C. STATO DELL'ARTE

A partire dal problema che cerchiamo di risolvere, ÃÍ importante iniziare a calarsi nel concreto, rendendosi conto a quali campi ispirarsi per costruire unâĂŹarchitettura funzionale alla soluzione della questione.

Web semantico: con questo termine, coniato dal suo ideatore Tim Berners-Lee, âĂIJsi intende la trasformazione del World Wide Web in un ambiente dove i documenti pubblicati (pagine HTML, file, immagini, e cosÃň via) sono associati ad informazioni e dati che ne specificano il contesto semantico in un formato adatto all'interrogazione e l'interpretazione (es. tramite motori di ricerca) e, piÃź in generale, all'elaborazione automatica.âĂİ [Wikipedia] Questo non significa limitare le numerosissime forme in cui ÃÍ possibile presentare i contenuti sul web, bensÃň ÃÍ un modo per fornire una struttura capace di rispondere a delle query, permettendo la lettura automatica di pagine e documenti. Risorse con questa caratteristica sono enormemente utili per la facilitÃă con la quale ÃÍ possibile estrarre informazioni e adattarle alle proprie necessitÃă; sono quindi queste la scelta piÃź ovvia come semplice punto di partenza per la ricerca di contenuti atti a stimolare la reminiscenza. Una delle realtÃă piÃź grosse che opera nel web semantico ÃÍ DBpedia, il cui approccio ÃÍ quello di parsare periodicamente i dump rilasciati da Wikipedia, estrarre le uniche informazioni strutturate delle pagine, le cosiddette infobox, unendo alcuni attributi delle versioni localizzate dello stesso articolo e rappresentando le risorse tramite Resource Description Framework (RDF). Ogni risorsa ÃÍ definita tramite un URI, un identificatore unico dellâĂŹentitÃă presentata. Il lavoro di DBpedia ÃÍ eccellente, ma ÃÍ involontariamente ostacolato dagli editori di Wikipedia, che per la compilazione delle infobox non hanno dei template ben definiti, finendo per rappresentare lo stesso attributo in articoli diversi con nomi diversi; DBpedia ÃÍ interrogabile tramite SPARQL, un linguaggio SQL-like costruito per leggere RDF. Dopo il rilascio del primo dataset al pubblico nel 2007, la situazione attuale ÃÍ quella di un grosso grafo di altri set di dati collegati tra di loro tramite lâĂŹaccoppiamento di risorse appartenenti a insiemi diversi, ma rappresentanti la stessa entitÃă. Pubblicare questi Linked Data rende molto piÃź facile la ricerca di informazioni sul web, rendendole molto piÃź precise e complete. Alcuni altri progetti di web semantico sono: Freebase, che a differenza di DBpedia ÃÍ un progetto proprietario e orientato al profitto; Uniprot, una knowledge base contenente dati liberamente accessibili sulle sequenze di proteine; GeoNames, un database geografico contentente oltre 10 milioni di luoghi, accessibile e scaricabile sotto licenza Creative Commons. A settembre 2011, lo stato dei collegamenti tra datasets semantici ÃÍ rappresentato dal grafo visualizzabile a (http://lod-cloud.net/versions/2011-09-19/lod-cloud.html). Un approccio diverso nellâĂŹattribuzione di un significato ai contenuti presenti sul web ÃÍ quello dei microformati, unâĂŹestensione di markup che, tramite lâĂŹutilizzo degli attributi HTML class, rel e rev, consente lâĂŹattribuzione di regole semantiche a normali pagine web: date le informazioni sul contatto di Nicola Parrello

```
<div>
    <div>Nicola Parrello</div>
    <div>University of Trento</div>
    <div>123-123123</div>
    <a href="mailto:hi@everybody.com">Email</a>
</div>
```

queste, se riscritte con il microformato hCard (specifico per i contatti), diventano

```
<div class="vcard">
    <div class="fn">Nicola Parrello</div>
    <div class="org">University of Trento</div>
    <div class="tel">123-123123</div>
    <a class="email" href="mailto:hi@everybody.com">Em
</div>
```

Attraverso questi microformati, ad esempio, un software come un browser puÃš estrarre facilmente informazioni e navigare le relazioni tra oggetti diversi, mantentenendo comunque la normale leggibilitÃă di una pagina web. Oltre ad hCard, solo hCalendar ÃÍ stato formalizzato: altri microformati, come hAtom, hMedia e hNews (rispettivamente per feed Atom, contenuti multimediali e notizie) sono solo piÃź o meno abbozzati, e non rappresentano quindi uno standard.

Web search e contestualizzazione dellâĂŹinformazione: Il passo successivo nella ricerca di una soluzione ad problema che ÃÍ tema di questo documento, dopo aver scelto da dove reperire i dati, ÃÍ quello di decidere come permettere a un eventuale utente di accedere ai contenuti che vogliamo proporre. LâĂŹapproccio che sembra piÃź naturale ÃÍ quello di costruire un piccolo motore di ricerca, in modo tale da creare una maschera che permetta una richiesta di risorse in maniera uniforme, anche se da fonti diverse; inoltre, per avere una ricerca rapida e reattiva, il buon senso suggerisce che ÃÍ una buona idea quella di indicizzare il materiale ricercabile. In questo campo ÃÍ difficile non considerare Google come lo stato dellâĂŹarte, essendo loro il search engine piÃź utilizzato al mondo. PerchÃl' utenti da ogni parte del globo possano utilizzare un servizio veloce ed efficiente, Google utilizza sette componenti dinamiche, che salvano e leggono dati in altre strutture; il procedimento puÃš essere riassunto cosÃň: Il Server URL parte da un URL e, leggendo il Document Index, invia gli URL ai Crawler. I Crawler scaricano le pagine web e le mandano nello Store Server. Lo Store Server comprime le pagine nel formato zlib (RFC1950), riducendo la loro dimensione a un terzo dellâĂŹoriginale, che vengono poi di un docId univoco e immagazzinate nel repository. Il repository viene letto dallâĂŹIndexer, che decomprime i documenti e li parsa, assegnando a ogni parola (a cui viene assegnato un wordId univoco) informazioni su posizione, grandezza del testo, numero di occorrenze e altro; ogni voce viene poi aggiunta ad un indice parzialmente ordinato. Dalle pagine lette, lâĂŹIndexer estrae anche i dati dei link in esse contenuti, oltre a servirsi del testo analizzato per costruire un lessico, utile per la vera e propria funzione di ricerca. Lo URL Resolver combina i documenti con i dati dei link, per costruire tabelle di coppie di docId, utilizzati per calcolare il PageRank. Il Sorter riordina lâĂŹindice (ordinato per docId) in un indice inverso (ordinato per wordId), aggiungendo altri dati per rendere la ricerca piÃź precisa. Infine viene calcolato il PageRank, una misura dellâĂŹimportanza di una pagina web calcolata in base a quali e quante sono le pagine che hanno dei link che puntano a essa.

Negli anni Google ÃÍ diventato colmo di pubblicitÃă, mostrata e scelta in base ai dati raccolti dalle ricerche degli utenti, ma certamente non ÃÍ lâĂŹunica possibilitÃă in quanto a motori di ricerca: un esempio su tutti e quello di Duck Duck Go che, a differenza di Google, ÃÍ un Semantic Search Engine. Il sistema quindi si occupa di valutare lâĂŹeffettivo significato dei termini di ricerca, eliminando in maniera piÃź precisa i risultati irrilevanti. Ma quello che lo contraddistingue in maniera maggiore dagli altri ÃÍ la sua attenzione alla privacy: Duck Duck Go infatti non conserva e non vende a terzi nessuna informazione sulle ricerche, permettendo una ricerca anonima e al sicuro da data leak, richieste legali da parte delle istituzioni e disonesti. (https://duckduckgo.com/privacy) Quello che puÃš infastidire gli utenti, cioÃÍ gli annunci di cui sopra, sono perÃš molto interessanti per la nostra ricerca, in quanto introducono un altro aspetto molto importante: perchÃÍ la ricerca sia semplice da utilizzare, e soprattutto piacevole, puÃš essere necessario renderla per cosÃň dire automatica, nascondendo lâĂŹazione manuale dellâĂŹinserimento di parametri e mostrando i risultati direttamente come contesto di qualche altra azione. Premiata nel 2012 dal magazine Popular Science come innovazione dellâĂŹanno (http://www.popsci.com/bown/2012/product/google-now), Google Now ÃÍ unâĂŹestensione dellâĂŹapplicazione mobile Google Search che, oltre ad essere un assistente vocale, analizza abitudini, ricerche e posizioni ricorrenti per fornire dati e risultati contestuali, mostrando allâĂŹutente le informazioni prima che egli ne faccia richiesta esplicita allâĂŹapp.

Algoritmi spazio-temporali: Abbiamo definito la raccolta e la ricerca, ma un ultimo punto rimane fumoso: in base a quale criterio scegliere un set di risorse rispetto a un altro, con la condizione che queste siano effettivamente rilevanti ed efficaci nella stimolazione della reminiscenza? Il vissuto di un individuo puÃš essere riassunto in una lista di eventi, di storie di vita; allo stesso tempo, un evento puÃš essere identificato da una coppia <t,s>, dove t ÃÍ la coordinata temporale e s descrive la posizione nello spazio. Utilizzando questo modello, il criterio di cui sopra diventa quello della distanza spazio-temporale tra due entitÃă, e il calcolo di questa distanza diventa il mezzo per effettuare unâĂŹindicizzazione preliminare delle risorse.

Un interessante esempio di sfruttamento estensivo delle informazione su tempo e spazio al fine di restituire dei risultati a una query ÃÍ TimeTrails. TimeTrails ÃÍ un sistema per lâĂŹestrazione e lâĂŹesplorazione di coordinate spazio-temporali che si possono trovare nei documenti di testo, composto di tre componenti principali: (1) una pipeline che, dopo aver letto i dati forniti da dei moduli che estraggono documenti di testo da varie sorgenti (e.g. la vetrina di Wikipedia (http://it.wikipedia.org/wiki/Wikipedia:Vetrina)), si occupa di estrarre le date e i luoghi contenuti nel testo, normalizzarli (e.g. da âĂIJ25 luglio 1991âĂİ a âĂIJ25/07/1991âĂİ per le prime e da âĂIJPergine Valsuganaâ€İ a âĂIJ46.06853620, 11.23528960âĂİ per i luoghi), e calcolare il numero e la posizione delle occorrenze trovate, in modo da verificare se la coordinata dello spazio e quella del tempo identificano un evento ben preciso oppure se sono due riferimenti senza alcuna correlazione. Il risultato dellâĂŹelaborazione, cioÃÍ il documento originale unito a una sequenza ordinata di tuple <t,s>, viene poi salvato in (2) un database ottimizzato per contenere dati su luoghi (http://postgis.net/), dal quale (3) unâĂŹinterfaccia permette la ricerca testuale di documenti, con un risultato che verrÃă mostrato su una mappa sotto forma di traiettoria, rappresentata da tutte le tuple lette dal db. Nel caso la query dovesse restituire piÃź di un documento, puÃš risultare interessante vedere dove e quando le traiettorie si incrociano: se i documenti sono biografie, lâĂŹintersezione delle traiettorie in uno dei punti segnati dalle tuple <t,s> puÃš indicare che i personaggi si sono incontrati.