

ExpLOD: un framework per la generazione di spiegazioni per recommender systems che utilizza le informazioni della Linked Open Data Cloud

Cataldo Musto, Pasquale Lops, Marco de Gemmis, Giovanni Semeraro

Dip. di Informatica - Università degli Studi di Bari

{cataldo.musto, pasquale.lops, marco.degemmis, giovanni.semeraro}@uniba.it

Abstract

In questo contributo presentiamo EXPLOD, una piattaforma che sfrutta le informazioni disponibili nella Linked Open Data (LOD) cloud per generare delle *spiegazioni* a supporto dei suggerimenti generati da un recommender system. La metodologia proposta è basata sulla costruzione di un grafo che connette gli item graditi all'utente con le raccomandazioni mediante le proprietà descrittive degli oggetti disponibili nella LOD cloud. A seguire le proprietà più rilevanti di questo grafo vengono utilizzate per costruire dinamicamente una spiegazione in linguaggio naturale che spieghi il comportamento del recommender system e motivi il suggerimento.

Questo contributo sintetizza i contenuti presentati sulla rivista International Journal of Human-Computer Studies [Musto *et al.*, 2019]. Come mostrato nell'articolo, tale sistema ha mostrato la sua efficacia in un esperimento che ha coinvolto oltre 300 utenti. I risultati dell'esperimento hanno dimostrato che il nostro framework è in grado di generare spiegazioni in grado di rendere più trasparenti ed affidabili i suggerimenti prodotti dal recommender system.

1 Introduzione

Le performance dei recommender systems sono generalmente valutate sulla base della capacità degli algoritmi di predire correttamente quali sono gli oggetti graditi ad un utente in un particolare dominio applicativo. Di conseguenza, la maggior parte delle metriche di valutazione a stato dell'arte premiano i sistemi in grado di massimizzare l'accuratezza predittiva degli algoritmi, a discapito di altri elementi. Questa ha portato nel corso degli anni alla definizione di algoritmi che sono diventati via via più accurati, ma anche meno interpretabili. Tale scelta penalizza però l'esperienza d'uso degli utenti, poichè essi si trovano spesso davanti a suggerimenti che sono ottenuti a valle di oscuri meccanismi di elaborazione che egli non è spesso in grado di capire o interpretare.

Per questo motivo Tintarev e Masthoff [Tintarev e Masthoff, 2012] hanno introdotto il concetto di *spiegazione* nei recommender systems, intesa come una *giustificazione* che

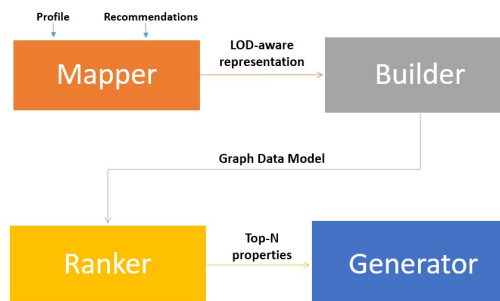


Figura 1: Organization of the framework

permetta all'utente di percepire la qualità del suggerimento ricevuto e di comprendere il comportamento dell'algoritmo. Nello specifico, vengono definiti sette possibili *obiettivi* alla base della generazione di una spiegazione, come trasparenza, scrutabilità, efficacia, capacità di persuasione, efficienza e soddisfazione.

In questo lavoro proponiamo un framework per la generazione di spiegazioni personalizzate che utilizza le informazioni disponibili nella Linked Open Data cloud, e, nello specifico, le proprietà disponibili in DBpedia¹, il mapping di Wikipedia in formato RDF. Come anticipato, l'algoritmo di spiegazione è basato su un grafo che connette gli item graditi all'utente con i suggerimenti generati dal recommender system per mezzo delle proprietà descrittive disponibili su DBpedia, come l'autore di un libro, l'attore di un film, e così via. Le proprietà più rilevanti vengono successivamente individuate attraverso delle formule matematiche e vengono inserite all'interno di una spiegazione in linguaggio naturale.

Nella prossima sezione andremo a dettagliare il comportamento del framework illustrandone le componenti principali e mostrando alcune schermate della piattaforma.

2 Descrizione del Framework

La pipeline eseguita da EXPLOD è mostrata in Figura 1. Il framework è strutturato in quattro moduli: Data MAPPER, Graph BUILDER, Property RANKER e un GENERATOR.

La componente Data MAPPER implementa dei meccanismi per *mappare* ciascun oggetto potenzialmente raccomandabi-

¹<http://wiki.dbpedia.org/>

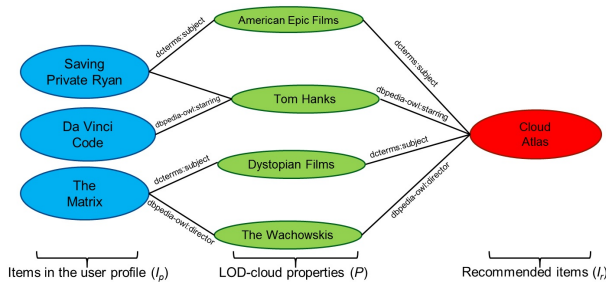


Figura 2: EXPLOD Data Model.

le (libri, film, musica, etc.) con degli elementi della LOD cloud. Ad esempio, se l'utente in passato ha gradito film come *The Matrix* e *Cloud Atlas*, il modulo andrà ad individuare gli elementi in DBpedia cui gli item fanno riferimento. Nel nostro caso, il mapping è effettuato implementando dei meccanismi di *matching* del nome dell'item con il titolo della pagina di Wikipedia cui il nodo DBpedia fa riferimento. L'output restituito dal MAPPER è una lista di URI in cui gli item raccomandati e quelli presenti nel profilo dell'utente sono stati mappati con elementi DBpedia. È importante sottolineare che questo passaggio preliminare è di fondamentale importanza per l'intero processo, poiché le spiegazioni sono generate proprio sfruttando le proprietà descrittive presenti in DBpedia. In assenza di una fase di mapping, non sarebbe possibile accedere all'informazione disponibile nella Linked Open Data cloud.

Successivamente, l'elenco degli item mappati con la LOD cloud è utilizzato per alimentare il graph BUILDER. Lo scopo della seconda componente della pipeline è di sfruttare le informazioni presenti in DBpedia per costruire un grafo che connetta gli item graditi con quelli suggeriti sulla base delle proprietà disponibili. Un esempio di tale grafo è riportato in Figura 2. In questo caso, l'utente ha gradito i film *Saving Private Ryan*, *The Matrix* e *The Da Vinci Code* e ha ricevuto come suggerimento il film *Cloud Atlas*. Sulla base di questo, quattro proprietà descrittive comuni agli item presenti nel profilo sono aggiunti al grafo². Le proprietà in comune sono etichettate come *proprietà candidate*, poiché rappresentano l'insieme preliminare di proprietà che *potrebbero* spiegare il suggerimento ricevuto dall'utente.

Il terzo passaggio della pipeline è eseguito dal modulo RANKER. Esso analizza il data model precedentemente costruito al fine di individuare quali tra le *candidate properties* siano quelle più adeguate a spiegare la raccomandazione. Nello specifico, il modulo assegna uno score di *rilevanza* a ciascuna proprietà. Tale formula è basata sull'intuizione che una *buona* spiegazione dovrebbe riuscire a dare risalto alle proprietà sono comuni al numero quanto più alto possibile di item presenti nel profilo. Data una proprietà candidata c , la formula implementata nel RANKER è la seguente

$$score(c, I_p, I_r) = \left(\alpha \frac{n_{c, I_p}}{|I_p|} + \beta \frac{n_{c, I_r}}{|I_r|} \right) * IDF_c \quad (1)$$

²È importante sottolineare che le proprietà presenti nella LOD cloud sono in realtà molte di più. Per vincoli legati alla dimensione del contributo abbiamo riportato solo un piccolo sottoinsieme

dove n_{c, I_p} descrive il numero di archi che collegano c con gli elementi del profilo, n_{c, I_r} è il numero di archi che collegano c con la raccomandazione, mentre α e β sono due parametri che regolano il peso dei diversi fattori della formula. Infine, IDF_c è un adattamento della classica Inverse Document Frequency [Manning *et al.*, 2008] su DBpedia. In generale tale formula serve a dare un maggior score a quelle proprietà che sono maggiormente collegate agli item nel profilo utente e alla raccomandazione. Applicando questa formula tutte le proprietà candidate vengono ordinate sulla base del loro score di rilevanza e le prime K sono utilizzate per generare la spiegazione. Ad esempio, facendo riferimento all'esempio in Figura 2, *Tom Hanks* è la proprietà più rilevante poiché descrive due dei tre film del profilo e la raccomandazione stessa.

Infine le proprietà restituite dal RANKER sono utilizzate dalle componente GENERATOR che si occupa di generare la spiegazione vera e propria a supporto della spiegazione. L'idea di base è le proprietà più rilevanti vadano a popolare un *template* di spiegazione. Ad esempio, la spiegazione generata a supporto del data model descritto in Figura 2 è: "*I recommend you CLOUD ATLAS since you often like movies starred by TOM HANKS as DA VINCI CODE and SAVING PRIVATE RYAN. Moreover, I recommend it because you sometimes like DYSTOPIAN MOVIES as THE MATRIX and AMERICAN EPIC FILMS as SAVING PRIVATE RYAN*". Tale spiegazione verrà poi mostrata all'utente attraverso un'interfaccia.

3 Conclusioni

Per motivi di spazio non ci è possibile fornire ulteriori dettagli sul framework e sulla sua valutazione. Per maggiori informazioni suggeriamo di fare riferimento a [Musto *et al.*, 2019] e allo screen recording che mostra l'interazione sulla piattaforma disponibile online³. Come mostrato nell'articolo, il nostro framework ha mostrato la sua efficacia in un esperimento che ha coinvolto oltre 300 utenti. I risultati dell'esperimento hanno dimostrato che il nostro framework è in grado di generare spiegazioni in grado di rendere più soddisfacenti ed affidabili i suggerimenti prodotti dal recommender system. Questo rende più trasparente e controllabile l'intero processo di raccomandazione, in linea con i dettami della recente normativa GDPR.

Riferimenti bibliografici

- [Manning *et al.*, 2008] C. Manning, P. Raghavan, e H. Schütze. Scoring, term weighting and the vector space model. *Introduction to Information Retrieval*, 100:2–4, 2008.
- [Musto *et al.*, 2019] Cataldo Musto, Fedelucio Narducci, Pasquale Lops, Marco de Gemmis, e Giovanni Semeraro. Linked open data-based explanations for transparent recommender systems. *International Journal of Human-Computer Studies*, 121:93–107, 2019.
- [Tintarev e Masthoff, 2012] N. Tintarev e J. Masthoff. Evaluating the effectiveness of explanations for recommender systems. *UMUAI*, 22(4-5):399–439, 2012.

³<http://bit.ly/2r4elgM>