1.1 Interpretare il testo

1.2 Un processo da seguire passo passo

L'aspetto più interessante, e per certi versi affascinante, dell'elaborazione automatica del linguaggio naturale è chiamato "estrazione del contenuto" di un testo.

Ma, ci chiediamo:

"Come può un computer comprendere un testo?"

e poi

"Come può un computer comprendere il significato di un testo?"

Un computer deve effettuare una serie di operazioni che per noi sono quasi automatiche, tanto che siamo in grado di capire velocemente la maggior parte dei testi senza neanche sapere come facciamo!

Limitandoci, per ora, alla prima domanda, vediamo i passi che segue un computer per estrarre il contenuto di un testo.

1.3 Alla ricerca della struttura sintattica

Il primo passo per capire un testo è la sua suddivisione in token, che possono corrispondere a singole parole, segni di interpunzione, o espressioni più articolate come date, ore del giorno, prezzi, ecc.

Il secondo passo è il POS-tagging, per assegnare ai token un tag, un'etichetta che ne descrive la funzione: nome, verbo, articolo, aggettivo e così via.

Quindi, terzo passo, si può analizzare la struttura sintattica del testo, con un'operazione nota col termine inglese di "parsing".

Il parsing permette di capire come sono costruite le frasi, che mettono insieme parole "piene" dette anche "parole lessicali" (come nomi, verbi e aggettivi), e parole "vuote", dette anche "grammaticali", che fanno da collante.

Ci sono molti modi di effettuare il parsing. Generalmente, questi algoritmi operano con una modalità "bottom-up". Vuol dire che invece di cercare di analizzare il testo sulla base di aspettative (come spesso facciamo noi), partono dai dati elementari, cioè dai singoli token e puntano a raggrupparli in strutture sempre più ampie, applicando le regole di una "grammatica".

1.4 L'analisi per costituenti

Ci sono diversi approcci per analizzare la struttura sintattica di un testo.

Il primo è l'analisi per costituenti, che corrisponde, in prima approssimazione, all'analisi sintattica insegnata a scuola.

Ecco questo metodo applicato alla prima strofa…

Questo schema rappresenta la struttura della frase come un albero rovesciato (cioè con la radice in alto). Leggendolo dall'alto in basso abbiamo:

- "S" ["esse"], cioè "sentence" che è la frase nel suo insieme.
- Poi, a un primo livello, una "frase nominale" ("NP", cioè "noun phrase", che corrisponde al soggetto, e una frase verbale ("VP", cioè "verb phrase").

Ciascuno di questi due "nodi" costituenti può essere la radice di altre ramificazioni. Infatti la frase nominale contiene un articolo e un nome: "La donzelletta".

Mentre la frase verbale contiene un verbo, una preposizione e un nome.

Attenzione, però: noi abbiamo letto la struttura dall'alto in basso, ma un algoritmo potrebbe partire dai singoli token e aggregarli progressivamente.

1.5 L'analisi di dipendenza

Un metodo alternativo è l'analisi di dipendenza, che prende in considerazione, appunto, le relazioni di dipendenza semantica, come quelle tra:

- il nome e i suoi modificatori (articoli, aggettivi, ecc.);
- il verbo, il soggetto e i complementi.

La nostra strofa "La donzelletta vien dalla campagna" verrebbe rappresentata così…

Questo tipo di analisi corrisponde all'analisi logica che si insegna alla scuola media.

1.6 Il chunking

Oltre alle tecniche di parsing completo, di cui abbiamo visto due esempi, si usano spesso algoritmi detti di "chunking", termine che in italiano si può tradurre con "tagliare a pezzi".

Il chunking è un'analisi sintattica di superficie che richiede:

- prima il POS-tagging;
- poi l'identificazione dei costituenti elementari di una frase (nomi, verbi, aggettivi, preposizioni, ecc.);
- infine il collegamento di questi costituenti elementari in unità più complesse, dette chunk, tipicamente gruppi nominali e verbali.

In sostanza, il chunking costruisce uno o più alberi sintattici parziali, di estensione e profondità limitata, senza pretendere di ricostruire l'albero sintattico completo.

L'analisi sintattica effettuata col chunking è chiamata "di superficie", ma, come vedremo fra poco, dà risultati utilissimi.

Per l'estrazione del contenuto, i chunk di maggior interesse sono quelli di tipo nominale (chiamati "noun chunks"), che corrispondono alle frasi nominali che abbiamo incontrato parlando di analisi per costituenti.

La ragione è semplice: dai chunk di tipo nominale possiamo capire qual è l'argomento di un certo testo!

1.7 Come funziona il chunking

Vale la pena di capire meglio come funziona un algoritmo di chunking…

Dopo la tokenizzazione e il POS-tagging della frase, il momento centrale è l'assemblaggio dei costituenti elementari in chunk. È un'operazione, chiamata "assiemamento", che può avvenire in diversi modi.

Uno dei più usati è il riconoscimento di specifici pattern (pattern matching), usando, per esempio, un insieme di espressioni regolari da provare una dopo l'altra.

Qui le espressioni regolari, che vanno applicate non al testo originario ma a una sequenza di POS-tag, assomigliano molto alle regole usate nell'analisi per costituenti.

In questo modo si crea una "chunk grammar", cioè un insieme di regole per riconoscere i chunk nominali di un testo.

Ma, come sono fatte queste regole?

1.8 Struttura delle regole per il chunking

Una chunk grammar è costituita da una serie di regole espresse così…

… dove "NP:" equivale a dire "un chunk nominale è così definito".

Le regole tra parentesi graffe vanno provate nell'ordine, una dopo l'altra, fino a trovare nel testo una serie di parole che messe insieme soddisfano la regola.

Ecco qualche esempio…

Come vedi, le regole sono rappresentate da un insieme di POS-tag, che si possono recuperare da un lessico morfologico.

In effetti, regole del genere sembrano particolarmente astruse, ma… clicca su ciascuna di esse per conoscerne il significato e avere qualche esempio.

Con un insieme di regole come queste, possiamo estrarre dal testo frammenti di informazione in modo rapido e robusto.

Per esempio, per analizzare un sito web e capire che vi si offrono servizi di traduzione.

Non è un esempio a caso: è così che funzionano i motori di ricerca, che usiamo quotidianamente!

1.9 Come costruire una grammatica per il chunking

Per costruire una "grammatica dei chunk", dobbiamo:

- definire tutte le regole che descrivono la struttura dei chunk che ci interessano;
- scegliere l'ordine in cui applicare tali regole, dato che un ordine diverso può dare risultati diversi, col rischio di non riconoscere quei chunk che corrispondono ai frammenti di contenuto più significativi.

E poi raffinare l'insieme procedendo per tentativi.

È un compito estremamente laborioso in cui, però, possiamo ricorrere ad algoritmi "addestrati" con tecniche di apprendimento automatico. È lo stesso modo di procedere che si adotta per la tokenizzazione e il POS-tagging:

- si prende un corpus, sufficientemente ampio, di testi rappresentativi del dominio che ci interessa;
- si compie, una volta per tutte, lo sforzo di marcare tutti i chunk rilevanti dal punto di vista del contenuto;
- si usa questo corpus per addestrare i chunker (cioè gli algoritmi, in grado di effettuare il chunking).

In questo modo si ottengono algoritmi particolarmente abili nell'assiemare gli elementi, scegliendo ogni volta la regola ottimale in base al contesto.

1.10 L'estrazione terminologica

Ora che abbiamo un'idea di come effettuarla, possiamo capire meglio a cosa serve l'analisi automatica di un testo o di un'ampia raccolta di testi.

Infatti, i chunk di un documento, o almeno parte di essi, hanno una funzione importante nell'estrarre informazioni da un file di testo o da un sito web.

In proposito, una delle operazioni più interessanti è la cosiddetta "estrazione terminologica", una sottocategoria dell'estrazione di informazioni che consiste nell'identificare automaticamente i termini rilevanti.

A questo punto, però, è il momento di chiederci cosa sono, nel nostro ambito, i "termini" e quali fra questi sono "rilevanti"!

1.11 Termini e termini rilevanti

Cosa si intende per "termine" in linguistica?

Il "Dizionario della Lingua Italiana" Sabatini Coletti, tra le tante accezioni della voce "termine", include questa:

“Vocabolo peculiare di una determinata disciplina, di un linguaggio settoriale; più generalmente: parola, voce.

Esempi di uso: termine tecnico; un termine dialettale”.

Ma un “termine” non è sempre una singola parola. Consideriamo “termini” anche sequenze di parole che richiamano un concetto.

Pensiamo a “calcio d’angolo”. Solo se messe insieme queste parole hanno un significato compiuto nel linguaggio sportivo.

Allo stesso modo, nel linguaggio settoriale della gestione d'impresa sono termini "socio in affari" o “processi di business”.

Tra i tanti termini che troviamo in un testo o in un corpus di testi quali vanno considerati “rilevanti”?

Sono quelli che si ritrovano con frequenza maggiore rispetto ai testi che provengono da altri ambiti.

Analizzando una serie di articoli di cronaca calcistica, troveremo come termini rilevanti:

“calcio d’angolo”, “calcio di rigore”, “palla”, “porta” o “palo”.

Naturalmente, non è impossibile che gli stessi termini siano presenti anche negli atti parlamentari o nei verbali delle assemblee di condominio, ma non con la stessa frequenza.

A questo punto, si incomincia a intravedere che l’estrazione dei termini ha già un’applicazione pratica.

Anzi, più d’una!

1.12 Di cosa si sta parlando?

Estrarre i termini rilevanti da un testo ci consente di dare una risposta alla prima delle domande con cui abbiamo iniziato il nostro ragionamento:

“Come può un computer comprendere un testo?”

Con l’estrazione dei termini rilevanti, il computer può capire se un testo afferisce a uno specifico settore disciplinare, intorno a quali concetti ruota, se afferma qualcosa a proposito di certi soggetti, luoghi o fatti.

Così se “calcio di rigore” è molto più frequente di “rigore finanziario” e di “carbonato di calcio”, che pure hanno parole in comune, l’algoritmo capisce che si parla di sport e non di politica economica o di chimica.

Questo è il primo passo per l'estrazione del contenuto di un testo. Cioè per rispondere alla nostra seconda domanda:

“Come può un computer comprendere il significato di un testo?”

1.13 Una questione di SEO

Tutti noi che usiamo spesso i motori di ricerca ce ne siamo accorti: i risultati che otteniamo inserendo determinate parole chiave sono sempre più precisi e coerenti.

Infatti, i motori di ricerca web non si limitano più a cercare meccanicamente singole parole, ma effettuano una duplice estrazione dei termini:

- nel testo, in fase di indicizzazione delle pagine web;
- nella stringa di ricerca, per selezionare i risultati all'interno di un certo ambito.

Così, andando al di là della singola parola, l'algoritmo capisce se l'argomento di un sito è proprio quello che stiamo cercando e mette in cima alla lista i risultati che incontrano meglio le nostre necessità.

Vedendo le cose da un altro punto di vista, mettiamoci nei panni del webmaster che gestisce un sito e ha tutto l'interesse che le sue pagine compaiano per prime.

Adottando una serie di tecniche chiamate SEO (Search Engine Optimization, cioè "ottimizzazione per i motori di ricerca"), inserisce ad arte certe parole chiave per dire al motore di ricerca quali, secondo lui, sono i termini più significativi.

In realtà, come abbiamo visto, i motori di ricerca applicano algoritmi di estrazione dei termini e sono perfettamente in grado di identificare di cosa parla una pagina web.

Il webmaster, con le sue tecniche di ottimizzazione, cerca solo di dare un "aiutino"!

1.14 Costruire dizionari

Oltre all'individuazione dell'argomento di un testo e alla facilitazione del lavoro dei motori di ricerca web, l'estrazione dei termini può avere una terza funzione: costruire una terminologia di settore.

Infatti, con l'analisi dei chunk di un corpus di documenti, possiamo evidenziare i termini (singole parole o espressioni più articolate) prodotti dai membri di una certa comunità sociale, scientifica o professionale.

Questo elenco dei termini che appartengono a un ambito specifico può essere poi rielaborato in forma di glossario: una raccolta sistematica che comprende definizioni ed esempi di uso in cui i singoli termini vengono, se possibile, messi in relazione tra loro.

Per concludere il nostro ragionamento, vale la pena di notare che un glossario condiviso non ha solo una funzione astrattamente documentale.

È, piuttosto, uno strumento importante per la coesione di una comunità, per facilitare la

comunicazione anche con chi non ne fa parte, per condividere l'uso di sistemi e strumenti.

Ma… non è tutto!

1.15 Named Entity Recognition (NER)

Ora che ci siamo addentrati nei meccanismi con cui un computer riesce a "comprendere" un testo digitale, è il momento di un passo ulteriore: la "named entity recognition" (NER), che possiamo tradurre con "estrazione delle unità nominate".

È un'operazione che consente di individuare tra i chunk nominali quelli che si riferiscono alle "entità" uniche cui di solito assegniamo un nome di battesimo per distinguerle da altre dello stesso tipo. Le chiamiamo "entità nominate", dall'inglese "named entities".

Si tratta, per esempio, di persone, organizzazioni, luoghi, eventi, mesi e giorni della settimana, modelli di automobili, farmaci, valute, unità di misura e così via.

Cosi, le espressioni "giorno della settimana" e "lunedì" sono entrambe chunk nominali, ma solo "lunedì" è anche un'entità nominata!

L'estrazione delle entità nominate è molto importante nell'analisi e comprensione di un testo.

Vediamo perché…

1.16 La funzione pragmatica del linguaggio

Il linguaggio non serve solo per esprimere concetti e fare affermazioni generali su qualche argomento.

Lo usiamo molto più spesso per comunicare conoscenze su fatti specifici e per influire, più o meno apertamente, sul comportamento degli altri con ordini, domande o semplici affermazioni.

Dove "gli altri", oggi, non sono solo gli umani, ma anche altri computer.

In prima approssimazione, i linguisti chiamano "pragmatica" questa funzione del linguaggio che permette di "capirsi".

La pragmatica è un aspetto del linguaggio particolarmente complesso.

Non presuppone solo competenze linguistiche sulla struttura della frase (morfologia e sintassi) e sulla semantica, ma anche la condivisione di un ampio "contesto" con l'interlocutore.

È il contesto che permette di decifrare il significato di un messaggio, e include molti aspetti:

- l'ambiente fisico (che possiamo sfruttare indicando le cose col dito);
- le esperienze comuni;
- le frasi che ci siamo scambiati finora;

- e tanta conoscenza condivisa di tipo "enciclopedico" che è proprio quella relativa alle "entità nominate".

Così se siamo in grado di interpretare questo breve testo giornalistico…

… è perché sappiamo già che l'ONU è un'organizzazione internazionale, che Cina, India e Stati Uniti sono stati, che Europa vuol dire Unione Europea e non una generica indicazione geografica, che CO2 significa anidride carbonica, il cui aumento causa cambiamenti climatici e via dicendo!

1.17 Come riconoscere le entità nominate

Le entità nominate, come dicevamo, costituiscono per lo più un sottoinsieme dei chunk nominali, che altre volte denotano concetti generali, cioè categorie di cose materiali o immateriali.

In un testo ci sono tanti indizi per individuare le entità nominate. Per esempio, in molte lingue (tra cui italiano e inglese) i nomi propri di persone, organizzazioni, marchi e luoghi iniziano con la maiuscola.

Inoltre, esistono in forma digitale enciclopedie, dizionari enciclopedici, annuari, indici di nomi geografici che consentono di verificare se una stringa di caratteri corrisponde al nome proprio di qualche entità nota.

1.18 Algoritmi di NER - 1

È facile capire che il riconoscimento di entità nominate si basa moltissimo sulle conoscenze accumulate, cioè sull'esperienza.

Mentre è possibile effettuare una rozza tokenizzazione del testo con un programma relativamente semplice e poche regole, gli algoritmi di named entity recognition devono essere addestrati a partire da grandi moli di dati: enciclopedie, corpora opportunamente annotati, flussi di notizie più o meno fresche, ecc.

Nei sistemi di analisi del testo si fa in modo che algoritmi diversi operino in sinergia, aiutandosi a vicenda.

Per esempio, verificare che questa sequenza di token…

… coincide con un nome inserito in un'enciclopedia dei luoghi del mondo supporta l'ipotesi che corrisponda a un chunk nominale.

Viceversa, se sto cercando di riconoscere le entità nominate da un testo, l'estrazione dei chunk nominali può essere un'utile pre-elaborazione.

1.19 Algoritmi di NER - 2

È possibile trovare sul web diversi dimostratori di applicazioni di named entity recognition.

Questo, per esempio, è basato su spaCy, uno dei più moderni ambienti per lo sviluppo di applicazioni di elaborazione del testo.

Come vedi, il sito propone già un testo di prova in inglese e consente di analizzarlo diversi "modelli" di lingua, dove per "modello" si intende il risultato dell'addestramento dei diversi algoritmi a partire da corpora annotati e non annotati.

Per l'inglese, si può scegliere tra modelli "small", "medium" o "large".

Questo è il risultato dell'estrazione col modello "large", che riconosce persone, organizzazioni, luoghi e date, anche quando sono espresse con espressioni come "a decade later", cioè "un decennio dopo"!

Prova a cercare con questa applicazione le "entità nominate" in brevi testi inglesi e italiani, copiandoli e incollandoli nell'apposito riquadro.

Per l'italiano, purtroppo è disponibile solo il modello "small", cosa che influisce negativamente sulla qualità dei risultati.

Vediamo, infatti, cosa succede se inseriamo il nostro breve testo giornalistico…

Notiamo subito che il programma individua i luoghi e i nomi, ma non l'unica data presente!

In generale, la scarsezza di risorse linguistiche utili per l'analisi della lingua italiana è un problema su cui ci si imbatte continuamente.

E che può essere affrontato solo con un ampio progetto di collaborazione che coinvolga pubblico e privato, grandi e piccole organizzazioni.