ANNO ACCADEMICO 2024/2025

Tecnologie del Linguaggio Naturale

Teoria - Di Caro

Altair's Notes

UNIVERSITÀ DI TORINO

DIPARTIMENTO DI INFORMATICA

Premessa

Licenza

Formato utilizzato

Box di "Concetto sbagliato":

Concetto sbagliato 0.1: Testo del concetto sbagliato

Testo contente il concetto giusto.

Box di "Corollario":

Corollario 0.0.1 Nome del corollario

Testo del corollario. Per corollario si intende una definizione minore, legata a un'altra definizione.

Box di "Definizione":

Definizione 0.0.1: Nome delle definizione

Testo della definizione.

Box di "Domanda":

Domanda 0.1

Testo della domanda. Le domande sono spesso utilizzate per far riflettere sulle definizioni o sui concetti.

Box di "Esempio":

Esempio 0.0.1 (Nome dell'esempio)

Testo dell'esempio. Gli esempi sono tratti dalle slides del corso.

Box di "Note":

Note:-

Testo della nota. Le note sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive.

Box di "Osservazioni":

Osservazioni 0.0.1

Testo delle osservazioni. Le osservazioni sono spesso utilizzate per chiarire concetti o per dare informazioni aggiuntive. A differenza delle note le osservazioni sono più specifiche.

1 Semantica Computazionale

1.1 Introduzione

1.1.1 Semantica Computazionale

La semantica computazionale (fig: 1.1) può essere divisa grossolonamente in tre parti:

- *Semantica lessicale:* consiste nello studio di *come* e *che cosa* denotano le parole di una lingua. Si analizzano:
  - *Significato letterale*.
  - *Polisemia:* parole con più significati.
  - *Relazioni semantiche:* sinonimia, antonimia, iponimia, etc.
  - *Composizione del significato*.
- *Semantica formale:* studia i modelli logico-matematici che definiscono formalmente i linguaggi. L'obiettivo è definire il significato in termini di condizioni di verità.
- *Semantica statistico-distribuzionale:* approccio computazionale e quantitativo al significato che combina metodi statistici e intuizioni linguistiche (in particolare il fatto che il significato delle parole possa essere inferito dalla loro distribuzione sui testi). Si analizzano grandi corpora per costruire rappresentazioni vettoriali delle parole (*embeddings*), in cui la vicinanza tra vettori (solitamente si usa la *cosine similarity*) riflette la somigliarità semantica.

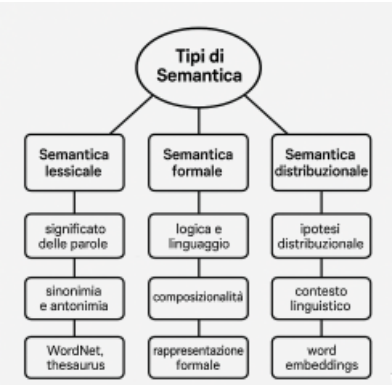

Figure 1.1: Tipi di semantica.

1.1.2 Origini del NLP

Inizialmente la linguistica computazionale e l'elaborazione del linguaggio naturale si occupavano del *question answering* (Q&A) ossia permettere a una macchina di leggere un testo (*what*) e rispondere a domande poste da un utente (*why*) attraverso l'impiego di codice e di risorse linguistiche (*how*). Con il passare del tempo le domande sono diventate sempre più complesse e variegate riguardando: fatti specifici, richieste di elenchi, definizioni, motivazioni, elenchi, etc.

Proprio per questo motivo è emersa una nuova area di ricerca appositamente per la caratterizzazione delle domande. L'obiettivo è quello di costruire tassonomie per modellare ogni possibile sfaccettatura che una domanda possa avere. Tutto ciò per aumentare l'efficacia dei sistemi di Q&A che necessitano, in primo luogo, di comprendere la tipologia di domanda.

Negli ultimi, grazie allo sviluppo dell'*intelligenza artificiale generativa* (e dei modelli GPT, LLaMA, etc.), il Q&A si è evoluto. Questi modelli possono estrarre domande da un testo e *generare* risposte articolate, sintetiche o creative. Possono:

- Rispondere a domande complesse.
- Gestire dialoghi multi turno.
- Tradurre le domande e le risposte.
- Spiegare le proprie risposte.

Note:-

Quindi è ancora più importante determinare il contenuto della domanda in modo da evitare *allucinazioni*. Storicamente il Q&A è sempre stato un task complesso, ma nel periodo post-ChatGPT sta vedendo il suo apice mediante il meccanismo chiamato *prompting*.

Gran parte della ricerca non si limita al Q&A, ma anche a ciò che ci sta dietro o in parallelo. Per esempio PoS, NeR tagging, iperonimi, word sense disambiguation, etc. Oppure casi come quello del *suggeritore automatico*, presente nelle tastiere degli smartphone, che usa un modello statistico.

1.1.3 Word Sense Induction

Definizione 1.1.1: Word Sense Induction

La Word Sense Induction (WSI) è il task che riguarda l'identificazione del senso di una parola polisemica in una frase, all'interno di un determinato contesto.

Questo task ha problemi di:

- *Specificità:* molti sensi attribuiti alle parole non vengono utilizzati perché troppo specifici e sono solamente *rumore*. Questo è criticato da vari studiosi che sostengono sia necessario aggregare alcuni sensi troppo simili (e.g. in WordNet).
- *Copertura:* ci sono molte zone di linguaggio non coperte.
- *Soggettività:* nonostante le decisioni siano prese collettivamente c'è sempre una componente soggettiva.

Differenze tra Word Sense Induction (WSI) e Word Sense Disambiguation (WSD):

- La disambiguazione ha necessità di un dizionario/sense inventory (e.g. WordNet) che contiene tutti i possibili sensi per ogni parola. Nel WSI non esiste un dizionario.
- Nel WSI ci si basa sull'effettivo uso della parola in grandi quantità di dati.
- La WSD, essendo fatta da linguisti, è basata sulla grammatica. La WSI è basata sull'uso delle parole, anche sgrammaticato.
- La valutazionw nel WSD è semplice (per esempio usando synsets gold), ma criticabile come visto in precedenza. Nel WSI è un po' più complicato.

Corollario 1.1.1 Pseudo-word

Il metodo della Pseudo-word è una tecnica per valutare algoritmi di WSI in assenza di risorse semantiche o annotazioni di senso. L'idea è quella di simulare l'ambiguità lessicale creando artificialmente delle parole ambigue e poi testare se il sistema è in grado di distinguerne i sensi sottostanti.

Fasi della Pseudo-word:

1. *Merging:* concatenazione di parole reali. Consiste nel fondere due o più parole esistenti in una parola ambigua. Queste parole devono avere significati distinti e usi in contesti diversi.
2. *Substitution:* sostituzione nei contesti. Tutte le occorrenze delle parole originali vengono sostituite nei testi dalla nuova parola create nel merging.
3. *Clustering:* identificazione dei sensi. Si applica un algoritmo di clustering (e.g. k-means, DB-SCAN, o modelli basati su embeddings) sulle rappresentazioni contestuali delle Pseudo-word per scoprire gruppi di usi distinti (sensi).
4. *Cluster-to-Class evaluation:* valutazione. Si valuta la qualità dei clusters ottenuti confrontandoli con le parole originali

Vantaggi e Limiti:

✓ Non sono richieste annotazioni manuali o risorse linguistiche.

✓ Può essere usato su grandi corpora in maniera automatica.

✗ I sensi creati non riflettono ambiguità reali.

✗ I contesti potrebbero essere troppo distinti e causare *overfitting*.

1.1.4 Comprensione del Linguaggio Naturale

- *Dizionari Elettronici:*
  - *Potere espressivo:* medio-alto, forniscono relazioni semantiche ricche.
  - *Scalabilità:* medio-bassa, solitamente sono costruiti manualmente (quindi difficilmente estendibili).
  - *Sorgente:* curata manualmente da esperti linguisti.
  - *Ambiguità e Soggettività:* ambiguità ridotta, soggettività media (su accezioni meno comuni).
- *Property Norms:*
  - *Potere espressivo:* alto per concetti concreti.
  - *Scalabilità:* bassa perché richiede raccolta tramite esperimenti psicologici o annotazioni.
  - *Sorgente:* spesso da studi cognitivi o crowd-sourcing/mechanical turk.
  - *Ambiguità e Soggettività:* alta, perché le persone non sono linguisti.
- *Frames:*
  - *Potere espressivo:* alto, cattura le strutture sintattico-semantiche.
  - *Scalabilità:* media, possono essere ampliati con annotazioni automatiche, ma richiede risorse linguistiche robuste.
  - *Sorgente:* tipicamente linguistica, contributi da annotatori esperti.
  - *Ambiguità e Soggettività:* media, perché c'è spesso intervento umano, ma si rimedia con strumenti automatici.
- *Senso Comune:*
  - *Potere espressivo:* molto alto, copre inferenze, aspettative sociali e causalità.

  - *Scalabilità:* media, dato che servono molte persone.
  - *Sorgente:* crowd-sourcing, scraping, machine learning.
  - *Ambiguità e Soggettività:* molto alta, molte conoscenze sono implicite, culturali o controverse.

- *Visual Attributes:*
  - *Potere espressivo:* medio, utile per oggetti visibili, ma limitato ad aspetti percettibili.
  - *Scalabilità:* medio-alta con dataset di immagini annotate.
  - *Sorgente:* dati visivi + annotazioni.
  - *Ambiguità e Soggettività:* medio-alta, le percezioni visive sono soggettive.

- *Word Embedding:*
  - *Potere espressivo:* molto alto in contesti distribuzionali.
  - *Scalabilità:* molto alta, addestrabili su grandi corpora.
  - *Sorgente:* dati testuali in grande scala.
  - *Ambiguità e Soggettività:* medio-alta, dipendono dal contesto, dalla lingua e possono riflettere eventuali bias.

- *Corpus Manager:*
  - *Potere espressivo:* dipende dal corpus, utile per esplorare usi reali del linguaggio.
  - *Scalabilità:* alta, può gestire ilioni/miliardi di parole.
  - *Sorgente:* testi reali.
  - *Ambiguità e Soggettività:* media, i dati sono "grezzi", quindi l'ambiguità linguistica è intrinseca.

Risorsa	Potere espressivo	Scalabilità	Sorgente	Ambiguità / Soggettività
Dizionari elettronici	Medio-alto	Medio-bassa	Manuale	Bassa
Property norms	Alto	Bassa	Esperimenti/Crowd	Alta
Frames	Alto	Media	Annotatori esperti	Media
Common-sense knowledge	Molto-alto	Alta	Crowd-sourcing/ML	Molto alta
Visual Attributes	Medio	Medio-alta	Immagini annotate	Medio-alta
Word/sense embeddings	Molto-alto	Molto alta	Testi su larga scala	Alta
Corpus manager	Variabile	Alta	Corpora reali	Media

Figure 1.2: Schema delle risorse.

1.2 Definizioni e Ricerca Onomasiologica

Nella linguistica la nozione di *definizione* è fondamentale. Una definizione è un testo progettato per guidare il lettore (o il *consumer*) verso un possibile significato associato a un termine all'interno di un contesto. Però bisogna tener presente che la relazione tra termini e significati non è univoca: un singolo termine può essere associato a una molteplicità di significati. Ogni significato può essere descritto da una definizione specifica o da più definizioni complementari.

Esistono diversi tipi di definizioni:

- *Genus-differentia:* identificano una categoria generale (*genus*) e specificano caratteristiche distintive.
- Definizioni basate su esempi.
- Definizioni tramite riferimenti ad altri concetti o termini già noti.
- Definizioni costruite tramite parafrasi, sinonimi o descrizioni operative.

Note:-

La qualità di una definizione dipende da chiarezza, accuratezza terminologica, adeguatezza rispetto al pubblico di riferimento, coerenza con il dominio e capacità di disambiguazione in contesti ambigui.

Definizione 1.2.1: Ricerca Onomasiologica

Partendo da un concetto o da un significato si deve identificare i termini o le espressioni linguistiche che possono denotarlo.

1.2.1 Definizione delle Definizioni

Domanda 1.1

Come si descrive un concetto?

Domanda 1.2

Quali caratteristiche sono più importanti?

Domanda 1.3

Che relazione c'è tra un termine da definire e il suo gruppo semantico più generale?

Domanda 1.4

Come si scrive una definizione? Come si valuta la qualità di una definizione? Quanto si è d'accordo? Quanto si utilizza lo stesso linguaggio e la stessa terminologia? Come varia ciò tra concetti astratti/concreti e generici/specifici?

1.2.2 Semasiologia e Onomasologia

Nella lessicografia si possono distinguere due approcci alla relazione tra forma e contenuto:

- *Approccio semasiologico:* si parte da un termine linguistico (una parola o una locuzione) e si vogliono determinare i possibili significati. Si tratta dell'analisi correlata ai tradizionali dizionari.
- *Approccio onomasiologico:* si parte da un concetto, un'idea o una definizione e si vogliono individuare i termini linguistici che possono esprimere ciò. Questo processo è anche noto come *lessicalizzazione di un concetto*.

Aspetti collegati alla ricerca onomasiologica:

- *Dizionari analogici:* permettono la ricerca a partire da un concetto.
- *Tip-of-the-tongue:* fenomeno per cui il parlante ha in mente un concetto ma non riesce a esprimerlo.
- *Meccanismo del Genus-differentia:* supporta la costruzione di descrizioni concettuali per una risalita onomasiologica.

2
Teorie del Significato

2.1 Panoramica

Inizialmente, i tasks di NLP, richiedevano un approccio mirato:

- Venivano costruiti sistemi a regole, grammatiche o modelli statistici.
- Si utilizzavano risorse linguistiche annotate manualmente (dizionari, corpora taggati, etc.).
- Si progettavano algoritmi ad hoc per ciascun compito (parsing sintattico, disabiguazione semantica, traduzione automatica, etc.).
- Si ricorreva all'apprendimento supervisionato o semi-supervisionato, con ingegnerizzazione manuale delle features.
- La valutazione richiedeva spesso il coinvolgimento umano: annotatori, crowdsourcing, etc.

Note:-

Questo modo pre-LLM ha permesso la crescita del NLP e contribuito alla costruzione di molte risorse linguistiche (WordNet, FrameNet, etc.).

Negli ultimi anni l'avvento dei Large Language Model (LLM), modelli basati su reti neurali profonde e addestrati su enormi quantità di testo, ha causato uno shift di paradigma. Oggi molti tasks non richiedono più modelli separati: un singolo modello, ben progettato e promptato, è in grado di riassumere, tradurre, fare sentiment analisys, Q&A, etc.

2.1.1 Definizioni di Base

Alcune definizioni di base:

- *Lessico:* corrisponde al dizionario, ovvero a tutti gli elementi che si hanno a disposizione per costruire una frase.
- *Sintassi:* studia come gli elementi del dizionario possono essere collegati tra loro attraverso una struttura che permette di costruire frasi.
- *Semantica:* interpretazione di una struttura lessico-sintattica a cui si attribuisce un significato.
- *Pragmatica:* disciplina linguistica che si occupa del rapporto tra parole e contesto.
- *Ambiguità:* proprietà del linguaggio che permette di esprimere e comunicare con un numero basso di parole. Tuttavia aumenta la difficoltà nella comprensione di parole con più interpretazioni.

- *Polisemia:* fenomeno per cui una parola può esprimere più significati.
- *Omonimia:* fenomeno per cui una stessa forma ortografica e fonologica esprime più significati.

Altri aspetti del linguaggio:

- *Comunicazione:* strumento per condividere i significati all'interno della nostra mente.
- *Convenzione:* meccanismo con cui si veicola il contenuto semantico attraverso dei simboli.
- *Granularità:* dimensione che caratterizza i modi con cui vengono concettualizzate le situazioni che si vogliono descrivere, muta il significato della parola in base a dei dettagli.
- *Soggettività:* il linguaggio è un'approssimazione delle immagini mentali, quindi è soggetto a errori.
- *Similarità:* meccanismo innato che permette di inferire il significato di un termine sconosciuto riconducendolo a un termine conosciuto.
- *Esperienza personale:* insieme di tutti gli eventi della vita di una persona che formano la conoscenza di un singolo individuo.
- *Senso comune:* convenzioni che stabiliscono il significato che la collettività dà ad alcuni termini.
- *Cultura:* il significato di alcune parole è legato alla convenzione della cultura nella quale ci si trova.

Note:-

Queste definizioni creano un'ontologia, non c'è interesse per il significato specifico dei singoli concetti, ma al significato condiviso che gli si attribuisce.

2.2 Il Significato delle Parole

Filoni di pensiero:

- *Primitive:* per rappresentare il significato di una parola lo si frammenta in piccoli contenuti semantici atomici.
- *Relazioni:* il significato di una parola non è frutto di combinazioni atomiche di primitive universali, ma nasce dalla relazione con altre parole. Nessuna parola ha un significato intrinseco se non impiegata all'interno di un contesto lessicale.
- *Composizioni:* una parola prende significato sia quando è inserita in un contesto sia quando è composta con altre parole vicine.

2.2.1 Triangolo Semiotico

Definizione 2.2.1: Triangolo Semiotico

Il Triangolo Semiotico (fig: 2.1) è un modello del significato per cui qualsiasi concetto che si ha in mente è rappresentabile attraverso un triangolo i cui poli indicano rispettivamente il concetto, il referente e la rappresentazione.

Note:-

Il referente è anche chiamato fenomeno o istanza.
Il concetto è anche chiamato significato o interpretazione.
La rappresentazione è anche chiamata segno, termine, simbolo.

Corollario 2.2.1 Concetto

Corrisponde a ciò che si ha in mente senza utilizzare una convenzione.

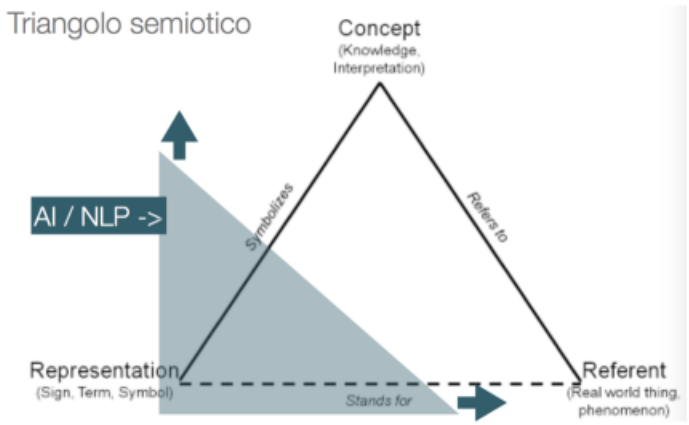

Figure 2.1: Triangolo semiotico.

Corollario 2.2.2 Rappresentazione

Si utilizza un simbolo convenzionale per comunicare il concetto.

Corollario 2.2.3 Referente

Un'istanza del concetto, ossia un elemento nel mondo reale.

Note:-

Per esempio il concetto di "gatto" in italiano e in inglese è lo stesso, ma la sua rappresentazione cambia (gatto/cat). Il referente è un qualsiasi gatto.

Domanda 2.1

Dove si collocano IA e NLP in questo triangolo e in quale direzione si muovono?

L'unico punto da cui si può partire è la rappresentazione perché nessun sistema informatico può prendere un concetto direttamente dalla nostra testa. Dall'insieme di testi presenti nel web si cerca di creare una concettualizzazione e si cerca di muoversi verso i referenti.

2.3 Multilinguismo e Granularità

2.3.1 Multilinguismo

Una delle sfide del NLP moderno è quella di trattare una pluralità di lingue, il ché è un'arma ha doppio taglio: è più difficile, ma fornisce molte più sfumature di significato. Analizzare un testo in più lingue permette di:

- Capire quali sono le informazioni semantiche più importanti, più certe o maggiormente condivise.
- Migrare informazioni semantiche da una lingua all'altra.

Osservazioni 2.3.1

Possibili problemi:

- I testi in lingue rare sono molto difficili da gestire, anche considerando che la maggior parte del web e dei testi scientifici è in inglese.
- Le lingue hanno sfumature che non possono essere tradotte direttamente.

2.3.2 Granularità

La Granularità può essere a livello di:

- *Parola:* complessità elevata.
- *Chunk:* composizione di parole (e.g. aggettivo + nome).
- *Discorso:* come per i chatBOTs.
- *Documento:* sistemi di sommarizzazione:
  - Estrattivi: estrapolano dal testo le parti più significative.
  - Astrattivo: generano nuove frasi a partire dal documento.
- *Collezione di documenti:* estrapolare gli argomenti principali (topic modelling).

2.4 Costruzione del Significato

2.4.1 Pustejovsky

Pustejovsky propose una teoria chiamata *generative lexicon* che utilizza una struttura basata su:

- *Argument Structure:* per esprimere il legame tra sintassi e semantica del concetto. In altre parola come mappare ciò che si vuole esprimere su un concetto mediante l'uso di lettere, parole e grammatica.
- *Event Structure:* per esprimere tutti i tipi di evento che coinvolgono quel concetto.
- *Qualia Structure:* per esprimere come sono definite le caratteristiche (qualia) di un concetto.
- *Inheritance Structure:* per collocare il concetto all'interno di una tassonomia per inferirne il significato.

Pustejovsky sostiene che per poter ragionare semanticamente in maniera precisa e completa abbiamo bisogno di formalizzare tutte queste strutture.

Definizione 2.4.1: Qualia

La qualia ha 4 ruoli:

- Costitutivo: esprime la parte di composizione del soggetto, riguarda il peso, la dimensione e le parti che lo compongono.
- Formale: esprime le caratteristiche che distinguono un concetto dagli altri dello stesso dominio.
- Telico: l'obiettivo o la funzione del concetto, il suo ruolo comportamentale.
- Agentive: tutte le entità (spesso umane) che rappresentano l'origine del concetto.

Note:-

Si tratta di una teoria formale in cui si assegna a ciascun elemento un ruolo e una struttura in base ai concetti definiti da Pustejovsky. In questo modo ogni frase può essere analizzata in modo formale. Il problema di questa teoria e la sua complessità che la rende difficile da implementare.

2.4.2 Hanks

Patrick Hanks enuncio una teoria del significato più semplice di quella di Pustejovsky: la *teoria delle valenze*. Questa teoria si basa sul concetto che il verbo sia la radice del significato: non esiste un'espressione di significato senza un verbo.

Definizione 2.4.2: Valenza

La valenza (fig: 2.2) è la cardinalità degli oggetti che compongono la struttura di cui il verbo è la radice. Un verbo può essere transitivo o intransitivo a vari livelli.

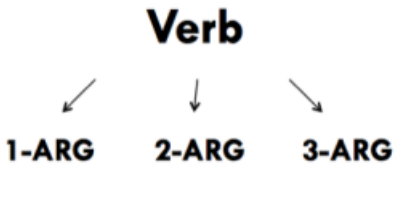

Figure 2.2: Valenze.

Note:-

Ogni valenza rappresenta un numero di argomenti chiamati slot e ogni possibile valore che possono assumere e chiamato filler.

Dati un verbo e una valenza si hanno (fig: 2.3):

- *Collocazione:* la combinazione di tutti i possibili filler.
- *Semantic Type:* delle macrocategorie che servono per raggruppare i vari filler.

Figure 2.3: Le due righe in verde rappresentano una valenza sintattica.

Domanda 2.2

Quali sono i Semantic Type? Quale deve essere il grado di generalizzazione?

- Non sempre si hanno sufficienti dati per tutte le parole, alcune sono rare e difficili da analizzare.
- I termini nei dati possono non sovrapporsi anche se sono simili.

3

Pre-LLM: Storia, Concetti e Task

3.1 Rappresentazioni

L'approccio classico alla linguistica computazionale è di tipo *top-down*, occupandosi di codificare il linguaggio naturale in qualcosa di eseguibile da un computer. Tuttavia di recente ha preso sempre più piega l'approccio statistico, *bottom-up*, sull'analisi qualitativa e quantitativa di fenomeni specifici per effettuare inferenze.

3.1.1 Rappresentazione Vettoriale

Nel *text mining* le parole vengono trattate come *token*: sequenze di caratteri prive di significato intrinseco. Un testo è considerato come un insieme di token ciascuno dei quali può comparire con una determinata frequenza.

Definizione 3.1.1: Vector Space Model (VSM)

Il Vector Space Model (VSM) introdotto da Salton è un modo per rappresentare i testi (fig: 3.1):

1. Si costruisce un dizionario con tutti i token distinti presenti nel testo associando a ciascuno un indice univoco.
2. Si calcolano le frequenze di ciascun token nel testo, ottenendo un vettore numerico che rappresenta il contenuto testuale in forma quantitativa. Se si hanno a disposizione più documenti si ottiene una matrice (generalmente sparsa) in cui:
   - Ogni riga rappresenta un documento.
   - Ogni colonna rappresenta un token del dizionario.

Note:-

Questa rappresentazione consente di reperire velocemente informazioni anche all'interno di collezioni di miliardi di elementi. Questo è reso possibile grazie all'uso del prodotto scalare tra vettori che permette di calcolare la similarità semantica tra documenti. In questo è molto usata la *cosine similarity*:

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2}\sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

3.1.2 Metodi Statistici

I metodi statistici applicati ai documenti si basano principalmente su:

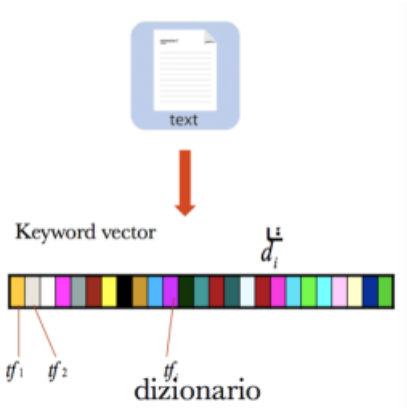

Figure 3.1: Rappresentazione testuale tramite VSM.

- *Frequenza dei termini:* rappresenta l'importanza o la rilevanza di una parola all'interno di un documento. Le frequenze assolute non sono sempre efficaci per cui si devono normalizzare. Un approccio utilizzato è il modello *TF-IDF* che combina due misure:
  - TF (Term Frequency): la frequenza di un termine normalizzata rispetto alla lunghezza del documento.
  - IDF (Inverse Document Frequency): inversamente proporzionale al numero di documenti in cui il termine appare, calcolata come rapporto logaritmico tra il numero totale dei documenti e il numero di documenti che contengono quel termine.

$$TF\text{–}IDF = \frac{n_{i,j}}{|d_j|} \times \log \frac{|D|}{|\{d : i \in d\}|}$$

- *Co-occorrenza delle parole:* misura la tendenza di due parole comparire insieme all'interno di un determinato contesto (frase, paragrafo, documento, corpus). Si basa sull'assunzione distribuzionale per cui termini semanticamente simili tendono a comparire in contesti simili. Solitamente le co-occorrenze sono rappresentate da una matrice quadrata $|D|X|D|$.

3.2 Task e Applicazioni Classiche

3.2.1 Tag Clouds

La prima applicazione sono le tag clouds (fig: 3.2) in cui l'associazione del peso di dominanza a una parola viene espressa attraverso la grandezza della parola stessa in una nuvola di parole. Volendo si può integrare la co-occorrenza mettendo vicine parole simili.

Figure 3.2: Esempio di Tag Clouds.

3.2.2 Document Clustering

Con *Clustering* (fig: 3.3) si intende qualsiasi approccio non supervisionato di separazione di documenti in sotto-gruppi più o meno omogenei. Non c'è bisogno di un training set, vengono utilizzate solo frequenze, pesi, co-occorrenze, etc. Esistono due concetti alla base del Clustering:

- Non esiste il Clustering perfetto.
- Non esite sempre una misura oggettiva per valutare la bontà di un Clustering.

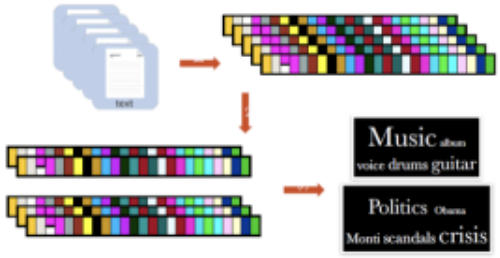

Figure 3.3: Pipeline con clustering e tag clouds.

3.2.3 Document Classification

Per la classificazione/categorizzazione (fig: 3.4) si vuole ricondurre un testo a una determinata etichetta all'interno di un set di etichette. Per la classificazione è facile effettuare una valutazione perché, avendo già una tassonomia "popolata" con alcune istanze, è possibile fare training con dei testi e poi valutare il modello con altri testi.

Figure 3.4: Classificazione di testi su label.

3.2.4 Document Segmentation

Questo task consiste nel separare diverse parti all'interno di un documento cercando di mantenere insieme aree semanticamente coerenti tra loro. L'algoritmo più famoso per svolgere il task è il *text tiling*: sull'asse x si ha l'indice del numero della frase, mentre sull'asse y si hanno le parole con la frequenza delle diverse frasi. Il text tiling è una tecnica semplice:

1. *Separazione:* del testo in finestre di lunghezza fissa.

2. *Calcolo della coesione intra-gruppo:* il valore di coesione è semplicemente quanto si usano le stesse parole tra blocchi successivi di frasi (o tokens).

3. *Ricerca:* di parti di testo a bassa coesione circondate da parti di testo ad alta coesione. I picchi più importanti sono quelli verso il basso, ossia i *break-points* (fig: 3.5).

4. *Riadattamento:* delle finestre rispetto al break-point più vicino.

Note:-

L'algoritmo è iterativo, quella descritta qui sopra è solo un'iterazione. Al termine delle iterazioni le finestre non hanno più una dimensione prestabilità perché sono state riadattate rispetto ai break-points.

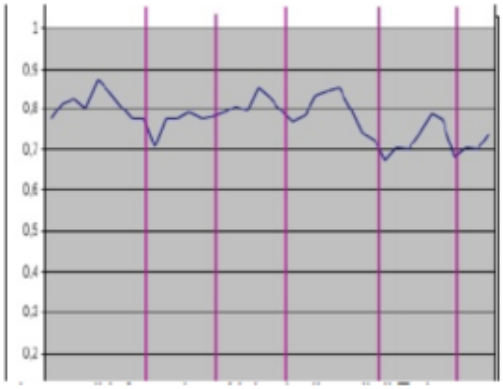

Figure 3.5: Break-points nel text tiling.

3.2.5 Document Summarization

Per ripassare brevemente questo task, esistono due metodi:

- *Estrattivi:* dato un testo o una collezione cercare di dare un valore di *salience* (importanza) alle frasi con l'obiettivo di creare un nuovo documento con quelle frasi. Uno degli algoritmi storici è il *text ranking* che genera un rank delle frasi sulla base della salience.

- *Astrattivi:* molto più complessi. Sono relativamente recenti e basati su reti neurali avendo in input molti documenti con i loro rispettivi riassunti.

Note:-

La valutazione di un riassunto è spesso effettuata mediante ROUGE (Recall-Oriented Understudy for Gisting Evaluation): un sistema che mappa gli ngrammi dei riassunti fatti da esseri umani con gli ngrammi dei riassunti generati.

3.2.6 Information Retrieval

Si basa sul recupero di un documento di interesse sulla base di un seti di keyword (query). Queste ultime possono anche essere collegate a ontologie o metadati. Inizialmente questo task era basato su un modello booleano: si cercava un match diretto. Al giorno d'oggi si effettuano analisi più sofisticate, tenendo in considerazione il contesto e la semantica dei documenti.

3.3 Semantica Distribuzionale

Un'evoluzione del text mining con l'aggiunta dell'uso del linguaggio e dei principi di linguistica.

3.3.1 Introduzione

Carrellata Storica

- 1954: Harris afferma che le parole che occorrono in contesti simili tendono ad avere significati simili.
- 1957: Firth afferma che una parola è caratterizzata da quelle che la accompagnano.
- 1983: Furnas sostiene che la congiunzione di varie parole permette di specificare più facilmente l'oggetto del discorso.
- 1990: Deerwester: introduce i concetti latenti, con cui afferma che esiste una struttura di base che viene parzialmente oscurata dalla scelta "casuale" delle parole che viene fatta per comporre un discorso.
- 2003: Bley introduce una visione probabilistica secondo cui il tema del documento influenza in modo probabilistico le parole utilizzate nel documento.
- 2003: Turney impiega il concetto di *coppie di parole*. Se una certa coppia (x, y) presenta più o meno gli stessi pattern che presenta la coppia (w, z) allora è possibile costruire una sorta di proporzione numerica nel significato delle due coppie. In questo modo si cattura la semantica relazionale.

La semantica distribuzionale ha avuto diversi nomi a seconda dei contesti:

- *Linguistica:* si parla di *Distributional Hypothesis*.
- *NLP: Distributional Semantics*.
- *Information Retrieval: Vector Space Models* o *Latent Semantic Analysis*.
- *Scienze cognitive: Conceptual Space* (i concetti cognitivi sono interpretati attraverso una serie di qualities) o *Graded Categorization* (tutti gli studi che si basano su come il cervello interpreti gli oggetti).
- *Psicometria: Hyperspace Analogue to Language*.
- *Graph Theory: Matrici di Adiacenza*.

3.3.2 Le Matrici

Domanda 3.1

Perché usare le matrici? Come mai sono state usate solo le matrici? Perché questo approccio funziona bene?

Risposta: la rappresentazione vettoriale è un'approssimazione, esattamente come lo è il linguaggio.

Le matrici sono una via di mezzo:

- *Rappresentazione simbolica:* propria dei metodi formali di rappresentazione, rende possibile fare inferenze logiche. Nella logica formale i *simboli* sono solo simboli che hanno un potere inferenziale all'interno di una base di conoscenza (KB).
- *Rappresentazione associazionistica/connessionistica:* teorie legate alle scienze cognitive in cui si pensa che tutto sia connesso con tutto. Il learning consiste nell'apprendere i pesi di tutte le connessioni.

Note:-

Le matrici facilitano la condivisione della conoscenza in cui il significato diventa una *regione geometrica*.

Esistono varie tecniche per utilizzare le matrici:

- *Similarità:* cosine similarity, Jaccard similarity, etc.
- *Trasformazioni matriciali:* SVD, NNMF, etc.
- *Clustering:* K-means, EM, etc.

Esistono procedure di pre-processing che dovrebbero essere sempre applicate:

- *Normalizzazione:* procedimento lessicale, sintattico o morfologico, ossia tokenizzazione, lemmatizzazione e stemming.
- *Denormalizzazione:* arricchimento semantico attuato mediante named entities, semantic roles e associazioni semantiche. Per esempio la WSD può essere considerata una tecnica di denormalizzazione.

Nel 2010 Peter Turney distingue tre configurazioni matriciali primarie:

- *Term-Document Matrix:* considera i documenti espressi come termini. Su ogni riga si ha un documento e su ogni colonna un termine. Questa configurazione viene utilizzata per effettuare il calcolo della similarità, il clustering, la classificazione, la segmentazione, parzialmente il Q&A, etc.
- *Term-Context Matrix:* è la generalizzazione della Term-Document. Su ogni riga si ha un contesto e su ogni colonna un termine. Un contesto non deve necessariamente essere un documento, ma può essere un paragrafo, una frase, una dipendenza sintattica, etc.
- *Pair-Pattern Matrix:* la proposta di Turney. Su ogni riga si hanno coppie di parole, mentre su ogni colonna si ha un pattern[1]. I pattern mettono in relazione tutte le possibili coppie di parole attraverso relazioni. Questa matrice viene usata per calcolare la *relational similarity*, la *pattern similarity*, la *relational clustering* e la *relational search*.

3.3.3 Il Ruolo della Similarità

La similarità è fondamentale, infatti la semantica distribuzionale è anche conosciuta come la semantica della similarità. Negli anni '60, Quine mette in luce il ruolo della similarità nell'apprendimento e nel pensiero poiché consente di categorizzare le cose. Per prevedere le funzionalità di oggetti sconosciuti l'essere umano fa riferimento a oggetti simili di cui conosce l'utilizzo/significato. Nella linguistica computazionale sono definiti vari tipi di similarità:

- *Semantic Similarity:* concetti che hanno quasi lo stesso significato (sinonimia).
- *Semantic Relatedness:* concetti che condividono proprietà, che hanno affinità semantica. Possono essere meronimi (costituente di qualcosa), antonimi (significato opposto di qualcosa), ma anche sinonimi. Ciò che è semanticamente simile è anche semanticamente relazionato, ma non è vero il contrario. Purtroppo questo tipo di semantica è quasi inutilizzabile perché troppo generica (dice che due concetti sono relazionati ma non il come).
- *Attributional Semantic:* concetti che condividono attributi (proprietà). In realtà si tratta di Semantic Relatedness.
- *Taxonomical Similarity:* concetti che condividono più iperonimi (concetti generici rispetto a dei concetti più specifici).
- *Relational Similarity:* lavora su tuple di concetti.
- *Semantic Association:* tratta concetti che co-occorrono frequentemente. Molto simile alla relatedness, ma orientata alla corpus analysis. Infatti è possibile avere concetti che non co-occorrono spesso ma che sono comunque relazionati.

Osservazioni 3.3.1

- La Distributional Semantic ha come cardine la similarità.
- Tale concetto è fragile perché persone diverse possono attribuire valori diversi alla similarità.

[1] Fortunatamente non un GoF.

3.4 Semantica Documentale

Definizione 3.4.1: Semantica Documentale

La semantica documentale riguarda tutte le analisi le ricerche effettuate a livello di collezione di documenti.

3.4.1 Topic Modelling

Domanda 3.2

Che cos'è un topic modelling?

Definizione 3.4.2: Topic Modelling

Un topic modelling è un modello statistico o probabilistico che analizza l'uso del linguaggio e individua automaticamente gli argomenti di una collezione di testi.

Note:-

Il topic modelling è un modello non supervisionato, non necessità di annotazioni manuali.

Un topic è rappresentato come una lista pesata di parole, non va pensato come un argomento strutturato (passaggio riservato a un'annotazione manuale). I topic vengono estratti con varie tecniche misurando quanto i termini vengono usati negli stessi contesti. È possibile che i topic estratti non siano utili e che siano solo coincidenze.

Definizione 3.4.3: Latent Semantic Analysis (LSA)

Prima tecnica di topic modelling, si tratta dell'applicazione di una fattorizzazione matriciale (*Singular Value Decomposition*) che prende in input una matrice, nel nostro caso l'insieme dei vettori del dizionario contenenti le frequenze normalizzate e crea in output tre matrici che approssimano quella di partenza.

Osservazioni 3.4.1

- La prima matrice è una nuova rappresentazione multidimensionale dei testi ma con nuove features chiamate *concetti latenti*.
- La seconda matrice è la più particolare: contiene 0 in tutte le celle tranne quelle diagonali in cui contiene i cosiddetti *Singular Value*.
- La terza matrice è una trasposta delle features latenti.

Dettagli di LSA:

- Data la matrice di partenza, con la frequenza delle parole, LSA riconosce le ridondanze.
- Nel caso di matrici Term-Document viene catturata l'informazione di co-occorrenza creando nuove features che accorpano tali co-occorrenze, dando vita a concetti latenti in un nuovo spazio vettoriale.
- I concetti latenti possono essere ordinati dal più importante al meno importante, in questo modo si può approssimare in matrici più piccole.
- Viene effettuata un'analisi delle varianze e una riorganizzazione del contenuto sotto forma delle varianze maggiori verso quelle minori.
- SVD applicata a matrici che rappresentano testi ha due vantaggi:
  - Permette di avere molte meno dimensioni.
  - Riduce la sparsità dei dati.

Esempio 3.4.1 (LSA)

Tratto dagli appunti del prof. Luigi Di Caro.
Immaginiamo di avere due documenti d_1 e d_2 all'interno di una grande base documentale. Se si effettua la Cosine Similarity tra d_1 e d_2 (le celle bianche in figura indicano valori uguali a 0) il risultato sarà zero, ovvero i due documenti saranno identificati come semanticamente dissimili. Ma se i termini in grigio iniziali di d_1 e i termini grigi finali di d_2 analizzati dal calcolo dell'SVD vengono accorpati all'interno di singoli concetti latenti (poiché co-occorrono all'interno di un contesto indiretto, cioè negli altri documenti che non sono nè d_1 nè d_2), diventando una nuova dimensione della matrice finale (in arancione, in figura). Su questo nuovo spazio vettoriale, la Cosine Similarity darà un risultato di similarità maggiore di 0, indice di una similarità positiva per quei due documenti.

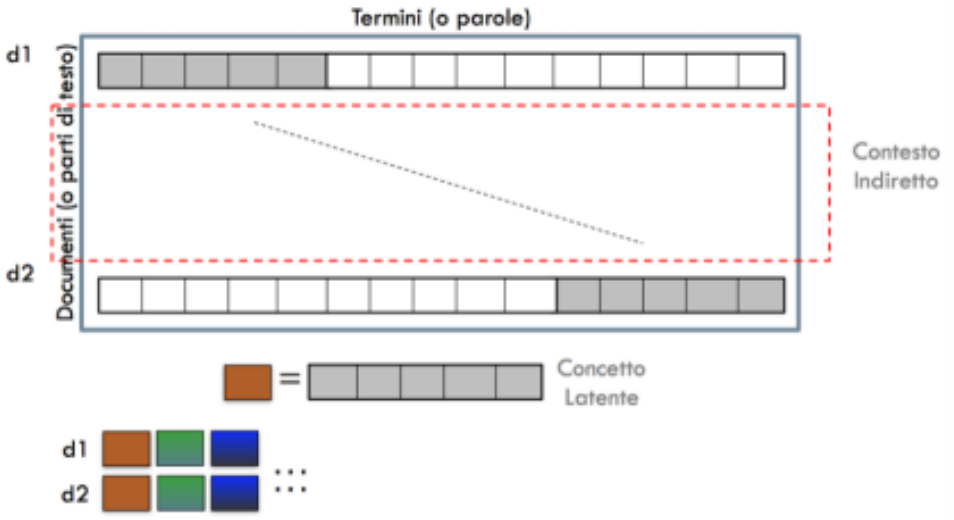

Figure 3.6: Contesti indiretti e concetti latenti con SVD.

LSA ha alcuni problemi:

- È un modello che non generalizza su documenti non visti: se si aggiungono documenti alla base documentale allora bisogna ricalcore tutto SVD.
- Valori negativi dopo la fattorizzazione sono difficilmente interpretabili.

Dopo la LSA:

- Un'evoluzione naturale si ha nella pLSA, una versione probabilistica.
- Successivamente si passa alla *Latent Dirichlet Allocation* (LDA) che sfrutta la *statistica Bayesiana*, parte dal presupposto che un documento sia un mix di topics e ogni parola ha una certa probabilità di appartenere a ogni singolo topic. Questa caratteristica ha una duplice funzione, da un insieme di parole:
  - È possibile dedurre il topic di appartenenza.
  - È possibile dedurre altre parole legate al topic.

3.4.2 Text Visualization

Un problema importante per il text mining e la semantica documentale riguarda il come rappresentare uno spazio a n dimensioni (i termini) in uno spazio bidimensionale.

Definizione 3.4.4: Text Visualization

Area di ricerca che si occupa di studiare come visualizzare il testo in modo tale da riuscire a trasferire un certo contenuto semantico a un utente mediante metodi grafici.

Approcci alla text visualization:

- *Parallel Coordinates:* ogni dimensione viene associata a una coordinata parallela alle altre. Un punto nello spazio multidimensionale diventa una retta che congiunge i valori per quelle dimensioni. Il problema di questo approccio è che se ci sono tanti dati bisogna comprimerli.

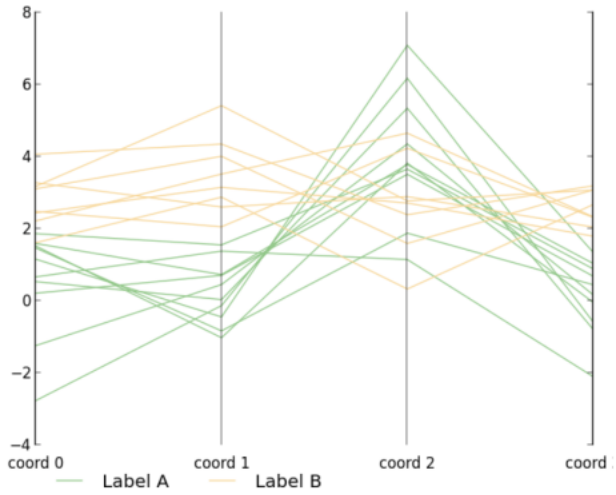

Figure 3.7: Parallel Coordinates a 4 dimensioni.

- *RadViz:* radial visualization. Le dimensioni vengono inserite all'interno di una circonferenza. I punti all'interno sono le istanze mentre la loro posizione deriva dall'attrazione (valore di importanza) delle features poste sulla circonferenza. Il problema è il conflitto gravitazionale tra features differenti che si può annullare a vicenda.

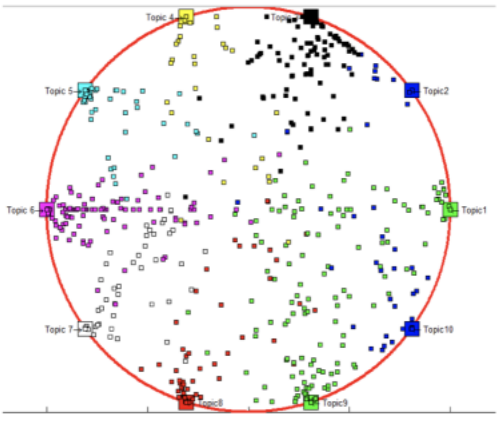

Figure 3.8: Visualizzazione con RadViz.

- *HeatMap:* i dati vengono espressi mediante una matrice e il colore o l'intensità del colore esprime il valore numerico.

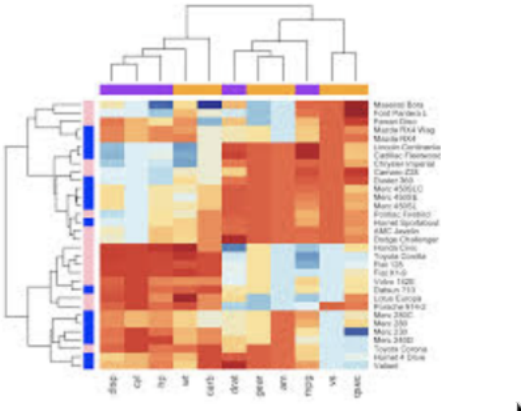

Figure 3.9: Visualizzazione con HeatMap.

- *Correlation Circle:* cerchi in cui sulla circonferenza vengono posti degli elementi e i collegamenti tra questi rappresentano la loro correlazione.

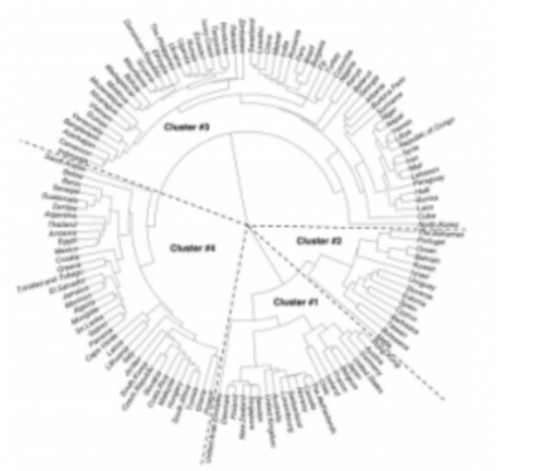

Figure 3.10: Visualizzazione con Correlation Circle.

3.5 Ontology Learning e Open Information Extraction

L'ontology learning è il task che consiste nel costruire un'ontologia con dati non strutturati.

3.5.1 Ontology Learning

Philipp Cimiano propone una visione dell'ontology learning come reverse engineering: data una conoscenza di un certo dominio, con la sua rappresentazione e la sua codifica si cerca di ritornare alla concettualizzazione di partenza. Questo processo ha due problemi principali:

- La conoscenza del mondo non è codificata e quindi non è uguale per tutti. Si possono avere visioni del mondo differenti.
- Non tutto quello che si sa del dominio viene effettivamente utilizzato: si potrebbe conoscere bene un sistema, ma quando lo si va a concettualizzare si pensa al suo utilizzo.

Sottospecializzazioni dell'ontology learning:

- *Ontology Population:* si ha a disposizione un'ontologia già costruita e si vuole analizzare la base testuale per trovare delle istanze da collocare all'interno dei concetti già esistenti dell'ontologia. Si cerca quindi di popolare l'ontologia con nuove istanze.
- *Ontology Annotation:* data un'ontologia già costruita e una base documentale l'obiettivo è quello di taggare il testo con delle informazioni concettuali come annotazione semantica all'interno di un testo.
- *Ontology Enrichment:* data un'ontologia già costruita e una base documentale l'obiettivo è dire qualcosa sull'ontologia stessa eventualmente ristrutturandola sia a livello di concetti che di relazioni.

Osservazioni 3.5.1

- Una rappresentazione più sofisticata è basata su un *thesaurus* (dizionario di sinonimi) come WordNet.
- Man mano che si passa a metodi più sofisticati la difficoltà aumenta.

Task dell'ontology learning:

- *Term Extraction:* trovare nomi per concetti e per le loro relazioni.
- *Synonim Extraction:* estrazione di paroli con lo stesso significato in determinati contesti.
- *Concept Extraction:*
  - *Intensionale:* si cerca di astrarre e rappresentare tutto ciò che un concetto può descrivere.
  - *Estensionale:* si vogliono enumerare tutti gli elementi che descrivono un determinato concetto.
- *Concept Hierarchies Induction:* si cerca di strutturare concetti già noti attraverso una tassonomia.
- *Relation Extraction:* come Concept, ma per le relazioni.
- *Population:* spesso fatto attraverso Named-Entity Recognition (NER) o Information Extraction.
- *Notazione di sussunzione:* meccanismi formali legati al campo della logica matematica che permettono di costruire automaticamente gerarchie.

Metodi per soddisfare i tasks:

- NLP: estrazione e uso di informazioni (PoS, NE), pre-processing, regole su alberi di parsing (CKY), informazioni statistiche, risorse lessicali, etc.
- *Formal Concept Analysis (FCA):* utilizza oggetti (concetti con associate features), attributi e l'incidenza (il fatto che un oggetto abbia o meno un determinato attributo). L'incidenza viene espressa come una matrice chiamata *formal context*. Sulla base del formal context si possono creare operatori per analizzare la matrice per dedurre informazioni. In particolare a ogni elemento si può applicare uno di questi operatori:
  - Up: sulle colonne, fornisce gli atributi che un oggetto possiede.
  - Down: sulle righe, specifica quali oggetti possidono un determinato attributo.
- Machine/Deep learning: addestrare una rete per la costruzione di strutture concettuali e ontologie.

3.5.2 Open Information Extraction

L'open information extraction (OIE) nasce dalla necessità di estrarre efficientemente grandi quantità di informazioni da corpora di grandi dimensioni. Queste operazioni son computazionalmente costose, soprattutto se si tratta di miliardi di documenti. Tipicamente si estraggono triple costituite da argomento, espressione verbale e ulteriore argomento. Il problema di questo tipo di estrazione è che si rischia di estrarre molto rumore. Alla fine questo è un approccio data-oriented che permette di effettuare estrazioni *pseudo-semantiche* su un insieme ridotto di dati testuali. Questo approccio è buono per fare Q&A senza utilizzare un LLM.

Problemi di OIE:

- Non esite un modo rigoroso o unico di estrarre le triple che quindi risultano disallineate in sistemi diversi.
- È difficile valutare e comparare questi sistemi perché non esiste un *gold standard*.

Note:-

I sistemi più famosi sono:

- ReVerb: basato su vincoli sintattici.
- KrankeN: utilizza WSD.
- ClausIE: non estrae solo triple.
- DefIE: combina parsing e WSD creando un grafo su cui si applicano pesatura degli archi e filtri di rilevanza.

4

Clustering e Topic Modelling, LLM e Prompting

4.1 Clustering Testuale con Embeddings

La pipeline per il Clustering Testuale si articola in tre fasi:

1. Conversione dei documenti in vettori (*embeddings*): su *Hugging Face* è disponibile un modello pre-addestrato chiamato *General Text Embeddings* (GTE).
2. Riduzione della dimensionalità degli embeddings: dato che gli embeddings hanno dimensioni elevate è necessario ridurli, per esempio con *UMAP*.
3. Clustering dei vettori in gruppi significativi: per raggruppare documenti simili, si può usare *HDBSCAN*.

Note:-

Successivamente c'è la fase di visualizzazione dei risultati (fig: 4.1), come visto in precedenza.

Figure 4.1: Visualizzazione dei dati.

4.1.1 BERTopic

Definizione 4.1.1: BERTopic

BERTopic è un framework moderno per il topic modelling. Si basa su:

- Una pipeline di clustering su embeddings testuali.
- L'estrazione di parole salienti per ogni cluster attraverso *class-based TF-IDF*.

È possibile accedere alle informazioni dei topic estratti con i comandi:

- `topic_model.get_topic_info()`.
- `topic_model.get_topic()`.

Sono supportate diverse funzioni di interrogazione e visualizzazione:

- Ricerca di topic associati a una query.
- Diagrammi interattivi:
  - `visualize_documents()`.
  - `visualize_barchart()`.
  - `visualize_hierarchy()`.
  - `visualize_heatmap()`.