

UN MOTORE DI RICERCA SEMANTICO PER LA GESTIONE DELLE POLITICHE DI INNOVAZIONE

Sommario

1. Sistemi basati su dati non strutturati.....	6
1.1.1 Sentence Detection.....	7
1.1.2 Tokenization	7
1.1.3 Stop Words Removal	8
1.1.4 POS Tagging	8
1.1.5 Lemmatization & Stemming.....	8
1.1.6 Keyphrase Detection	9
1.2 Motori di ricerca classici.....	10
1.2.1 Indice invertito.....	11
1.2.2 Modello Bag of Words.....	12
1.2.3 Il Modello Booleano	13
1.2.4 Term Frequency e Inverse Document Frequency	14
1.2.5 Il Vector Space Model	15
1.3 Modelli Semantici Distribuzionali	18
1.3.1 Random Indexing	19
1.3.2 Motori di ricerca semantici.....	20
2. Il progetto TALIA	22
2.2.2 Motori di ricerca	27
2.2.3 Gestione dei metadati.....	27
2.3 Panoramica sui servizi del sistema di supporto alle decisioni	30
3. Servizi RESTful per la ricerca semantica.....	33
3.1 Introduzione alle architetture RESTful	33
3.2 Servizi RESTful impiegati.....	34
3.2.1 Creazione motore di ricerca semantico.....	34
3.2.1.1 NLP Pipeline	34
3.2.1.2 Estrazione delle phrases	35
3.2.1.3 Analyze	37
3.2.2 Gestione delle collezioni in modo automatizzato	39
3.2.3 Summarization.....	42
3.2.4 Related Concepts	42
3.2.5 Arricchimento con metadati.....	43
4 Implementazione del servizio di ricerca semantica	45
4.1 L'importanza di una buona interfaccia.....	45
4.2 Integrazione dei servizi del Semantic Framework	46

.....	52
<i>4.3 Visualizzazione geografica dei risultati di ricerca.....</i>	<i>53</i>
5. Conclusioni e sviluppi futuri	55
Riferimenti bibliografici.....	59

Introduzione

I sistemi di supporto alle decisioni sono sistemi informatici che hanno lo scopo di aumentare l'efficacia e l'efficienza dei processi decisionali in qualunque tipo di organizzazione. Il valore aggiunto apportato da questi sistemi ai processi decisionali consiste nell'offrire agli stakeholder la possibilità di tenere in considerazione informazioni e conoscenze relative al dominio di interesse che non sarebbero disponibili senza di essi, per poter compiere decisioni o organizzare processi in contesti parzialmente o completamente variabili e di difficile prevedibilità. I processi di ritrovamento e di elaborazione dell'informazione sono svolti in modo automatico a causa della grande quantità di informazioni e conoscenze prodotte nelle organizzazioni di medie e grandi dimensioni, che è, peraltro, conservata sempre più frequentemente solo in formato digitale.

La maggior parte dei sistemi di supporto alle decisioni impiegano risorse informatiche per elaborare informazioni strutturate, ossia dati che descrivono il dominio di interesse che solitamente conservati in grandi basi di dati o *data warehouses*. Queste analisi sono svolte tramite tecniche di *data mining* che hanno il fine di estrarre conoscenza precedentemente ignota a partire dai dati, applicando metodi algoritmici o statistici.

Una tipologia alternativa di sistemi di supporto alle decisioni è quella basata sull'estrazione di informazioni e conoscenza a partire dai dati non strutturati, ossia dai comuni testi scritti in linguaggio naturale. I sistemi più diffusi in questa categoria sono indubbiamente i motori di ricerca, che hanno l'obiettivo di reperire e mostrare all'utente i documenti di una collezione che sono rilevanti rispetto ad un particolare bisogno informativo. Il criterio con il quale questo genere di sistemi valuta la corrispondenza tra un documento e un bisogno informativo si basa solitamente sulla

valutazione dell'occorrenza nel documento di specifiche parole con cui esso è espresso. È anche possibile, però, applicare ai documenti tecniche di elaborazione che tengono in considerazione il significato dei termini. Attraverso queste tecniche è possibile sia creare motori di ricerca più sofisticati, sia progettare e sviluppare servizi basati sulla rappresentazione semantica del testo. Questo genere di servizi dà all'utente la possibilità di esaminare il contenuto dei documenti e fare confronti su di essi sfruttando il processo di analisi automatica. Ad esempio, è possibile individuare i documenti in una collezione che trattano di un tema, individuare i concetti simili tra loro sulla base di come vengono usati all'interno di una collezione, o, ancora, trovare i documenti di una collezione che parlano di temi simili.

Il seguente lavoro di tesi mira a costruire un sistema di analisi di tipo semantico ai fini del soddisfacimento del bisogno informativo dell'utente, tenendo anche in considerazione l'esperienza d'uso della piattaforma per garantire un elevato tasso di usabilità. L'estrazione di informazioni è stata effettuata utilizzando alcune tecniche di trattamento del testo che andremo a trattare in seguito all'interno della tesi. Attraverso l'aggiunta di ulteriori informazioni, organizzati all'interno di un database, si è puntato a migliorare i risultati delle ricerche effettuate dall'utente, passo fondamentale per la effettiva soddisfazione del bisogno informativo.

1. Sistemi basati su dati non strutturati

La maggior parte delle informazioni attualmente disponibili in formato digitale, siano esse di pubblico dominio o appartenenti a un'organizzazione, sono organizzate in forma non strutturata, ossia sottoforma di testi o frammenti di testo.

Dal punto di vista della rappresentazione della conoscenza, il testo è un formato radicalmente diverso rispetto a quello strutturato con cui sono memorizzate le informazioni in una base di dati, o rispetto alle regole logiche con le quali è codificata la conoscenza su un dominio in una base di conoscenza. Il testo è, infatti, una forma di rappresentazione della conoscenza prodotta da e per le persone, perciò esso va processato in modo opportuno affinché il suo contenuto informativo possa essere acquisito, processato e reso disponibile automaticamente, e non soltanto tramite la lettura. Esistono due livelli di rappresentazione digitale del testo: il più superficiale è basato solo sulla *forma* del testo e non sul suo *significato* (ossia, ad esempio, le parole “cappello” e “berretto”, pur avendo un significato molto simile, sono considerate diverse perché hanno una forma diversa); diversamente da questo, il livello di rappresentazione dei testi più profondo e completo tiene conto del significato delle parole.

Le possibilità aperte dal poter operare automaticamente sui testi sono significative: diventa possibile gestire grandi collezioni di testi, contenenti anche decine di milioni di parole. Tramite i motori di ricerca è possibile reperire documenti utili al loro interno cercando tramite parole chiave o a partire da “concetti”, se la rappresentazione del testo ne preserva la semantica.

1.1 Elaborazione del linguaggio naturale

I passi compiuti per preparare il testo ad essere rappresentato in forme appropriate per le operazioni di ritrovamento dell'informazione fanno parte di un processo detto di "elaborazione del linguaggio naturale" o Pipeline di Natural Language Processing (NLP). Esso comprende un insieme di passi che hanno il fine di aumentare l'efficacia delle tecniche di rappresentazione e analisi del testo, tanto eliminando le ridondanze tipiche di questo tipo di rappresentazione della conoscenza, quanto mettendone in risalto la peculiare ricchezza informativa. Le operazioni eseguite in questo processo sono alla base sia della rappresentazione "superficiale" del testo, sia di quella che tiene conto del significato dei termini.

1.1.1 Sentence Detection

Il primo passo nel processo di elaborazione del testo è l'individuazione delle frasi. Esse vengono individuate all'interno di un testo considerando che sono solitamente separate dal punto fermo. Tuttavia, è necessario distinguere questo da altri usi del punto che possono essere presenti nei testi considerati (ad esempio, per abbreviare termini, per separare porzioni di un numero, eccetera).

1.1.2 Tokenization

Il secondo passo nella pipeline di NLP è la suddivisione delle frasi in parole o "token", eliminando i segni di punteggiatura e i caratteri speciali. Anche in questo caso, l'euristica base con la quale si identificano i token è che essi sono generalmente separati da spazi; bisogna tuttavia tener conto di token in forme particolari, come le date e l'ora, i nomi propri di persona, ecc., che sono composti da più parti che non vanno considerate separatamente.

1.1.3 Stop Words Removal

Le *stop words* sono parole molto comuni in una lingua o in un particolare ambito, al punto tale che il loro contenuto informativo è considerato molto basso o nullo; per questa ragione un testo si può rappresentare efficacemente senza comprenderle risparmiando spazio di rappresentazione e tempo di computazione. Per le lingue più comuni esistono liste di *stop words* che vengono usate per filtrare le parole rilevanti nei documenti.

1.1.4 POS Tagging

Il Part Of Speech Tagging è l'operazione che associa a ciascun token la sua categoria grammaticale all'interno della frase. Questa operazione non è banale, poiché molto spesso uno stesso termine può avere più di un significato, o perché alcuni elementi sintattici nella frase sono sottintesi nell'uso comune della lingua. La disambiguazione tra i diversi ruoli sintattici e l'individuazione di quello corretto avviene considerando sia le definizioni del termine considerato, che, soprattutto, le parole assieme alle quali occorre e il loro ruolo all'interno della frase.

L'operazione di POS Tagging è particolarmente utile per la costruzione di una rappresentazione della sintassi del testo e per l'individuazione della corretta semantica associata ai termini, nel caso in cui questi abbiano un diverso significato a seconda del loro ruolo sintattico all'interno della frase.

1.1.5 Lemmatization & Stemming

Nei documenti sono spesso presenti diverse forme di una parola che hanno dei significati simili (democrazia, democratico, democratizzazione). La lemmatizzazione è un'operazione di semplificazione dei token che trasforma ciascuno di essi nella sua forma grammaticale di base (ossia il suo

lemma) trasformando il suffisso proprio della specifica inflessione presente nel testo in quello della forma base dell'elemento grammaticale. Il processo di trasformazione non è banale, in quanto va identificata correttamente la forma base di ogni token; in particolare, ciò è più difficile nel caso di termini che hanno la stessa forma ma funzioni grammaticali e significati differenti (es. un'ancora, egli ancora, ancora qui). In questi casi gli algoritmi di lemmatizzazione possono ricondurre il termine al lemma corretto grazie al contesto in cui occorre e al risultato del processo di POS tagging.

Lo stemming è un'operazione di semplificazione dei token più radicale: essa tronca il suffisso di ogni token riducendo il termine alla sua radice. Uno degli algoritmi più usati per compiere questa operazione è l'algoritmo di Porter [1].

1.1.6 Keyphrase Detection

Questo passo di analisi del testo consiste nell'identificare i concetti più rilevanti all'interno di un testo, che ne descrivono dunque l'argomento e il contenuto informativo; essi possono essere espressi anche con più di una parola (ad es. "information retrieval"). Un approccio semplice, che impiega una tecnica di apprendimento non supervisionato, si basa sull'identificazione delle coppie, triple o n-uple di termini che co-occorrono con una frequenza significativa [2].

1.1.7 Chunking

Il Chunking è un'operazione di analisi sintattica del testo che può essere vista come un'operazione di tokenizzazione più elaborata, o come un parsing più semplice (è, infatti, anche detta "shallow parsing"). Essa consiste nell'individuare gli elementi grammaticali che sono composti da più termini, come i nomi propri di luoghi o di persone o le forme verbali

che comprendono verbi ausiliari o modali. Ciò è utile perché è più appropriato considerare queste forme grammaticali in modo unitario per avere contezza del loro significato; ad esempio, dire che in un testo si parla di Sud Africa è diverso dal dire che si parla

1.2 Motori di ricerca classici

Uno degli strumenti più comuni e più utili che si può costruire su una collezione digitale di documenti, processati attraverso le operazioni appena descritte, è il motore di ricerca. Esso è un sistema che, dato un insieme di parole – detto “query” – che esprime un bisogno informativo dell’utente, ha l’obiettivo di restituire ad esso i documenti della collezione che sono più rilevanti rispetto ai termini della query. Questo task cambia significativamente in relazione al tipo di rappresentazione del testo adottata. Nel caso in cui i documenti siano rappresentati tenendo conto della semantica dei termini al loro interno, infatti, i documenti più rilevanti per una query saranno quelli che contengono parole dal significato più simile a quelle date in input. Nel caso più semplice in cui i documenti siano rappresentati senza tener conto del significato dei termini, il calcolo della similarità tra documenti e query si basa sull’occorrenza delle parole della query nei documenti, ed eventualmente sulla frequenza con cui queste occorrono.

In questo paragrafo sono considerati i motori di ricerca “classici”, ossia basati solo sulla forma dei termini e non sul loro significato. Esistono diversi approcci per creare un motore di ricerca classico; essi differiscono principalmente per il modo in cui sono rappresentati i documenti all’interno del sistema e per il modo in cui viene calcolata la similarità tra essi e le query dell’utente. Sono presentate di seguito le strutture dati e le

assunzioni sulla rappresentazione dei documenti comuni a diversi tipi di motori di ricerca; viene poi brevemente presentato il modello Booleano e il più diffuso Modello a Spazio Vettoriale.

1.2.1 Indice invertito

L'operazione più basilare ed intuitiva che un motore di ricerca classico deve compiere per risolvere una query è individuare quali sono i documenti della collezione che contengono le parole presenti in essa. L'insieme delle parole presenti in tutti i documenti di una collezione, associati alla loro frequenza assoluta, è detto "vocabolario" della collezione. Per compiere l'operazione di ritrovamento dei documenti in modo efficiente i motori di ricerca operano su una struttura dati chiamata "indice invertito". Il suo nome deriva dal confronto con il normale indice di un libro, che mette in relazione i capitoli o i paragrafi di un testo con le pagine in cui si trovano, e quindi con le parole contenute in essi. L'indice invertito è, invece, una struttura dati che associa ad ogni parola del vocabolario i documenti nei quali essa compare, consentendo così di individuare i sottoinsiemi di documenti in cui sono presenti tutte o molte delle parole in una query.

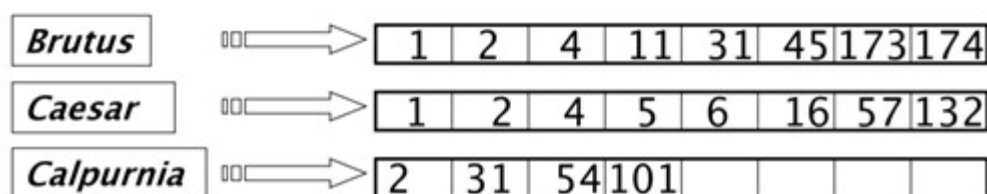


Figura 1.1 Un esempio di indice invertito, nel quale i numeri identificano i documenti della collezione

1.2.2 Modello Bag of Words

Il modello Bag of Words è un metodo di rappresentazione di un testo che tiene conto del numero di occorrenze di ogni parola nel testo ma non delle posizioni che ciascuna occupa in esso. Ciò implica che due o più testi che contengono le stesse parole ma in diverso ordine, e che hanno quindi complessivamente significati diversi, saranno rappresentati nello stesso modo. Questo modello si basa sull'assunzione forte che la posizione di una parola nelle frasi del testo in cui compare non sia rilevante per stabilirne il significato e, dunque, per interpretare il contenuto informativo del testo. Sebbene questa assunzione sia errata, e si punti, infatti, a sopperire alle sue mancanze tramite le operazioni di analisi sintattica del testo, come il Chunking e il POS Tagging, il motivo per cui si adotta è che la posizione delle parole in un testo non risulta empiricamente rilevante per il ritrovamento dei documenti attinenti alla richiesta espressa con una query.

Document at index 0		
Document index	(0, 46)	1
	(0, 48)	1
	(0, 81)	1
	(0, 28)	1
	(0, 15)	1
	(0, 27)	1
	(0, 17)	1
	(0, 23)	2
	(0, 59)	1
	(0, 53)	1
	(0, 56)	1
	(0, 6)	1
Term index	Term count	

Figura 1.2 Rappresentazione secondo il modello Bag of Words di un documento identificato dall'indice 0

1.2.3 Il Modello Booleano

Adottando una rappresentazione conforme al modello Bag of Words, un documento sarà dunque rappresentato come un multiinsieme di parole, ossia un insieme nel quale gli elementi possono apparire più volte. Il modello booleano [3, 4] implementa nel modo più semplice possibile questa rappresentazione, considerando, cioè, la semplice presenza o assenza di un termine in un documento senza considerarne la frequenza d'uso. Un documento è rappresentato, quindi, come un vettore di variabili booleane, nel quale ogni elemento rappresenta un termine del vocabolario e assume il valore *vero* se la parola appare nel documento, e il valore *falso* altrimenti. Una query per un motore di ricerca costruito secondo questo modello può essere espressa collegando i termini con operatori booleani (es. *termine1 AND termine2 OR termine3*) e la risposta ad essa corrisponde al sottoinsieme di documenti della collezione la cui rappresentazione vettoriale soddisfa le condizioni espresse nella formula logica della query. Ogni documento sarà dunque semplicemente rilevante o non rilevante per una query e non vi sarà associata una misura di rilevanza; per questo motivo, i risultati di una ricerca non vengono restituiti secondo un ordinamento significativo.

Pur essendo un modello che consente di esprimere interrogazioni con precisione, la forma logica delle query costituisce sia un ostacolo sintattico per l'utente, che deve saper esprimere il proprio bisogno informativo in questa forma, sia un vincolo spesso troppo – o troppo poco – stringente nel processo di individuazione dei risultati rilevanti, in quanto i documenti che corrispondono precisamente alla query possono essere molto pochi, quando la query è molto specifica, o troppi, se comprende molti casi ed è espressa tramite un insieme condizioni disgiunte.

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calpurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0

Figura 1.3 Rappresentazione di una collezione (documenti sulle colonne) secondo il modello Booleano (termini sulle righe)

1.2.4 Term Frequency e Inverse Document Frequency

Per arricchire la rappresentazione di un documento utilizzata nel modello booleano si può tenere conto di due parametri relativi ad ogni termine del vocabolario: la frequenza con cui ogni termine appare in ciascun documento e la rarità di un termine rispetto all'intera collezione. L'idea alla base dell'introduzione di queste misure è che un documento sarà più rilevante rispetto ad una query se uno o più termini della query sono molto frequenti nel documento, oppure se uno o più termini che si trovano nella query sono presenti solo in quel documento o in pochi altri. La frequenza di una parola in un documento è detta Term Frequency [5]; matematicamente essa è definita come la frequenza assoluta di occorrenza del termine nel documento. Generalmente, questa misura non viene direttamente impiegata per calcolare la rilevanza di un documento rispetto a una query, poiché porterebbe a far aumentare il peso di un termine in un documento in modo spropositato, facendo risultare il documento di una rilevanza rispetto alla query maggiore di quella reale (un documento in cui un termine appaia 10 volte non dovrebbe essere 10 volte più significativo di uno in cui esso appare una volta sola). Per questa ragione, si impiega la funzione logaritmo per ridurre l'impatto della crescita del Term Frequency

sulla funzione di scoring; l'operazione applicata è detta *sublinear tf-scaling* ed è definita matematicamente come segue:

$$sub_tfs(t, d) = \begin{cases} 1 + \log_{10} tf_{t,d}, & tf_{t,d} > 0 \\ 0, & tf_{t,d} \leq 0 \end{cases}$$

dove t è un generico termine e d è un generico documento considerato.

La rarità di un termine in una collezione è definita matematicamente sulla base del concetto di Document Frequency di un termine, ossia il numero di documenti in cui il termine appare. Questa misura è, ovviamente, inversamente proporzionale alla rarità di un termine; la rarità viene quindi misurata da una misura chiamata Inverse Document Frequency [6], che è definita matematicamente come:

$$idf_t = \log\left(\frac{N}{df}\right),$$

dove N è il numero di documenti nella collezione e df è definito come il Document Frequency del termine t .

Ad ogni termine in un documento può essere associato, dunque, un peso che è direttamente proporzionale sia alla frequenza del termine nel documento stesso, sia alla rarità del termine nella collezione; esso è il prodotto tra Term Frequency e Inverse Document Frequency del termine, ed è chiamato peso *tf-idf* del termine [7]:

$$w_{t,d} = sub_tfs(w, d) \cdot idf_t$$

1.2.5 Il Vector Space Model

Il Modello a Spazio Vettoriale [8] è una tecnica di rappresentazione dei testi nella quale documenti e query sono rappresentati come vettori in uno stesso spazio multidimensionale, nel quale ogni dimensione rappresenta un

termine nel vocabolario della collezione. I testi sono quindi rappresentati da entità matematiche di tipo diverso rispetto al modello booleano, e di queste entità vengono sfruttate le proprietà per il calcolo della similarità tra query e documenti. Il modello integra, inoltre, le misure presentate nel paragrafo precedente: infatti, l'i-esimo elemento di un vettore che

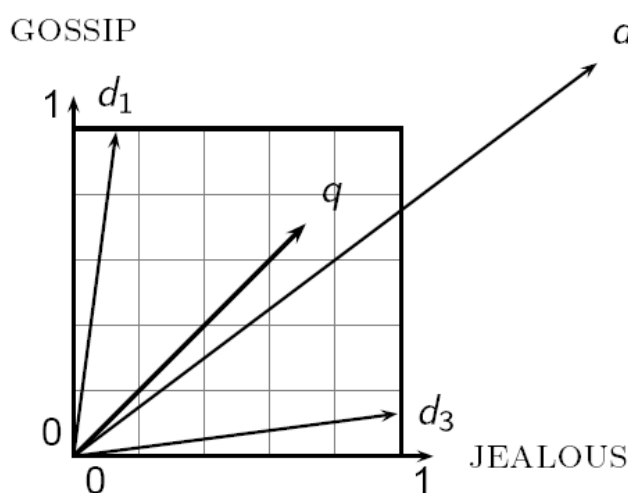


Figura 1.4 Rappresentazione dei documenti come vettori di lunghezza non normalizzata

rappresenta un documento (o una query) ha come valore il peso associato al termine corrispondente rispetto a quel documento, calcolato in funzione della sua Term Frequency ed Inverse Document Frequency come descritto nel paragrafo precedente.

Nella rappresentazione secondo il Modello a Spazio Vettoriale è possibile comparare documenti e query utilizzando operazioni vettoriali. Una prima possibilità per calcolare la similarità tra due documenti è considerare la loro distanza Euclidea nello spazio multidimensionale. Questa è definita come segue:

$$d(\vec{q}, \vec{p}) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2},$$

dove q_i e p_i sono gli elementi in posizione i-esima dei generici vettori \vec{q} e \vec{p} . Usando questa metrica, però, due documenti, contenenti gli stessi termini

ma di dimensioni diverse a causa della ripetizione di alcuni termini in uno di essi, saranno considerati diversi perché distanti nello spazio. È necessario, quindi, compiere una normalizzazione della dimensione dei vettori. Questa operazione è compresa nel calcolo della misura di similarità del coseno, che valuta la distanza tra documenti sulla base dell'angolo compreso tra i vettori che li rappresentano. La metrica è definita come segue:

$$\cos(\vec{q}, \vec{d}) = \frac{\vec{q} \cdot \vec{d}}{|\vec{q}| |\vec{d}|} = \frac{\vec{q}}{|\vec{q}|} \cdot \frac{\vec{d}}{|\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sum_{i=1}^{|V|} q_i^2 \sum_{i=1}^{|V|} d_i^2}.$$

La normalizzazione rispetto alla lunghezza consiste nel dividere il prodotto interno tra i vettori per le loro norme.

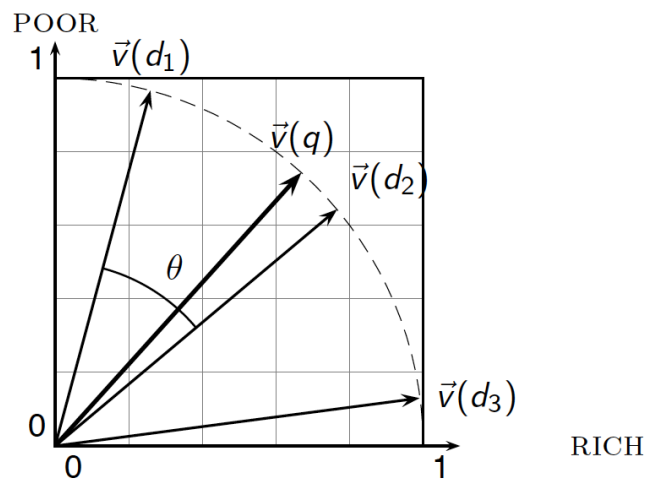


Figura 1.5 Rappresentazione della distanza tra vettori normalizzati basata sull'angolo compreso tra essi

Il Vector Space Model presenta comunque alcune problematiche:

1. come in tutte le tecniche di rappresentazione basate sul modello Bag of Words, la posizione dei termini non viene presa in considerazione;
2. documenti molto lunghi possono essere penalizzati nel processo di ritrovamento poiché tenderanno ad avere score di similarità più bassi a causa dell'operazione di normalizzazione delle lunghezze;

3. poiché viene usato il prodotto interno per confrontare i vettori, il modello può essere penalizzato dalla sparsità dei dati.

Inoltre, il modello a spazio vettoriale non può rilevare la similarità tra due documenti che abbiano un contenuto semanticamente simile ma contengano parole differenti, poiché la rappresentazione dei documenti è basata solo sulla forma delle parole che contengono e non sul loro significato.

1.3 Modelli Semantici Distribuzionali

Rappresentare il testo tenendo conto del significato delle parole ha un impatto significativo sulle operazioni di ritrovamento di documenti rilevanti rispetto a una query. Avere a disposizione una rappresentazione del significato delle parole usate in una collezione di documenti permette di calcolare la similarità tra una query e un documento non soltanto in base alle parole della query che occorrono nel documento, ma anche considerando le parole in esso che hanno un significato simile a quelle date in input.

Per creare una rappresentazione del significato delle parole si utilizzano spesso modelli matematici che le rappresentano come vettori in uno spazio multidimensionale, nel quale quelle che hanno un significato simile sono rappresentate tramite vettori “vicini” tra loro; lo spazio multidimensionale viene detto Word Space o Semantic Space. In questo spazio si può misurare la similarità tra concetti calcolando il coseno dell’angolo compreso tra essi. L’assunzione fondamentale su cui si basano tutti i modelli semantici distribuzionali è che i termini che occorrono frequentemente in contesti simili – e raramente in contesti diversi – hanno un significato simile [9].

Esistono diversi metodi per creare lo spazio vettoriale nel quale rappresentare il significato delle parole; di seguito è presentata la tecnica chiamata Random Indexing.

1.3.1 Random Indexing

Il Random Indexing [11] è una delle tecniche di creazione di un Word Space su una collezione di documenti. Essa ha il vantaggio di generare una rappresentazione del significato delle parole senza utilizzare fonti di conoscenza esterne (come dizionari o ontologie) e in modo incrementale, ossia riuscendo ad aggiornare la rappresentazione in modo semplice all'aggiunta di nuovi documenti nella collezione; l'unica operazione di preprocessing del testo necessaria per costruire lo spazio è la tokenizzazione.

La tecnica consiste in due passi principali: il primo consiste nell'assegnare a ogni termine del vocabolario di una collezione un vettore ternario e sparso generato casualmente, ossia un vettore i cui elementi possono assumere i valori -1, 0 e 1, nel quale la maggior parte degli elementi hanno valore 0 e nel quale le posizioni degli elementi non nulli sono scelte in modo casuale. Il secondo passo consiste nel generare un "vettore contesto" per ogni termine nel vocabolario sommando i vettori precedentemente associati alle parole che occorrono frequentemente insieme ad esso nella collezione, in base ad una soglia di vicinanza prefissata. Il vettore contesto di un termine è calcolato secondo la formula seguente:

$$\overrightarrow{cv_i} = \sum_{d \in C} \sum_j \vec{r}_j, \quad -c < j < c, i \neq j,$$

dove C è la collezione di documenti, c è la soglia di vicinanza dei termini prefissata che delimita il contesto di ogni parola e \vec{r}_j è il vettore casuale

assegnato ad ogni parola presente nel contesto. Il vettore contesto di un termine è di fatto la rappresentazione del suo significato nello spazio multidimensionale, ed è un'implementazione algebrica del principio alla base dei modelli semantici distribuzionali enunciato in precedenza.

La creazione di un vettore contesto per ogni parola corrisponde formalmente ad un'operazione di riduzione della dimensionalità della matrice di co-occorrenza dei termini in una collezione. Questa operazione genera uno spazio di dimensionalità ridotta nel quale le distanze tra i vettori che rappresentano le parole restano proporzionali alle distanze che le parole avevano nello spazio originale [10].

1.3.2 Motori di ricerca semantici

Usando la tecnica di Random Indexing è possibile rappresentare termini e documenti nello stesso spazio vettoriale multidimensionale. Si può generare, infatti, una rappresentazione vettoriale di un documento tramite la somma pesata di tutti i vettori che rappresentano le parole che contiene, usando come pesi gli indici di Inverse Document Frequency associati ad ogni parola; in questo modo è aumentata la rilevanza nella rappresentazione di un documento dei termini che sono meno comuni nella collezione.

Questa rappresentazione vettoriale di un documento nello stesso spazio in cui sono rappresentati i termini permette di calcolare la similarità semantica tra un termine e un documento o tra due documenti, oltre che tra due termini. Questa possibilità è alla base del funzionamento di un motore di ricerca semantico, ossia che tiene conto del significato dei termini. Se è possibile rappresentare un documento nello spazio vettoriale, infatti, si può rappresentare anche la query di un utente come il vettore somma dei

termini presenti in essa. Si può, dunque, calcolare la rilevanza dei documenti presenti nella collezione rispetto alla query come similarità del coseno tra il vettore che rappresenta la query e quelli che rappresentano i documenti; si può poi, come di consueto, ordinare i documenti in base alla rilevanza rispetto alla query, mostrando i primi k risultati all'utente.

2. Il progetto TALIA

Il progetto TALIA – acronimo di “Territorial Appropriation of Leading-edge Innovation Actions” – è un progetto del quale è capofila la Regione Puglia, finanziato dalla Commissione Europea nell’ambito di una call del programma Interreg-Med. Gli obiettivi più ampi della call erano la promozione dello sviluppo sostenibile supportato dalla tecnologia nell’area del Mediterraneo, e, più specificatamente, l’aumento della capacità di comunicazione e cooperazione tra gli attori principali nei più importanti settori socioeconomici nell’UE.

Il programma Interreg-Med comprende diverse decine di progetti denominati “verticali”, portati avanti da imprese, enti pubblici e associazioni in diverse nazioni dell’area del mediterraneo. Questi progetti sono raggruppati, sulla base dei temi che affrontano, in nove gruppi chiamati “Community”: Blue Growth, Green Growth, Social and Creative, Efficient Buildings, Renewable Energy, Urban Transports, Sustainable Tourism, Biodiversity Protection, Governance. A loro volta, le Community sono raggruppate in tre Assi, ciascuno dei quali comprende le Community con obiettivi comuni: Innovation Axis, Low Carbon Economy Axis, Natural and Cultural Resources Axis.

2.1 Obiettivi del progetto e soluzione adottata

Il progetto TALIA è trasversale ai progetti verticali del programma Med che afferiscono alla Community Social and Creative. Esso è stato sviluppato partendo dall’assunzione che fosse necessario andare oltre l’analisi dei singoli progetti verticali ed esplorare le loro possibilità di scalabilità mettendo ciascuno in relazione con gli altri. Per scalabilità si intende la capacità dei

risultati dei progetti di raggiungere un numero maggiore di beneficiari nel tempo e nello spazio. Attraverso il confronto dei risultati ottenuti dai diversi progetti e delle prassi adottate in ciascuno di essi, in relazione al contesto territoriale in cui sono inseriti, il progetto mira al miglioramento e alla crescita di ciascuno, anche tramite l'instaurazione di partnership e contaminazioni tra progetti simili, ma potenzialmente svolti in due luoghi distanti nell'area del mediterraneo.

Per raggiungere questi obiettivi, nell'ambito del progetto TALIA si è scelto di sviluppare un sistema di supporto decisionale basato sull'analisi intelligente del contenuto testuale dei documenti scritti in ogni progetto finanziato nell'ambito del programma Interreg-Med. Il sistema di supporto decisionale è basato sul Semantic Framework, una piattaforma per l'indicizzazione e l'analisi semantica dei documenti. Esso ha come utenti potenziali diversi tipi di soggetti coinvolti nell'ambito del programma Med:

- in primo luogo, i policy maker europei, chiamati a decidere l'indirizzo da prendere nelle call che stanzeranno in futuro i fondi comunitari destinati a quest'area di sviluppo; per questi, può essere rilevante tener conto dell'analisi automatica dei documenti prodotti nell'ambito di una o più Community del programma Med per comprendere, ad esempio, come queste abbiano operato, quali siano i temi principali intorno ai quali si è concentrata l'azione dei Partner di ogni progetto, e come diversi partner e progetti possano in futuro cooperare tra loro.
- In secondo luogo, i partner di ciascun progetto verticale possono beneficiare dai servizi di analisi automatica dei documenti per confrontare il proprio operato con quello svolto da altri soggetti in progetti

con obiettivi simili ai propri, per poter migliorare il proprio operato o valutare la possibilità di stabilire una collaborazione con essi.

- Infine, il sistema di supporto decisionale può supportare gli stakeholder di ciascun progetto, che possono essere enti pubblici, privati o singoli cittadini, a valutare l'andamento e l'impatto dei progetti operanti in aree di loro interesse specifico; ciò permette, dunque, al progetto TALIA di avere una maggiore capacità di coinvolgimento degli stakeholders sui territori, aumentando anche le potenzialità di ciascun progetto di innescare processi di crescita socioeconomica nelle aree in cui esso opera.

In particolare, il progetto TALIA, grazie al sistema di supporto alle decisioni basato sul Semantic Framework, mira a costruire e sviluppare la comunità Social and Creative del programma Interreg-Med; essa ha l'obiettivo di promuovere cluster di innovazione distribuiti tra le nazioni dell'area del mediterraneo fornendo strumenti che permettono la connessione di progetti trans-nazionali del programma Med con le comunità locali, a partire dalle regioni dei partner di ogni progetto.

2.2 Semantic Framework: architettura e funzionalità

Lo scopo del Semantic Framework è quello di rendere accessibile la conoscenza e le informazioni presenti nei deliverable di progetto prodotti nei progetti del programma MED. Il Framework si può considerare la parte di back-end del sistema di supporto alle decisioni sviluppato nel progetto TALIA, poiché offre le funzionalità per l'estrazione di informazioni che verranno poi utilizzate per soddisfare i bisogni informativi degli utenti.

Il Framework consente di gestire collezioni di documenti e di applicare ad ogni documento le operazioni di elaborazione del linguaggio naturale e di rappresentazione vettoriale e semantica descritte nel capitolo precedente. Esso è stato sviluppato secondo un'architettura client-server che permette l'accesso ai risultati del processo di elaborazione dei documenti tramite Web API sviluppate secondo il protocollo REST.

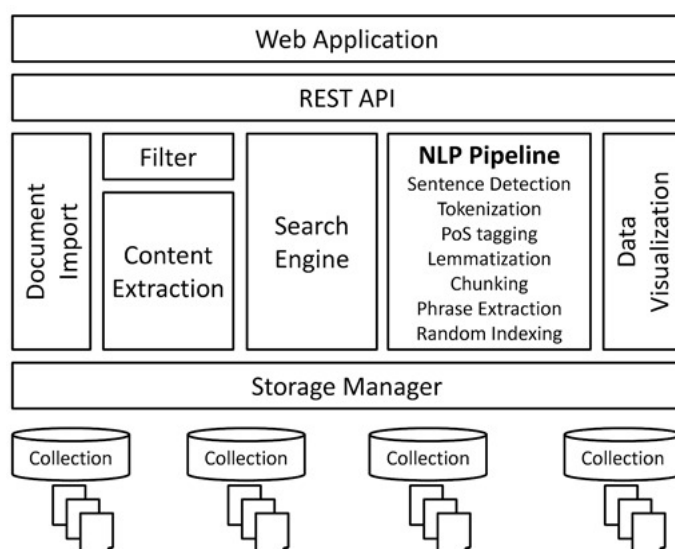


Figura 2.1 Architettura del Semantic Framework

2.2.1 Gestione delle Collezioni

La componente alla base del Semantic Framework, chiamata Storage Manager, è quella che consente di creare e gestire collezioni di documenti, ed è stata implementata grazie alla nota libreria Java Apache Lucene per la creazione di motori di ricerca. Nell'ambito dell'applicazione del Framework al progetto TALIA è stata creata una collezione di documenti per ogni Community relativa al programma Interreg-Med, e ciascuna di esse contiene tutti i deliverable di progetto che sono stati scritti nell'ambito di tutti progetti afferenti a ciascuna Community.

I deliverable di ogni progetto sono stati recuperati attraverso l'ausilio di un servizio software esterno al Semantic Framework che ha effettuato il crawling e lo scraping delle pagine web ufficiali del programma Med, scaricando i file dei deliverable e i metadati relativi a ogni progetto codificandoli in formato json. Il caricamento dei documenti in ciascuna collezione è avvenuto in seguito a un processo di selezione dei documenti significativi per ogni Community, al fine di ottenere collezioni contenenti esclusivamente documenti in lingua inglese o italiana, con un contenuto informativo utile, e privi di immagini, condizioni necessarie affinché il processo automatico di estrazione e analisi del testo desse risultati corretti, comprensibili e significativi. Per facilitare il processo di selezione dei documenti e di caricamento di quelli significativi nella relativa collezione è stato sviluppato un ulteriore servizio web esterno alla piattaforma, che permette di visualizzare i documenti estratti tramite il processo di crawling, selezionare manualmente quelli di interesse dell'utente e caricarli automaticamente nel Semantic Framework attraverso i servizi REST di gestione delle collezioni.

Attraverso i servizi REST esposti dal Framework, ogni collezione può essere creata specificandone il nome e la lingua – informazione necessaria, quest'ultima, durante il processo di NLP. Oltre a poter creare nuove collezioni vuote e poi aggiungervi documenti, è possibile generarne di nuove attraverso la fusione di due o più collezioni già esistenti. Sono state create in questo modo, ad esempio, le collezioni di documenti relative agli Assi tematici del programma MED. Per ogni collezione possono essere aggiunti o rimossi documenti; al momento del caricamento in una collezione, ogni documento può essere strutturato in sezioni (come "titolo", "autore", "corpo", ecc.), grazie alle funzionalità messe a disposizione dalla

libreria Lucene. Un documento può essere aggiunto in formato CSV o PDF; in entrambi i casi, sarà conservato all'interno della piattaforma come un file contenente solo testo, che, nel caso dei file PDF, viene estratto automaticamente tramite la libreria Apache Tika.

2.2.2 Motori di ricerca

Per poter estrarre informazioni e conoscenza dai documenti presenti nelle collezioni, su ogni documento è possibile eseguire tutti i passi della pipeline di NLP descritti nel paragrafo 1.1. Questi generano all'interno della piattaforma i file contenenti i risultati del processo di elaborazione del linguaggio naturale; per ogni documento viene creato un file contenente i token estratti da esso, il lemma corrispondente, il POS Tag associato ad essi e la parte che occupano nel "chunk" di cui fanno parte. Inoltre, viene creato un file contenente le key phrases di ogni collezione.

Sulla base dei risultati delle operazioni di elaborazione del testo, la piattaforma è in grado di compiere operazioni di indicizzazione dei documenti basate sul modello a spazio vettoriale, presentato nel paragrafo 1.2. Inoltre, è possibile creare una rappresentazione semantica dei testi attraverso la tecnica di creazione di un modello semantico distribuzionale chiamata Random Indexing, presentata nel paragrafo 1.3.1. In seguito alle operazioni di preprocessing, indicizzazione e creazione di un Semantic Space associato ai documenti è possibile creare e associare a ciascuna collezione un motore di ricerca classico o semantico, a seconda delle operazioni di elaborazione dei documenti svolte.

2.2.3 Gestione dei metadati

Nell'ambito del progetto TALIA, oltre a beneficiare della conoscenza estratta dai documenti, è risultato utile sfruttare le informazioni strutturate disponibili relative ad ogni community e progetto.

Inizialmente, le informazioni strutturate, reperite durante il processo di crawling che ha permesso di ottenere anche i deliverable dei progetti del programma Med, erano state gestite impiegando le funzionalità offerte dalla libreria Lucene. In particolare, la libreria per la creazione e gestione di motori di ricerca dà all'utente la possibilità di suddividere ciascun documento in più sezioni (o campi), al fine di permettere di compiere una ricerca mirata sul contenuto di alcuni campi scelti (ad esempio, ricerca sui titoli o sugli abstract dei documenti). Le informazioni più importanti relative al dominio e a ciascun documento erano state quindi inizialmente memorizzate all'interno del Semantic Framework come campi associati ai documenti.

Il dominio del progetto TALIA, che coinvolge diversi progetti verticali del programma Med, è, tuttavia, troppo complesso per una memorizzazione delle informazioni di questo genere, che presentava diversi problemi:

- in primo luogo, la quantità di informazioni memorizzabili relativamente a ciascuno documento era limitata, soprattutto rispetto all'abbondanza di informazioni disponibili sul dominio dei progetti Med. Aggiungere troppi campi ad un documento faceva sì che esso occupasse più spazio di memorizzazione del necessario, oltre a far sì che non fosse rispettato il fine originale dei campi di un documento, poiché ben pochi campi erano adatti a poter compiere una ricerca su di essi.
- In secondo luogo, molte informazioni erano ridondanti, poiché erano ripetute in diversi documenti, nonostante esistessero relazioni tra le entità del dominio tenendo conto delle quali si sarebbe potuta semplificare la rappresentazione delle informazioni (ad esempio, ad

un Progetto corrispondono diversi Deliverable, dunque le informazioni relative al progetto possono non essere ripetute in ogni deliverable relativo ad esso, il che invece accadeva memorizzando i metadati come campi del documento).

- In terzo luogo, i vincoli di integrità esistenti tra le informazioni del dominio erano difficilmente preservabili con un modello che le mettesse in relazione soltanto con il documento cui si riferivano, e non le non mettesse, invece, in relazione tra loro rispecchiando le connessioni realmente esistenti tra le entità. Sarebbe stato, dunque, molto difficile rilevare errori o incongruenze nei dati.
- Infine, non potendo codificare le relazioni esistenti tra le entità del dominio, non era possibile neanche compiere ricerche più sofisticate sui dati presenti, prescindendo dall'associazione di essi con un particolare documento della collezione.

Per far fronte a questi problemi, è stato integrato nel Semantic Framework un modulo di gestione dei metadati relativi ad ogni collezione. Esso collega il Semantic Framework ad una base di dati relazionale progettata per contenere e organizzare i dati relativi ai documenti presenti nelle collezioni, grazie ad una struttura che rispecchia le relazioni esistenti tra le entità coinvolte nel dominio.

Il modulo è, dunque, in grado di rendere disponibili informazioni più dettagliate riguardo ogni documento presente nelle collezioni gestite nel Framework, e soprattutto relativamente al contesto in cui esso è inserito nell'ambito del programma Med. Ad esempio, oltre a quelle relative ad ogni singolo documento, sono disponibili informazioni dettagliate relative al progetto verticale cui ogni deliverable afferisce, ai partner e agli stakeholder

coinvolti in esso. La disponibilità di queste informazioni aggiuntive sul dominio in forma strutturata è stata utile ad arricchire ed ampliare i risultati ottenuti utilizzando i servizi del sistema di supporto alle decisioni, affiancando la potenzialità informativa offerta dalle funzionalità di analisi semantica del testo.

2.3 Panoramica sui servizi del sistema di supporto alle decisioni

Nell'ambito del progetto TALIA, le funzionalità del Semantic Framework sono state utilizzate per l'implementazione dei due principali servizi del sistema di supporto alle decisioni: un motore di ricerca semantico e una matrice di correlazione tra concetti.

I due servizi hanno l'obiettivo di contribuire al miglioramento del coordinamento e della cooperazione tra gli attori principali della Community Social and Creative facilitando la scoperta di informazioni utili e connessioni esistenti tra le azioni concrete svolte dai diversi attori, supportando le attività degli utenti come descritto nel paragrafo 2.1.

2.2.1 Motore di ricerca semantico

Il motore di ricerca semantico ha l'obiettivo di soddisfare i bisogni informativi degli utenti del sistema di supporto alle decisioni, offrendo la possibilità di compiere una ricerca sulle collezioni di documenti relativi a una Community o ad un Asse del programma Interreg-Med.

I risultati di una ricerca sono arricchiti grazie alle funzionalità esposte dal Semantic Framework e grazie alla disponibilità di metadati relativi ai documenti: infatti, l'utente riceve, in risposta ad una query, l'elenco dei documenti più significativi rispetto ad essa, un elenco dei concetti

maggiormente correlati a quello cercato e un insieme di metadati che forniscono informazioni più dettagliate relative al documento.

2.3.2 Matrice di correlazione tra concetti

La matrice di correlazione dà all'utente del sistema di supporto alle decisioni la possibilità di esplorare in modo interattivo il contenuto informativo dei documenti di una collezione, beneficiando dell'analisi semantica dei testi svolta grazie al Semantic Framework.

L'esplorazione di una collezione consiste nel poter visualizzare, data una coppia di concetti, un elenco contenente i concetti maggiormente correlati ad essi all'interno della collezione considerata. Grazie alla presentazione della funzionalità di correlazione sottoforma di matrice è possibile compiere questa analisi su diverse coppie di concetti contemporaneamente. Assegnando un concetto ad ogni riga e ad ogni colonna, infatti, ciascuna cella della matrice corrisponderà ai concetti associati alla riga e alla colonna che identificano la cella; essa sarà quindi riempita con i concetti della collezione maggiormente correlati ad essi. I concetti corrispondenti alle dimensioni della matrice possono essere, inoltre, sia concetti presenti nei documenti della collezione, sia definiti dal creatore della matrice a partire da quelli già esistenti.

La matrice di correlazione tra concetti, servizio più elaborato tra quelli messi a disposizione dal sistema di supporto alle decisioni, permette all'utente di compiere analisi più dettagliate e più flessibili rispetto al servizio di ricerca semantica. Ciascuna matrice, infatti, definita dalle sue dimensioni e dai concetti associati ad esse, può essere vista come una specifica analisi compiuta sul dominio rappresentato dalla collezione (e Community) sulla base della quale vengono calcolati i concetti correlati.

Ogni analisi può essere creata da un utente oppure progettata da un esperto; questa può essere quindi utilizzata da altri utenti, sia così com'è stata creata, che modificando le definizioni dei concetti in base alle proprie esigenze. I concetti maggiormente correlati a quelli definiti, mostrati nelle celle della matrice, possono essere poi interpretati in diversi modi, disponendo di un'adeguata conoscenza del dominio e dello strumento usato. Infatti, essi possono essere considerati sia come una "fotografia" dello stato attuale dei progetti operanti nella Community scelta, sia interpretati con il fine di prescrivere possibili azioni da compiere per cambiare i risultati ottenuti, e, di conseguenza, la situazione reale che si ritiene il servizio stia rappresentando.



	Work	Creativity	Innovation
Tourism	Airport Hotel Restaurant	Cinema Festival Music	Co-housing Car sharing Experience
Entrepreneurship	Company Employee Product	Publisher Film producer Open innovation	Industry 4.0 Co-Working Startup accelerator

[Fill matrix](#)

Figura 2.2 Una immagine di mockup della matrice di correlazione tra concetti

3. Servizi RESTful per la ricerca semantica

L'implementazione dei servizi del sistema di supporto alle decisioni ha richiesto l'utilizzo delle funzionalità offerte dal Semantic Framework esposte tramite servizi RESTful. Per ogni servizio è stato necessario utilizzare e implementare servizi specifici, sia per realizzare le funzionalità richieste, che per aumentarne l'efficacia.

3.1 Introduzione alle architetture RESTful

Lo stile architetturale RESTful è una tecnica di sviluppo di servizi Web concepita nel 2000 da Roy Fielding, allora studente di dottorato all'Università della California a Irvine.

Le architetture REST [13] (ossia REpresentational State Transfert), impiegano il protocollo HTTP per trasferire tutte le informazioni in una comunicazione tra host client e server. Nel protocollo HTTP ogni entità che si può richiedere ad un host server è considerata una "risorsa", identificabile e localizzabile univocamente tramite il suo indirizzo URL [12].

Un servizio web sviluppato secondo questo stile architetturale, dunque, è accessibile a un computer host tramite una semplice richiesta HTTP all'host server che espone il servizio. A seconda che il servizio richiamato abbia lo scopo di modificare o ottenere informazioni presenti sull'host server, la richiesta HTTP dev'essere formulata tramite i metodi HTTP progettati per il particolare scopo dell'operazione; ad esempio, il metodo GET è specifico per la richiesta di una risorsa senza modifica, mentre i metodi POST e PUT sono specifici per le modifiche delle risorse. [12]

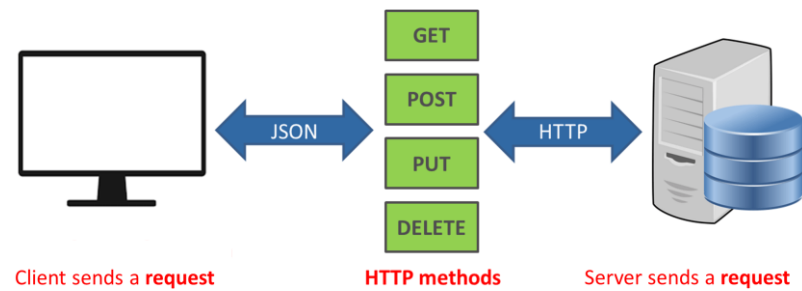


Figura 3.1 Rappresentazione del protocollo REST

3.2 Servizi RESTful impiegati

Uno dei casi d'uso prevedeva la costruzione di un'interfaccia che mettesse a disposizione una ricerca semantica usabile per gli utenti del sistema del supporto alle decisioni.

I servizi RESTful messi a disposizione dal semantic Framework vengono utilizzati sia per la costruzione di un motore di ricerca classico, che per quella di un motore di ricerca semantico. Esse effettuano varie operazioni sui documenti di una collezione, sulla quale gli utenti della piattaforma potranno compiere operazioni di ricerca fornendo in input un insieme di parole chiave.

3.2.1 Creazione motore di ricerca semantico

Il processo di costruzione di un motore di ricerca semantico viene effettuato richiamando le API REST esposte dal Semantic Framework, illustrate nei paragrafi seguenti.

3.2.1.1 NLP Pipeline

Prima di poter creare il motore di ricerca semantico è necessario effettuare le operazioni di NLP (Natural Language Processing) sui documenti della collezione presa in considerazione. Queste operazioni sono state descritte nel capitolo uno e sono:

- Sentences Detection

- Tokenization
- Rimozione delle stopWord
- POS Tagging
- Lemmatizzazione
- Phrase detection

Il semantic Framework fornisce un metodo REST per applicare la Pipeline NLP ad ogni documento. Come parametri il metodo richiede il nome della collezione e il nome di un documento presente in essa su cui effettuare le operazioni di NLP.

https://localhost:9001/v1/nlp/nome_della_collezione/nome_documento

3.2.1.2 Estrazione delle phrases

L'operazione di estrazione delle phrases [3] può essere realizzata attraverso due metodi REST presenti all'interno del Semantic Framework.

Il primo metodo è basato sull'idea che le parole che occorrono molto spesso insieme e molto raramente in contesti diversi possono essere considerate *key phrases*. Queste rappresentano dei gruppi di termini che esprimono un singolo concetto; ad esempio "Information Retrieval" è una espressione che denota un particolare ambito di ricerca informatico e viene considerata una *key phrase* poiché i due termini che la compongono assumono un significato diverso da quello che avrebbero se considerati singolarmente.

I vantaggi che presenta questo metodo sono due:

1. Essendo non supervisionato non ha bisogno di risorse esterne come dizionari o ontologie.
2. Il metodo è molto semplice infatti è basato sulla frequenza delle parole.

Tutti i bi-grammi sono valutati attraverso la seguente formula:

$$score(w_i, w_j) = \frac{count(w_i, w_j) - minCount}{count(w_i) \cdot count(w_j)}$$

dove w_i e w_j sono due termini che occorrono nella collezione oggetto di analisi, e $count()$ è la funzione che conta il numero di occorrenze delle parole all'interno della collezione, applicabile sia ad un singolo termine ($count(w_j)$), sia ai bi-grammi ($count(w_i, w_j)$). Inoltre, $minCount$ è un valore che previene la possibile formazione di phrases con parole non molto frequenti.

Sebbene questa tecnica sia in grado di individuare soltanto bi-grammi, ossia *key-phrases* composte da due soli termini, è possibile individuare n-grammi concatenando coppie di bi-grammi o bi-grammi.

Il metodo REST tramite il quale si può richiamare questo servizio richiede come parametri il nome della collezione sulla quale riconoscere le phrases, una soglia massima di bi-grammi da restituire e un numero che rappresenta la frequenza minima che un bi-gramma deve avere per poter essere restituito come *key-phrase*.

GET https://localhost:9001/v1/phrase/CollectionName1/w2p/30/4

Il secondo metodo di riconoscimento delle phrases impiega un automa a stati finiti in grado di riconoscere sequenze di parole categorizzate all'interno di Wikipedia; si può quindi intuire che questo approccio è utile quando abbiamo liste di concetti predefiniti. Le liste di concetti che vengono prese in considerazione sono formate da n-grammi formati da un massimo di sei parole e un minimo di due; esse verranno utilizzate per la costruzione dell'automa che sarà in grado di riconoscere phrases all'interno di testi. L'elemento fondamentale di questo approccio sono le liste di concetti

etichettate da Wikipedia che vengono considerate significative ai fini del riconoscimento delle phrases.

Come parametri il metodo REST richiede unicamente il nome della collezione sulla quale effettuare l'estrazione.

GET https://localhost:9001/v1/phrase/CollectionName1/FSA

3.2.1.3 Analyze

L'analisi statistica delle parole all'interno di grandi testi è in sempre più utilizzata grazie all'aumentare del potere computazionale disponibile. I DSM sono modelli semplici per costruire spazi geometrici di concetti. Essi sono conosciuti anche come Word Space e sono in grado di esaminare grandi testi ed estrarre il contesto d'uso delle parole presenti in essi. Nello spazio risultante la somiglianza semantica tra concetti viene espressa come vicinanza dei punti, visti come parole, nello spazio. In questo modo la somiglianza semantica viene calcolata come il coseno dell'angolo tra i due vettori che rappresentano le parole.

I DSM possono essere costruiti utilizzando diverse tecniche. Un approccio comune è il Latent Semantic Analysis [3], il quale si basa sulla decomposizione a valore singolare della matrice di co-occorrenza di parole. Tuttavia, esistono molti altri metodi che prendono in considerazione l'ordine delle parole [14] oppure metodi che effettuano predizioni [15].

Queste operazioni di analisi semantica vengono realizzate attraverso una API REST fornita dal Semantic Framework che richiede come parametri il nome della collezione sul quale effettuare le operazioni e la lingua.

GET https://localhost:9001/v1/analyze/nomeCollezione/lingua

3.2.1.4 Creazione SE e Random Indexing

Dopo aver effettuato tutte le operazioni sul testo e sulle collezioni si può effettivamente creare il motore di ricerca di tipo semantico.

Lo scopo del *search engine* è quello di ritrovare documenti che abbiano uno score di rilevanza rispetto alla query molto alto. Lo score di rilevanza può essere calcolato utilizzando varie metodologie:

- 1) Classical search: questo approccio realizza il classico Vector Space Model discusso nel capitolo uno e restituisce documenti che contengono almeno una delle parole presenti nella query dell'utente.
- 2) Semantic Search: questo approccio si basa sul WordSpace model che viene realizzato attraverso il processo di Random Indexing il quale permette di ottenere i vettori semantici per tutti i termini di una collezione. All'interno del Semantic Framework l'approccio utilizzato per la costruzione del *semantic search engine* si basa su una semplice ed efficace somma tra i vettori semantici delle parole. Quindi i documenti vengono visti come somma dei vettori di ogni singola parola presente in essi e, durante il processo di ritrovamento, vengono confrontati con la query, anch'essa rappresentata come somma dei vettori associati ai termini che la compongono.

Per la creazione del Search Engine e l'esecuzione Random Indexing per ogni collezione, il Semantic Framework fornisce servizi REST che vengono richiamati solo dopo aver effettuato tutte le operazioni di NLP e estrazione delle phrases dai testi dei documenti delle collezioni.

L'API REST per la creazione del Search Engine richiede come parametro il nome della collezione su cui creare e sincronizzare il motore di ricerca.

PUT <https://localhost:9001/v1/createSE/CollectionName>

È possibile anche controllare l'esistenza di un motore di ricerca su una specifica collezione attraverso l'API REST chiamata '*checkSE*' che richiede come parametro il nome della collezione:

GET https://localhost:9001/v1/checkSE/CollectionName

Per la creazione del Random Indexing, invece, il servizio REST richiede come parametro il nome della collezione, la grandezza dei vettori mediante i quali saranno rappresentati termini e documenti (*vetTermSize*), e la grandezza del vocabolario (*vocSize*).

PUT https://localhost:9001/v1/createRI/nome_collezione/vetTermSize/vocSize

3.2.2 Gestione delle collezioni in modo automatizzato

Per una questione di comodità è stata realizzata un'applicazione web, scritta principalmente in PHP, che gestisce l'aggiunta di nuovi documenti alle collezioni o la creazione di quest'ultime eseguendo tutte le chiamate REST per la costruzione di un motore di ricerca, descritte in precedenza, in modo automatizzato.

La schermata principale permette di scegliere il percorso nel quale si trovano i documenti delle collezioni estratti dal crawler (Figura 3.2).

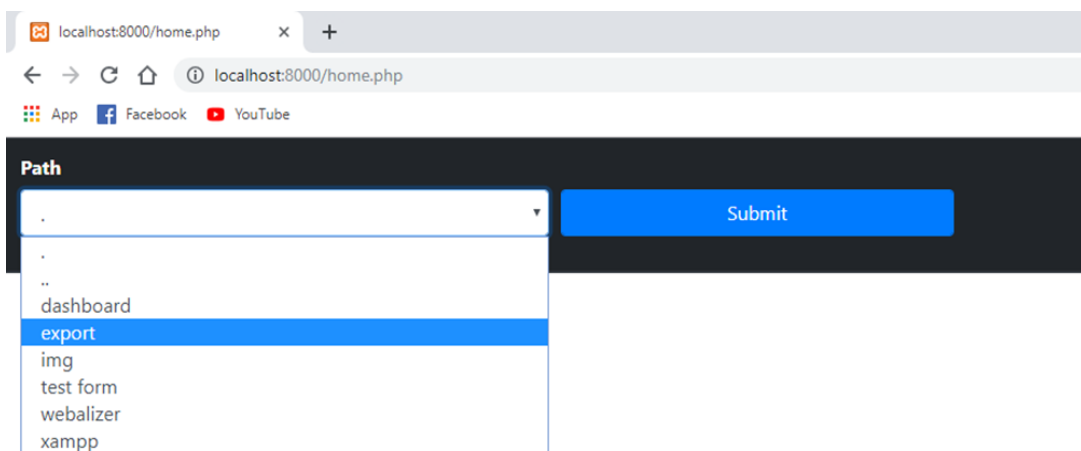


Figura 3.2 Interfaccia iniziale per la scelta della cartella contenente le collezioni

Selezionando la cartella e cliccando su “submit” il sistema andrà ad esplorare il percorso della stessa e visualizzerà i documenti al suo interno ottenendo una schermata simile a quella mostrata in Figura 3.3.

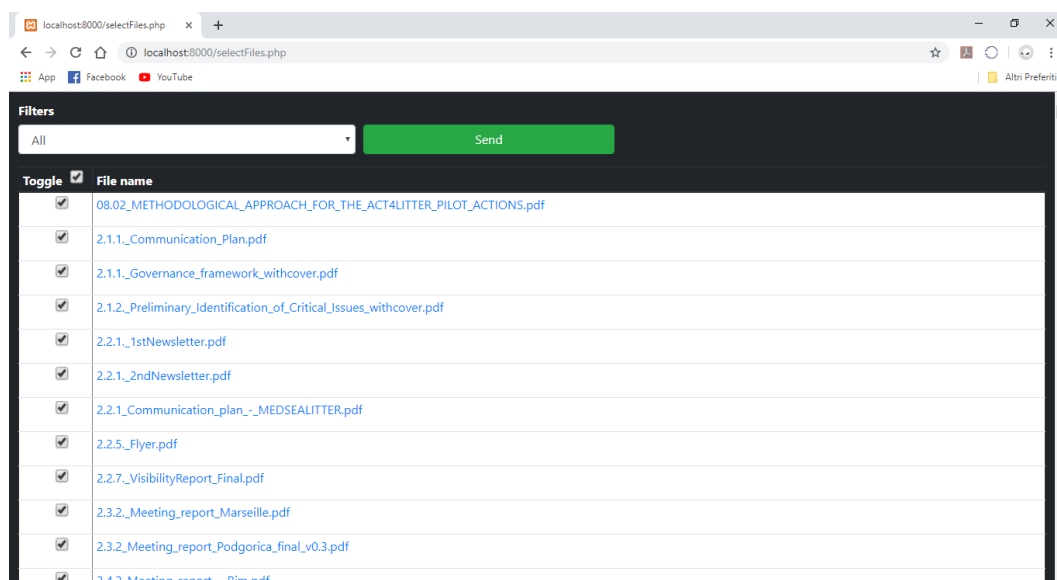


Figura 3.3 Lista di tutti i documenti ritrovati nella cartella selezionata precedentemente

I documenti vengono organizzati in una tabella dove ogni riga è un documento al quale è associata una checkbox di selezione. Essendoci numerosi documenti, per poterli ritrovare più facilmente è stata integrata una combobox dove ogni elemento corrisponde ad una collezione presente nella cartella estratta dal crawler e permette, dunque, di visualizzare solo i documenti presenti in essa.

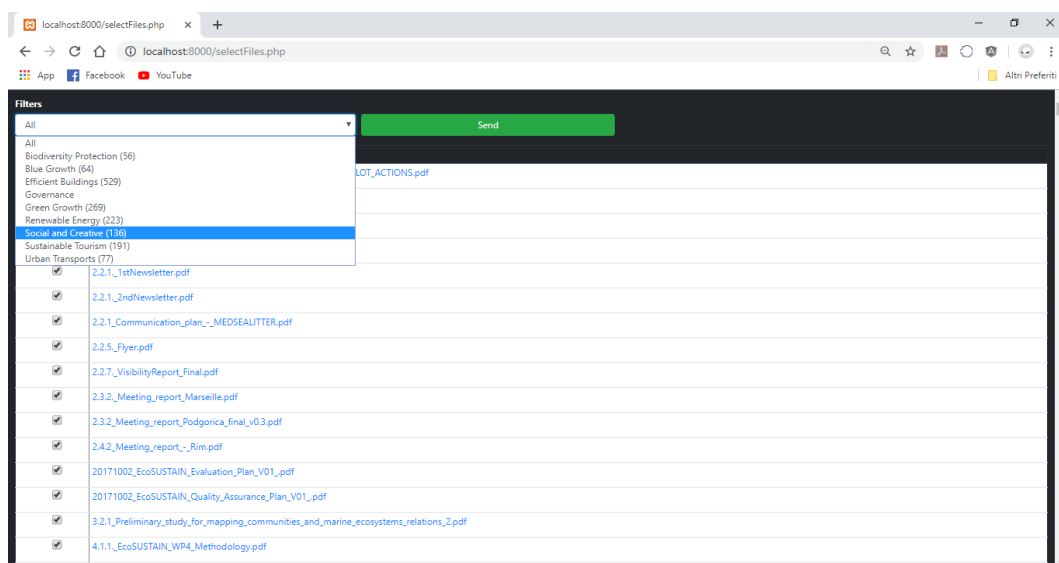


Figura 3.4 Combobox per filtrare i documenti per collezione

Una volta selezionati tutti i documenti da inserire in una nuova (o già esistente) collezione, cliccando il bottone “send” verrà mostrato un popup simile a quello mostrato in Figura 3.5.

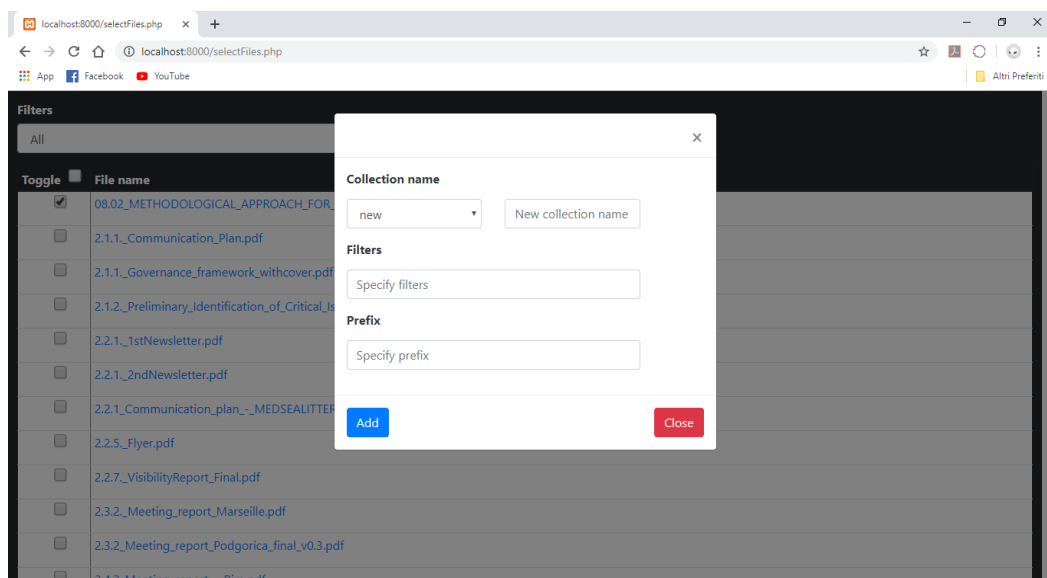


Figura 3.5 Popup per la selezione/creazione della collezione nel quale aggiungere i documenti selezionati

A questo punto si può inserire il nome della nuova collezione o scegliendo tra la lista di quelle esistenti e, successivamente, cliccando sul bottone “Add” tutti i documenti selezionati verranno processati, al fine di eseguire

tutte le operazioni di preprocessing del testo su di essi, ed infine inseriti nella collezione scelta su cui verrà creato il motore di ricerca attraverso i metodi REST del Semantic Framework; tutto questo verrà effettuato in modo automatizzato.

3.2.3 Summarization

Una volta creato il motore di ricerca esso verrà utilizzato per effettuare richieste da parte dei Policy Maker(paragrafo su query). I risultati delle query saranno i documenti che avranno uno score di rilevanza rispetto alla richiesta dell'utente molto alto.

Per aiutare l'utente a capire effettivamente se il documento restituito durante la ricerca contiene informazioni utili per soddisfare il suo bisogno informativo, vista la lunghezza dei documenti, si è scelto di avvalersi di un servizio che, dato il corpo di un documento, effettua un riassunto automatico del testo. Questo servizio è esposto tramite una API REST di tipo POST esterna al semantic Framework. Come parametri nel corpo della chiamata esso richiede un numero che rappresenta la lunghezza del riassunto da svolgere; questa si può esprimere come percentuale della dimensione del documento fornito oppure come numero di frasi che si vuole siano presenti nel documento, in base alle preferenze dell'utente. Inoltre, è possibile specificare uno o più concetti a cui dare particolare rilevanza nella creazione del riassunto [16].

POST <http://90.147.102.57:9003/getSummary>

3.2.4 Related Concepts

La possibilità di individuare, dato un concetto, quelli più correlati ad esso è una funzionalità molto importante del Semantic Framework. Infatti, questa è in grado di aiutare l'utente a soddisfare il proprio bisogno informativo estendendo la ricerca con parole molto simili a quelle della query iniziale.

Questo servizio REST è implementato all'interno il Semantic Framework e ha come compito quello di restituire una lista di parole simili a quelle della query. Per aumentarne l'efficacia sono stati aggiunti alcuni filtri al metodo; infatti, sono state eliminate dalla lista dei risultati tutte le stop-word, i concetti che fanno parte della query e, inoltre, la lista viene composta per metà da n-grammi e per metà da parole singole.

Il metodo REST richiede come input il nome della collezione su cui effettuare la ricerca, la grandezza della lista di output contenente i concetti simili e la query dell'utente.

GET https://localhost:9001/v1/simw/nome della collezione /dimensione_lista /query

3.2.5 Arricchimento con metadati

Come ulteriore arricchimento informativo su una ricerca effettuata da un utente si è scelto di aggiungere ad ogni risultato informazioni strutturate relative al documento e al progetto di cui fa parte.

Le informazioni strutturate relative al dominio del progetto TALIA sono state organizzate all'interno di un database (Figura 3.6) che è stato integrato al Semantic Framework tramite l'aggiunta di un modulo finalizzato alla gestione dei metadati. Per la costruzione del database è stato impiegato il DBMS (Database Management System) MySQL. I dati relativi ad ogni progetto comprendono varie informazioni: il budget dello stesso, i partner che vi hanno investito e i suoi documenti. Le informazioni relative a questi ultimi includono le parole chiave del documento, la data di creazione e una piccola descrizione.

Le tabelle e gli attributi ad esse relativi all'interno del database hanno la stessa rappresentazione delle classi presenti all'interno del Semantic

Framework. Inoltre, per il recupero dei dati dal database sono state aggiunte delle classi che effettuano accesso al database con le relative REST.

In particolare, durante lo sviluppo dell'interfaccia ci siamo resi conto che le numerose chiamate alle API REST finalizzate all'estrazione dei dati dal database, rallentavano notevolmente la visualizzazione dei risultati, quindi è risultato opportuno diminuire il numero delle chiamate REST.

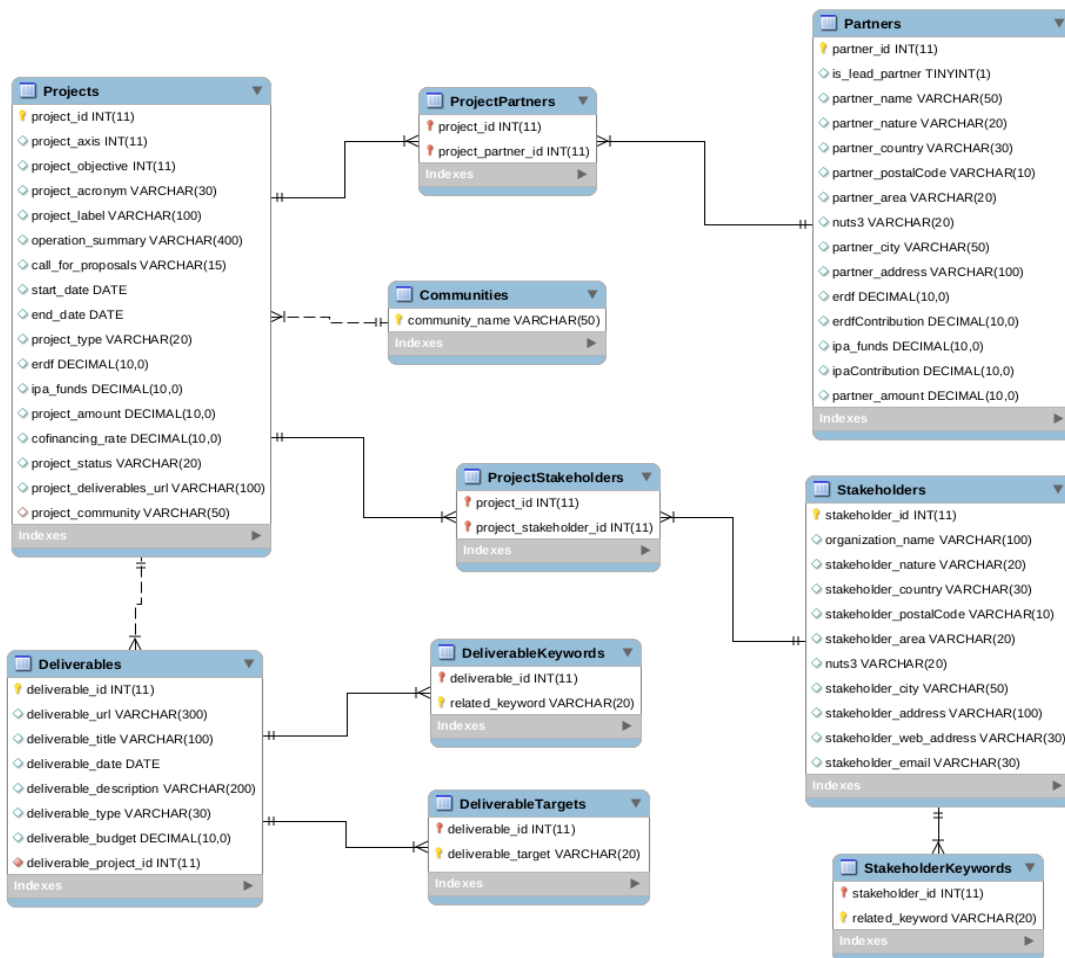


Figura 3.6 Modello relazionale database del Semantic Framework

4 Implementazione del servizio di ricerca semantica

Il caso d'uso trattato in questa tesi consiste nella costruzione di un'interfaccia web per l'utilizzo del motore di ricerca semantico fornito dal Semantic Framework. Per la costruzione dell'interfaccia sono stati utilizzati diversi tipi di linguaggi di markup e scripting. Inoltre, per il funzionamento del sistema gli elementi dell'interfaccia effettuano chiamate alle API REST definite all'interno del Semantic Framework. Lo scopo dell'interfaccia è quello di rendere usabili le funzionalità del motore di ricerca affinché l'utente finale possa utilizzarle al fine di soddisfare il proprio fabbisogno informativo nel modo più semplice possibile.

4.1 L'importanza di una buona interfaccia

La scelta della costruzione di una applicazione web per l'utilizzo dei servizi esposti dal Semantic Framework è stata compiuta per far fronte all'esigenza di utilizzo della piattaforma da parte di utenti che non hanno una formazione tecnica.

Come già detto nell'introduzione al capitolo l'interfaccia utente di un software gioca un ruolo fondamentale nel garantire la semplicità d'utilizzo dello stesso. Una buona interfaccia si costruisce attraverso l'implementazione di un sistema che risponda alle definizioni di usabilità e affordance. Un sistema sviluppato secondo questi criteri, infatti, garantisce all'utente un'interazione con il software che gli permetta di raggiungere i propri obiettivi in modo semplice.

Il concetto di usabilità è definito nel modo seguente:

l'usabilità di un prodotto è il grado con cui esso può essere usato da specifici utenti per raggiungere specifici obbiettivi con efficacia, efficienza e soddisfazione in uno specifico contesto d'uso [17].

Il concetto di affordance è definito nel modo seguente:

Con il termine affordance si denota la proprietà di un oggetto di influenzare, attraverso la sua apparenza fisica, il modo in cui viene usato [18].

4.2 Integrazione dei servizi del Semantic Framework

Come già anticipato il compito dell'applicazione web è quello di rendere disponibili le funzionalità della parte back-end all'utente. Per fare ciò gli elementi dell'interfaccia effettuano le chiamate alle API REST del Semantic Framework. Per effettuare una ricerca nella Smart search va inizialmente inserita la query da cercare e la collezione su cui ritrovare i documenti più rilevanti. L'elenco delle collezioni mostrate all'utente mediante una combobox viene richiesto al Semantic Framework, al momento del caricamento della pagina HTML, tramite un'API REST.

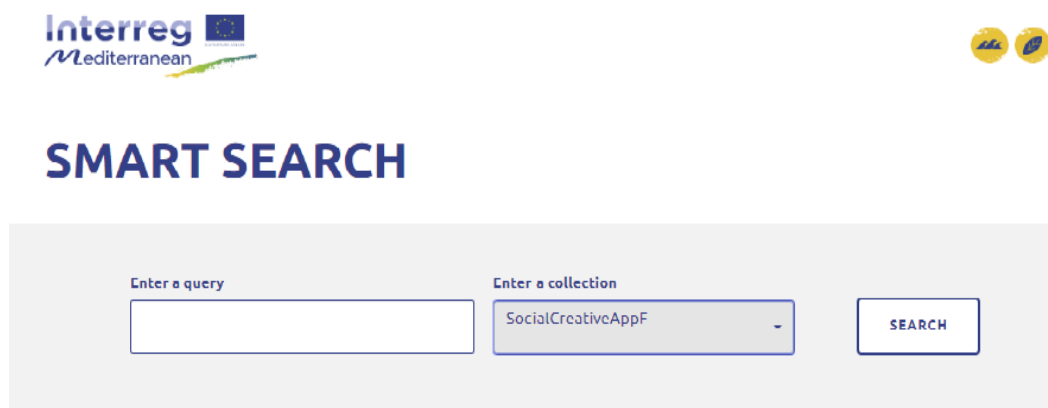


Figura 4.1 Interfaccia di ricerca

Cliccando il bottone search viene effettuata la chiamata al metodo REST, che, data la query, compie la ricerca semantica all'interno della collezione:

<http://localhost:9100/ivp/v2/semsearch/collezione/numrisultati/query>

I parametri di questa chiamata REST sono esattamente i valori delle due caselle di testo dell'interfaccia e un numero che rappresenta il massimo

numero di documenti rilevanti da restituire (passato come parametro statico via codice).

I risultati ottenuti dalla ricerca vengono visualizzati come mostrato in Figura 4.2.

The screenshot displays the Interreg Mediterranean search interface. At the top, there is a search bar with the text 'Enter a query' and a dropdown menu for 'Enter a collection' set to 'socialAndCreative'. Below these are 'SEARCH' and 'SHOW MAP' buttons. A section titled 'related concepts:' lists various tags: mostar_faculty, agriculture_livestock, food_technology, food_processing, food_security, mostar, fisheries, livestock, cataloniainistry, and government. Below this, it states '54 results'. The main content area shows three project entries, each with a title, description, project name, budget, and three buttons: 'Summarization', 'Deliverable details', and 'Partners details'.

Interreg Mediterranean

Enter a query: Enter a collection:

related concepts:

mostar_faculty agriculture_livestock food_technology food_processing food_security mostar fisheries livestock cataloniainistry government

54 results

D.3.2.1_REGIONAL_ANALYSIS_PP5 PROJECT NAME: CHIMERA
 Regional analysis on CCIs sector/subsectors and main aspects of innovation supporting system in CCIs sector/subsectors in Catalonia PROJECT BUDGET: €2,412,326

Summarization Deliverable details Partners details

D.3.2.1_REGIONAL_ANALYSIS_PP10 PROJECT NAME: CHIMERA
 Regional analysis on CCIs sector/subsectors and main aspects of innovation supporting system in CCIs sector/subsectors in Albania PROJECT BUDGET: €2,412,326

Summarization Deliverable details Partners details

D.3.2.1_REGIONAL_ANALYSIS_BASILICATA PROJECT NAME: CHIMERA
 Regional Analysis by Basilicata Region PROJECT BUDGET: €2,412,326

Summarization Deliverable details Partners details

Figura 4.2 Risultati di una ricerca

Come già accennato nel capitolo tre, una funzionalità utile esposta dal Semantic Framework sono i related concept. Essi sono singole parole o n-grammi con un significato molto simile a quello dei termini usati nella query dell'utente, secondo una misura di similarità calcolata sui concetti della collezione sulla quale si effettua la ricerca. Essi hanno il compito di

espandere i risultati ottenuti dalla ricerca semantica sui documenti per soddisfare il fabbisogno informativo dell'utente in modo più completo.

I related concepts nell'interfaccia realizzata vengono mostrati assieme ai risultati di una ricerca, prima dei documenti ritrovati. Cliccando su uno di essi, verrà nuovamente richiamato il servizio di ricerca semantica passando come parametri la stessa collezione della ricerca effettuata in precedenza e la query contenente il related concept cliccato.



Figura 4.3 Concetti simili alla query dell'utente

I documenti risultanti dalla ricerca vengono organizzati in una tabella, nella quale ogni riga corrisponde ad un documento considerato rilevante rispetto alla query e le righe sono ordinate in base allo score di rilevanza associato a ciascun documento. Il singolo documento viene visualizzato come mostrato in Figura 4.4.

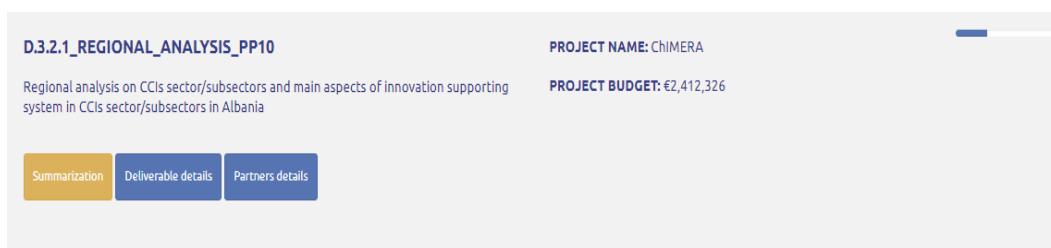


Figura 4.4 Formattazione singolo risultato

Di tutte le informazioni disponibili su ciascun documento, si è scelto di mostrare inizialmente solo una parte. Alcuni metadati, infatti, vengono nascosti per una questione di ordine, leggibilità dei risultati e utilità delle informazioni, e vengono mostrati solo su richiesta dell'utente attraverso i bottoni DELIVERABLE DETAILS e PARTNERS DETAILS.

Gli elementi mostrati per ogni deliverable al caricamento dei risultati sono: il nome del documento (D.3.2.1_REGIONAL_ANALYSIS_PP10); una breve descrizione del suo contenuto; il nome del progetto di cui fa parte il documento (ChiMERA) e il budget di progetto. Tutti questi metadati vengono caricati dal database attraverso le apposite API REST al momento della visualizzazione dei risultati.

Cliccando il bottone DELIVERABLE DETAILS verranno mostrate informazione aggiuntive relative al documento ritrovato, come, ad esempio, le parole chiave di un documento o la data di creazione dello stesso.

D.3.2.1_REGIONAL_ANALYSIS_PP5

Regional analysis on CCI sector/subsectors and main aspects of innovation supporting system in CCI sector/subsectors in Catalonia

PROJECT NAME: Chimera

PROJECT BUDGET: €2,412,326

Summarization Deliverable details Partners details

Date: 2017-03-29

Keywords: arts, awareness-raising, clustering, cultural heritage, economic cooperation, innovation capacity

Target Audience: Business support organisation, Higher education and research, Regional public authority, SME

Deliverable budget: €8,000

Figura 4.5 Singolo risultato ritrovato in cui mostriamo i dettagli del documento

Il bottone PARTNERS DETAILS, invece, ha come compito quello di visualizzare la lista dei partners coinvolti nel progetto assieme alle corrispettive caratteristiche per ognuno di essi, come il nome e il budget investito.



Figura 4.6 Singolo risultato ritrovato in cui mostriamo i dettagli dei Partners

Il recupero di tutti i metadati dal database inizialmente rallentava il sistema di ricerca a causa delle numerose chiamate ai metodi REST del Semantic Framework di connessione al database. Per risolvere questo problema abbiamo limitato il numero di chiamate alle API REST creandone solo 2 che estraggono tutti i metadati, la prima per i metadati del deliverable e la seconda per i metadati dei partners.

Tutti i metadati relativi a ogni documento presenti all'interno del database verranno poi utilizzati per la costruzione di un servizio di visualizzazione geografica dei risultati di ricerca ottenuti che sarà trattato in seguito.

Un altro elemento significativo nell'interfaccia mostrato per ogni documento ritrovato è la barra di rilevanza. Essa rappresenta lo score di rilevanza numerico che viene associato a ciascun documento nel processo di ritrovamento, di valore compreso tra zero e uno (una barra completamente blu rappresenta il massimo tasso di rilevanza del documento rispetto alla query). Il valore di score di ogni documento viene restituito assieme al titolo del documento stesso dalla funzione che effettua la ricerca semantica (*semsearch*).



Figura 4.7 Barra di rilevanza di un documento rispetto alla query

Infine, come già accennato nel capitolo tre, per ogni documento si può effettuare un riassunto cliccando il bottone “Summarization” il quale apre un popup uguale a quello mostrato in Figura 4.8.

4.8 Popup per il per effettuare il summary di un documento inserendo un numero di frasi che rappresenta la lunghezza del riassunto

In questo popup potremo scegliere se effettuare il riassunto specificando la sua lunghezza attraverso il numero di frasi che deve contenere (number of sentences) oppure attraverso un valore in percentuale (rate of sencences) che rappresenta lunghezza del riassunto rispetto al documento.

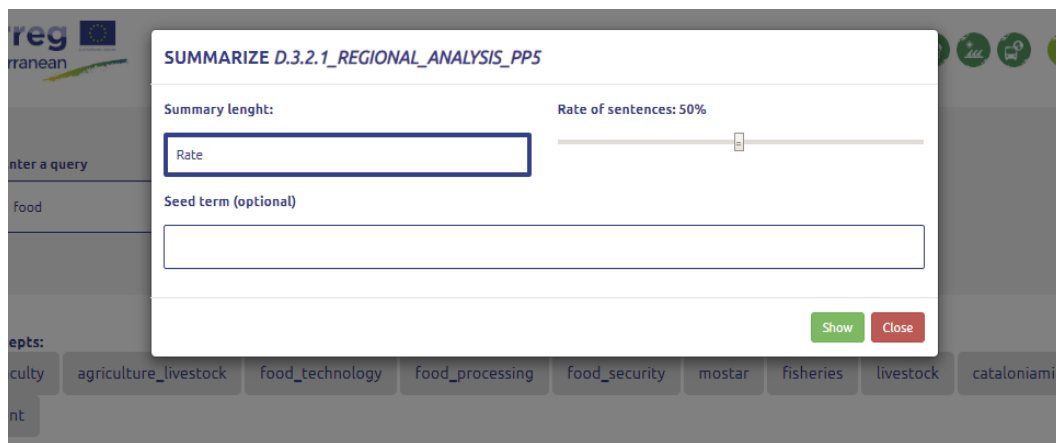
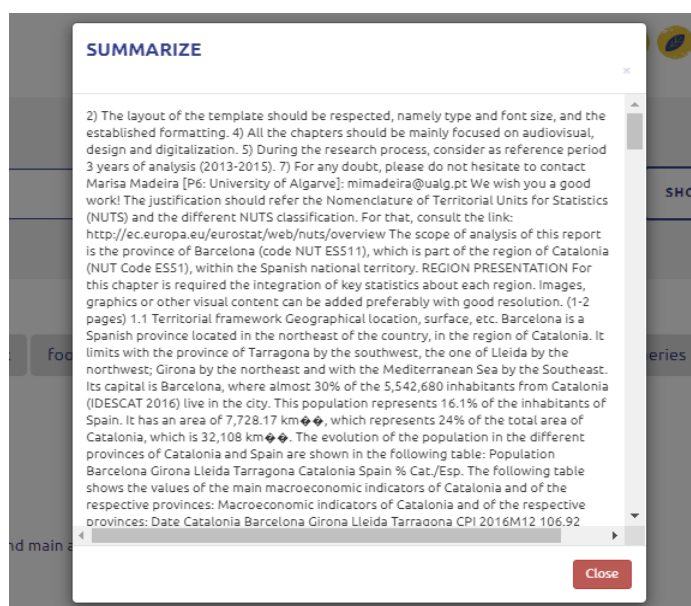


Figura 4.9 Popup per il per effettuare il summary di un documento inserendo un numero in percentuale

Inoltre, come campo opzionale, possiamo inserire alcuni termini dentro la casella di testo “Seed term” che rappresentano i concetti ai quali dare maggiormente rilevanza nel riassunto da effettuare.

Una volta inseriti tutti i valori, cliccando il bottone Show sarà richiamato il metodo Summarization, descritto nel capitolo tre, e verrà mostrato il riassunto da esso prodotto. Il metodo è implementato dal servizio REST:

<http://90.147.102.57:9003/getSummary>



4.10 Risultato servizio Summarization

4.3 Visualizzazione geografica dei risultati di ricerca

Una funzionalità che estende il caso d'uso della ricerca semantica è la visualizzazione geografica dei risultati ottenuti da una ricerca al fine di identificare i luoghi in cui operano i partner attivi su un particolare tema ricercato, rappresentando relazioni tra essi in modo grafico.

Nel sistema di smart search una volta effettuata una ricerca, quando vengono visualizzati i risultati, appare accanto al bottone search un altro pulsante denominato “Show map”.

The screenshot displays the smart search interface. At the top, there are two input fields: 'Enter a query' with the text 'food' and 'Enter a collection' with a dropdown menu showing 'socialAndCreative'. To the right of these fields are two buttons: 'SEARCH' and 'SHOW MAP'. The 'SHOW MAP' button is circled in red. Below the input fields, there is a section titled 'related concepts:' with a horizontal list of tags: 'mostar_faculty', 'agriculture_livestock', 'food_technology', 'food_processing', 'food_security', 'mostar', 'fisheries', 'livestock', 'cataloniainistry', and 'government'. Below the tags, it says '54 results'. Further down, there is a section titled 'D.3.2.1 REGIONAL_ANALYSIS_PPS' with a subtitle 'Regional analysis on CCIs sector/subsectors and main aspects of innovation supporting system in CCIs sector/subsectors in Catalonia'. To the right of this section, there is a progress bar and two buttons: 'Summarization' and 'Deliverable details'. Below the progress bar, there is a section titled 'PROJECT NAME: CHIMERA' and 'PROJECT BUDGET: €2,412,326'. At the bottom, there is a section titled 'Partners details'.

Figura 4.11 Bottone per usufruire del servizio per le mappe

Cliccando sul pulsante “Show map” appare un popup che dà la possibilità di scegliere tra tre diversi modelli di rappresentazione geografica dei risultati di ricerca: il primo tipo di mappa rappresenta i partner che hanno scritto i documenti risultanti dalla ricerca svolta; il secondo tipo di mappa rappresenta il budget che ciascun partner ha speso nell'ambito del tema ricercato; il terzo tipo di mappa rappresenta geograficamente gli stakeholder coinvolti nei progetti cui afferiscono i documenti risultanti dalla ricerca.

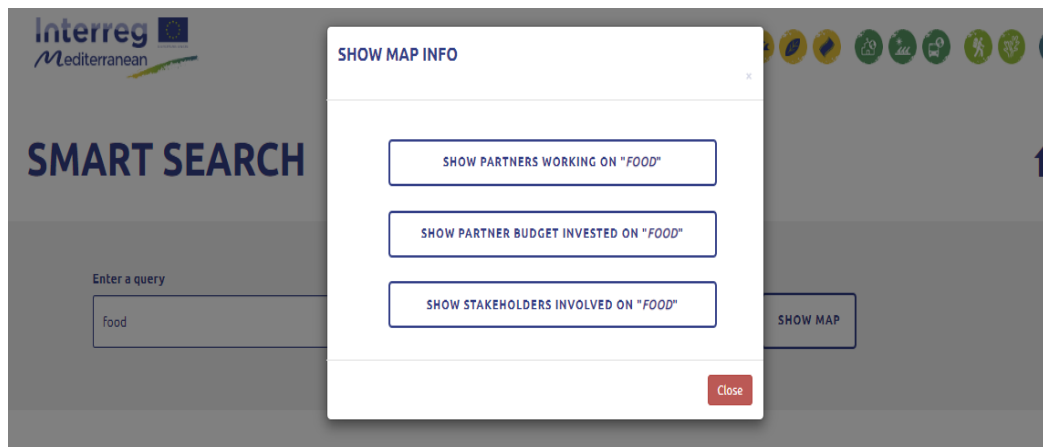


Figura 4.12 Popup per la scelta dei metadati da utilizzare per le mappe

Ogni bottone presente all'interno del popup richiama una diversa API REST presente all'interno del modulo di gestione dei metadati del Semantic Framework, che raggruppa e restituisce le informazioni necessarie per la successiva costruzione delle mappe. Ogni metodo REST restituisce un file di tipo JSON il quale viene poi trasmesso al servizio esterno di visualizzazione delle mappe.

5. Conclusioni e sviluppi futuri

Il progetto TALIA ha affrontato l'obiettivo di migliorare il coordinamento e la comunicazione tra soggetti operanti nell'area del Mediterraneo attraverso lo sviluppo del sistema di supporto alle decisioni illustrato nel presente lavoro di tesi. In particolare, il sistema di supporto alle decisioni, basato sul Semantic Framework, offre all'utente la possibilità di acquisire conoscenza nel dominio dei progetti del programma Med attraverso l'utilizzo del motore di ricerca semantico fin qui illustrato, di un servizio di analisi delle correlazioni esistenti tra i concetti di una collezione, e delle informazioni strutturate sul dominio integrate nel Semantic Framework.

Lo sviluppo della funzionalità di ricerca semantica come servizio web ha assunto un ruolo importante nel rendere tale servizio accessibile agli utenti in modo più semplice. Le funzionalità di ritrovamento di concetti simili a quelli presenti nella query dell'utente e la possibilità di esplorare le collezioni attraverso le ricerche svolte a partire dai concetti ritrovati, può supportare notevolmente l'utente nel soddisfare il proprio bisogno informativo esplorando le connessioni semantiche presenti nelle collezioni. La rappresentazione semantica dei testi è stata, inoltre, utilmente sfruttata nell'implementazione del servizio di "Summarization", che permette all'utente di visionare una parte ridotta del contenuto di ogni documento.

Il caso d'uso della matrice di correlazione tra concetti dà all'utente la possibilità di esplorare in maniera ancora più ampia il contenuto informativo presente nelle collezioni considerate, mettendo in evidenza, a prescindere dai singoli documenti presenti in esse, quali siano le connessioni tra i temi trattati in una intera collezione.

Infine, l'integrazione di dati non strutturati relativi al dominio delle collezioni, organizzati in una base di dati relazionale, ha ulteriormente arricchito la disponibilità di informazioni a disposizione dell'utente del sistema, pur mantenendo flessibile rispetto alle sue esigenze la quantità e il tipo di informazioni di volta in volta mostrate, grazie alla progettazione di una interfaccia utente attenta alle esigenze di usabilità del software. L'integrazione di un modulo di gestione dei dati strutturati ha permesso, inoltre, l'implementazione del caso d'uso di visualizzazione geografica dei risultati di ricerca, consentendo un accesso ai metadati più flessibile rispetto alla tecnica inizialmente adottata, che consisteva nel memorizzarli assieme al corpo del documento come parti di esso, secondo la strutturazione dei documenti disponibile nella libreria Lucene.

Gli sviluppi futuri del progetto di tesi sono molteplici, sia dal punto di vista del miglioramento tecnico delle funzionalità del Semantic Framework, che da quello di possibili ulteriori scenari nei quali il sistema potrebbe essere impiegato.

Occorre, innanzitutto, evidenziare che tanto la piattaforma "Semantic Framework", quanto i servizi del sistema di supporto alle decisioni, offrono funzionalità declinabili alle esigenze di una qualsiasi organizzazione e a qualsiasi dominio applicativo. Le funzionalità di gestione di collezioni di documenti e di analisi semantica del testo offerte dal Semantic Framework sono adatte a all'implementazione di molti servizi che necessitino di un'analisi di testi approfondita. Analogamente, come è stato illustrato inizialmente, i servizi di ricerca semantica e correlazione tra concetti costituiscono funzionalità importanti in un sistema di supporto alle decisioni basato su dati testuali sviluppato per un qualsiasi dominio.

Nonostante il progetto TALIA avesse come obiettivo quello di favorire lo sviluppo della comunità Social and Creative, è evidente – anche dalle considerazioni appena fatte – che il sistema di supporto alle decisioni possa essere facilmente esteso alle collezioni di documenti attinenti alle altre Community, in modo tale da supportarne la crescita.

È opportuno, tuttavia, che in futuro il sistema sia integrato in modo più efficiente con le fonti documentali e informative relative al dominio dei progetti del programma Med. Oltre a semplificare il processo di ritrovamento e aggiunta dei documenti alle collezioni, e di inserimento dei metadati nella base di dati, il sistema potrebbe anche beneficiare della presenza di una maggiore quantità di documenti, che renderebbero le operazioni di analisi dei documenti ancor più significative.

Dal punto di vista dei servizi esposti dal sistema di supporto alle decisioni, è possibile implementare alcuni miglioramenti tecnici. Nel motore di ricerca semantico, ad esempio, può essere integrata una funzione di relevance feedback che permetta all'utente di notificare al sistema quando un documento non è rilevante rispetto alla query che ha svolto; ciò porta al raffinamento della funzione di scoring del motore di ricerca e, dunque, ad ottenere dei risultati di ricerca maggiormente rilevanti.

I risultati attualmente ottenuti tramite il servizio di correlazione tra concetti potrebbero essere confrontati con quelli ottenuti implementando metodi diversi da quello utilizzato per combinare le liste di concetti simili ritrovati dai metodi implementati all'interno del Semantic Framework. L'efficacia di questo caso d'uso può essere, inoltre, misurata effettuando sperimentazioni che coinvolgano esperti del dominio chiamati ad interpretare i risultati

ottenuti dalla correlazione dei concetti definiti, così da poter identificare, ed eventualmente correggere, la presenza di risultati indesiderati.

Infine, il modulo di gestione dei metadati potrebbe essere reso indipendente dal dominio in modo tale da poter gestire le informazioni strutturate attinenti alle collezioni gestite indipendentemente dalla struttura della base di dati nella quale sono memorizzate.

Riferimenti bibliografici

- [1] M.F. Porter, (1980) "An algorithm for suffix stripping", Program, Vol. 14 Issue: 3, pp.130-137
- [2] Thomas K Landauer and Susan T Dumais, "A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge.," Psychological review, vol. 104, no. 2, pp. 211, 1997
- [3] Information Retrieval On-Line. F. W. Lancaster and E. G. Fayen. Los Angeles: Wiley-Becker & Hayes (1974)
- [4] Baeza-Yates, R., Ribeiro-Neto, B. (2011). Modern Information Retrieval. Harlow, England: Pearson Addison Wesley. ISBN: 978-0-321-41691-9
- [5] Luhn, Hans Peter (1957). "A Statistical Approach to Mechanized Encoding and Searching of Literary Information" (PDF). IBM Journal of Research and Development. 1 (4): 309–317.
- [6] Spärck Jones, K. (1972). "A Statistical Interpretation of Term Specificity and Its Application in Retrieval". Journal of Documentation. 28: 11–21
- [7] J. Ramos et al. (2003), "Using tf-idf to determine word relevance in document queries", in Proceedings of the first instructional conference on machine learning
- [8] G. Salton, A. Wong, C. S. Yang, A vector space model for automatic indexing, in Communications of the ACM, Volume 18 Issue 11, Nov. 1975, pp. 613-620
- [9] Harris, Z. (1954). Distributional structure. Word. 10 (23), pp. 146–162

- [10] Johnson, W. and Lindenstrauss, J. (1984) Extensions of Lipschitz mappings into a Hilbert space, in Contemporary Mathematics. American Mathematical Society, vol. 26, pp. 189–206.
- [11] Magnus Sahlgren, “An Introduction to Random Indexing,” in Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE, 2005, vol. 5
- [12] Mark Massè. REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces. O'REILLY, 2012.
- [13] Roy Thomas Fielding and Richard N. Taylor. Architectural styles and the design of network-based software architectures. Doctoral dissertation: University of California, Irvine, 2000. Vol. 7.
- [14] Michael N Jones and Douglas JK Mewhort, “Representing word meaning and order information in a composite holographic lexicon,” Psychological review, vol. 114, no. 1, pp. 1, 2007.
- [15] Trevor Cohen, Roger Schvaneveldt, and Dominic Widdows, “Reflective random indexing and indirect inference: A scalable method for discovery of implicit connections,” Journal of biomedical informatics, vol. 43, no. 2, pp. 240–256, 2010.
- [16] G. Rossiello, P. Basile, e G. Semeraro. Centroid-based text summarization through compositionality of word embeddings. In Proc. of the Workshop on Summarization and Summary Evaluation Across Source Types and Genres, pages 12–21, 2017.
- [17] R. Polillo. Facile da usare. Apogeo, 2010, pp. 66-67
- [18] *Ivi*, pp. 63