamo adottato uno stile espositivo il più divulgativo possibile e raccomandiamo la lettura sistematica del capitolo per poter meglio apprezzare quelli successivi relativi al disegno, alla ingegnerizzazione e alla implementazione del modello teorico.

I riferimenti alle formule hanno una valenza solo formale e per la comprensione concettuale a cui ci si è riferiti sopra non rappresentano un ostacolo. Non occorre alcune malizia matematica per la presentazione quasi didascalica del discorso espositivo.

La CA per chi non la vuole capire ma la vuole usare

In questo studio introduttivo tratteremo i seguenti spunti :

- Punteggio di documenti (Scoring documents)
- Frequenza di Termini (TF o Term frequency)
- Statistica delle collezioni (Collection statistics)
- Schemi pesati (Weighting schemes)
- Schemi posizioneli (TPos scanning)
- Punteggio dello spazio di vettori Vector space scoring

Questi temi presumono ovviamente di essere contestualizzati e noi possiamo pensare ad essi come a dei metodi e delle tecniche di calcolo che vorreo poi utilizzare in un ambito di Classificazione di Documenti.

La inferenza categorica (Ranked retrieval)

Fino ad oggi abbiamo usato le queries dell 'SQL basate sull'algebra di Boole (da cui inferenze booleane) . Il punto di limite maggiore di questo approccio riguarda il modello *ruled based* con il quale si può investigare sui dati. La logica AND/OR seppure aperta e nidificabile fino ad un ordine di dimensione il cui unico limite è la fantasia, di fatto è vincolata alla architettura del database relazionale.

Teoria booleana: Banchetto o Carestia

Il Ranked Retrieval del SQL risponde alla richiesta / inferenza con un esito di tipo ***Match*** oppure ***NoMatch***
In alcuni contesti di nicchia dove chi ricerca informazione è esperto questo può andare bene, ma la maggiorna parte degli utenti che cercano articoli a fronte di pochi indizi (parole chiave), si aspettano un ritorno :

adeguato, aderente, affidabile, pertinente e ordinato per rilevanza e/o affinità

Le queries booleane ritornano poche (se non =0) o troppi (1000) risultati (*feast or famine psycosis*). Chi ha una esperienza di SQL non ipocrita sa che in genere la AND porta a pochi la OR a troppi. E cosa c'è in mezzo ?

Caso 1 : "Articoli Azioni AND Banco di San Paolo AND estero"
Caso 2 : "Articoli Azioni AND Banco di San Paolo estero AND caduta"

Chi prende? l'Istituto di credito o il santo ?

Quelle sopra riassunte sono le ragioni per le quali le nuove tecnologie di servizio IT che si occupano di contenuti elettronici devono adottare una strategia CA. Non solo ragioni di risparmio e ROI ma anche, e soprattutto, maggiore *Customer Satisfaction* e livello di standard.

La logica del Bag of Words (BoW)

Per superare i limiti del costrutto booleano la CA usa modelli di calcolo in grado di convertire il contenuto testuale in un vettore delle caratteristiche o Features Vector. Questa rappresentazione è basata sulla rilevanza delle occorrenze di un termine o parola nel testo. Il conteggio di una parola viene definiti Term Frequency o TF ed è alla base della logica di

contronto per un Information Retrival System che sia in grado di graduare la similarità di contenuti (o gruppi di contenuti).

Il TF all'interno del documento può essere ulteriormente affinato considerando le occorrenze non solo di parole uniche ma anche doppie e triple. Questo approccio viene definito *n-gram* (dove n = 2,3 > 2-*gram* e 3-*gram*). Insieme le generazioni di n-grams si chiamano DATAGRAM del corpus.

Pensiamo a :

Monte dei Paschi di Siena > *syntax Lemming/pruning preprocess* > **Monte Paschi Siena**

Dopo il pruning di lemmizzazione di cui accennato in precedenza le TF troveranno le seguenti generazioni nel DATAGRAM :

1-*gram* : **Monte**; 2-*gram* : **Monte.Paschi**; 3-*gram* : **Monte.Paschi.Siena**

Risulta piuttosto intuitivo che la concorrenza dei tre termini se fosse ripetuta anche solo due volte, sarebbe un forte fenotipo per la categoria cui il testo è associato, es. Economia, Banche, Istituti di Credito.

Complessivamente considerati gli elementi dei paragrafi precedenti fanno parte di quella che viene denominata logica del Bag of Words o anche BoW. Vedremo come in seguito tutte le applicazioni teoriche degli algoritmi applicati dalla CA parta da questo fondante presupposto di calcolo.

Logica posizionale

In CAIMANS la piattaforma di CA utilizza le rappresentazioni iper-spaziali geometriche del testo analizzato o Corpus, tuttavia questi algoritmi (BoW) non considerano l'ordinalità dei termini nel testo. Si è pertanto deciso di utilizzare anche un indice di posizione con il quale poter eleborare anche speculazioni in ordine allo stile del costrutto dell'articolo.

Altrove abbiamo spiegato come nella CA gli algoritmi di lemmizzazione e stemming (implementati in CAIMANS) intendono fornire indicazioni semantiche in quanto processano il testo sulla base di meta-template sintattici e linguistici, il caso della logica posizionale contribuisce ulteriormente e questi approcci.

Per capire meglio riferiamoci all'esempio delle frasi :

"**Giovanni è più veloce di Maria**" vs "**Maria è più veloce di Giovanni**"

Da un punto di vista vettoriale i due testi sarebbero uguali mentre un algoritmo posizionale o AlgoTPos è in grado di discriminare.

Dal Database alla Base di conoscenza

Essenzialmente il paradigma di ricerca deve essere applicato ad un repository "analogico" superando la logica digitale (TRUE/FALSE) a vantaggio di quella dello Score. Una inferenza potrà quindi essere graduata per affinità e pertinenza fornendo in outoput un elenco oggettivo e progressivo di risultati.

In questa logica ogni documento ha un punteggio o indice di affinità con campo di appartenenza [0-1] che può quindi essere espresso anche in pertentuale (es. 0,67 --> 67%) rispetto alla categoria alla quale il contenuto del documento risulta associato.

Un eventuale documento non classificato viene quindi confrontato in termini di similarità dove 0= totalmente differente e 1= identico. Nella pratica il calcolo della filosofia BoW non raggiunge mai i valori di limiti perché produce calcoli "razionali" (nel senso valori matematici frazionari e continui).
Quando si ricorre alla CA non si dirà mai che un Articolo "***è qualcosa***" piuttosto che "***dovrebbe essere qualcosa***" (Coefficiente di Jaccard).

Nonostante quello scritto sopra in alcuni casi di algoritmimi molto quotati come il Naive Bayes Classifier o NBC, le assunzioni per il riconoscimento del testo sono totalmente diverse e non tengono minimamente conto della semantica e della posizione di un termine o parola nel testo. Vedremo che questa estrema diversità nell'approccio teorico alla CA/TC è una delle ragioni per le quali non è possibile scegliere di lavorare con un solo algoritmo ma bisogna sempre compendiare con un ulteriore tecnica di calcolo che tenga contro di più classificazioni scegliendo al run-time quella che premia di più. Parleremo infattio di Podio delle Classificazioni.

Teoria Aracnocentrica : la semantica supervizionata

Naturalmente tutto ciò che abbiamo considerato nel caso di un singolo Corpus o articolo, può essere classificato su un gruppo di documenti che sono classificati e/o associati ad una categoria (nel senso che appartengono alla Classe).
Pertanto il salto di qualità nella classificazione è quello di avere dei DATAGRAMs di Classi e non singoli articoli.

Il termine Classe da cui classificazione indica un criterio di raggruppamento semantico che per noi, in questo contesto di studio della piattaforma CAIMANS può essere indifferentemente il Settore/SubSettore, la categoria, l'Autore, Il profilo Cliente della Rassegna stampa, l' Argomento, l'Avvenimento ecc..

Si conprende pertanto che la classificazione deve iniziare da alcuni Training Set (gruppi peculiari di articoli) selezionati da esperti in grado di valutarne pertinenza, completezza, aggiornamento e affidabilità rispetto ad una classe.

Questo approccio si chiama Costrutto Semantica Supervisionata (CSS) e costituisce il primo imprinting per una base di dati Relazionale che venga sottoposta alla CA.

Quando il numero di DATAGRAMs delle classi (Categorie) è sufficientemente elevato in ragione del numero di documenti presenti nel database, la CA offre la possibilità di intraprendere la fase di Automated Text Categorisation o ATC.

La ATC sfrutta i la teoria della Self-Learning Machine che si basa su tecniche di algoritmi di calcolo di meta conoscenza basate su un KB (Knowledge Base) per fare in modo che nuovi articoli (Unstructured Contents Feeder UCF) non solo vengano associati ad Classi di articoli già classificati, ma che aggiungano nuove terminologie peculiari della evoluzione linguistica/culturale della materia considerata.

Da ultimo, e certamente più entusiasmante, nel caso di nuovi ambiti e materie la candidatura e la promozione di nuove classi.

Supponiamo di aver classificato in 1000 categorie/argomenti una base di dati di 1 milione di articoli nell'ambito della Economia e Finanza. Se un domani si acquisisse una commessa per trattare una banca dati testuale del Centro di Studi Vaticanensi, CAIMANS potrebbe indicare e quindi classificare automaticamente senza supervisione gli articoli creando da solo le classi categoriche.

Vediamo schematicamente alcune altre rilevanti applicazioni

- Riconoscimento della paternità nella letteratura scientifica
- Distribuzione automatica di documenti
- Codifiche automatiche di raccolte dati e censimenti
- Text Filtering a supporto di attività di intelliggence e investigazioni
- Indicizzazione automatica di librerie digitali

Il training Set : quando il piccolo conta più del grande

Il Training Set o TS è quindi un insieme di documenti etichettati da esperti che li attribuiscono ad una o più Classi o Categorie. Questa categorizzazione, definite rispettivamente *Single* e M*ulti labeling* o **SL** ed **ML**, sono quindi una collezione di documenti coerentemente associati in base a criteri di completezza, appartenenza e pertinenza valutati dall'uomo (*supervisioning*) .

Ultima finalità di questo scenario quella di creare una Base di Conoscenza o Knowledge Base in acronimo **KB.**
Quando non si conosce la CA si è portati a credere che più una KB è grande e piena di milioni di articoli più la classificazione sarà efficace e affidabile.

E' essenziale chiarire perché non sia cosi, e al contrario è vero il contrario. Ci sono primcipalmente, ma non esaustivamente, due ragioni le quali congiuntamente considerate impattano proprio sulla efficacia e sulla affidabilità della scelta categorica.

Prima di trattare le due fattispecie assumiamo due convenzioni puntuali circa i termini efficace e affidabile (*effective* e *trustable*).

Con efficacia si allude alla capacità del Motore CA di processare una grande numero di documenti calcolando il maggiorn numero di Indici o Matching Scores nel più breve tempo possibile.

Con affidabilità si intende la capacità del motore di puntare e scegliere la categoria di appartenenza in modo pertinente e puntuale con un altro grado di confidenza qualitativa nella scelta; tale scelta dovrebbe essere credibile al punto di avere una buona certezza di aver selezionato articoli effettivamente aderenti al contenuto e non in eccesso (*overbooking*). Quella che in altre parti della nostra documentazione abbiamo chiamato la Sindrome del Buco Nero e della Nebulosa (NdA).

Vediamo le due ragioni per scegliere un TS piccolo e non gigantesco :

1) **RAGIONI DI PERFORMACE**. I tempi di classificazione in un processo produttivo quale quello che si intende affrontare devono essere conciliabili con l'analisi delle esigenze progettuali concordate tra PM e progettista. Tipicamente in CAIMANS ci siamo prefissati un ordine di processo medio intorno agli 800ms per un Corpus medio con 1200 termini/parole;
2) **RAGIONI DI PULIZIA**. La eccessiva campionatura di articoli per una determinata Classe o Categoria, porta statisticamente al fenomeno conosciuto come "Deriva della media delle distribuzioni" che in pratica indica una sovrapposizione di risultati con articoli che non dovrebbero essere inclusi.

Il punto 2) può essere meglio comprenso pensando all'aumento del rumore di fondo (*noise*) fenomeno che si accentua con l'aumentare delle dimensioni del campione. In modo analitico la curca di apprendimento di una KB è quella di una funzione logaritmico. Questa funzione raggiunge un plateau che è come dire che oltre un numero di articoli del TS non si migliora la capacità di riconoscimento della CA e si rischia di introdudurre solo rumore e quanto meno ridondanza.

Il training Set :una bugia detta per far bene

Adesso che abbiamo trattato un certo numero di nozioni e il quadro si fa più nitido, ci si accorge che rimane in fondo all'angolo un dubbio… Un fastidioso granello di incertezza che non ci dà tregua e che persiste sotto forma di una domanda ancora senza risposta conclusiva : ma cosa è questo Training Set in pratica ?

Il Viagra del motore CA : la white e le black list

Affrontiamo da ultimo l'aspetto più controverso e realisticamente ineludibile del progetto. Come si dice di solito secondo gli antichi : "Alla fine della fiera, 'sto <**coso**> funziona o no ?

Chi ha studiato per anni sa' che non esiste una Panacea assoluta, non esiste cioè un unico algoritmo vincente per il semplice fatto che istruire un Training Set introduce di perse una valutazione soggettiva, non oggettiva (in statistica Bias di Selezione (Stanford University A2,[12]).
In pratica, l'insieme dei **TSs** costituirà la **KB** di riferimento per la **CA** ed è verosimile che per la selezione degli articoli per una determinata categoria e quindi per uno specifico TS, si siano impegnati esperti diversi specializzati proprio in quella categoria.

E' essenziale rendersi conto del fatto che quale che sia la capacità di "calcolo ragionamato" (C-Learning Machine), la classificazione sarà in grado di selezionare pertinentemente in base alla qualità e alla pertinenza dei contenuti scelti in modo soggettivo. Come dire che il calcolo puro di un algoritmo non può "aggiungere" significato ad un testo se non c'è. Con una metafora già sfruttata, possiamo pensare di cucinare un buon pasto solo se abbiamo fatto bene la spesa e gli alimenti sono freschi e flagranti.

Veniamo alla bugia detta per fine di bene. Fino ad ora abbiamo parlato di un TS, ma in realtà di TS non ne esiste uno solo ma almeno tre !

Peraltro, l'ordine con il quale sono elencati riflette quello delle fasi di lavoro di CAIMANS e, al di là della traduzione dall'inglese, il loro significato è piuttosto ovvio. Rilevante nella notazione sopra il significato della freccia che indica il fatto che questa sequenza di **TS** è in realtà un ciclo virtuoso che si ripete nel tempo. Infatti, le indicazioni del **TS** ricavato dalle procedure di Test, forniscono nuovi elementi/termini che vanno in produzione e sono "imparati" dalla **KB** per essere utilizzati al primo RUN di confronto con un nuovo contenuto (*new entry text matching*)

La fase di costruzione di un motore CA ha proprio lo scopo di usarli tutti e tre per confezionare una piattaforma i controllo delle variabili di funzionamento del motore una volta che è messo in produzione perché continuamente venga aggiornato ed impari a correggere/aggiungere nuovi punteggi di terminologia sempre più affinati.

Veniamo al senso del paragrafo. Per quanto raffinata e completa, la KB, ossia l'insieme di tutti i TS di ogni categoria clasificata, non può prevedere l' "evoluzione delle specie". Intendiamo con ciò il fatto che ogni dominio di competenza, settore, categoria e/o argomento che sia, conosce periodicamente una evoluzione che segue quella culturare e antropologica della editoria e della stampa.

Periodicamente, la KB dovrà essere alimentata con ingresso di un nuovo profilo di termine (datagram). Appare molto semplice da acquisire ma spieghiamo perché tecnologicamente è molto problematico da fare.

La CA impara dai contenuti ma li rappresenta in un modello geometrico-spaziale; questo individua uno spazio-significato (*meaning Hyperplan*) che tiene conto di tutti i termini "aurei" di quel contenuto.

L'implicazione di questo è che ogni volta che introduco un termine in un documento/articolo (e quindi per tutto il TS delle relative categorie associate), l'intero assetto dello spazio-significato deve essere ricomputato !

In effetti una KB è una struttura sempre cangiante, e una pletora di processi in costante attività, scandisce sistematicamente la KB per aggiornare i punteggi di score nel DATAGRAM.

Quando sopra riconsiderato, esiste nel flusso di processo della CA un ultimo elemento (anch'esso un algoritmo) che possiamo assimilare al Filtro delle Stop Words (vedi sopra); si tratta del modulo delle Black or White lists (BWL).

Pertanto, un ultima gestione del processo di classificazione ad alto livello comporta una sorta di enfatizzatore di peso o "weigth enhancer" che agisce tui punteggi di similarità/confidenza del Classificatore statistico/probabilistico modulandoli ulteriormente. In estrema semplificazione, diciamo che aumenterà il significato di una parola se questa risulta nella WhiteList (**WL**) mentre lo abbasserà se si trova nella BlackList (**BL**).

Evidentemente le WL e le BL contengono parole/termini che sono scelti in modo manuale esupervisionato dall'esperto e/o dal revisore. La loro rilevanza è soprattutto apprezzata quando la classificazione non devesse riconoscere il documento o divesse confonderlo tra diverse categorie.

PROGETTO TECNOLOGICO

Dopo il capitolo relativo al modello teorico e alla descrizione degli algoritmi di calcolo utilizzati nel progetto CAIMANS, questo capitolo si focalizza sugli aspetti di formalizzazione progettuale delle tecnologie di software per lo sviluppo.

Questa parte dello studio è comunque espresso in modo divulgativo mentre per i contenuti riferiti elettivamente alla programmazione sia essa di sistema o applicativa, si ricorre al sistema articolato di allegati alcuni dei quali per motivi di tutela del diritto di autore, sono secretati e/o vincolati nella stampa cartacea.

Nella progettazione architetturale di un Classificatore (l'algoritmo implementato) il primo passo di una CA è la costruzione del modello che essenzialmente è basato sulla preparazione e l'analisi dei campioni di documenti o del gruppo di campioni raccolti e distinti in modo supervisionato : il Training Set o **TS**

Nel secondo passo troviamo la fase di Testing o Classificazione propriamente detta durante la quale il modello di calcolo implementato nella base di conoscenza del primo passo di costruzione viene inferito con campioni nuovi e/o esterni per saggiare le calibrazioni di Pattern Maching o riconoscimento dei contenuti talchè si possa ottenere una selezione di risposte alla inferenza che siano ordinate secondo un ordine di similarità (o probabilità di verosimiglianza) [A1 - 7,8].

Il Classificatore deve capire del documento testuale ad esempio il formato, l'idioma, lo stile tipografico. (vedi prec Cap. Bag of Words, **BoW**). Ognuno di questi aspetti o caratteristiche (*Features vector*) l Classificatore implemeta diverse tecniche e algoritmi che possano trasformare in una rappresentazione geometrica spaziale denominata *Vector Space Model* [A1 - 4,6,9].

In CAIMANS le principali tecniche studiate per la CA sono :

- Nearest Neighbour classifier (KNN)
- Likelihood ratio (Laplace correction)
- Bayesian Classification
- Logistic Regression
- Support Vector Machine (§)
- Centroid Based Classification
- Decision Tree Induction (§)

Notare che le tecniche contrassegnate con (§) sono solo state valutate ma non implementate.

Complessivamente considerate le metodiche sopra descritte definiscono l'ambito del Supervised Machine Learning o **SML.** Per i dettagli teorici si rimanda alla bibliografia applicabile mentre, nei paragrafi che seguono vedremo riprendiamo ognuno di questi modelli di calcolo delineando il "come" e il "perché cosa" sono stati implementati nella piattaforma.

Si partirà cioè dalla descrizione di un aspetto problematico per volta mostrando come venga risolto praticamente. Ovviamente la spiegazione rimane sul livello concettuale e non tratta gli aspetti implementativi del codice di sviluppo software.

Modellazione della piattaforma

La piattaforma è strutturata in diversi moduli e *frameworks* e ognuno dei due insiemi di software può avere applicazioni custom/verticali, console GUI-less
Una introduzione più dettagliata è stata descritta secondo un schema a blocchi nel documento "*Schema fattuale del progetto preliminare*" (Sinopsi allegati)

Una importante premessa per la lettura che segue riguarda la definizione della Generazione Testuale.

In sintesi abbiamo la seguente sequenza di forme testuali a partire dall'documento originale

(1.0) Corpus → Generazione Zero → Generazione 1 → Generazione 2 → Generazione 3

Il primo passaggio riguarda l'applicazione di algoritmi di filtro quali : estrazione citazioni, lemmizzazione e Stemmizzazione. Queste tecniche sono descritte in dettaglio più avanti.

Pertanto dal testo originale o Corpus si ottiene una collezione di termini basilare o Generazione Zero. Il risultato di questo processo subisce quindi in cascata una serie di trasformazioni successive che producono una serie di collezioni di parole o termini singoli, a coppie e a triplette. Questi elementi sono anche definiti uno-gram, due gram e tre –gram. L'insieme delle tre collezioni, complessivamente considerate forma il DATAGRAM del documento considerato e viene persistito nel tracciato record di un database.

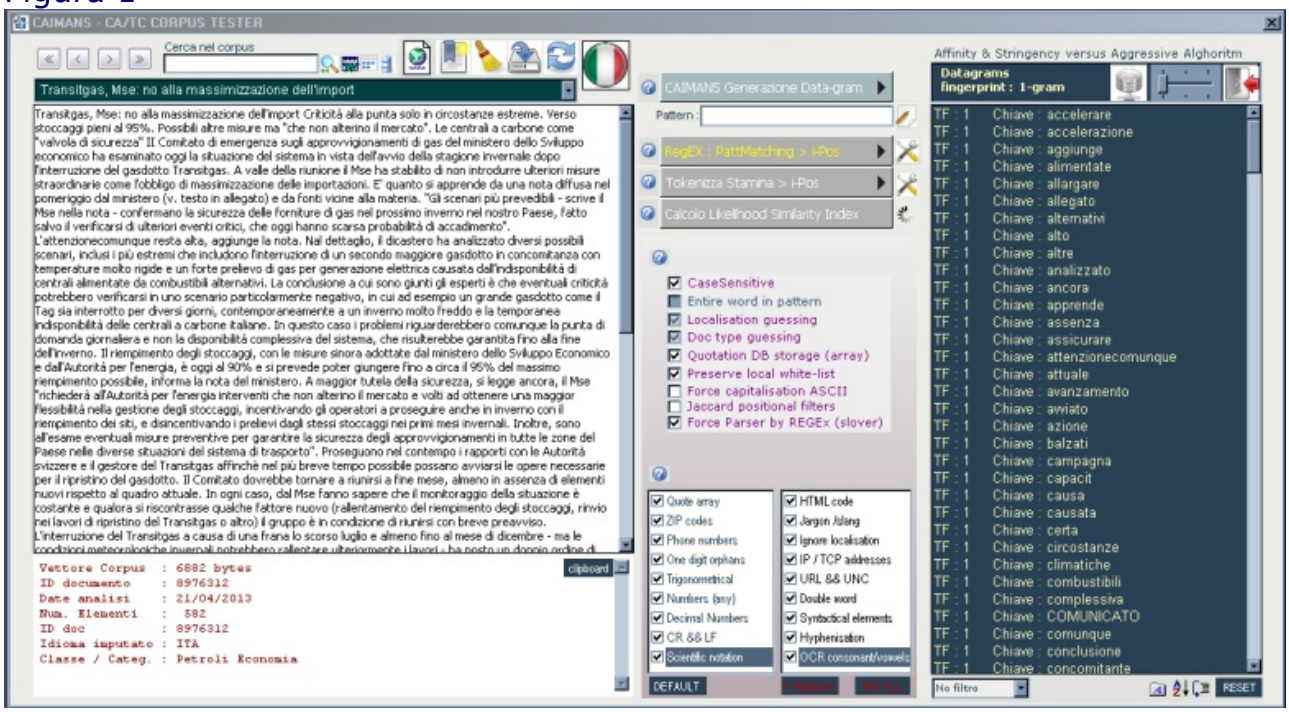
IL MODULO CORPUS TESTER CA/TC

Dal tempo della versione 1.2.0344 in poi di questo documento è stato distribuito anche un video relativo al Modulo Tester CA/TC visualizzabile dal sito dell'autove (vedi Sinopsi degli Allegati)

Il **TesterCA** è una interfaccia preliminare orientata alla analisi di un singolo articolo/corpus. Questo modulo è essenziale per costruire il Pattern di algoritmi che costituiranno i processi di filtro e calcolo espressi in seguito dal motore di classificazione che in CAIMANS abbiamo chiamato ClassificatoreTS (dove TS indica TrainingSet) e che vedremo in seguito.

Il front-end del **TesterCA** è diviso in tre pannelli o aree. In quella di sinistra si gestisce il Corpus di un articolo secondo caratteristiche strutturali (Es.: lunghezza, n. elementi ossi termini crudi e idioma candidato)

Figura 1



Passiamo in rassegna le descrizioni delle funzionalità implementate nel Modulo Corpus CA/TC motivandone le scelte degli algoritmi utilizzati e i principi di funzionamento.

Nazionalizzazione : Idioms Guesser

Caimans è dotato di un modulo CA in grado di stabilire l'idioma linguistico dei testi trattati. Questo viene realizzato con un semplice algoritmo che scandisce le parole estratte dal testo pulito (Vedi Testo Generazione Zero sopra)
Il Guesser utilizza l'algoritmo di Porter che a sua volta si appoggia a dizionari di StopWord utilizzati anche per la Lemmizzazione e lo Stemming (Figura 2, Porter-2004).

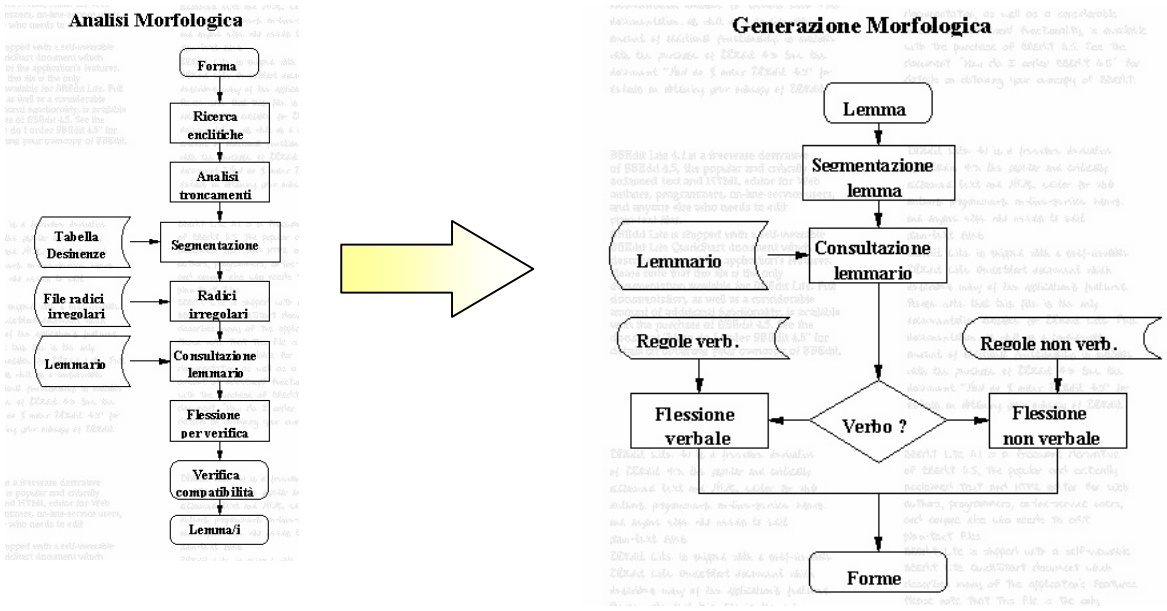
L'idiom Guesser si applica ancora prima che il Corpus originale intraprenda il passaggio di cui alla notazione (1.0) per applicare l'algoritmo di Porter con successo visto che bisogna rispettare il testo originale con costrutti sintattici peculiari della lingua.

In CAIMANS l'algoritmo *tout court* è stato costruito a scopo di validazione rispetto ad alcuni testi di riferimento dalla letteratura, tuttavia non implementato perché incompatibile con le esigenze di performance.

Come equivalente soluzione (sovrapponibile fino al 98% del Lemmizzatore di Porter) stato adottato un processore di filtro incrementale basato su Dizionario di *stop words*. Questa tecnica è estremamente più veloce pur rimanendo efficace nella individuazione della lingua. Nella Figura 1 relativa al TesterCA l'idioma viene indicato con una icona corrispondente alla bandiera associata. Su un campione di 88 articoli tutti i dizionari instrumentati in CAIMANS hanno riconosciuto con successo la lingua localizzata.

Allo stato rimane solo un margine residuo di incertezza sulla distinzione dell' Inglese Britannico verso quello statunitense o superficialmente definito internazionale. Questa sfumatura è comunque marginale rispetto agli scopi classificativi semantici delle categorie

Figura 2



Estrazione delle citazioni

Un altro processo preliminare alle Generazioni delle collezioni n-Gram è quello della estrazione delle Citazioni.
E' noto che in un testo potrebbero essere quotate delle frasi riportate in modo letterale. Questo tipo di contenuti deve essere identificato prima della applicazione al testo di algoritmi di elaborazione perché altrimenti andrebbero perduti.

Un domani ricercando con la nostra inferenza sarà sempre utile poter risalire ad una ricerca diretta. Se cioè in vece di una query per parole chiave si intende ricercare esattamente un testo corrispondente alla citazione, sarà sempre possibile un matching di tipo diretto.

Sappiamo che un periodo riportato come citazione tipicamente è delimitato dalle virgolette ("…"), già dopo il primo passo di sequenza generazionale del testo (**G0**) tutti gli elementi sintattici e lessicali del documento originale vanno perduti.

Preprocesso lessicale : Pruning and cleaning

La prima analisi di filtro applicata al documento di partenza comporta la rimozione delle parole e dei simboli sintattici e lessicali.

Seppure semplice da definire, questo primo algoritmo è molto critico perché le scelte operate possono poi risultare un un sitema di Information Retrieval più o meno efficiente.

Tipicamente nel progetto CAIMAS si sono filtrati i seguenti elementi perché non sono indicativi della morfolia e del contenuto testuale, sia esso semantico che deterministico :

- Punteggiatura e notazioni (&$%£?"! ecc)
- Lemmi sintattici (articoli, pronomi, verbi asusiliani ecc)
- Indirizzi IP e indirizzi email
- Numeri (incluso notazioni esponenziali e unità di misura, es.: E-23)
- Stop Words (dizionari idioma specifici tratti da WordNet Princeton University)

La procedure di PreProcesso per il filtering del contenuto originale è molto critica perché va ad impattare sul tempo di computazione ed è onerosa in ragione del numero di algoritmi di Pruning eCleaning adottati, tuttavia è importante rimuovere gli elementi che non saranno in grado di aggiungere una Caratteristica peculiare semantica e nella logica dello spazio vettoriale come rappresentazione geometrica del contenuto testuale, in fase di calcolo puro dei TF e del peso di ogni termine, chiaramente economizza nella parte del processo vero e Proprio.

Anticipiamo in questo tema anche la presenza di un PostProcesso, altrettanto critico perché deputato alla validazione dei vettori e ad un ulteriore filtraggio per la scrematura dei risultati.

IL MODULO GETS

Da un punto di vista Modulare la piattaforma è dotata di funzionalità separate seppure integrate e interoperabili. Abbiamo visto sopra il Modulo del Tester CA/TF che produte una tabella DATAMART con gli "stampini" vettoriali del nostro articolo.

Abbiamo anche spiegato che lo scopo ultimo del modello CA è quello di eleborare un Training Set, ossia trattare i DATAGRAMs di svariati articoli di campione. Questo è ciò che viene deputato al modulo GETS il cuo acronimo esploso vuol dire appunto Gestione Training Sets

Il modulo GETS lavora sul prodotto del Modulo Trasponder che linearizza la struttura dati di origine in un formato DATAMART. Il tabellone risultante contiene records di articoli ridondanti per categorie/classi e per caratteristiche accessorie (Autori, settori ecc.)

Quando si usa un approccio DATAMART per creare gruppi di TS essenzialmente si inferenziano dei sub-recordset che contengono gruppi di articoli