GENERAZIONE DELLE CATEGORIE DI WIKIPEDIA ATTRAVERSO IL CLUSTERING

CAZZARO DALLA CIA LOVISOTTO VIANELLO

INTRODUZIONE

- descrizione del problema in generale e del constesto
  - delineazione degli obiettivi:
  - gli articoli di wikipedia sono clusterizzabili?
  - rapporto tra cluster e le categorie?
  - cosa succede con varie tecniche di clustering?

DATASET E ANALISI PRELIMINARE

Il dataset che abbiamo utilizzato per effettuare questa analisi non è altro che un dump di wikipedia. Tale Dump è composto da un JSON contenente centomila articoli di wikipedia in versione inglese. Per ogni articolo di wikipedia abbiamo a disposizione il titolo, il testo, l'id e le categorie dell'articolo.

La nostra analisi mira a valutare gli articoli e le loro categorie quindi prima di effettuare tale analisi si è deciso di eliminare tutti gli articoli i quali non risultano essere associati a nessuna categorie. Tale pagine sono dovute al fatto che in wikipedia sono presenti della pagine di disambiguazione e quindi non sono utili per i nostri scopi.

//todo analisi delle categorie (sort e distribuzione delle categorie)

RAPPRESENTAZIONE DEL DATASET

Prima di procedere con qualsiasi operazione sull'intero corpus si è deciso di preprocessare il data set eliminando le stop words presenti nel testo e lemmatizzando del corpus rimanente. Per l'eliminazione delle stop words ci siamo basati su una lista di parole fornita dal sito http://www.ranks.nl/stopwords.

VETTORIALIZZIONE DEGLI ARTICOLI

Per interagire con i più comuni algoritmi di clustering, ad esempio K-means, si è reso necessario trasformare gli articoli di Wikipedia in vettori. Per fare ciò abbiamo adottato una tecnica nota in letteratura con il nome Word2Vec. Tale algoritmo ideato da Tomas Mikolov non è altro che una rete neurale a due strati il cui scopo è quello di trasformare parole del linguaggio

naturale in vettori. Nello spazio vettoriale generato le parole semanticamente più simili saranno più vicine, viceversa parole semanticamente diverse risulteranno distanti.

Tale funzionalità è già implementata nella suite software di Spark, per sfruttarla è necessario inizializzare alcuni parametri di settaggio. Tra questi uno dei più importanti è sicuramente la dimensione del vettore in uscita. In letteratura si è valutato che una dimensionalità nel ordine dei 100/300 [1] è sufficiente a rappresenta un buon compromesso in termini di performance. Uno volta settati i parametri l'algoritmo di Word2Vec necessita di essere allenato. Tale allenamento è stato fatto su tutto il corpus.

Per trasformare un articolo è stato sufficiente effettuare la media vettoriale di tutte le parole presenti nel testo di un'articolo. Tale operazione è stata eseguita attraverso il BLAS (Basic Linear Algebra Subprograms) di Spark per eseguire tali conti nella maniera più efficiente possibile.

BAG OF WORDS

//todo: la trasformazione in bag of words

CLUSTERING

Tecniche di Clustering

Hopkins Statistic

Riportiamo lo score che ci dice che il nostro dataset è ben clusterizzabile

Kmeans

Kmeans e il suo score con un bel grafico

Altri metodi

Abbiamo provato anche il clustering gerarchico e il Gaussian Mixture Model ma non abbiamo abbastanza potenza di calcolo

Latent Dirichlet Allocation

Vediamo cosa viene fuori e un bel grafico

Valutazioni del Clustering

Simple Silhouette

Utilizzo della versione semplificata di Silhouette con i centroidi Rimozione dei cluster con un solo articolo dal punteggio Magari buttiamoci un peso a sta metrica

Normalized Mutual Information

Il problema di individuare una funzione obiettivo che utilizzi le categorie, le quali non formano una partizione in quanto overlapping Pulizia delle categorie con con idf Modifica dell'algoritmo

RISULTATI

Numero di cluster

Il K selezionato dalle due tecniche Kmeans e LDA Speriamo sia simile!

Validazione con Simple Silhouette

L'andamento sempre crescente della Silhouette Un bel grafico lineare

Confronto tra i Cluster ottenuti con Normalize Mutual Information

Confronto tramite NMI delle due tecniche Kmeans e LDA

CONCLUSIONI

Le conclusioni generali dalle analisi effettuate
Proposte di punti da approfondire in studi futuri

REFERENCES

[1] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. ICLR Workshop, 2013.