Sistema di raccomandazione per utenti di Twitter basato su Time Weight Collaborative Filtering

Introduzione

In questo progetto sono stati presi in esame due algoritmi di temporal collaborative filtering proposti nel libro Recommender Systems di Charu C. Aggarwal per valutarne l'efficacia nel suggerire nuovi hashtag ad utenti di Twitter.

Dataset

Il dataset di partenza è stato raccolto in un arco di 36 giorni ed è composto da circa 1.500.00 di tweet scritti da 1176 utenti. Ciascuna entry del dataset è composta dai seguenti attributi:

- messageId: identificativo del messaggio
- userId: identificativo dell'utente che ha scritto il tweet
- username: username dell'utente che ha scritto il tweet
- content: contenuto del tweet
- creationTime: timestamp di inserimento del tweet
- replyToPostId: identificativo del tweet a cui si risponde (assume valore -1 se non è un tweet di risposta)
- replyToUsername: username dell'utente a cui si risponde (assume valore -1 se non è un tweet di risposta)
- retweetedFromPostId: identificativo del tweet in cui si effettuato il retweet (assume valore -1 se non è stato retweettato)
- retweetCount: numero di retweet
- retweetedFromUserName: Username di chi ha effettuato il retweet.

Da questo dataset ne è stato derivato un altro estraendo da ciascuna entry gli hashtag contenuti nel campo content e associando a ciascuno di essi l'identificatore dell'utente e la data di creazione. Durante questo processo gli hashtag più volte utilizzati dallo stesso utente sono stati filtrati prendendo in

considerazione soltanto l'ultimo utilizzato e riportandone la data. Questa è stata trasformata opportunamente, partendo dal formato yyyy-mm-dd si ottiene ad un intero compreso fra 0 e 36 corrispondente al periodo di crawling espresso in giorni. Infine da quest'ultimo dataset si è costruita la matrice di rating utente/hashtag utilizzando come metrica per il rating un tf-idf dato dalla seguente formula:

$$TFIDF_{ij} = (u_{ij}/h_i) * log(U/p_j)$$

u_{ij} = numero di volte l'utente i ha utilizzato l'hashtag j.
h_i = numero di hashtag utilizzati dall'utente i.U = numero di utenti totali.
p_j = numero di utenti che hanno usato l'hashtag j.

La matrice di rating ottenuta è formata da circa 175.008 record.

Algoritmi

I due algoritmi esaminati fanno entrambe parte della categoria temporal collaborative filtering che a differenza dei tradizionali algoritmi di collaborative filtering utilizzano informazioni temporali per migliorare l'efficacia. I due algoritmi inoltre risiedono nella tipologia degli item-based, ovvero considerano la similarità tra item piuttosto che tra utenti per calcolare le predizioni. La metrica scelta per calcolare la similarità tra item è la PCC (Pearson correlation coefficent) che risulta essere migliore rispetto alla similarità del coseno quando applicata ad item.
Il primo algoritmo è del tipo decay-based, ovvero utilizza una funzione di decadimento per calcolare un peso il cui valore dipende dalla distanza temporale tra l'istante in cui viene effettuata la predizione e l'istante di inserimento del rating. L'idea è di utilizzare questo peso nel calcolo della predizione per dare maggiore importanza ai rating più vicini temporalmente al tempo target di predizione rispetto a quelli più lontani.

$$w_{uj}\,(t_f\,) = exp[-\lambda * (tf - tuj)]$$

$w_{uj}(t_f)$ = peso per il rating espresso dall'utente u per l'item j calcolato al tempo t_f.
t_f = tempo target per il rating.
t_{uj} = istante di tempo il cui il rating espresso dall'utente u per l'item j è stato inserito.
λ: è il grado di decadimento che regola l'importanza del tempo.

Il peso così calcolato viene poi utilizzato nella formula per predire il rating di un utente per un certo item.

$$\hat{r}_{uj}(t_f) = \mu_j + \frac{\sum_{v \epsilon P_u(j)} w_{uj} \cdot Sim(j,v) \cdot (r_{uj} - \mu_v)}{\sum_{v \epsilon P_u(j)} w_{uj} \cdot |Sim(j,v)|}$$

$\hat{r}_{uj}$= rating predetto dell'utente u per l'item j al tempo t_f.
$P_u(j)$ = insieme di item simili all'item j.
w_{uj} = peso calcolato al passo precedente.
Sim(j,v) = similarità tra l'item i e quello v.
r_{uj} = rating dell'utente u per l'item j.
t_f = tempo target di predizione.
μ = rating medio di un item.

Il secondo algoritmo è una variante del primo in cui invece di calcolare un peso per ciascun item tramite la decay function, si calcola un discount factor che rappresenta l'errore futuro atteso e si assegna a ciascun item un peso pari all'inverso di questo errore.

$$D_{ui} = \left(1 - \frac{|O_{ui} - r_{ui}|}{r_{max} - r_{min}}\right)^{\alpha}$$

D_{ui} = discount factor per l'item i rispetto all'utente u.
O_{ui} = rating medio dell'utente u per item simili all'item i.
r_{ui}= rating dell'utente u per l'item i.
r_{max}, r_{mim}= rating massimo e minimo complessivi.
α= parametro di controllo.

L'idea sviluppata è che la differenza tra il rating di un utente per un item e il rating medio dello stesso utente per item simili rappresenti l'errore dovuto all'evoluzione temporale.

La formula del primo algoritmo viene così modificata utilizzando il discount factor:

$$\hat{r}_{uj} = \frac{\sum_{i \epsilon Q_j(u)} D_{ui} \cdot Sim(i,j) \cdot r_{ui}}{\sum_{i \epsilon Q_j(u)} D_{ui} \cdot |Sim(i,j)|}$$

$\hat{r}_{uj}$= rating predetto dell'utente u per l'item j.
$Q_j(u)$ = insieme di item simili all'item j per cui l'utente u ha espresso un rating.
D_{ui} = discount factor per l'item i rispetto all'utente u.
Sim(i,j) = similarità tra l'item i e quello j.
r_{ui} = rating dell'utente u per l'item i.

Risultati

Entrambe gli algortimi hanno mostrato un discreto funzionamento nel suggerire nuovi hashtag simili agli utenti, tuttavia risulta difficile valutare quale dei due risulti essere quello più efficace. Questo è dovuto sia alla scarsa dimensione del dataset che non permette di operare su dati raccolti in un lungo periodo temporale, che alla difficoltà intrinseca di valutare quale siano i suggerimenti migliori. Bisogna inoltre considerare che i due algoritmi sono stati costruti a partire da un algoritmo di KNN-Item della libreria LibRec e potrebbero necessitare di ulteriori perfezionamenti.

Esempi

Di seguiti sono riportati alcuni esempi non significativi di suggerimenti per un utente prodotti utilizzando entrambe gli algoritmi:
utente "142052768" i cui hashtag utilizzati sono:

[itunes, vertex, klout, cleanweb, happybirthday, dallas, tx, gome, labchat, waywardradio, npr, grammar, musicplayce, gotanproject, astronomy, astrophysics, hmxbs, genius, rq, msnbc, blossomdearie, americanidol, fox, boom, dropsmic, dianakrall, kayahcesariaevora, sunday, tropical, beryl, phase7, horror, pontypool, disappointing, movie, netflix, qblog, dogs, melodygardot, paulinhomoska, redrush, mazzystar, citizencope, radiohead, maroon5, enchantment, smcdallaa, guykawasaki, kloutforgood, throughthewormhole,

ancientaliens, relaxingsunday, paranormal, fringe, fightthefuture, worldsapart, theuniverse, hailpocalypse]

Il primo algoritmo suggerisce i seguenti hashtag:

[cnn, ff, fb, nowplaying, smcdallas, quarantine2]

Mentre il secondo suggerisce:

[cnn, hillaryclinton, ff, equinox, quarantine2, nowplaying]

Si può notare che alcuni suggerimenti sono uguali per entrambe gli algoritmi e che in generale riguardano serie tv, musica e sport che sono gli argomenti utilizzati dall'utente.

Utente "100258169" con hashtag:

[polish, armenia, gauck, r, altmaier, nkorea, polen, catherine, polishart, napi, nato, afghanistan, hietzingerabendsalon, iran, webinar, humantrafficking, euro, chicago, 321, golf, publichealth, journalists, fbi, rendition, smem, 2012its, socialmedia, globaldev, polska, pl, globalhealth, mrsa, antibiotic, biosecuri, joplin, youtube, forcedmarriage, v, news, video, galicia, www, bernatowicz, council, global, wgf, economy, politics, exhibition, dictator, azerbaijan, eurovision, eurovision2012, eur, secretjustice, ahmedkhaledmueller, european, beirut, faz, china, tajikistan, syrian, hula, homs, eu, us, homophobia, ukraine, freedomofassembly, prideparade, gay, far, conference, ngo, internationalaid, civ, usa, nauka, human, trafficking, universities, asia, egypt, conspiracy, immigration, guant, 9, memorialday, multiplesclerosis, globalrace, obama, history, tahrir, india, nigeria, guantanamo, icrc, scottish, library, preparedness, abcdrbchat, poles, greeneconomy, wed2012, frus, aq, jihadi, libi, bachelet, violenceagainstwomen, arctic, macroregional, risk, eusbsr, muslims, newyork, newjersey, edip, nigerian, italy, lewandowski, greece, panorama, deraa, africa, europe, ecology, teliasonera, turkcell, sweden, fi, bosnia, uk, askfs, disaster, nlm, peoplelocator, indianapolis, hattip, jos, diplomacy, mennonite, em2012, bigdata, digitaldata, ict4d, refugee, switzerland, israel, givebl, warsaw, bsr2012, childsurvival, gardening, polishfamily]

Il primo algoritmo suggerisce i seguenti hashtag:

[heu, egypreselex, belarus, drc, bahrain]

Mentre il secondo suggerisce:

[egypreselex, euro2012, belarus, poland, pakistan]

Anche in questo secondo esempio si possono riscontrare delle uguaglianze tra i suggerimenti forniti dai due algoritmi che rispecchiano gli interessi dell'utente.