UNIVERSITÀ DEGLI STUDI
DI NAPOLI FEDERICO II

spaCy

Contest II Intelligenza artificiale

Tweet analysis with SpaCy

Pasquale Angelino N46003194
Giuseppe Capasso N46003195
Daniele Cerasuolo N46003481
Lamberto Fulgione N46002003

05/2019

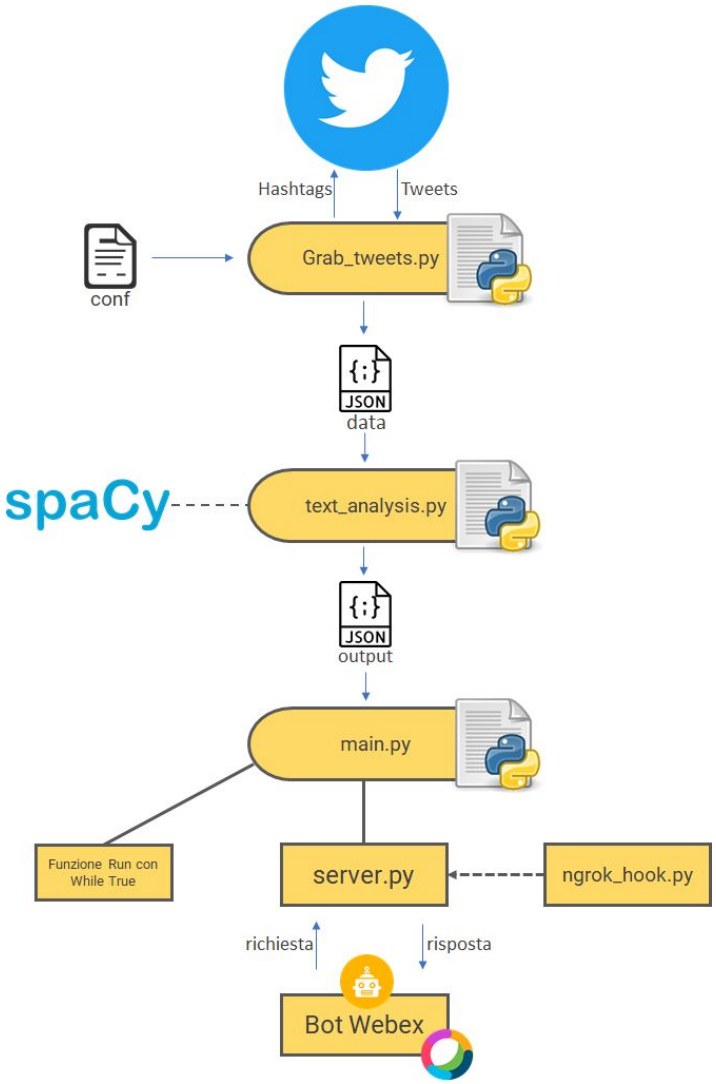
Sommario

Introduzione

Il presente documento tratta il secondo contest di Intelligenza Artificiale basato sulla rappresentazione dell'informazione, si è scelto di svolgere la seconda tipologia di elaborato al fine di estrapolare informazioni attraverso l'analisi di testi prelevati da social network.

Descrizione

Come specificato precedentemente si è scelto di sviluppare la seconda tipologia di elaborato che prevede l'utilizzo di un framework, SpaCy, per effettuare Natural Language Processing (NLP).



Sistema

A tal fine è stato sviluppato un sistema software che fornisce informazioni e notizie di calciomercato dato in input il nome di un personaggio calcistico di interesse. E' possibile effettuare interrogazioni sia a riga di comando che attraverso un bot presente sulla piattaforma **Cisco Webex Teams** che fornisce servizi di online meeting, web conferencing and videoconferencing.

L'architettura del sistema software si compone dei seguenti moduli:

- **grub_tweets.py:** Questo modulo si occupa della ricerca e prelievo dei tweet presenti sul social network attraverso l'interrogazione per hashtag, eseguita attraverso le API messe a disposizione da Twitter. Gli hashtag di interesse vengono inseriti in righe successive all'interno di un file di configurazione che viene fornito in input al modulo, ognuno di essi è seguito dal numero di tweet da prelevare per quel

determinato hashtag. Il risultato dell'esecuzione dello script è un file con estensione "json": un formato adatto allo scambio di dati fra applicazioni soprattutto tramite la rete. Il file ".json" contiene informazioni sui tweet divisi per campi quali:

- Screen_name autore
- Extended text
- Entità quali gli hashtag utilizzati
- Data di pubblicazione

In particolare lo script accetta come primo parametro il file di configurazione che deve trovarsi nella cartella **settings.** Se non specificato il valore di default è 'conf'. Il secondo parametro è il nome del file con estensione json che sarà creato nella cartella **results,** che sarà creata dallo stesso script se non esistente. Se non specificato il valore di default è 'data.json'

- **text_analysis.py:** Il file di output dello script precedente è utilizzato come input per il presente modulo, il quale sfrutta il framework SpaCy per effettuare analisi testuale sui tweet. Lo script inizializza un matcher addestrato con due concetti definiti attraverso una lista di lemmi. I concetti sono "indiscrezione", intesa come incertezza della permanenza del giocatore nella squadra di appartenenza, o la "partenza", inteso come cambio squadra. Lo script apre un file di ingresso (uscita di 'grub_tweets.py**')** e per ogni tweet cerca di effettuare il matching con uno dei due concetti di indiscrezione o partenza. Produce un file di output ".json" contenente le seguenti informazioni:
  - concetto riconosciuto dal matcher
  - persone riconosciute da spaCy
  - organizzazioni riconosciute da spaCy
  - testo processato da spaCy

In particolare lo script accetta come primo parametro il file di output prodotto dallo script precedente**.** Se non specificato il valore di default 'è data.json'. Il secondo parametro è il nome del file in cui sarà scritto l'output e sarà salvato nella cartella **results.** Se non specificato il valore di default è results.json'

- **main.py**: modulo contenente due funzioni, una per la ricerca di tweet dato un nome in input “get_tweet_by_content(content)” e una funzione main contenente un while True che permette di eseguire il software a riga di comando. La funzione ritorna la percentuale di ‘acquisto’ e ‘indiscrezione’ rilevate dal matcher e il numero di tweet per cui è stata misurata la percentuale

- **server.py:** un modulo basato sulla libreria http.server di Python che consente di implementare un piccolo web server. In particolare, all’avvio il server si sincronizza con un eventuale processo di ngrok(se non presente il server si avvia in locale) da cui prende l’url pubblico e crea un webhook (si mette in ascolto per l’arrivo di un messaggio) attraverso le API di Webex Teams. In sostanza, il server reagisce a dei messaggi HTTP di tipo POST e in base al contenuto di quest’ultimo utilizza il modulo main.py per fornire l’informazione richiesta. Il server utilizza la porta 8080.

- **ngrok_hook.py:** questo modulo fornisce delle funzioni di utilità che, attraverso webex teams sdk, creano webhook cancellando tutti quelli precedentemente creati. In particolare viene usato come libreria dal server come supporto alla sincronizzazione con ngrok. Il modulo non è stato sviluppato da noi, ma può essere scaricato qui.

Esecuzione

L ’esecuzione del sistema software si compone delle seguenti fasi (o passi):

1. Lanciare lo script **grub_tweets.py** assicurandosi che esista un file conf nella cartella settings. Richiede una connessione ad Internet.
2. Lanciare lo script **text_analysis.py** per analizzare i tweet con spacy. Non richiede una connessione ad internet.
3. Per eseguire il programma in locale (a riga di comando) basta lanciare lo script **main.py.**
4. Se si vuole eseguire il programma attraverso il bot su Webex Teams (richiede una connessione ad Internet):

a. Iniziare una sessione di **ngrok**
b. lanciare lo script **server.py**
c. avviare uno spazio su webex con il bot (cercare: **Calciomercato@webex.bot**)

Strumenti

Per eseguire il codice sorgente si utilizza la versione 3.6.7 di Python con prompt dei comandi in Windows 10. Python è un linguaggio di programmazione ad alto livello, orientato agli oggetti, adatto, tra gli altri usi, a sviluppare applicazioni distribuite, scripting, computazione numerica e system testing.

SpaCy è una libreria open source per l'elaborazione del linguaggio naturale, scritta in Python e Cython. La libreria è rilasciata sotto licenza MIT ed attualmente implementa modelli statistici di reti neurali in inglese, tedesco, spagnolo, portoghese, francese, italiano, olandese e greco; inoltre offre funzionalità di NER e di tokenizzazione per diverse altre lingue. Per questo progetto è stato utilizzato il dizionario multilingua.

Ngrok è un tool che permette di avere un Url pubblico a partire dall'indirizzo di loopback che consente la connessione tra bot e software. Fornisce un'interfaccia utente Web accessibile solo dall'host stesso (http://127.0.0.1:4040) e delle API attraverso i quali è possibile analizzare e monitorare il traffico HTTP che transita sull'URL creato. Attraverso il free plan è possibile ottenere un URL pubblico (non fisso, ma cambia ad ogni esecuzione) e un tunnel TLS per 40 connessioni al minuto.

Conclusioni

I risultati ottenuti risultano discretamente accurati, sia dal punto di vista dell'elaborazione testuale sia per quanto riguarda le percentuali ottenute relative ai rumors. E' stato osservato che molti nomi possono non essere interpretati correttamente da SpaCy data la diversa nazionalità dei giocatori e l'ambiguità tipica del linguaggio umano. Questo problema potrebbe essere risolto usando delle API per ottenere una lista completa di giocatori, allenatori e società calcistiche e insegnando al modello in uso di spacy i nuovi termini per arricchire il dizionario.

Il sistema può essere facilmente modificato per analisi testuali riguardanti ambiti diversi dal calciomercato, modificando il file di configurazione degli hashtag e addestrando il matcher per concetti utili al contesto in esame.