UNIVERSITÀ DEGLI STUDI DI TORINO

DIPARTIMENTO DI INFORMATICA

CORSO DI LAUREA IN INFORMATICA

Laurea Magistrale

Tecnologie del linguaggio naturale

Relazione Content To Form

Stefano Locci

ANNO ACCADEMICO 2018/2019

Motivazioni

I dizionari comuni (*forward dictionary*) vengono utilizzati per ricercare il significato di una parola, quindi si ha in input una parola e in output un significato. Un problema molto comune riguarda, però, l'opposto, ovvero si conosce il significato del concetto, ma non si riesce a ricordare la parola con quel significato. Questo problema è noto come "Tip of the Tongue problem".
Solitamente, quando ci si trova in queste situazioni, si cerca di descrivere il concetto che si vuole esprimere creando un contesto composto da definizioni, esempi e sinonimi in modo tale da farsi venire in mente la parola corretta. L'obiettivo di questo progetto è quindi quello di fornire uno strumento che risolva il problema del Tip of the Tongue attraverso l'utilizzo di un dizionario analogico, un particolare tipo di dizionario il quale, dato un input corrispondente ad una o più definizioni, restituisce in output la parola alla quale le definizioni fanno riferimento.
In particolare verranno eseguiti alcuni esperimenti volti a dimostrare la "fragilità" delle definizioni e la complessità della costruzione di un dizionario analogico.

Approccio al problema

Il codice è stato sviluppato in Python con l'ausilio di due librerie: Pandas[1], usata per l'accesso e la memorizzazione dei dati (file .xlsx contentente le definizioni) e Nltk[2] per accedere a WordNet. Per il task richiesto l'input corrisponde a 8 parole con 12 definizioni ciascuna. L'algoritmo dovrà elaborare le definizioni e, sfruttando WordNet, restituire la parola che si adatta meglio alle definizioni date. Per reperire i dati necessari per la costruzione del dizionario inverso è stata utilizzata la risorsa online WordNet: un ricco database lessicale che organizza le parole in gruppi di sinonimi chiamati "synsets" i quali vengono connessi tra di loro attraverso delle relazioni semantiche e gerarchiche. Inoltre, WordNet fornisce per ogni synset una definizione ed un esempio di utilizzo della parola all'interno di una frase.
Per risolvere il problema sono state adottate diverse strategie di pre-processing delle definizioni e differenti misure di similarità tra le definizioni date in input e quelle presenti in WordNet, nelle sezioni seguenti verranno analizzate singolarmente.

Lexical Overlap

In questo primo tentativo si è deciso di testare una misura di similarità molto semplice che calcola il numero di parole che compaiono nelle due definizioni da confrontare normalizzando sulla lunghezza minima tra le due definizioni (inteso come numero di parole della definizione).

$$\frac{|A \cap B|}{min\{len(A), len(B)\}},\ con\ A, B\ = definizioni$$

Questa misura di similarità non produce buoni risultati poiché fa un semplice controllo lessicale e non semantico. Ciò significa che esisteranno molte definizioni all'interno di WordNet che avranno più match con la definizione in input (corrispondente ad un'unica stringa pari alla concatenazione delle 12 definizioni), andando ad "inquinare" il risultato. I risultati seguenti mostrano due esecuzioni, una senza pre-processing delle definizioni e una applicando uno stemming[3] ciascuna esecuzione è stata effettuata sia considerando tutti i synset di WordNet, sia restingendo lo spazio di ricerca considerando solamente i nomi comuni:

- No stemming, tutti i synset:
  - **Justice**: [(Synset('rightfully.r.01'), 1.0), (Synset('thereon.r.01'), 1.0), (Synset('thereto.r.01'), 1.0), (Synset('dextral.a.01'), 0.8), (Synset('strung.a.01'), 0.8)]

[1]Pandas: https://pandas.pydata.org/
[2]Nltk: http://www.nltk.org/
[3]Porter Stemmer NLTK: http://www.nltk.org/howto/stem.html

- **Politics**: [(Synset('demotic.s.02'), 1.0), (Synset('home.s.03'), 1.0), (Synset('inside.r.02'), 1.0), (Synset('thereon.r.01'), 1.0), (Synset('thereto.r.01'), 1.0)]
- **Greed**: [(Synset('lavish.s.01'), 1.0), (Synset('low-level.s.01'), 1.0), (Synset('fiery.s.03'), 1.0), (Synset('bad.s.02'), 1.0), (Synset('consuming.s.01'), 1.0)]
- **Radiator**: [(Synset('stuffed.s.01'), 1.0), (Synset('watery.s.01'), 1.0), (Synset('thereto.r.01'), 1.0), (Synset('subpart.n.01'), 1.0), (Synset('up.v.01'), 1.0)]
- **Patience**: [(Synset('clean.s.14'), 1.0), (Synset('trouble-free.s.01'), 1.0), (Synset('angrily.r.01'), 1.0), (Synset('unpleasingness.n.01'), 1.0), (Synset('nastiness.n.03'), 1.0)]
- **Food**: [(Synset('cardboard.s.01'), 1.0), (Synset('in_perpetuity.r.01'), 1.0), (Synset('in_vivo.r.01'), 1.0), (Synset('thereto.r.01'), 1.0), (Synset('person.n.01'), 1.0)]
- **Vehicle**: [(Synset('between.r.02'), 1.0), (Synset('thereon.r.01'), 1.0), (Synset('thereto.r.01'), 1.0), (Synset('thing.n.04'), 1.0), (Synset('demotic.s.02'), 0.8333333333333334)]
- **Screw**: [(Synset('hand_in_hand.r.01'), 1.0), (Synset('double.r.02'), 1.0), (Synset('thereon.r.01'), 1.0), (Synset('thereto.r.01'), 1.0), (Synset('rest.n.06'), 1.0)]

- No stemming, solo synset con POS nome:
  - **Justice**: [(Synset('publicity.n.02'), 0.875), (Synset('unpleasingness.n.01'), 0.8), (Synset('cloudiness.n.03'), 0.8)]
  - **Politics**: [(Synset('common_good.n.01'), 1.0), (Synset('third_estate.n.01'), 1.0), (Synset('baronetage.n.02'), 0.8)]
  - **Greed**: [(Synset('acquisitiveness.n.01'), 1.0), (Synset('longer.n.01'), 1.0), (Synset('slice.n.01'), 1.0)]
  - **Radiator**: [(Synset('subpart.n.01'), 1.0), (Synset('shaker.n.03'), 0.875), (Synset('air_pump.n.01'), 0.8)]
  - **Patience**: [(Synset('unpleasingness.n.01'), 1.0), (Synset('nastiness.n.03'), 1.0), (Synset('becomingness.n.01'), 1.0)]
  - **Food**: [(Synset('person.n.01'), 1.0), (Synset('organism.n.01'), 0.5333333333333333), (Synset('object.n.01'), 0.45454545454545453)]
  - **Vehicle**: [(Synset('thing.n.04'), 1.0), (Synset('location.n.01'), 0.8333333333333334), (Synset('person.n.01'), 0.6666666666666666)]
  - **Screw**: [(Synset('rest.n.06'), 1.0), (Synset('strip.n.05'), 1.0), (Synset('receptacle.n.01'), 0.8181818181818182)]

- Con stemming, solo synset con POS nome:
  - **Justice**: [(Synset('legal_right.n.01'), 1.0), (Synset('raison_d'etre.n.01'), 1.0), (Synset('occasion.n.03'), 1.0), (Synset('subpart.n.01'), 1.0), (Synset('contempt_of_court.n.01'), 0.875)]
  - **Politics**: [(Synset('common_good.n.01'), 1.0), (Synset('dictate.n.02'), 1.0), (Synset('point.n.07'), 1.0), (Synset('third_estate.n.01'), 1.0), (Synset('commissioner.n.01'), 1.0)]

- **Greed**: [(Synset('extravagance.n.03'), 1.0), (Synset('personhood.n.01'), 1.0), (Synset('acquisitiveness.n.01'), 1.0), (Synset('exorbitance.n.01'), 1.0), (Synset('longer.n.01'), 1.0)]
- **Radiator**: [(Synset('heater.n.01'), 1.0), (Synset('metalwork.n.01'), 1.0), (Synset('producer.n.03'), 1.0), (Synset('subpart.n.01'), 1.0), (Synset('inflation.n.04'), 0.857142 8571428571)]
- **Patience**: [(Synset('unpleasingness.n.01'), 1.0), (Synset('nastiness.n.03'), 1.0), (Synset('becomingness.n.01'), 1.0), (Synset('duration.n.03'), 1.0), (Synset('hand.n.04'), 1.0)]
- **Food**: [(Synset('person.n.01'), 1.0), (Synset('crotophaga.n.01'), 1.0), (Synset('life.n.10'), 0.6666666666666666), (Synset('dwarf.n.03'), 0.625), (Synset('organism.n.01'), 0 .6)]
- **Vehicle**: [(Synset('thing.n.04'), 1.0), (Synset('location.n.01'), 0.8333333333333334), (Synset('escape.n.05'), 0.8333333333333334), (Synset('article.n.02'), 0.8), (Synset('p erson.n.01'), 0.6666666666666666)]
- **Screw**: [(Synset('gaze.n.01'), 1.0), (Synset('rest.n.06'), 1.0), (Synset('strip.n.05'), 1.0), (Synset('subpart.n.01'), 1.0), (Synset('connection.n.06'), 1.0)]

- Con stemming, tutti i synset:
  - **Justice**: [(Synset('half-and-half.s.01'), 1.0), (Synset('single-minded.s.01'), 1.0), (Synset('alike.r.01'), 1.0), (Synset('chiefly.r.01'), 1.0), (Synset('morally.r.01'), 1.0)]
  - **Politics**: [(Synset('demotic.s.02'), 1.0), (Synset('home.s.03'), 1.0), (Synset('puissant.s.01'), 1.0), (Synset('stiff.s.02'), 1.0), (Synset('based.s.01'), 1.0)]
  - **Greed**: [(Synset('plummy.s.01'), 1.0), (Synset('undesirable.a.01'), 1.0), (Synset('lavish.s.01'), 1.0), (Synset('calculating.s.01'), 1.0), (Synset('low-level.s.01'), 1.0)]
  - **Radiator**: [(Synset('stuffed.s.01'), 1.0), (Synset('watery.s.01'), 1.0), (Synset('objectively.r.01'), 1.0), (Synset('thereto.r.01'), 1.0), (Synset('heater.n.01'), 1.0)]
  - **Patience**: [(Synset('clean.s.14'), 1.0), (Synset('creepy.s.01'), 1.0), (Synset('slumberous.s.01'), 1.0), (Synset('time-honored.s.01'), 1.0), (Synset('trouble-free.s.01'), 1. 0)]
  - **Food**: [(Synset('accordant.s.02'), 1.0), (Synset('cardboard.s.01'), 1.0), (Synset('in_perpetuity.r.01'), 1.0), (Synset('in_vivo.r.01'), 1.0), (Synset('needlessly.r.01'), 1.0)]
  - **Vehicle**: [(Synset('elsewhere.r.01'), 1.0), (Synset('between.r.02'), 1.0), (Synset('objectively.r.01'), 1.0), (Synset('thereon.r.01'), 1.0), (Synset('thereto.r.01'), 1.0)]
  - **Screw**: [(Synset('employed.s.02'), 1.0), (Synset('utilized.s.01'), 1.0), (Synset('sought.a.01'), 1.0), (Synset('hand_in_hand.r.01'), 1.0), (Synset('double.r.02'), 1.0)]

Analisi dei risultati e considerazioni

Come si può notare, nessuna delle parole è stata identificata correttamente. Gli unici risultati che si avvicinano alla parola da trovare si trovano nel caso in cui è stato applicato lo stemming alle parole ed è stato ristretto il campo di ricerca ai synset di WordNet aventi il POS "nome", ovvero: Justice $\longrightarrow$ legal_right.n.01 e Radiator $\longrightarrow$ heater.n.01.

Questo primo metodo, sebbene abbia una scarsa performance, dà un'idea della complessità del problema e fornisce qualche indizio utile: restringere il campo di ricerca e pre-processare in qualche modo le definizioni porta dei miglioramenti.

Word Mover's Distance

Visti i deludenti risultati ottenuti con la Lexical Overlap semplice si è deciso di adottare una misura di similarità più fine: la Word Mover's Distance (WMD). La WMD misura la dissimilarità (distanza) tra due documenti di testo come "la distanza minima che le parole *embedded* impiegano per "raggiungere" le parole *embedded* dell'altro documento"[4]. Questa misura utilizza quindi il word embedding e calcola la distanza euclidea tra i vettori ovvero, a partire da un corpus non annotato, memorizza le informazioni semantiche e sintattiche andando a costruire uno spazio vettoriale in cui i vettori delle parole sono più *vicini* se le parole occorrono negli stessi contesti linguistici.
Per questo tentativo è stata utilizzata la libreria python *Gensim*[5] che ha permesso di caricare uno dei pre-trained word embeddings di tipo Word2Vec più diffusi: *GoogleNews-vectors-negative300*. In questo modo è stato possibile caricare in tempi brevi (circa 1 minuto) il modello che verrà utilizzato per calcolare le WMD tra le definizioni in input e le definizioni dei synset di WordNet.
Anche per i seguenti test sono stati effettuati differenti pre-processing sia direttamente sulle definizioni sia sullo spazio di ricerca, ovvero:

- Spazio di ricerca:
  - Su tutti i synset di WordNet
  - Solo sugli iperonimi dei synset delle parole della definizione in input
  - Sugli iperonimi uniti agli iponimi delle parole della definizione in input
- Definizioni:
  - Rimozione delle stopwords
  - Rimozione delle definizioni che contengono parole come "opposite" e "without" (definizioni date spiegando il significato dell'opposto non sono buone definizioni)
  - Accorpare le definizioni in un'unica definizione e ridurla alle sue parole "rilevanti" (parole che compaiono più di 1 volta nell'insieme di definizioni)

Per ogni definizione viene calcolata la Word Mover's Distance con la definizione del synset preso in considerazione e infine viene salvato il synset la cui definizione, in combinazione con una delle definizioni in input, ha minimizzato la distanza. Il risultato migliore è stato ottenuto restringendo lo spazio di ricerca agli iperonimi uniti agli iponimi delle definizioni in input pre-processate rimuovendo le stopwords e le definizioni non buone descritte in precedenza. Di seguito vengono mostrati i risultati ottenuti includendo le definizioni che hanno portato a tale misura di similarità:

- WMD considerando iperonimi e iponimi:

[4]From Word Embeddings To Document Distances: http://proceedings.mlr.press/v37/kusnerb15.pdf
[5]gensim: https://pypi.org/project/gensim/

- synset for word **Justice** is (0.418, Synset('justice.n.01')) because "the quality of being fair and reasonable" is the most similar definition of "Justice" to "the quality of being just or fair"
- synset for word **Politics** is (0.428, Synset('strategics.n.01')) because "the science or art of political government" is the most similar definition of "Politics" to "the science or art of strategy"
- synset for word **Greed** is (0.517, Synset('greed.n.01')) because "intense desire to acquire or possess more than needed" is the most similar definition of "Greed" to "excessive desire to acquire or possess more (especially more material wealth) than one needs or deserves"
- synset for word **Radiator** is (0.681, Synset('fuel.n.01')) because "something that can produce heat" is the most similar definition of "Radiator" to "a substance that can be consumed to produce energy"
- synset for word **Patience** is (0.613, Synset('hush.v.02')) because "ability to be quiet or tolerate problems" is the most similar definition of "Patience" to "cause to be quiet or not talk"
- synset for word **Food** is (0.621, Synset('diamagnet.n.01')) because "substance that animals eat" is the most similar definition of "Food" to "a substance that exhibits diamagnetism"
- synset for word **Vehicle** is (0.607, Synset('wheeled_vehicle.n.01')) because "machine usually with wheels and engine, used for transporting people or goods on land, especially on roads" is the most similar defini tion of "Vehicle" to "a vehicle that moves on wheels and usually has a container for transporting things or people"
- synset for word **Screw** is (0.629, Synset('module.n.04')) because "objects that is used to look and join other components" is the most similar definition of "Screw" to "a self-contained component (unit or item) that is used in combination with other components"

- WMD considerando iperonimi e iponimi usando le relevant words:
  - synset for word **Justice** is (0.962, Synset('tolerate.v.02')) because "'laws', 'people', 'right', 'abstract', 'rights', 'concept', 'idea', 'fairness', 'fair', 'respect'" is the most similar definition of "Justice" to "recognize and respect (rights and beliefs of others)"
  - synset for word **Politics** is (0.85, Synset('politics.n.05')) because "'activities', 'state', 'governance', 'government', 'area', 'science', 'entity', 'population', 'associated', 'country'" is the most similar definition of "Politics" to "the activities and affairs involved in managing a state or a government"
  - synset for word **Greed** is (0.796, Synset('itch.v.04')) because "'money', 'typical', 'strong', 'attachment', 'desire', 'something', 'non', 'excessive'" is the most similar definition of "Greed" to "have a stron g desire or urge to do something"
  - synset for word **Radiator** is (0.882, Synset('heater.n.01')) because "'item', 'object', 'temperature', 'hot', 'room', 'heat', 'water', 'used'" is the most similar definition of "Radiator" to "device that heats water or supplies warmth to a room"
  - synset for word **Patience** is (0.952, Synset('long_run.n.01')) because "'problems', 'time', 'wait', 'quiet', 'ability', 'period', 'tolerate', 'able', 'capacity'" is the most similar

definition of "Patience" to "a period of time sufficient for factors to work themselves out"

- synset for word **Food** is (0.924, Synset('organic_phenomenon.n.01')) because "'people', 'plants', 'needed', 'living', 'edible', 'life', 'substance', 'something', 'animals', 'eat', 'absorb', 'thing'" is the most similar definition of "Food" to "(biology) a natural phenomenon involving living plants and animals"
- synset for word **Vehicle** is (0.874, Synset('wheeled_vehicle.n.01')) because "'point', 'another', 'objects', 'thing', 'object', 'people', 'engine', 'wheels', 'move', 'goods', 'things', 'land', 'especially', 'tra nsporting', 'moving', 'transport', 'used', 'transportation'" is the most similar definition of "Vehicle" to "a vehicle that moves on wheels and usually has a container for transporting things or people"
- synset for word **Screw** is (0.964, Synset('straight_pin.n.01')) because "'two', 'object', 'things', 'pointed', 'fix', 'raised', 'around', 'join', 'thin', 'item', 'objects', 'parts', 'metal', 'thread', 'helical', 'together', 'pin', 'running', 'used'" is the most similar definition of "Screw" to "pin consisting of a short straight stiff piece of wire with a pointed end; used to fasten pieces of cloth or paper together"

In questo caso, ridurre le definizioni in relevant words peggiora il risultato, trovando qualche parola che si avvicina al concetto originale e trovando correttamente solo il synset per "politics". Aumentando a "più di 2" il numero di occorrenze per considerare una parola rilevante, i risultati peggiorano drasticamente poiché le parole selezionate per il confronto con le definizioni dei synset si riducono ad un paio per ogni gruppo di definizioni risultando così troppo poche per trovare una corrispondenza precisa e sbagliando di conseguenza tutte le parole. Probabilmente per un approccio che conta le relevant word sarebbero necessarie molte più definizioni rispetto alle 12 date in input.

Analisi dei risultati

La misura Word Mover's Distance impiega decisamente più tempo nell'esecuzione rispetto alla lexical overlap semplice (dai 2 ai 4 secondi per gruppo di definizioni) mantenendo un tempo tra i 10 e 20 secondi per insieme di definizioni, con un picco di circa 45 secondi per le definizioni relative a "Screw" dovuto alla maggiore lunghezza delle definizioni. I risultati ottenuti, però, sono decisamente più soddisfacenti:

- Tre parole sono state identificate esattamente:
  - **Justice** $\longrightarrow$ Synset('justice.n.01')
  - **Greed** $\longrightarrow$ Synset('greed.n.01'))
  - **Vehicle** $\longrightarrow$ Synset('wheeled_vehicle.n.01'))
- Due parole sono state identificate attraverso sinonimi o parole "vicine":
  - **Politics** $\longrightarrow$ Synset('strategics.n.01'))
  - **Screw** $\longrightarrow$ Synset('module.n.04'))
- Tre parole sono state associate a synset completamente errati:

- **Radiator** $\longrightarrow$ (0.681, Synset('fuel.n.01'))
- **Patience** $\longrightarrow$ (0.613, Synset('hush.v.02')
- **Food** $\longrightarrow$ (0.621, Synset('diamagnet.n.01'))

Considerazioni sull'uso delle definizioni

In generale le definizioni non sono uno strumento affidabile per descrivere un concetto. Esse, infatti, presentano diversi problemi:

- **Ciclicità**: è possibile che alcune definizioni contengano dei termini la cui definizione contiene il termine di partenza creando così un ciclo di definizioni senza darne una chiara.
- **Definizioni opposte**: Alcune definizioni vengono date descrivendo ciò che il concetto "non è". Anche in questo caso risulta difficile capire il significato del concetto, soprattutto per il task content-to-form, poiché, i termini opposti vanno ad "inquinare" il campo di ricerca con synset che sono lontani dalla parola che si sta cercando, ma che potrebbero avere un'alta similarità con le definizioni della parola da trovare. Ad esempio, considerando la parola "Patience", la sua definizione "feeling opposed to anger" risulta molto simile alla definizione del synset *Synset('irascibility.n.01')* ovvero "a feeling of resentful anger", synset che sarà sicuramente preso in considerazione quando si considerano sia gli iperonimi sia gli iponimi delle parole delle definizioni. Per questo motivo si è deciso di escludere le definizioni che contengono parole che indicano un opposto.
- **Generalità**: Spesso risulta complesso decidere il grado di generalità di una definizione, infatti con una definizione troppo generica non si riesce a identificare precisamente il significato del concetto.
- **Specificità**: Come per la generalità anche una definizione troppo specifica non è una buona definizione, in quanto usare ad esempio termini troppo tecnici per descrivere un concetto renderebbe il significato non comprensibile se non attraverso ulteriori ricerche riguardo i singoli termini che compongono la definizione stessa.

Osservazioni generali

Data la fragilità e la scarsa affidabilità dell'uso delle definizioni per descrivere concetti, risalire alla forma partendo dal contenuto risulta un task complesso. Tecniche di similarità come la lexical overlap non bastano per confrontare le definizioni, sono necessarie misure di similarità più fini che tengono in considerazione anche le informazioni semantiche delle definizioni. L'utilizzo dei Word Embeddings ha infatti portato notevoli miglioramenti sull'individuazione dei concetti. Esistono diverse misure di similarità basate sul word embedding e, per questo progetto, si è scelto di mostrare i risultati ottenuti con la Word Mover's Distance in quanto risulta di immediata applicazione e porta a risultati in tempi accettabili (anche se, immaginando un applicativo online completamente funzionante, i tempi risultano estremamente lunghi).
Un ulteriore esperimento è stato effettuato utilizzando la Smooth Inverse Frequency, ma si è preferito non mostrare i risultati a causa dei tempi eccessivamente lunghi per ogni gruppo di definizioni (oltre 25 minuti) e performance identica alla WMD.