Università degli Studi di Padova

Dipartimento di Matematica "Tullio Levi Civita"

Corso di laurea in Informatica

Applicazione di un sistema di raccomandazione in ambito BTB

Relatore
Aiolli Fabio
Università di Padova

Laureando
Daniel Rossi

Ringraziamenti

Ecco raggiunto un altro traguardo molto importante carico di felicità che vorrei condividere con coloro che mi sono stati vicino durante questi anni.

Prima di tutto vorrei porgere i miei ringraziamenti al Professor De Giovanni per il supporto fornitomi durante il periodo di stage e durante la stesura della tesi.

Ringrazio l'azienda Trans-Cel per l'opportunità di stage offertomi, grazie a Filippo, Nicola e Beatrice per le tante ore passate insieme.

Ringrazio la mia famiglia ed in particolare mia madre per lo sforzo fatto nello spronarmi a ricercare un futuro in cui mi sentissi realizzato.

Ai miei compagni di Università, per le giornate passate insieme in Torre Archimede, nostra seconda casa, per le risate, le giornate brutte e quelle belle, grazie di avermi fatto sentire apprezzato e coinvolto, grazie per il tempo dedicatomi, in particolare grazie a Victor e Mihai per gli importanti momenti passati insieme a studiare, sarà difficile trovare migliori compagni di corso di voi.

Infine voglio ringraziare Anna, non ultima per importanza, per avermi sempre sostenuto durante tutti questi anni, per avermi ascoltato nei momenti di difficoltà e aver condiviso con me la felicità dei momenti belli, non penso potrò mai ringraziarti abbastanza per l'aiuto nella revisione di questo documento.

Padova, Dicembre 2018 *Daniel Rossi*

Abstract

Il seguente documento vuole essere un report della collaborazione svoltasi tra l'Università di Padova, nella persona del Professor Aiolli, e l'azienda Estilos SRL nel periodo intercorso tra marzo e agosto 2021.
Lo scopo della collaborazione è stato quello di applicare un sistema di raccomandazione ad un e-commerce BTB, dove i clienti sono intermediari che poi rivendono i prodotti a clienti terzi.
Il tipo di prodotti venduti nell'e-commerce non segue le classiche logiche di vendita in famosi e-commerce come può essere Amazon, quindi si è reso necessario ragionare sui diversi tipi di prodotto disponibili. Nello specifico i dati su cui abbiamo lavorato provengono da un'e-commerce di un'azienda cliente di Estilos e non sono basati, come nei classici problemi di raccomandazioni, su recensioni o valutazioni, bensì sullo storico vendite dell'azienda nel canale e-commerce.
Gli obiettivi del progetto prevedevano di rielaborare lo storico vendite in modo d'avere i dati in forme più usuali, a cui poi applicare gli approcci più popolari al momento nell'ambito dei sistemi di raccomandazione, come il collaborative filtering e il content based filtering.
Più nello specifico si vuole raccomandare a ciascun cliente una lista di prodotti che si ritiene possano interessarlo.
Ci si proponeva inoltre di trovare prodotti simili e correlati dato uno di partenza.
Infine ci si proponeva di vagliare approcci ibridi che permettessero di combinare informazioni che descrivono l'interesse di un cliente verso un prodotto rispetto diverse prospettive quali per esempio la quantità totale acquistata e la recentezza dell'acquisto.

Indice

Lista delle figure

Lista delle tabelle

1 Introduzione

1.1 Contesto progetto

Nel mondo dei software ERP *(Enterprise Resource Planning)*, ossia prodotti software pensati per le aziende che permettono la gestione e il controllo dei procesi e delle funzioni aziendali, uno dei più famosi è di certo il gestionale SAP, il quale è sviluppato in moduli integrabili e a seconda delle esigenze dell'azienda utilizzatrice se ne possono attivare in qualunque combinazione.
Uno di questi moduli è l'e-commerce hybris, utilizzato dalle aziende come canale di vendita online e alcune delle sue potenzialità sono che è altamente personalizzabile e perfettamente integrato con i sistemi SAP, come per esempio con il modulo CRM (*(Customer Relationship Management)*), il quale si occupa di tutte le modalità di gestione delle relazioni con il cliente.

1.2 L'idea di progetto

L'idea nasce, in un'ottica di innovazione del prodotto, all'interno di un progetto aziendale che mira all'ampliamento e miglioramento delle funzionalità di hybris, dove uno degli aspetti su cui si vuole lavorare è quello della personalizzazione dei prodotti mostrati agli utenti dell'e-commerce, si vuole sperimentare racco-

mandazioni sui prodotti basate sullo storico vendite e non sui feedback lasciati dall'utente in quanto la loro raccolta non è prevista dal sistema. Partendo quindi dallo storico vendite di un'azienda Cliente, con hybris configurato in versione BTB, l'obiettivo era quello di utilizzare i dati disponibili per raccomandare a ciascun cliente una lista di prodotti $TopN$ che gli risultassero interessanti e per ciascun prodotto una lista di prodotti simili ad esso, sempre interessanti per il cliente a cui viene mostrato quello specifico prodotto.
Come detto solitamente si parte da feedback impliciti / espliciti dati dagli utenti ai prodotti, ma non essendo disponibili si cercherà di estrarre informazioni relative l'interesse del cliente verso un prodotto rispetto diversi punti di vista, quali può essere la quantità acquistata, la recentezza dell'acquisto, il numero di fatture in cui compare o la spesa totale per quello specifico articolo.
Una volta organizzate le informazioni in delle matrici cliente-prodotto grezze, si voleva eseguire una sorta di preprocessing su di esse, andando a trasformarle in dei rating rispetto una scala comune che fornisse una misura d'interesse del cliente rispetto il prodotto. Si è voluto applicare le tecniche più popolari usate nei sistemi di raccomandazione, quali il collaborative filtering alle rating matrix ottenute dal preprocessing descritto precedentemente e il content-based filtering ai dati descrittivi dei prodotti. Data la non disponibilità di rating si è poi pensato di considerare il problema anche come una next basket recommendation, dove si vanno a vedere le sessioni d'acquisto e in base a queste si predice quella finale, questo approccio potrebbe funzionare nel caso i clienti acquistino spesso gli stessi prodotti.

1.3 Organizzazione del testo

Di seguito viene riportata per ogni capitolo una piccola descrizione delle tematiche trattate:

- **Capitolo 2**: viene mostrato come sono inizialmente organizzati i dati, come sono stati trattati e quali informazioni si è potuto ricavare;
- **Capitolo 3**: viene fatto un breve riepilogo della teoria sui sistemi di raccomandazione, spiegando meglio gli approcci del collaborative filtering e del content based filtering, oltre che spiegando il funzionamento degli algoritmi utilizzati e delle metriche;

- **Capitolo 4**: vengono spiegate le diverse tecniche di preprocessing utilizzate per trasformare i dati grezzi in valutazioni.
- **Capitolo 5**: viene riportata una descrizione della libreria Cornac, dove sono implementati modelli e metriche per l'esecuzione di test;
- **Capitolo 6**: vengono mostrati i risultati delle metriche rispetto i diversi algoritmi applicati al preprocessing dei dati;
- **Capitolo 7**: vengono mostrati i risultati delle metriche rispetto i diversi algoritmi applicati al preprocessing dei dati nella loro versione combinata;
- **Capitolo 8**: vengono mostrati i risultati delle metriche considerando il problema come un next basket recommendation;
- **Capitolo 9**: vengono riportate le conclusioni del lavoro svolto, andando a delineare problemi risolti, criticità e sviluppi per il futuro;

1.4 Convenzioni tipografiche

Il testo adotta le seguenti convenzioni tipografiche:

- ogni acronimo, abbreviazione, parola ambigua o tecnica viene spiegata e chiarificata alla fine del testo;
- ogni parola di glossario alla prima apparizione verrà etichetta come segue: $parola^{[g]}$;
- ogni riga di un elenco puntato terminerà con un ; a parte l'ultima riga che si concluderà con un punto.

2
Analisi dei dati

2.1 Principali tabelle

Come detto precedentemente i dati sono lo storico vendite dell'azienda Cliente estratti dal modulo hybris, questi sono organizzati secondo le tabelle SAP, avremo quindi lo storico delle fatture, dove avremo una tabella per la testata della fattura, ossia la parte descrittiva dove viene riportato l'acquirente, e una tabella per le posizioni della fattura, ossia la parte dovve vengono riportati i materiali acquistati. Abbiamo inoltre due tabelle che riportato rispettivamente l'anagrafica cliente e materiali, dove possiamo trovare informazioni aggiuntive che li descrivono.
Mi è stata inoltre fornita anche una tabella glossario che riportava per ciascuna tabella una breve spiegazione di ogni campo.
Tutte queste tabelle sono state fornite in formato Excel.
Quindi ricapitolando le principali tabelle disponibili sono le seguenti:

- **VBAK**: testata della fattura;
- **VBAP**: posizioni della fattura;
- **MARA**: anagrafica prodotto;
- **KNA1**: anagrafica materiali.

Andiamo ora a vedere per ciascuna di queste tabelle i campi annessi e alcune informazioni di natura statistica:

2.1.1 Tabella VBAK

La tabella VBAK contiene la testata di circa 35000 fatture, datate dall'anno 2016 fino a maggio 2021.
Ciascuna riga della tabella è la testata di una fattura e ne riporta il suo codice identificativo (**VBELN**) insieme con il codice del cliente a cui è associata (**KUNNR**). Inoltre ciascuna fattura riporta data e ora (**ERDAT**, **ERZET**) in cui è stata emessa e l'importo totale e valuta corrispondente (**NETWR**, **WAERK**).

2.1.2 Tabella VBAP

La tabella VBAP è la tabella che contiene le posizioni delle fatture, riporta per ogni fattura il numero prodotti acquistati riportando diverse informazioni, ci sono circa 250000 posizioni. Ciascuna riga della tabella riporta quindi il codice identificativo (**VBELN**) della fattura e il codice identificativo del prodotto acquistato (**MATNR**), poi vengono riportati per quel prodotto il prezzo unitario (**NETPR**), la quantità acquistata (**KWMENG**), la spesa totale con la valuta (**NETWR**, **WAERK**) e il codice gerarchia prodotto storico (**PRODH**), ossia quello salvato in MARA al momento del'emissione della fattura.

2.1.3 Tabella KNA1

La tabella KNA1 riporta l'anagrafica cliente, abbiamo salvati circa 3000 clienti, per ciascuna riga abbiamo il codice cliente (**KUNNR**), il codice paese d'origine (**LAND1**), il nome dell'azienda (**NAME1**), la località (**ORT01**) e la regione (**REGIO**).

2.1.4 Tabella MARA

Come detto la tabella MARA è quella che riporta l'anagrafica dei materiali, nella nostra futura trattazione considereremo questi materiali come prodotti in quanto sono acquistabili all'interno dell'e-commerce.

Ciascuna riga della tabella riporta quindi un prodotto univoco composto dal suo usuale codice identificativo (**MATNR**), dal codice della gerarchia prodotto (**PRODH**) e del gruppo merceologico (**MATKL**) e un breve descrizione testuale (**MAKTX**), poi abbiamo delle informazioni su dimensioni, volumi e peso, per le dimensioni abbiamo il campo (**GROES**) che fornisce le dimensioni nel formato (lunghezza X larghezza X altezza), oppure possiamo trovarli nei campi (**LAENG**, **BREIT**, **HOEHE**), rispettivamente lunghezza, larghezza e altezza con l'unità di misura delle dimensioni (**MEABM**), poi abbiamo per il volume il campo (**VOLUM**) con l'unità di misura (**VOLEH**), e per il peso il campo (**NTGEW**) con l'unità di misura (**GEEWI**).

2.2 Prodotti

Da questo momento in poi faremo riferimento ai materiali chiamandoli prodotti come detto in precedenza.
In totale nella tabella anagrafica materiali (MARA) sono presenti circa 75000 prodotti diversi, mentre i prodotti effettivamente venduti risultano essere molti meno attestandosi all'incirca verso gli 8000.
Abbiamo però due campi interessanti che riguardano la gerarchia prodotto (PRODH) e il gruppo merceologico (MATKL), questi due campi ci permettono di studiare la similarità dei prodotti.

2.2.1 Gerarchia prodotto (PRODH)

Il campo gerarchia prodotto (PRODH) è un campo numerico di 18 cifre utile per separare i prodotti rispetto le diverse categorie su più livelli. Nella tabella secondaria T179 vengono definiti i livelli di gerarchia e le diverse categorie. Vediamoli di seguito:

- **PRODH**: codice gerarchia prodotto;
- **STUFE**: livello gerarchia;
- **VTEXT**: descrizione testuale;

Ciascun codice PRODH contenuto nella tabella T179 avrà rispettivamente il seguente numero di cifre in base al livello di gerarchia (STUFE):

- **STUFE = 1**: 1° livello della gerarchia, il codice sarà di 5 cifre.
- **STUFE = 2**: 2° livello della gerarchia, il codice sarà di 10 cifre, dove le prime 5 identificano la categoria di 1° livello a cui appartengono mentre le restanti 5 indentificano la sotto-categoria di 2° livello.
- **STUFE = 3**: 3° livello della gerarchia, il codice sarà di 18 cifre, dove le prime 10 identificano la categoria di 2° livello a cui appartengono mentre le restanti 8 indentificano la sotto-categoria di 3° livello.

Ciascun prodotto sarà quindi provvisto di un codice di 18 cifre che identificherà una categoria per ogni livello.
Nella tabella VBAP ci sono alcune posizioni dove a parità di codice prodotto (MATNR) si hanno codici PRODH diversi, questo è dovuto al diverso momento temporale in cui sono stati acquistati. Infatti nella tabella VBAP il codice PRODH è storico, ho provveduto per semplicità ad aggiornarli tutti al codice PRODH più recente riportato nella tabella anagrafica materiali MARA. Il numero di prodotti interessati sono circa 100 su 10000 posizioni.

Selezione categorie di 1° livello

PRODH	#tot	#sold	titolo
00010	28	0	categoria1
00020	0	0	categoria2
00030	0	0	categoria3
00040	5	0	categoria4
00050	2	0	categoria5
00090	2	0	categoria6
00100	1117	173	CATEGORIA1
00200	645	130	CATEGORIA2
00250	31	11	CATEGORIA3
00300	405	92	CATEGORIA4
00400	525	36	CATEGORIA5
00500	1715	70	CATEGORIA6
00600	334	6	CATEGORIA7
00700	1	0	CATEGORIA8
00900	70441	7702	CATEGORIA9
00950	28	1	CATEGORIA10
09999	215	0	ALTRO

Nella tabella vengono mostrati i codici PRODH delle categorie di 1° livello, nella colonna #tot il numero di prodotti diversi per quella categoria, nella colonna #sold il numero di prodotti diversi acquistati almeno una volta appartententi a quella categoria ed infine il titolo della categoria. Come possiamo vedere le prime sei categorie con titolo in minuscolo hanno pochi prodotti catalogati in MARA e nessun prodotto venduto.

Chiaramento il fatto che siano tutti a zero è dovuto all'aggiornamento dei codici PRODH di cui abbiamo parlato precedentemente, a prescindere da ciò il numero di posizioni che prima riportavano codici appartenenti alle categorie prese in considerazione non superava la decina, quindi non considerare queste categorie in quanto si è smesso di usarle sembra la scelta più logica.
La categoria ALTRO (09999) non è stata considerata in quanto riporta prodotti

che non sono disponibili sull'e-commerce. Inoltre le categorie SISTEMI RICICLO (00700) e CHEMICALS (00950), dato il basso numero di prodotti presenti in MARA e le basse vendite, si è preferito non considerarle.

Overview categorie di 1° livello

PRODH	*#posizioni*	$\sum_{KWMENG}$	$\mathbb{E}_{KWMENG}$	$\mathbb{E}_{NETPR}$ (€)	$\sum_{NETWR}$ (€)	$\mathbb{E}_{NETWR}$ (€)	*titolo*
00100	5908	15461	2.61	1613.44	3865.97	22840167.61	CATEGORIA1
00200	1936	3219	1.66	5898.09	8640.59	16745496.87	CATEGORIA2
00250	333	2949	8.85	552.99	3772.04	1377422.72	CATEGORIA3
00300	745	1390	1.86	4353.24	5364.34	3996430.94	CATEGORIA4
00400	389	1651	4.24	708.17	2404.47	940777.84	CATEGORIA5
00500	1133	12984	11.46	175.75	1034.63	1172236.58	CATEGORIA6
00600	153	494	3.23	448.52	1338.80	83501.04	CATEGORIA7
00900	239740	1070334	4.46	26.67	66.61	15968039.56	CATEGORIA9
	250339	1108493.51	4.72	1706.86	3225.75	63124073.16	valori riassuntivi

Nella tabella per ogni categoria $PRODH$ di 1° livello possiamo vedere:

- $\#posizioni$: numero di posizioni in cui compaiono prodotti di quella categoria in fattura;
- $\sum_{KWMENG}$: quantità totale di prodotti acquistati appartenenti a quella categoria;
- $\mathbb{E}_{KWMENG}$: quantità media per fattura di prodotti acquistati appartenti a quella categoria;
- $\mathbb{E}_{NETPR}$: prezzo medio per fattura di prodotti acquistati di quella categoria;
- $\sum_{NETWR}$: spesa totale per prodotti di quella categoria;
- $\mathbb{E}_{NETWR}$: spesa totale media per fattura di prodotti di quella categoria;

Dalla tabella possiamo vedere come la categoria RICAMBI & ACCESSORI riporti un prezzo medio per fattura molto più basso rispetto alle altre categorie, questo è dovuto al fatto che i pezzi di ricambio ed accessori non sono macchine o sistemi da usare per fornire un servizio quanto un prodotto per riparare quanto già si possiede, possiamo vedere che in termini di posizioni l'acquisto di pezzi di ricambio copra una cospicua parte delle posizioni in fattura, oltre che valere un importante parte del fatturato per l'azienda. Le altre categorie vendono macchine e sistemi per la pulizia quindi i prezzi medi per prodotti sono molto maggiori e per i clienti finali questi prodotti rappresentano un investimento. Da quanto detto finora si vengono a creare due macro categorie di prodotti:

- **Macchine**: questa macro categoria racchiude le seguenti categorie di 1° livello: 00100, 00200, 00250, 00300, 00400, 00500, 00600;
- **Ricambi**: questa macro categoria invece racchiude la sola categoria 00900.

Dobbiamo dare un'ultima precisazione infine, la maggior parte delle categorie di 2° e 3° livello risultano essere di prodotti della macro categoria delle macchine, quindi la gerarchia è molto più densa orizzontalmente per le macchine rispetto che i pezzi di ricambio, infatti per le macchine le categorie risultano essere i diversi modelli di macchinari disponibili e i prodotti a catalogo di quella categoria sono le varianti dello stesso macchinario. Per i pezzi di ricambio abbiamo solo poche categorie contenitore che li raggruppano tutti.

2.2.2 Gruppo merceologico (MATKL)

Il gruppo merceologico (MATKL) non è organizzato come una gerarchia, come lo è invece la gerarchia prodotto (PRODH), bensì come un insieme di prodotti, in totale abbiamo circa 160 gruppi, dove uno di questi contiene tutti i prodotti che prima abbiamo classificato come macchine. Rispetto il codice PRODH, il gruppo merceologico è più divisivo rispetto i ricambi, questo ci può aiutare in quanto ora siamo in grado di categorizzare anche i ricambi.

2.2.3 Dimensione, volume e peso

I campi rigurdanti dimensione, volume e peso potrebbero essere utili per ricercare una similarità tra i prodotti.
Le informazioni sulle dimensioni, come lunghezza, larghezza e altezza sono praticamente ridondanti nei campi GROES e (LAENG, BREIT, HOEHE) se non per alcuni prodotti dove le informazioni sono esclusive di uno dei due campi.
Per peso e volume abbiamo i rispettivi campi numerici e altri due campi che riportano le unità di misura, per il volume possono essere i metri cubi o i millimetri cubi, per il peso o i kilogrammi o i grammi. La criticità riguardo queste misure sono sulla loro scarsità, infatti su 75000 prodotti avremo informazioni su volume e peso rispettivamente solo sul 20% e 39%, mentre sui prodotti acquistati almeno una volta sul 19% e 5%.

3

Sistemi di raccomandazione

3.1 Introduzione

Uno dei campi più popolari al momento verso cui si rivolge una particolare attenzione è quello dei sistemi di raccomandazione, da ora in poi RS, in quanto l'attività online sta aumentando sempre più e nascono sempre più spesso nuovi servizi che permettono di scegliere oggetti, siano questi prodotti, video, musica, film o molto altro, da cataloghi vastissimi. I sistemi di raccomandazione permettono di navigare questi cataloghi andando a cercare gli oggetti che risultino più interessanti per l'utente.

3.2 Preliminari

In generale possiamo dire che un RS si compone di diversi elementi, in primo luogo abbiamo i cosidetti "attori" del problema, gli user, ossia gli utenti del sistema, e gli item, ossia gli oggetti che si vuole consigliare. Abbiamo a disposizione inoltre informazioni riguardo l'interazione tra user e item solitamente sotto forma di feedback implicito o esplicito, questa misura viene definito rating. Questi vengono utilizzati dal RS, insieme con eventuali dati legati al contesto di user e item, per effettuare raccomandazioni.

3.2.1 Feedback impliciti / espliciti

Solitamente le informazioni che legano user e item, ossia i rating, possono essere di due tipi:

- Implicito: 1 se c'è stata interazione tra lo user e l'item, 0 se non c'è stata;
- Esplicito: valutazione numerica intera in una scala da 1 a N, 0 se non c'è stata interazione.

Nel nostro caso di studio però ci ritroviamo a metà strada in quanto, la quantità per esempio potrebbe essere considerata come un dato esplicito ma non è definita su una scala discreta, mentre se lo considerassimo implicito trascureremmo delle informazioni che possono in qualche modo fornire una misura di interesse.

3.3 User-item interaction matrix

		Items					
		1	*2*	...	*i*	...	*m*
Users	*1*	5	3		1	2	
	2		2				4
	:			5			
	u	3	4		2	1	
	:					4	
	n			3	2		

I rating sono organizzati in matrici, dette appunto user-item interaction matrix o semplicemente matrici dei rating (R), dove sulle righe abbiamo gli user mentre sulle colonne abbiamo gli item, nell'incrocio abbiamo riportato il rating. Può essere di rating sia impliciti che espliciti e le celle vuote corrispondono allo 0. Quando scriviamo r_{ui} dove u è lo user e i è l'item.

3.4 Task

L'obiettivo del sistema può essere quello di consigliare ad uno user una lista di N item, detta *TopN* che si ritiene possano interessargli, oppure dato un item si può trovare una lista di item che si considerino simili allo stesso.

3.5 Approcci

Definito quindi il task abbiamo diversi modi per poter soddisfare il nostro obiettivo, in generale abbiamo due principali categorie di RS:

- **Non Personalizzato**: andiamo a consigliare i prodotti che globalmente risultano più popolari, ossia che abbiamo complessivamente ricevuto più valutazioni, o quelli con rating più alto. Questo approccio non va a considerare le informazioni relative il singolo user;
- **Personalizzato**: ci sono diversi approcci che vedremo nelle sezioni successive, in generale si fanno raccomandazioni basate sulle informazioni dello user.

I due approcci più famosi negli RS sono il collaborative filtering, dove si cerca di consigliare item ad uno user basandosi su user simili, mentre nel content-based filtering si cerca di raccomandare item simili a quelli con cui si ha già interagito;

3.5.1 Collaborative filtering

Collaborative filtering è un approccio agli RS basato sulla similarità, raccomandiamo ad uno user item interessanti per altri user simili ad esso, e viceversa item simili ad altri item per cui ha dimostrato interesse. La similarità può essere quindi di due tipi: item-based, basata quindi sulla similarità tra prodotti o user-based ossia su quella tra user. Ci sono due approcci possibili al collaborative filtering:

- Memory-based: utilizziamo la matrice dei rating per calcolare la similarità tra user e item, metodi basati sull'algoritmo K nearest neighbour;
- Model-based: utilizziamo dei modelli che attraverso degli algoritmi permettono di predire il rating su item non valutati.

UserKnn

UserKnn è un metodo memory-based che fa uso della matrice dei rating, ogni user avrà quindi un proprio "profilo", ossia la propria riga nella matrice dei rating. L'idea è quella di calcolare la similarità tra tutti gli user e fatta questa operazione è possibile calcolare il rating previsto per ogni item non valutato rispetto ad uno user. Per fare ciò andiamo a selezionare i k user con similarità più alta con il nostro user target e calcoliamo la media pesata dei loro rating usando come pesi la similarità.

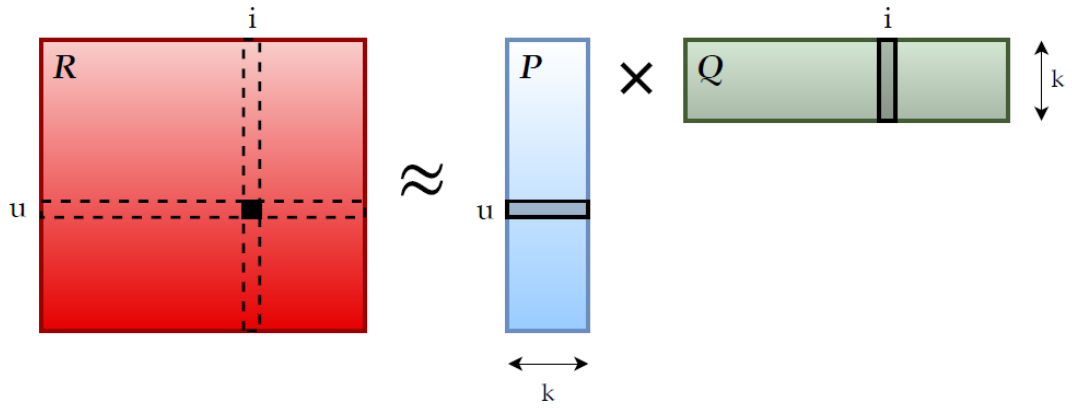
Fatto questo si procede ad ordinare per ciascuno user tutti i prodotti secondo i rating ottenuti e si ottiene così la lista $TopN$ degli item più interessanti.
Questo metodo funziona senza informazioni relative alle caratteristiche degli user o item e può gestire rating espliciti o impliciti con formule leggermente diverse.

ItemKnn

ItemKnn è un metodo memory-based molto simile al precedente, qui si va a considerare però gli item da raccomandare e il loro "profilo" è la propria colonna della rating matrix. Si calcola la similarità tra tutti gli item e dato uno user si procede a calcolare il rating stimato sugli item che non ha valutato andando a trovare per ciascuno di essi la lista di K item che ha valutato più simili ad esso, poi calcola la media pesata dei rating dei K item selezionati usando come peso la similarità.

Matrix Factorization (MF)

Nell'approccio model-based Matrix Factorization (MF) è uno dei modelli più famosi, questo si basa sul concetto che si possa mappare user e item verso uno spazio delle feature comune di una certa dimensionalità K, possiamo quindi creare due matrici P e Q, dove P è una matrice avente sulle righe gli user e come colonne le K feature, mentre Q è una matrice avente sulle righe le k feature e sulle colonne gli item, quello che vogliamo ottenere è una approsimazione della rating matrix attraverso la moltiplicazione di P e Q, ossia $R \approx P \cdot Q^T = \hat{R}$.



Quello che otteniamo è quindi un profilo sia per gli user che per gli item rispetto lo stesso spazio delle feature, il problema maggiore risulta però nell'ottenere queste

due matrici, possiamo farlo creando una funzione di loss apposita e andando ad allenare in modo alternato le matrici P e Q, cercheremo quindi di ridurre dopo ciascuna iterazione la differenza tra R e $\hat{R}$.
Una volta che il modello è allenato possiamo predire il rating dato da uno user u su un item i moltiplicando i profili corrispondenti $\hat{r}_{ui} = P_u \cdot Q_i^T$.

Variational Auto-Encoder for CF (VAECF)

Ancora da scrivere.

3.5.2 Content-based filtering

Il content-based filtering è un approccio che si basa sull'idea di consigliare item simili a quelli con cui si è già interagito.
Ciascun item possiede delle feature, per esempio nel caso si abbia come item dei film queste potrebbero essere i generi, l'insieme delle feature può essere mappata su di un vettore delle feature, per ogni item quindi si riporta nel vettore 1 se possiede quella feature, 0 altrimenti, questo permette di definire un profilo per l'item. Una volta fatto ciò si procede a creare anche un profilo per lo user, ci sono diversi modi per farlo ma per esempio si può considerare tutti i profili degli item con cui si è interagito e farne quindi la media pesata basata sui rating corrispondenti. Ottenuto un profilo anche per lo user si può calcolare la similarità tra di esso e quello degli item per poi ordinarli secondo similarità ottenendo così la lista $TopN$.

3.6 Valutazione

Quanto abbiamo visto finora sono metodi che ci permettono di effettuare le raccomandazioni, vogliamo trovare però anche il modo per poterle valutare. Per prima cosa dobbiamo dividere le matrici dei rating in training e test set.
Per fare ciò andiamo ad eseguire uno shuffle delle coppie (user,item) e si va ad assegnare al training set l'80% delle coppie e le restanti al test set, questo sistema non ci assicura che uno user sia presente nel training set. Date le raccomandazioni fornite dal RS vogliamo valutarle rispetto due aspetti principali: il rating e il ranking. Il primo aspetto riguarda semplicemente la diversità tra i rating

stimati da quelli reali delle coppie (user,item) del test set, queste metriche non vengono solitamente usate in quanto non è un buon modo per valutare un RS perché non ci permette di capire se un'item consigliato sia rilevante. Andando invece a considerare il concetto di ranking ci rendiamo conto che sia più legato a ciò che vogliamo andare a valutare, le metriche annesse considerano rilevanti gli item presenti nel test set e vanno a verificarne la posizione nella lista $TopN$. In generale le metriche si applicano a ciascuno user e il risultato finale è la media dei singoli risultati.

3.6.1 AUC

L'AUC (area under the curve) è un metrica che permette di valutare un RS basandosi sul numero di coppie di item presenti nella $TopN$ che sono in ordine sbagliato, vediamo di seguito la formula:

$$AUC = \frac{1}{|N^+| \cdot |N^-|} \sum_{i \in N^+} \sum_{j \in N^-} [s(i) > s(j)]$$

Il termine N^+ è l'insieme degli item presenti nel test set, mentre N^- sono tutti gli item rimanenti. $s(i)$ è la valutazione data dal RS sull'item i, quello che si fa è andare a calcolare il numero di coppie di item non in ordine nella $TopN$ andando a vedere lo score assegnatogli.

3.6.2 nDCG@k

La metrica nDCG@k (normalized Discount Cumulative Gain) è una metrica che ci permette di calcolare una misura basata sulla posizione degli item rilevanti, ossia quelli del test set.

$$DCG@k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{log_2(i+1)} \qquad IDCG@k = \sum_{i=1}^{|REL_i|} \frac{2^{rel_i} - 1}{log_2(i+1)}$$

Possiamo vedere come la nDCG sia in realtà la normalizzazione - daconcludere

$$IDCG@k = \frac{DCG@k}{IDCG@k}$$

Glossario

- **Agile**: modello di ciclo di vita nato per sopperire alla rigidità dei modelli precedenti, caratterizzato da rilasci rapidi e incrementali, usato per rispondere velocemente alle richieste dei clienti in termini di nuovi requisiti.

- **Best Bound**: è una soluzione fornita da un modello dopo che si è raggiunto il tempo limite di esecuzione, questa soluzione non è ottima ma rappresenta il miglior risultato disponibile.

- **Brainstorming**: incontri di gruppo creativi utili per risolvere problemi o individuare nuove idee. Servono più di due partecipanti in modo che vi sia una discussione arbitraria e quindi utile.

- **Ciclo Di Vita**: insieme degli stati che il prodotto assume dal concepimento al ritiro.

- **Euristica**: algoritmo progettato per risolvere un problema velocemente, spesso una strada obbligata per risolvere problemi molto difficili.

- **GIL**: meccanisco usato dagli interpreti dei linguaggi di programmazione per sincronizzare i thread in modo che ve ne sia in esecuzione in coda.

- **Modello Incrementale**: modello di ciclo di vita che prevede rilasci multipli e successivi, dove ciascuno di essi realizza un incremento di funzionalità. I requisiti vengono trattati per importanza, prima quelli di maggior importanza in modo che possano stabilizzarsi con il rilascio delle versioni fino a quelli minori.

- **Milestone**: punto nel tempo al quale associamo un insieme di stati di avanzamento.

- **Open Source**: termine utilizzato per riferirsi ad un software di cui i detentori dei diritti sullo stesso ne rendono pubblico il codice sorgente.

- **Project Management**: gestione delle attività di analisi, progettazione, pianificazione e realizzazione degli obiettivi di un progetto, compito svolto dal project manager di un'azienda attraverso anche strumenti idonei.

- **Slack**: tempo aggiuntivo ad un'attività che ha lo scopo di evitare ritardi nella produzione del prodotto.
- **Soluzioni**: costrutto software e termine usato per indicare l'insieme di pacchi da disporre e le loro coordinate che permettono di collocarli nel contenitore ed il contenitore stesso..
- **Solver**: software commerciale o open source che permette di risolvere problemi di programmazione lineare.
- **Stackable**: termine utilizzato per indicare se un pacco può avere sopra di sé altri pacchi.
- **Time Limit**: termine usato per indicare un tempo limite entro il quale può essere ricercata una soluzione dal solver.
- **Vehicle Routing**: famiglia di problemi che trattano tutti gli aspetti della gestione di una flotta di veicoli nell'ambito della logistica.

Lista degli acronimi

2D ..... Modello 2D

2DR ..... Modello 2D con rotazione

2DRS ..... Modello 2D con rotazione e sequenza di scarico

3D ..... Modello 3D

CBC ..... Coin-or branch and cut

GIL ..... Global Interpreter Lock