Progetto Elaborazione Linguaggio Naturale: Tecniche di Clustering

Giuseppe De Palma

Alma Mater Studiorum - Università di Bologna
giuseppe.depalma@studio.unibo.it
Matricola: 854846

Sommario Ciaone

1 Introduzione

Il *clustering* (o analisi dei gruppi) è una forma di *machine learning* non supervisionato che permette di raggruppare in *cluster* elementi non annotati dati in input. Un cluster è una collezione di oggetti "simili" tra loro che sono "dissimili" rispetto agli oggetti degli altri cluster. Questo tipo di machine learning è ottimo per partizionare un insieme di dati in diverse "categorie", quindi poter eseguire diverse analisi ed ottenere nuove informazioni. Applicazioni tipiche in cui il clustering viene molto usato è il riconoscimento di email di spam (le email a scopi pubblicitari o di frode), oppure per l'aggregazione di notizie (Google News ne è un esempio).

Il clustering trova possibili applicazioni anche nel campo dell'elaborazione del linguaggio naturale. Oltre alle nuove possibili analisi sui corpora ed al fornire una visualizzazione pittografica delle parole raggrupate, un interessante utilizzo è quello della **generalizzazione** delle parole.

Possiamo considerare i vari cluster delle classi di equivalenza. Per questo motivo, se avessimo un dataset su cui comporre i cluster fatto di frasi e parole, allora si potrebbe assumere che una qualche parola che compare in una frase può essere sostituita con un'altra dello stesso cluster lasciando intatta la correttezza della frase. Ad esempio, se avessimo nel nostro dataset "per Lunedì", "per Martedì", "per Mercoledì", "per Sabato", "per Domenica", senza avere "per Giovedì" e "per Vernedì", e avessimo un cluster in cui i giorni della settimana sono raggruppati insieme, allora potremmo generalizzare l'utilizzo della preposizione "per" con Giovedì e Vernedì.

Il clustering, quindi, può essere molto utile anche nell'elaborazione del linguaggio naturale. Nel progetto in studio vengono testate le capacità di alcune tecniche di clustering da cui si derivano dei risultati per mostrarne le differenze, i pregi e i difetti. I dati utilizzati negli esperimenti, comunque, non sono parti di testo, ma semplici dataset di vettori numerici 2D in modo tale da poter facilmente visualizzare i grafici relativi ai cluster e determinare le caratteristiche di ogni tecnica.

1.1 Outline

[**SCRIVERE OUTLINE**]

2 Clustering

Ci sono numerosi algoritmi per effettuare clustering, ma essi sono classificabili in poche tipologie: il clustering gerarchico e il clustering partizionale. Clustering partizionale consiste nell'ottenere dei cluster, di solito in modo iterativo, ma spesso senza determinare una vera relazione tra gli elementi. Si inizia con un insieme di cluster iniziale ed iterativamente si riassegnano gli oggetti nei giusti cluster. Il clustering gerarchico, invece, forma un albero (la gerarchia) degli elementi dove un nodo rappresenta un sotto-cluster del nodo padre e le foglie sono i singoli oggetti.

Un'altra importante distinzione tra gli algoritmi di clustering è il *soft clustering* e *hard clustering*. Nel primo caso, ogni oggetto può essere assegnato a più cluster secondo un qualche grado di appartenza, mentre nel secondo caso ogni oggetto è assegnato ad un unico cluster. In questo progetto vedremo quattro diversi algoritmi, due della classe di clustering gerarchico, due del clustering partizionale. I primi tre eseguono hard clustering mentre l'ultimo soft clustering.

Di seguito sono elencati i metodi implementati e testati:

- Clustering **gerarchico**
  1. Aggregativo
  2. Divisivo
- Clustering **partizionale**
  1. K-Means
  2. EM (soft clustering)

2.1 Gerarchico

Andando più in dettaglio sulle diverse tecniche, abbiamo detto che la prima classe di clustering permette di creare degli alberi con i cluster e sotto-cluster. Questo può essere ottenuto con un approcco *bottom-up* che è il clustering **aggregativo**, il quale inizia dai singoli oggetti e ne raggruppa i più simili, per poi raggruppare i gruppi più simili e così via, fino ad ottenere un unico gruppo che sarà la radice dell'albero. Un altro approccio è quello *top-down*, il clustering **divisivo**, che in modo inverso dal precedente parte dal gruppo comprendente tutti gli elementi e lo divide in sotto-gruppi in modo da massimizzare la similarità intrinseca dei gruppi, fino ad arrivare ai singoli elementi.

Aggregativo Il clustering agglomerativo è un algoritmo **greedy**, che prende diversi cluster contenenti ognuno un singolo elemento e ad ogni passo determina i due cluster più simili. each step, the two most similar clusters are determined (81, and merged into a new cluster (9). The algorithm terminates when one large cluster containing all objects of S has been formed, which then is the only remaining cluster in C (7). Let us flag one possibly confusing issue. We have phrased the clus- tering algorithm in terms of similarity between clusters, and therefore we join things with maximum similarity (8). Sometimes people think in terms of distances between clusters, and then you want to join things that are the minimum distance apart. So it is easy to get confused between whether you re taking maximums or minimums. It is straightforward to produce a similarity measure from a distance measure d, for example by sim(x,y) = l/(1 + d(x,y)).

Divisivo

2.2 K-Means

2.3 EM

3 Sessione Sperimentale

4 Conclusioni