Arquitectura

Se procesan T y H

T: t1, t2, t3,.. tn

H: h1, h2,… hn

1. Creamos las entidades buscando modificadores a los sustantivos o verbos

Cada sustantivo o verbo se trata como una entidad a la cual se va a comparar la contención de cada uno de los atributos que contenga. Para este proceso no se realiza ninguna eliminación de tokens ya que se realiza la búsqueda a través del árbol de dependencias.

Se almacena el resultado en un diccionario

2. Se realiza un proceso de eliminación de stopwords, pero se mantienen las negaciones:

El hombre no se resbalo en el césped

----

hombre no resbalo césped

Una de las cosas que hay que tomar en cuenta es el orden de las palabras, debido a que mantenemos el sentido del sujeto y el objeto

A continuación para cada palabra de la frase obtenemos bolsas de sinonimia de cada una.

Hombre no resbalo cesped

persona

BoW Relaciones

Para cada palabra restante en T y H

3. Calculamos la distancia de Jaro Winckler a traves de esas bolsas de palabras de hiperonimias

relaciones\_generales1=["is\_a","part\_of","used\_for", "capable\_of", "at\_location","etymologically\_related\_to","manner\_of","has\_a","derived\_from","has\_property","form\_of","causes","has\_prerequisite","has\_subevent","has\_first\_subevent"]

4. Regresamos al proceso de contención de las entidades generadas en el paso 1, pero con respecto a los a las BoW de palabras para identificar la contención de cada uno de los atributos de la entidades

Cada vez que identificamos que una entidad se encuentra contenida en otra, EH en ET.

Aqui obtenemos nuestras primeras características que nos da una primer probabilidad de etiqueta

-relation: -1 si es contra, 2 si es neutral y 1 si es posible Entailment

-overlap\_ent: relación entre el total de tokens de H

-no\_matcheadas: relación entre el total de tokens de H

-contradiction: relación entre el total de tokens de H

5. Si hay alguna forma de negar los conceptos

-negH

-negT

-jaccard: este coeficiente se calcula a nivel token

6. Se realiza otro procesamiento para quedarnos solo con tipos de palabra que a continuación se detallan, para construir la matriz de alineamiento con sus words embeddings normalizados y no normalizados.

if token.pos\_ in ["NUM","PROPN","NOUN","VERB","ADJ","ADV"] or token.dep\_ =="neg":

if token.lemma\_ !="be":

words.append(token.lemma₎

7. Construimos las matrices con los tokens anteriores de alineamiento, de información mutua y distancia wassereint, a traves de sus words embeddings, por lo que las matrices contienen información cruzada de cada uno. Estas son nuestras matrices iniciales.

- max\_info\_t = ma.max().sum()/ma.shape[1]

- sumas\_t = ma.sum().sum()/todo

- mearts\_t = m\_eart.min().sum()/los tokens de h

- mutinf\_t = m\_mi.max().sum()/tokens de H

- entropia\_t = cada uno de los valores redondeados a un decimal d toda la matriz y ahi aplicamos la entropia

- listT tokens de Texto

- listH tokens de Hipótesis en la matriz

8. uno de los primeros procesos a la matriz es validar si existen coincidencias en el texto y la hipotesis a traves de sus tokens. Hay que dar cuenta de que aquí no importa el orden es a nivel léxico. POR LO QUE HAY QUE RETOMAR ESTO.

Si los tokens coinciden, se eliminan los tokens correspondientes en H

9. Se reconfigura el calculo de entropia sobre los grupos encontrados de las relaciones que existen entre T y H

Grupo 1 de generalidad y similitud → 1

Grupo 2 de contradiction → 0

Grupo 3 de neutralidad → 2

Grupo 4 no relación → 3

Solo se eliminan los tokens de H que se encuentran en el grupo 1 y 3.

También se contabilizan

rel [1 1 1 1 1 1 0 3 3]

A esto se le calcula su probabilidad y su entropia

10. calculamos lo restante

- entropia de la matriz restante todos los valores como arriba

- KL\_divergence : diferencia entre las entropias total y restante

- max\_inf

-sumas: matriz restantes

- mearts: matriz restante

- mutinf: matriz restantes

-diferencias: relacion de tokens restantes entre el total de T

- distancias: distancias de los vectores de palabras totales

- semantics: calculo de centroide de las palabras restantes con respecto a las mas comunes en T

- nlp\_semantics: similitud coseno

- list\_com

-list\_incomp

-rel\_conceptuales

- list\_m

las ultimas son meramente conteos