Marco Ottina 795130
Davide Sgaramella 938632

A.A. 2019/2020

Università degli Studi di Torino – polo Scienze della Natura
Laurea Magistrale in Informatica
Corso: Tecnologie del linguaggio naturale
Parte di: Luigi Di Caro

Esercitazione 2

L'esercitazione richiede di formulare un algoritmo che, dato un insieme di definizioni espresse sotto forma di singola frase, restituisca il synset che meglio corrisponda a tali definizioni, similarmente a quanto una persona sa fare.

Algoritmo

Possiamo dividere l'algoritmo in 3 fasi principali:

1. ***Inizializzazione***;
2. ***Ricerca***;
    a. Ricerca dei *genus*;
    b. Ricerca dei *synset*;
3. ***Elaborazione risultato***.

Per prima cosa si inizia con la lettura del contenuto del CSV, lo si divide in base ai concetti che stiamo cercando di trovare, in modo tale da avere degli insiemi di frasi e poi di parole relativi ad un concetto.
Su ognuna delle frasi si è effettuata la classica operazione di ***preprocessing*** attraverso cui rimuovere la *punteggiatura* e le *stopwords*.
Dopo questo passaggio, che possiamo considerare una sorta di preparazione dei dati prima dell'esecuzione dell'algoritmo, si passa a creare gli insiemi di parole su cui l'algoritmo lavorerà: per questo si uniscono tutte le parole delle frasi relative ad un concetto in una lista relativa al concetto stesso; al termine di questo passaggio si otterrà la lista *candidate_genus* che conterrà le liste di parole relative ai concetti obiettivo (8 sotto-liste per 8 concetti).
Successivamente inizia la seconda fase ovvero l'algoritmo vero e proprio, in quanto parte la **ricerca del *genus.*** La ricerca viene effettuata in modo da circoscrivere il dominio delle parole su cui si lavorerà successivamente, sarà scelto quindi il 10% delle parole più comuni nella lista di un concetto da cui estrarre poi il genus tra quelle con maggiore frequenza.

Si passa in seguito alla **ricerca dei synset** all'interno di **WordNet** relativi ai genus trovati nel passaggio precedente. Si cercano quindi i synset associati al genus per poi trovare quello più adeguato, ovvero il synset da restituire in output. Ed è qui che parte la terza ed ultima fase dell'algoritmo: la **fase di ricerca del best synset.**

La terza fase quindi risulta essere quella cruciale: in questo contesto lavoriamo sui possibili genus trovati all'inizio (*candidate_genus*) ed i synset dei *genus* trovati. Per trovare il risultato finale si cercheranno nella tassonomia di WordNet, relativa al genus su cui si sta lavorando, gli iponimi del genus stesso; una volta trovati, avviene il processamento degli stessi per capire quale sia, e quindi scegliere, il synset più adeguato.
A questo punto si introducono gli *esempi* e le *definizioni* (sempre forniti da WordNet) degli iponimi relativi al genus e si cerca il valore massimo di intersezione tra la lista di parole in *candidate_genus* e la lista di parole preprocessate ottenuta dall'unione della definizione e degli esempi relativi ad ogni iponimo.

Risultati

Il risultato finale sarà il synset con valore di intersezione maggiore.

I risultati prodotti dall'algoritmo sono elencati nella seguente tabella.

Output Atteso	**Output ottenuto**
Justice	Synset('right.n.01')
Patience	Synset('ancients.n.01')
Greed	Synset('greed.n.01')
Politics	Synset('politics.n.05')
Food	Synset('embryo.n.02')
Radiator	Synset('latent_heat.n.01')
Vehicle	Synset('air_transportation_system.n.01')
Screw	Synset('solder.n.01')