Relazione esercizio Topic Modelling

L'esercitazione 5 ha visto la definizione di un algoritmo di Topic Modelling, un altro task del NLP che abbiamo inquadrato nel contesto della semantica documentale. Durante il corso abbiamo affrontato 2 approcci distinti al topic Modelling:

- **LSA** (Latent Semantic Analisys)
- **LDA** (Latent Diritchlet Allocation)

Nel caso di studio è stato considerato un approccio LDA, reso più semplice grazie all'uso della libreria Gensim che, appunto, ne fornisce una implementazione già pronta. Il Topic Modelling affronta il problema di recuperare, a partire da una base documentale, il topic, ovvero una lista di parole riassuntive di ciò che si sta parlando. LDA considera ogni documento come una collezione di topic sotto una certa proporzione, e ogni topic è un insieme di parole chiave che, sotto una distribuzione di probabilità rimandano ad un tema. LDA affronta il problema con un approccio probabilistico. Consiste in un modello generativo che riceve in input il numero di topic da ricercare. Consideriamo alcuni esempi presi da https://blog.echen.me/2011/08/22/introduction-to-latent-dirichlet-allocation/ :

- I like to eat broccoli and bananas.
- I ate a banana and spinach smoothie for breakfast.
- Chinchillas and kittens are cute.
- My sister adopted a kitten yesterday.
- Look at this cute hamster munching on a piece of broccoli.

Dando in input a LDA due topic, l'algoritmo potrebbe restituire che le prime 2 frasi parlano al 100% separatamente dei topic, mentre le altre hanno delle probabilità frammentate. L'idea di base di LDA è che i documenti si possono vedere come un mix di topic e ci sono alcune parole riassuntive di tali argomenti che vengono restituite in output a formare il topic. Siccome l'approccio implementato da LDA è probabilistico, inizialmente, dato in input k topic e una collezione di documenti, LDA assegna in modo casuale ogni parola nei testi ad uno qualsiasi dei topic, in questo modo si ha da subito la distribuzione di probabilità dei topic e quella delle parole nei topic. Ciò vuol dire che ogni topic k ha una sua distribuzione sull'intero vocabolario di termini, quindi se so che in un certo documento si parla di scienza, la probabilità della parola "argon" sarà più alta della probabilità di "cane". In seguito queste probabilità vengono ricalcolate, quindi si aggiorna l'assegnamento con una nuova distribuzione di probabilità. Dopo aver ripetuto questi passi per un certo numero di volte si arriva ad una situazione in cui le probabilità cambiano di poco e tali risultato le probabilità finali di ogni topic.

I passi fondamentali per il topic modelling sono:

1. **Preprocessing dei dati**
2. **Creazione del dizionario e del corpus necessari per il topic modelling**
3. **Costruzione dei topic**

Il dataset considerato è un file testuale presente nei file di progetto. Consiste in un documento in cui sono convogliati 4 gruppi di paragrafi di diversi testi:

- Pagina di wikipedia dell'università di torino
- Pagina di wikipedia del linguaggio python
- Paragrafi e articoli della guerra in Ucraina
- Pagina di wikipedia che parla della frutta

Nelle cartelle del progetto è presente il file "**DataPreprocessing.py**" che include alcune funzioni per processare i dati rimuovendo i simboli di punteggiatura, stopwords, spazi bianchi multipli e infine per costruire il bag of words di ogni frase del dataset.

Per la generazione dei topic, la libreria Gensim riceve 2 parametri:

- Un dizionario
- Il corpus

Per analizzare il testo, la libreria gensim richiede che i tokens preprocessati siano convertiti associando loro degli identificativi univoci, quindi per far ciò si avvale dei dizionari di python. Ogni token è mappato con un id unico. La funzione "**corpora.Dictionary(tokens)"** si occupa di far questo. Questo dizionario verrà usato come base di partenza per la costruzione del corpus che sostanzialmente è un mapping tra id univoci del dizionario e la frequenza della parola all'interno del documento originale, nel caso di un singolo documento il mapping è fra id unico e posizione paragrafo nel testo.

Per quanto riguarda la costruzione dei topic, la libreria gensim offre la funzione **"gensim.models.ldamodel.LdaModel()"** che riceve in ingresso il corpus, il dizionario e il numero di topic che deve trovare. Studiando la libreria gensim e la letteratura a riguardo su internet, un modo semplice per visualizzare i topic costruiti è usare la libreria "pyLDAvis". Quest'ultima offre una serie di primitive che consentono di rappresentare graficamente il risultato della computazione dell'algoritmo LDA. In particolare è stata invocata la funzione "**pyLDAvis.gensim_models.prepare**" i risultati sono stati salvati in un file .html.

Risultati

Di seguito viene riportato il grafico costruito a termine del processo di topic modelling. I topic individuati sono:

- Topic 1 = frutta
- Topic 2 = Linguaggio di programmazione
- Topic 3 = Guerra in Ucraina
- Topic 4 = Università