Università degli studi di
Milano-Bicocca

Text Mining and Search

Extractive summarization con LSA e TextRank

Autori:

Matteo Campironi - 801850 - m.campironi@campus.unimib.it
Serena Di Maggio - 821063 - s.dimaggio2@campus.unimib.it

Sommario

A partire dalla seconda metà del secolo scorso, alcuni ricercatori hanno posto la loro attenzione sullo sviluppo di tecniche di Text Summarization. Si tratta del processo automatizzato attraverso cui, dati dei documenti testuali, si cerca di estrarne i contenuti e le informazioni principali, ottenendone delle sintesi finali. È stato proprio questo l'obiettivo di tale progetto, in cui sono stati generati dei riassunti di tipo estrattivo in modo automatico, partendo da un dataset di articoli di giornale estratti dai siti web della CNN e DailyMail. A questo scopo sono state utilizzate due diverse metodologie: LSA, che fa parte dei Topic Representation methods e TextRank, che invece è un Indicator Representation method. In particolare, è stato utilizzato un adattamento della Maximal Marginal Relevance (MMR) per il ranking delle frasi in base alla loro importanza e sono stati comparati i risultati ottenuti dai due metodi.

Indice

1 Introduzione

La realizzazione di riassunti a partire da lunghi testi è una pratica molto diffusa e utilizzata in svariati ambiti, basti pensare ad uno studente che legge e analizza un libro o ad un ricercatore che estrapola le informazioni principali da differenti documenti, o ancora ai giornalisti che cercano di sintetizzare notizie fondamentali in brevi ed incisivi articoli. Inoltre, la notevole quantità di informazione presente attualmente sul web per qualsiasi argomento rende necessaria la capacità di saperne sintetizzare i punti chiave. Questi sono solo alcuni degli esempi dell'importanza del ruolo rivestito dai riassunti e che hanno portato allo sviluppo di tecniche di sintesi automatiche dei testi, sfruttando il Text Mining e il Machine Learning per svolgere uno dei più difficili task di Natural Language Processing. In particolare, esistono due approcci alternativi di text summarization automatica:

- *Extractive*, che utilizza frasi contenute nel testo originale per creare un riassunto;
- *Abstractive*, che rielabora i contenuti principali presenti nel testo, esprimendoli con parole diverse dalle originali.

Nonostante il secondo metodo sia quello che più si avvicina al modo di agire umano e quindi il più interessante, risulta anche di notevole difficoltà. In questo progetto, dunque, si è deciso di svolgere Extractive Text Summarization, sfruttando due diversi metodi:

- uno degli Indicator Representation methods, TextRank, che rappresenta ogni documento come una rete di nodi (frasi) interconnesse;
- uno dei Topic Representation methods, la Latent Semantic Analysis, che permette di ottenere una rappresentazione implicita della semantica dei testi considerati, basandosi sulla co-occorrenza osservata di parole.

2 Dataset

Il dataset utilizzato [1], contiene diversi articoli di giornale estratti dal sito della CNN e del DailyMail riguardanti notizie degli ultimi anni e scritti in lingua inglese. Ad ogni notizia sono assegnati dei riassunti prodotti manualmente, quindi non di tipo estrattivo. Dal momento che il dataset totale era composto da circa 287.000 articoli, si è deciso di lavorare su un campione casuale composto da 20.000 articoli, di cui 10.000 estratti da CNN e 10.000 da DailyMail. Le motivazioni che hanno spinto all'utilizzo di solo una parte dei dati sono l'elevata richiesta computazionale dovuta al preprocessing dei testi e l'utilizzo di metodi per l'estrazione di riassunti che non avrebbero beneficiato di un numero maggiore di testi.

2.1 Preprocessing

Prima di applicare i metodi per la text summarization è necessario effettuare un lavoro di preprocessing dei dati [2]. Il dataset, contenendo articoli di stampo giornalistico, non richiede particolare attenzione per quanto riguarda eventuali errori grammaticali o linguaggio abbreviato come può accadere su testi estratti da social network, ma nonostante questo necessita di diversi accorgimenti.

Per prima cosa è stato necessario separare gli articoli dai riassunti, in quanto il dataset proviene da uno scraping dei siti web di CNN e DailyMail. Per questo motivo sono presenti delle stringhe contenenti l'autore e la data di pubblicazione che sono state rimosse tramite l'utilizzo di espressioni regolari, come è possibile osservare nella Tabella 1. Successivamente è stata fatta una normalizzazione del testo ed in particolare è stato convertito tutto in lettere minuscole, sono stati rimossi numeri e punteggiatura, sono state sostituite le contrazioni della lingua inglese con la loro forma estesa e sono state eliminate le stop-words.

Testo originale	Articolo estratto
By Shari Miller. PUBLISHED:. 05:45 EST, 22 November 2013. |. UPDATED:. 08:39 EST, 22 November 2013. Some people can literally wait a lifetime before they find true love - and for one 83-year-old Canadian more than seven decades would pass before she married her childhood sweetheart.	Some people can literally wait a lifetime before they find true love - and for one 83-year-old Canadian more than seven decades would pass before she married her childhood sweetheart.

Tabella 1: Testo prima e dopo aver rimosso delle stringhe dovute allo scraping delle pagine web di CNN e DailyMail.

Infine sono state effettuate la tokenizzazione e la lemmatizzazione del testo. Per quanto riguarda quest'ultima è stato utilizzato il lemmatizer che viene fornito dalla libreria nltk. In particolare è stato effettuato il POS Tagging per identificare le categorie lessicali e passarle come parametro al lemmatizer, in modo che i risultati fossero più accurati.

Articolo estratto	Articolo preprocessato
Some people can literally wait a lifetime before they find true love - and for one 83-year-old Canadian more than seven decades would pass before she married her childhood sweetheart.	people literally wait lifetime find true love one yearold canadian seven decade would pass marry childhood sweetheart.

Tabella 2: Articolo prima e dopo aver applicato le tecniche di preprocessing.

3 Approccio Metodologico

3.1 Rappresentazione vettoriale

Entrambi i metodi utilizzati per l'estrazione di riassunti presuppongono che gli articoli vengano suddivisi in frasi e che queste siano rappresentate sotto forma di vettori. Si è quindi deciso di utilizzare il TfidfVectorizer di sklearn per ottenere la rappresentazione *TF-IDF*, che permette di assegnare un peso alle parole all'interno del documento considerato, basandosi su due elementi:

- La frequenza $tf_{t,d}$ del termine t all'interno del documento d;
- Il numero df_t di documenti che contengono il termine t.

Si definisce quindi il *TF-IDF* come:

$$\textit{TF-IDF} = (tf_{t,d}) \times \log_{10}\left(\frac{N}{df_t}\right)$$

con N = numero totale di documenti [3]. Il suo valore aumenta tanto più il termine è presente nel documento, ma ha una bassa frequenza negli altri e, viceversa, diminuisce quando il termine è abbastanza comune nella collezione generale dei vari documenti.
Una volta ottenute le rappresentazioni vettoriali delle frasi degli articoli, le matrici dei pesi *TF-IDF* risultanti sono state processate in modo diverso a seconda dell'approccio usato.

3.2 Latent Semantic Analysis

La Latent Semantic Analysis (LSA) [4] è una tecnica non supervisionata che riesce a catturare le relazioni tra vari termini e frasi, consentendo di esprimere implicitamente il contenuto semantico degli articoli. Si prende in considerazione una matrice A di dimensione $n \times m$, le cui righe corrispondono a n parole e le colonne a m frasi di un dato documento e ogni entrata a_{ij} rappresenta il peso della parola i nella frase j. Specificatamente, viene utilizzata la matrice dei pesi *TF-IDF*, alla quale viene poi applicata la funzione TruncatedSVD della libreria sklearn, consentendone una riduzione della dimensionalità. Questa funzione si basa sul metodo algebrico della decomposizione ai valori singolari, o SVD (Singular Value Decomposition), che

permette di fattorizzare una qualsiasi matrice $n \times m$ reale o complessa A nella forma:

$$A = U\Sigma V^T,$$

con U e V^T matrici di dimensione rispettivamente $n \times m$ e $m \times m$, Σ matrice diagonale $m \times m$, i cui elementi sulla diagonale sono detti valori singolari non negativi e sono disposti in ordine decrescente. Nel caso specifico di analisi testuale, ogni colonna di U rappresenta un topic e la matrice riporta i pesi delle singole parole per ogni topic; i valori singolari della matrice diagonale Σ invece rappresentano i diversi pesi dei topic; infine, la matrice V^T esprime in ogni riga il peso di una frase per ogni specifico topic.

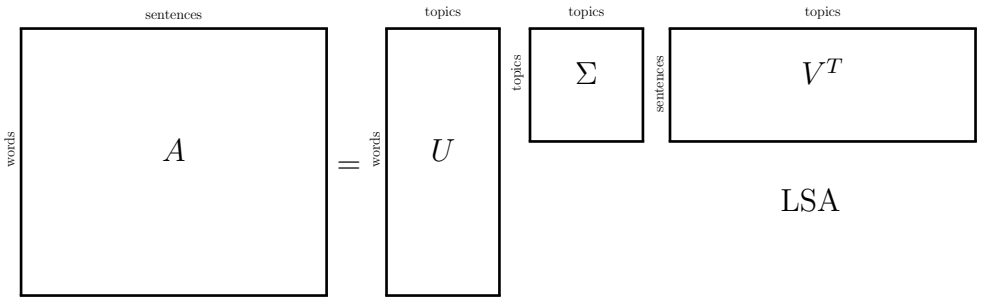

Figura 1: Decomposizione ai valori singolari (SVD).

La funzione TruncatedSVD sfrutta questo concetto matematico applicandone una versione differente, detta appunto troncata, la quale considera solo le k colonne di U e le k righe di V^T corrispondenti ai k più grandi valori singolari di Σ. Questo parametro è stato fissato a 10 e rappresenta il numero di componenti principali, ossia il numero di quelle che saranno considerate le frasi rilevanti di un testo. Il risultato finale della LSA è dunque costituito da frasi che catturano gli aspetti salienti degli articoli considerati e che, inoltre, non risultano ridondanti, proprietà che deriva dall'indipendenza lineare dei vettori singolari della matrice Σ.

3.3 Graph-based: TextRank

TextRank è un modello graph-based che può essere utilizzato per identificare le frasi più importanti nel testo. Il concetto di base di TextRank è quello di assegnare un punteggio a ciascuna frase in base alla loro importanza e quindi riordinarle di conseguenza. L'algoritmo è composto da diversi passi: il primo è quello di dividere il testo in frasi e a questo scopo è stata utilizzata la funzione sent_tokenize della libreria nltk. Successivamente è necessario scegliere una rappresentazione vettoriale per quest'ultime e come spiegato precedentemente la scelta è ricaduta sui pesi *TF-IDF*. Utilizzando la *cosine similarity* è possibile costruire una matrice di similarità, che viene sfruttata come base di partenza per la costruzione di un grafo i cui nodi rappresentano le frasi e gli archi la loro similitudine. [5]

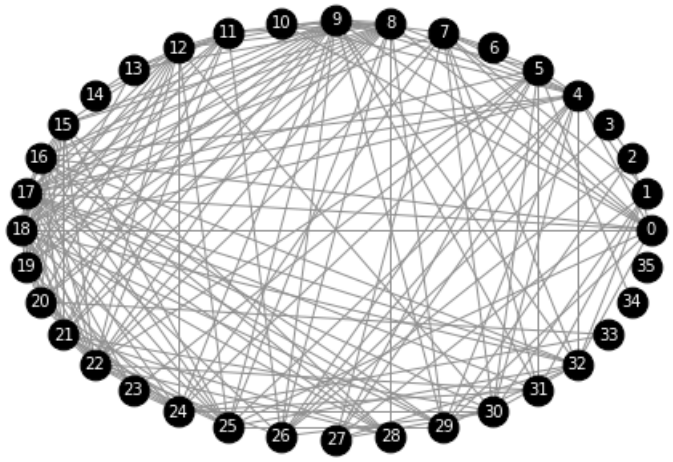

Figura 2: Grafo di un testo composto da 36 frasi.

Utilizzando l'algoritmo PageRank viene assegnato un punteggio ad ogni frase in base alla sua connessione con le altre.

3.4 Selezione delle frasi più importanti

La soluzione più comune per la scelta delle frasi da inserire nel riassunto è quella di tenere le prime n, ordinate per importanza. In questo modo però non si tiene conto di eventuali ripetizioni di concetti all'interno delle varie frasi. Per questo motivo si è scelto di implementare una versione adattata

a riassunti generici della Maximal Marginal Relevance. [6] Questo approccio è stato originariamente proposto per i riassunti query-based, ma è possibile adattarlo andando a sostituire la query con l'intero input D. [7]

$$\text{MMR}(s_i) = \lambda \, \text{sim}(s_i, D) - (1 - \lambda) \max_{s_j \in D}(\text{sim}(s_i, s_j))$$

Infine sono state selezionate le 4 frasi con MMR maggiore fissando $\lambda = 0.5$, in quanto si è osservato che in media ogni riassunto di riferimento ne conteneva circa 4.

4 Risultati e valutazione

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) è uno strumento per il calcolo di una serie di metriche di valutazione, basate sulla presenza di elementi comuni nel riassunto e nei modelli, quali n-gram, sequenze o coppie di termini. Per questo dataset in particolare i riassunti di riferimento non sono stati ottenuti in maniera estrattiva, ma sono stati scritti dagli autori degli articoli. Per questo motivo si è scelto di utilizzare il ROUGE-1 e il ROUGE-2, che vanno a considerare rispettivamente il numero di parole singole e di bigrams che sono presenti sia nel riassunto di riferimento che in quello prodotto. [8] I risultati ottenuti sono i seguenti:

	ROUGE-1	ROUGE-2
LSA	0.25	0.08
TextRank	0.26	0.09

Tabella 3: Risultati ottenuti con i metodi LSA e TextRank.

Entrambi i metodi producono risultati molto simili. I valori medi non sono molto elevati, ma questo dipende anche dalla natura stessa dei riassunti di riferimento, come è possibile osservare in Tabella 4. ROUGE è una metrica con diversi svantaggi, tra cui quello di non tener conto di eventuali sinonimi in quanto misura corrispondenze sintattiche piuttosto che semantiche e questo può portare a valori bassi.

Riassunto di riferimento	Riassunto prodotto
Prince was pictured holding the bear as he landed at Kensington Palace. Promised girl he would pass on bear to the Duke and Duchess of Cambridge. Was given the present during visit to brain injury charity in Nottingham	It comes as Prince Harry's romance with girlfriend Cressida Bonas is reportedly 'on the rocks' because she is not ready to marry. 'On the rocks': Prince Harry's romance with girlfriend Cressida Bonas is reportedly in trouble. A friend told the paper that Harry wants to get married and settle down with Cressida. True to his word: Prince Harry pictured holding the blue bear he promised to give to his unborn niece or nephew.

Tabella 4: Nonostante in questo caso il ROUGE-1 sia pari a 0.23, il riassunto prodotto contiene i punti salienti dell'articolo, mentre quello di riferimento risulta meno leggibile poiché strettamente legato alle informazioni fornite nel titolo.

5 Conclusioni

In questo lavoro sono stati messi a confronto due metodologie differenti per la text summarization, LSA e TextRank. Nonostante a livello di risultati non ci siano state grosse differenze, il primo metodo ha richiesto uno sforzo computazionale maggiore, dovuto alla complessità dell'algoritmo di SVD. Sebbene i valori di ROUGE-1 e ROUGE-2 non siano particolarmente elevati, i riassunti risultano essere comunque apprezzabili, sintomo di come questa misura non sia in grado di cogliere tutte le sfaccettature di un testo.

Riferimenti bibliografici

[1] Deepmind q&a dataset. [Online]. Available: https://cs.nyu.edu/~kcho/DMQA/

[2] Text processing in python. [Online]. Available: https://towardsdatascience.com/text-processing-in-python-29e86ea4114c

[3] H. Christian, M. Agus, and D. Suhartono, "Single document automatic text summarization using term frequency-inverse document frequency (tf-idf)," *ComTech: Computer, Mathematics and Engineering Applications*, vol. 7, p. 285, 12 2016.

[4] Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," in *SIGIR '01*, 2001.

[5] An introduction to text summarization using the textrank algorithm. [Online]. Available: https://www.analyticsvidhya.com/blog/2018/11/introduction-text-summarization-textrank-python/

[6] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '98. New York, NY, USA: Association for Computing Machinery, 1998, p. 335–336. [Online]. Available: https://doi.org/10.1145/290941.291025

[7] A. Nenkova and K. McKeown, *A Survey of Text Summarization Techniques*. Boston, MA: Springer US, 2012, pp. 43–76. [Online]. Available: https://doi.org/10.1007/978-1-4614-3223-4_3

[8] The ultimate performance metric in nlp. [Online]. Available: https://towardsdatascience.com/the-ultimate-performance-metric-in-nlp-111df6c64460