Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE

Corso di Laurea Magistrale in Informatica

Recommender Systems: Integrazione dell'influenza nei social e della *Community Similarity* nei modelli di raccomandazioni

Relatore:
Chiar.mo Prof.
Giovanni Rossi

Presentata da:
Mattia Ballo

Sessione III
Anno Accademico 2017/2018

Dedicata a tutti quelli che credono
che nessuna impresa sia impossibile,
fino a che c'è anche un solo sciocco
che combatte per essa.

Introduzione

Il grande successo riscosso dall'*e-commerce*, e la crescita esponenziale dei suoi utilizzatori, ha portato le aziende che operano in quel settore ad intraprendere azioni di marketing, mirate a massimizzare le vendite ed aumentare i guadagni, specifiche per ogni utente. Questo è stato fatto sviluppando i *recommender system*, sistemi che potevano fornire consigli di acquisto sui prodotti agli utenti in base alle preferenze espresse. Tuttavia questi sistemi non erano in grado di fornire una previsione con precisione. La grande diffusione dei *social network*, insieme alla quantità di dati che viene generata dagli utenti su queste piattaforme, ha attirato l'attenzione delle aziende verso nuovi orizzonti. Questo ha fatto si che questi sistemi, da semplici piattaforme per la condivisione di interessi e contenuti, divenissero vere e proprie fonti di dati.
Non passò molto tempo per cui le aziende, resesi conto dell'enorme potenzialità derivante dall'utilizzo di questi dati per fini commerciali, iniziarono a mettere in piedi dei veri e propri modelli che sfruttassero questi dati per effettuare campagne di marketing più accurate, utilizzando gli stessi dati che gli utenti avevano deliberatamente deciso di condividere.
La natura sociale delle piattaforme e degli utilizzatori ha, inoltre, favorito la formazione di *communities* di persone che condividono gli stessi interessi e opinioni verso un particolare argomento o articolo.

Lo scopo di questa tesi è valutare i vari approcci utilizzati per costruire un recommender system basato sui social network, analizzando i vari algoritmi utilizzati e valutandone le performance, comparandoli gli uni con gli

altri attraverso delle metriche definite. Successivamente verrà proposto un indicatore per valutare la bontà di una community per l'applicazione di tali modelli.

Nella prima parte viene fatta una panoramica sul mondo dell'e-commerce, approfondendo i recommender systems esistenti e le due metodologie principalmente utilizzate. Viene inoltre fatta una panoramica sui social network e il ruolo che essi hanno nella vita quotidiana di ogni consumatore. Successivamente vengono introdotti alcuni approcci per lo sviluppo di un recommender system basato su queste piattaforme, analizzando alcuni modelli e valutando come l'influenza degli amici sugli utenti possa essere presa in considerazione nell'equazione per migliorare la precisione. Infine viene introdotto il concetto di community e viene proposta una metrica per valutare quanto una community sia adatta ad essere utilizzata per fornire raccomandazioni.

Nella seconda parte vengono presentati degli insiemi di dati su cui verranno effettuati gli esperimenti empirici applicando i modelli sopra descritti, sia per i singoli utenti che per i vari tipi communities. Successivamente verrà caratterizzato il *Community Similarity Degree* per la valutazione della bontà della community. Infine vengono effettuati degli esperimenti empirici su di esso al fine di analizzare il suo comportamento al variare di alcuni fattori.

Indice

Elenco delle figure

Elenco delle tabelle

Parte I

Stato dell'Arte e Metodologie

Capitolo 1

Una panoramica sui Social Network

Ormai è assodato che si possa affermare che l'avvento dei *social network* abbia inequivocabilmente dato il via ad una vera e propria rivoluzione tecnologica per quanto riguarda l'utilizzo di internet da parte degli utenti.
Al giorno d'oggi le piattaforme di microblogging sono fra le più versatili e popolari tecnologie presenti nella vita quotidiana di ogni persona. Questa presenza è tanto importante al punto da aver creato un nuovo paradigma per quanto riguarda le relazioni interpersonali che sempre più spesso trovano gran parte della loro formazione online. Si stima infatti che nel 2017 circa 2,5 miliardi di persone nel mondo abbiamo utilizzato regolarmente almeno una piattaforma di *Social Networking*. Dato, questo, che è destinato ad aumentare (figura 1.1).
Oltre all'aspetto sociale, queste piattaforme hanno favorito la proliferazione dei così detti *user-generated contents*[1]. Questa enorme mole di dati di varia natura, spesso contiene un significato semantico riguardante il *sentiment* che un particolare utente prova verso un argomento di cui pubblica un contenuto. L'analisi di questo sentiment è mirata ad identificare le informazioni

[1]Qualsiasi tipo di contenuto creato dagli utenti e pubblicato in Internet, spesso reso fruibile tramite le piattaforme di social networking.

soggettive degli utenti come le opinioni, i punti di vista e le sensazioni verso un particolare contenuto.
Questo significato, facilmente interpretabile dall'uomo ma molto complesso da identificare per una macchina, ha fatto si che venissero a formarsi naturalmente delle *communities* di persone con interessi e opinioni comuni e condivisibili. Secondo un report stilato da `Google`, nel 2010 era possibile identificare circa 620 milioni di communities create dagli utenti di `Facebook`. Con il passare del tempo questo numero è aumentato in modo esponenziale, portando con sè una quantità enorme di user-generated contents verso il quale le aziende hanno mostrato sempre più interesse.
All'interno di queste communities hanno, inoltre, iniziato ad affermarsi delle figure di spicco le cui opinioni e tendenze erano condivise dalla maggior parte del bacino di utenza della community stessa. Queste figure, chiamate *influencers*, hanno così permesso lo sviluppo di una nuova metodologia di marketing basata sull'influenza mediatica degli stessi verso il resto degli utenti.
La grande diffusione dell'e-commerce e lo sfruttamento di queste piattaforme ha quindi permesso la proliferazione di realtà di business interamente basate su di esse. Tuttavia queste rapide diffusioni hanno portato alla luce un problema di *data overloading*, sia per le aziende che devono sviluppare delle campagne di marketing ottimali, sia per gli utenti che si ritrovano immersi in un mare pieno di informazioni che può portare ad un disorientamento e quindi a un peggioramento del risultato della strategia di marketing stessa.
Per questo motivo si è iniziato a sviluppare dei sistemi di recommendation che potessero filtrare le informazioni, seguendo determinati approcci di varia natura, in modo da poter migliorare la precisione.
Con la promulgazione e l'adozione del *GDPR*[2], la quantità di dati personali a cui le aziende hanno accesso è diminuita vertiginosamente e le analisi so-

[2]**General Data Protection Regulation**, ufficialmente regolamento (UE) n. 2016/679, è un regolamento dell'Unione europea in materia di trattamento dei dati personali e di privacy adottato il 27 aprile 2016.

pra citate sono aumentate ulteriormente di complessità. Per questo motivo al giorno d'oggi si cercano nuovi approcci in modo da poter affinare questi particolari sistemi in conformità con le attuali normative EU.
Nei prossimi capitoli verrà mostrato il funzionamento dei recommender system, come abbiano contribuito alla diffusione dell'e-commerce e come questi siano evoluti con l'avvento dei social network.

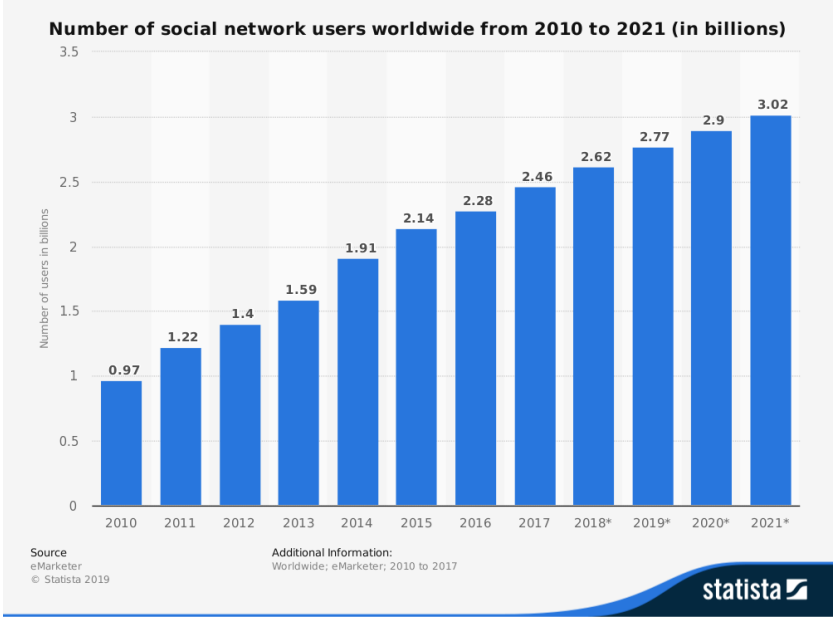

Figura 1.1: Numero di utenti che utilizzano almeno un Social Media

Capitolo 2

E-commerce e Recommendation Systems

Con la grande diffusione di Internet e il continuo aumento della fiducia dei consumatori nell'utilizzo di piattaforme elettroniche per gli acquisti, l'e-commerce è diventato uno dei business più diffusi e remunerativi a livello mondiale.
Come si evince dalla figura 2.1, nel 2017 il volume complessivo di proventi generati da questo particolare commercio è stato pari a circa 2 300 miliardi di dollari americani ed è stimata una crescita che può raggiungere quota 4 900 miliardi nel 2021[3].
Dalla figura 2.2, si può notare come circa $1,66$ miliardi di persone hanno effettuato almeno un acquisto online nello stesso periodo, pari a circa il 20% della popolazione mondiale. Con la previsione di una ulteriore crescita dovuta proprio all'aumento di fiducia dei consumatori prima discusso.
Questo tipo di commercio si è infatti guadagnato una quota pari a circa il 10% del commercio retail mondiale, destinata a crescere in futuro (figura 2.3).

Tuttavia, nonostante le ottime premesse mostrate dalle statistiche, come

[3]fonte: *statista* (https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/)

ogni mercato anch'esso soffre dei problemi che affliggono ogni altro mercato economico. Quello con più impatto è la saturazione[4], processo il quale, anche se in maniera poco accentuata, sta già avvenendo come mostra la figura 2.4. La crescita percentuale infatti, nonostante sia ancora molto elevata, si sta riducendo con la previsione di perdere più di 5 punti percentuali nei prossimi anni.
Per questo motivo un efficacie recommendation system sarà fondamentale per poter ottimizzare le strategie di vendita e di marketing negli anni a venire.

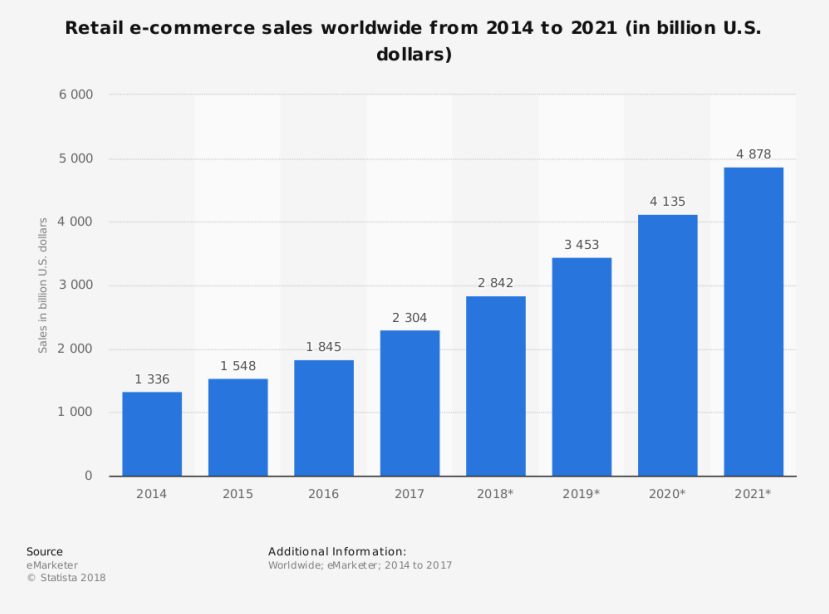

Figura 2.1: Proventi da commercio e-commerce.

[4]La saturazione di un mercato si ha quando il potenziale produttivo delle imprese supera la capacità di assorbimento del mercato stesso.

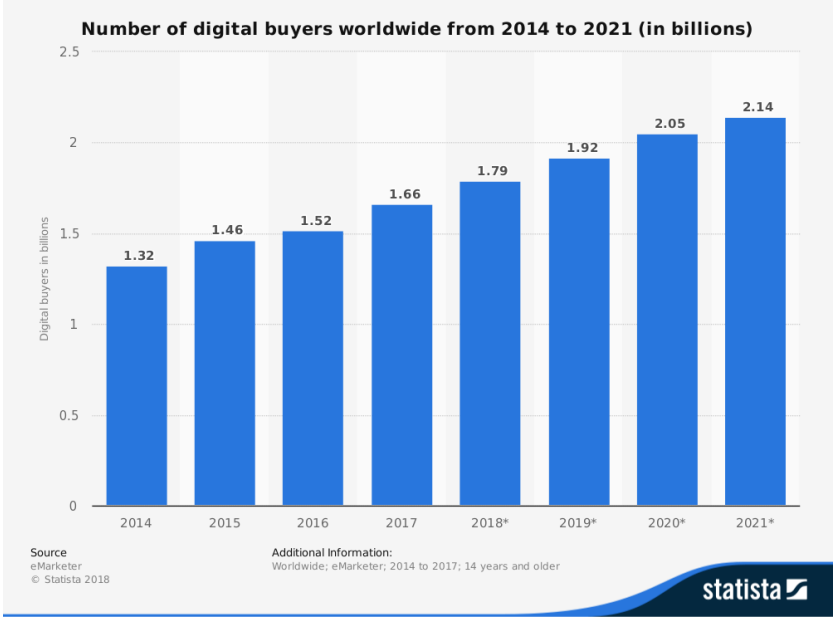

Figura 2.2: Consumatori che utilizzano l'e-commerce.

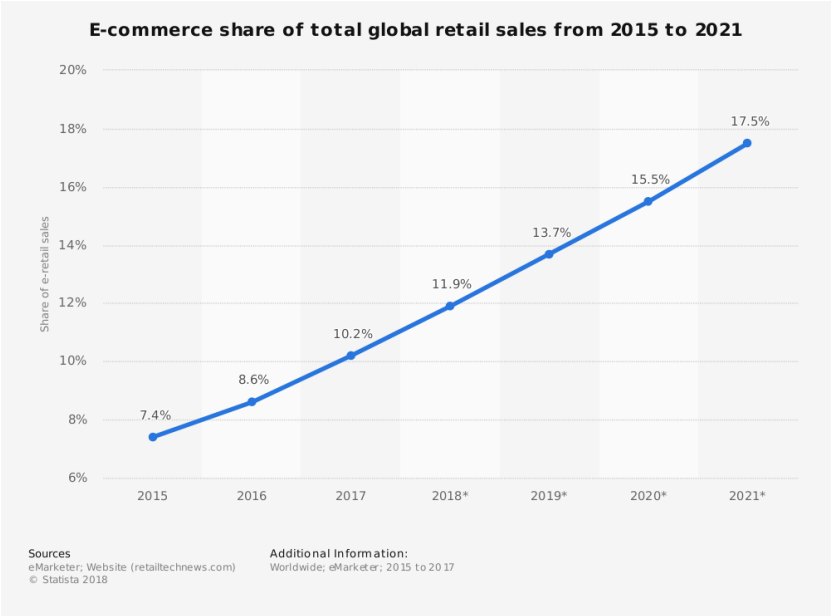

Figura 2.3: Percentuale dell'e-commerce nel business retail mondiale.

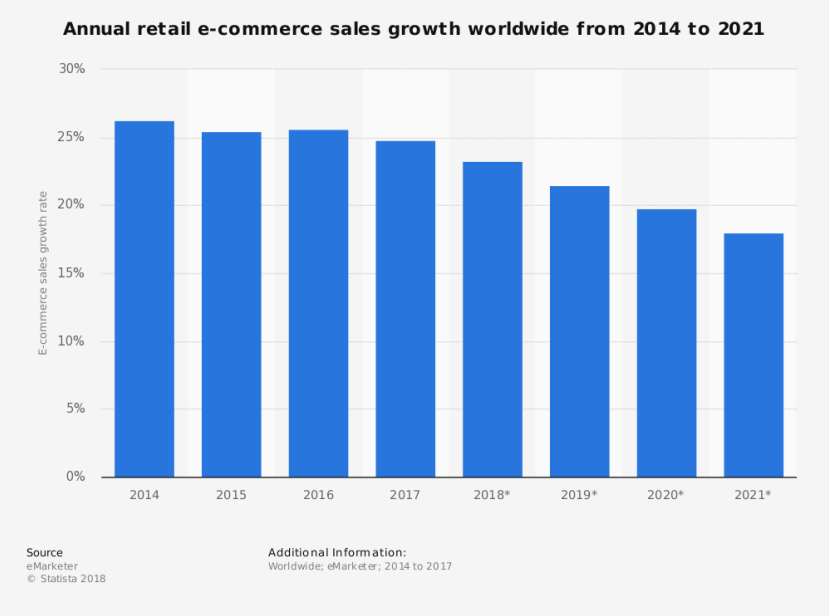

Figura 2.4: Crescita percentuale dei proventi da e-commerce.

2.1 Recommendation Systems

Un **Recommendation System** o **Recommender System** è una sottoclasse di *information filtering system*[5] atta a predire la valutazione che un determinato utente potrebbe dare ad un determinato articolo o argomento. Questo tipo di sistemi trovano utilizzo in una moltitudine di aree. Dai social network al commercio retail, passando per i servizi di intrattenimento, finanziari, la ricerca di specialisti e così via.
Questi sistemi, generalmente, producono una lista di risultati basandosi principalmente su due approcci: *collaborative filtering* e *content-based filtering*. Nelle sezioni successive si andranno ad approfondire entrambi.
Alcune ricerche hanno portato alla luce come un approccio ibrido, che combina i due metodi sopra citati, possa essere più efficace in alcuni casi di studio. Comparando infatti i risultati è emerso che questo tipo di approccio ottenga

[5]Sistema che rimuove informazioni irrilevanti o duplicate prima della presentazione all'utente in modo da ridurre il sovraccarico di informazioni.

dei risultati più accurati rispetto agli approcci "puri", risolvendo al tempo stesso alcune problematiche che vengono a presentarsi.
Un esempio famoso di utilizzo di questo sistema è il caso di `Netflix`[1]. Esso infatti applica delle raccomandazioni sulla base delle abitudini di ricerca e visualizzazione di contenuti di utenti simili (collaborative filtering) in combinazione con la scelta di contenuti simili a quelli a cui l'utente ha assegnato una valutazione positiva in precedenza (content-based filtering).

2.1.1 Content-based Filtering

Questo approccio è basato sulla descrizione di un *articolo* e di un *profilo di preferenze* relativo ad un utente.[2, 3]
In questi sistemi vengono utilizzate delle parole chiave (*keywords*) per descrivere sia l'articolo sia le preferenze stesse. Questo sistema raccomanda gli articoli simili a quelli che l'utente ha apprezzato in passato. In particolare viene utilizzata una metrica per assegnare un valore ad ogni oggetto, sulla base delle preferenze, e vengono proposti quegli che ottengono il risultato maggiore. Le principali entità prese in considerazione dall'algoritmo sono le seguenti:

1. Un modello delle preferenze dell'utente.
2. La storia delle interazioni dell'utente con il recommender system.

In parole povere, questo metodo utilizza un profilo per l'articolo contenente attributi che lo caratterizzano con il sistema. Contemporaneamente utilizza un vettore pesato di caratteristiche definite dall'utente. Ogni peso rappresenta l'importanza della caratteristica per l'utente stesso.
L'approccio semplice utilizza la media dei valori degli articoli valutati mentre altri approcci più sofisticati utilizzano varie tecniche come *Classificatori Byesiani, analisi dei cluster, alberi di decisione* e *reti neurali artificiali* per valutare l'affinità.[4]
Un problema fondamentale di questo approccio è come il sistema sia in grado di "imparare" le preferenze dell'utente in base alle sue azioni riguardanti una

sorgente di contenuti e sia in grado di utilizzarle verso altri tipi. Quando si limita a consigliare prodotti dello stesso tipo di quelli valutati dall'utente, il valore del sistema è significativamente inferiore rispetto a quando debba consigliare contenuti di altra natura. Per esempio raccomandare news in base agli articoli letti dall'utente può essere utile ma sicuramente il valore è minore rispetto a raccomandare musica, film e prodotti sulla base delle preferenze sulla cronaca.
Alcuni esempi di sistemi che utilizzano questo approccio sono `Pandora Radio`[6], `IMDB`[7] e `Rotten Tomatos`[8]. Alcune applicazioni si possono trovare anche per quanto riguarda l'ambito della prevenzione sanitaria.[5]

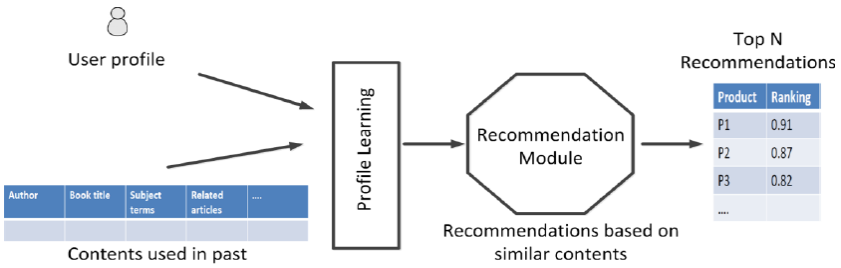

Figura 2.5: Esempio di content-based filtering.

2.1.2 Collaborative Filtering

A differenza alla tecnica precedentemente descritta, questa metodologia fornisce previsioni automatiche (filtering) degli interessi di un utente sulla base delle preferenze o gusti degli altri (collaborative). L'assunzione alla base di questo approccio è che se un utente A ha la stessa opinione su di uno specifico argomento di un utente B, allora è probabile che l'utente A sarà più propenso ad avere opinioni simili a B su di un altro argomento rispetto

[6]Servizio di streaming musicale che riproduce canzoni sulla base della prima scelta dall'utente.
[7]Servizio che consiglia film in base alla valutazione dei film già visti.
[8]Servizio simile a IMDB.

ad un utente scelto in maniera casuale.
In un senso più generale, il collaborative filtering è un processo di selezione delle informazioni utilizzando tecniche che includono la collaborazione di più agenti, punti di vista, sorgenti dati, etc.[6]
I passi fondamentali che compie questo approccio sono:

1. Identificare gli utenti che mostrano lo stesso pattern di preferenze con l'utente in esame.

2. Utilizzare le preferenze di quegli utenti per predire la preferenza dell'utente.

Questa tecnica risiede nella categoria dei così detti *user-based collaborative filtering* (figura 2.6). Un applicazione specifica è il *Nearest Neighbour algorithm*[9].
Un alternativa è l'*item-based collaborated filtering* (figura 2.7), dove l'algoritmo è incentrato sull'articolo e dove:

1. Viene costruita una matrice *item-item* che determina le relazioni fra i vari articoli.

2. Vengono predetti i gusti esaminando questa matrice e comparandola con i dati dell'utente.

Queste informazioni vanno poi filtrate ulteriormente seguendo la logica di business che si vuole applicare al sistema.
Alcune delle sfide più grandi da risolvere quando si utilizza questo approccio sono:

- **Data Sparsity**: questi approcci sono basati su dataset di enormi dimensioni, generando una matrice che mette a dura prova le performance del sistema. Uno dei problemi principali causati è il *cold-start problem*, ovvero la difficoltà nel fare previsioni per quegli utenti per cui non

[9]In ambito di riconoscimento di pattern, è un metodo non parametrico utilizzato per la classificazione e la regressione.[7]

si hanno ancora sufficienti informazioni per fornire risultati accurati. Lo stesso problema si ha per i nuovi articoli che devono ancora essere valutati.

- **Scalability**: nel momento in cui il bacino di utenza aumenta, così come i prodotti, anche una complessità lineare risulta troppo elevata per fornire risultati in tempi brevi. Per questo alcune compagnie tendono a creare dei *cluster*[10] per poter ovviare a questo problema.

- **Synonyms**: si riferisce alla tendenza ad avere prodotti molto simili ma con nomi differenti che il sistema quindi tratta come articoli distinti.

- **Gray Sheep**: si riferisce a quel bacino di utenza le cui preferenze sono difficilmente identificabili, per esempio quegli utenti che risultano non essere mai in accordo o in disaccordo con un gruppo.

- **Shilling Attacks**: quando un utente da recensioni molto positive ai suoi prodotti e molto negative a quelli dei competitors, in modo da favorire la sua attività.

- **Diversity e Long Tail**: le aspettative di questo approccio sono quelle di aumentare la varietà dei prodotti in quanto esso aiuta a scoprire nuovi prodotti. Tuttavia potrebbe, non intenzionalmente, favorire i prodotti più valutati a discapito di quelli meno valutati, dove per esempio non si hanno abbastanza informazioni per la computazione (*rich-get-richer effect*).

Nel prossimo capitolo si andranno a valutare vari approcci e paradigmi proposti negli ultimi anni per migliorare la tecnica di collaborative filtering sopra descritta, sfruttando i dati presenti sui social network.

[10] In informatica un computer cluster, o più semplicemente un cluster (dall'inglese grappolo), è un insieme di computer connessi tra loro tramite una rete telematica.

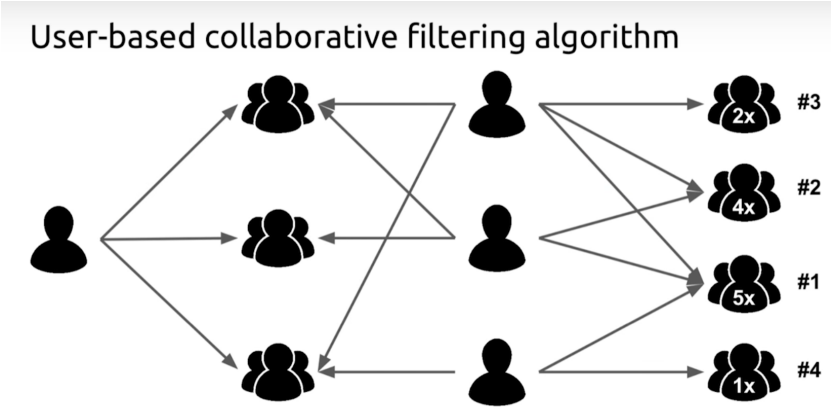

Figura 2.6: Esempio di user-based collaborative filtering.

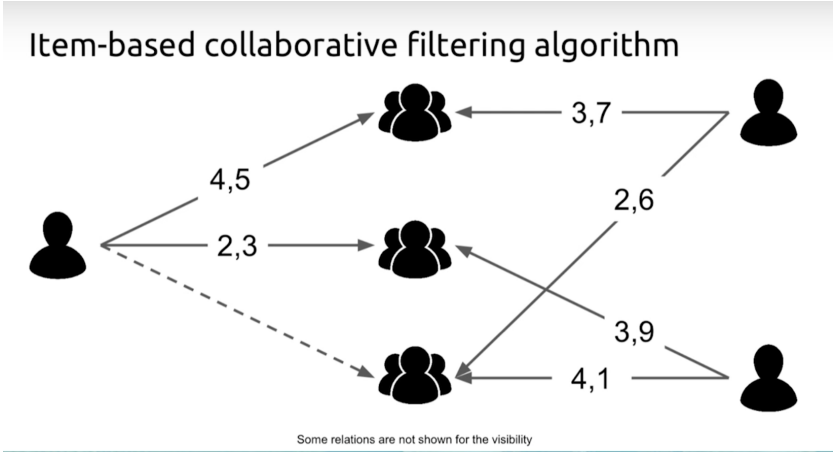

Figura 2.7: Esempio di item-based collaborative filtering.

Capitolo 3

Social Network e Recommendation Systems

L'influenza dei social media gioca un ruolo molto rilevante nelle campagne di marketing dei prodotti. Tuttavia essa è stata raramente presa in considerazione per lo sviluppo di un recommendation system fino a pochi anni fa. Negli ultimi anni, infatti, sono stati approfonditi alcuni paradigmi che prendono in considerazione le informazioni generate sui social network dagli utenti, come l'influenza degli utenti stretti, per valutare nuovi modelli di raccomandazioni. Uno dei problemi che si è venuto a creare è stato quello dell'*information overload*, letteralmente un "sovraccarico di informazioni" dovuto alla natura stessa di queste piattaforme. Per questo si è cercato di filtrare ulteriormente questa mole di dati in modo da utilizzare solo quelli ritenuti rilevanti.
Di seguito verranno approfonditi alcuni approcci basati su collaborative filtering, sia item-based che user-based, che sfruttano le informazioni derivanti dai social per migliorare le performance del recommender system.

3.1 Social Network-Based Recommender System

Molti recommender system, basati sul collaborative filtering, assumono che tutti gli utenti siano indipendenti, ignorando il ruolo dell'influenza sociale nelle decisioni di acquisto delle persone.
Per poter analizzare al meglio questo tipo di approccio si pensi ad uno scenario in cui esso possa essere applicato.
Andrea vuole vedere un film. Dalle sue preferenze risulta che il suo genere preferito è il drammatico. Tramite una ricerca su internet è riuscito a trovare due risultati interessanti: Il Curioso Caso di Benjamin Button e Seven. Entrambi i titoli hanno ottime valutazioni. Non sapendosi decidere chiama Marco con cui passa molte serate e gli chiede consiglio. Marco consiglia Seven e gli propone di passare una serata assieme guardando il film. Andrea accetta felice la proposta ed entrambi guardano quel film.
Analizzando questo scenario si possono notare tre fattori fondamentali che hanno contribuito alla selezione del contenuto. Per prima cosa le preferenze dell'utente lo hanno portato a prendere in considerazione risultati di quel genere. In secondo luogo le recensioni positive fornite dagli altri utenti hanno permesso di restringere la scelta a soli due titoli. Infine il consiglio dell'amico ha permesso la scelta finale. Molte scelte nella vita quotidiana sono influenzate da questi tre fattori. Questa influenza è molto più efficace della semplice pubblicità. Infatti se tutti questi fattori sono positivi, sarà molto probabile che l'utente compi questa scelta (figura 3.1).
L'impatto di questa influenza non è limitato solo agli amici più stretti ma anche ad alcuni più distanti. Per questo motivo vengono proposte due tecniche per valutare questi tipi di influenze: *immediate friend inference* e *distant friend inference*.

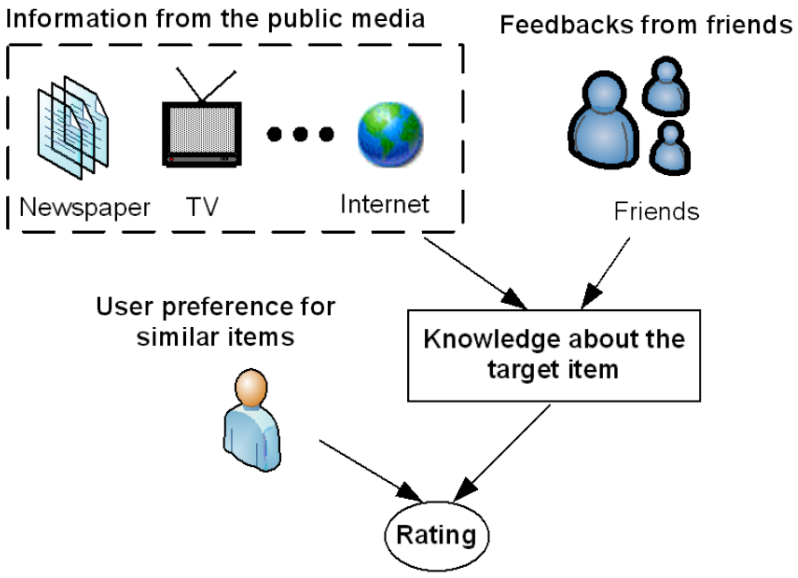

Figura 3.1: Esempio di processo di scelta basato sui 3 fattori.

3.1.1 Immediate Friend Inference

Formalmente un social network può essere considerato come un grafo[11] $\mathcal{G} = (\mathcal{U}, \mathcal{E})$ dove U rappresenta i nodi, ovvero gli utenti, e $\mathcal{E}$ rappresenta l'insieme degli archi, ovvero le relazioni sociali fra utenti. Ogni utente $u \in \mathcal{U}$ possiede un insieme di caratteristiche $\mathcal{A}_u$ e un insieme di *immediate neighbors*, ovvero gli amici stretti, $N(u)$ tali che $v \in N(u), (u, v) \in \mathcal{E}$. Il sistema inoltre contiene un record delle preferenze degli utenti che può essere definito come una relazione $\mathcal{T} = (\mathcal{U}, \mathcal{I}, \mathcal{R})$ dove $\mathcal{U}$ è l'insieme degli utenti del social network, $\mathcal{I}$ l'insieme dei prodotti o servizi in cui ogni elemento i possiede un insieme di attributi $\mathcal{A}'_i$ e $\mathcal{R}$ l'insieme delle valutazioni tale che $\mathcal{R}_{ui}$ rappresenta la valutazione dell'utente u per l'articolo i. Questa valutazione consiste in un valore numerico k. Si definisce inoltre con $I(u)$ l'insieme degli articoli per cui l'utente u ha espresso una preferenza.[8]
Definiti questi formalismi, lo scopo di questo approccio è predire la distribu-

[11] In matematica, la configurazione formata da un insieme di punti (*vertici* o *nodi*) e un insieme di linee (*archi*) che uniscono coppie di nodi.

zione di probabilità delle preferenze dell'utente in oggetto su di un articolo, dati gli attributi dell'articolo, le preferenze dell'utente e l'insieme di valutazioni dei suoi amici. Si assume che queste tre informazioni siano indipendenti fra loro. Nonostante questa assunzione possa sembrar semplificare la correlazione, è stato dimostrato come il modello *Naive Bayes* sia risultato efficiente in molte applicazioni.[9] Fatte queste assunzioni la probabilità può essere definita come

$$Pr(\mathcal{R}_{ui} = k|\mathcal{A}' = a'_i, \mathcal{A} = a_u, \{\mathcal{R}_{vi} = r_{vi} : \forall v \in U(i) \cap N(u)\})$$

dove $Pr(\mathcal{R}_u = k|\mathcal{A}' = a'_i)$ è la probabilità condizionata che l'utente u dia una valutazione k ad un articolo che ha le stesse caratteristiche dell'articolo i. Non sempre è presente un insieme di caratteristiche, per questo motivo, in quel caso, si approssima con $Pr(\mathcal{R}_u = k)$. Prendendo in considerazione quindi il comportamento dell'utente, per esempio quanto è generosa con le valutazioni una persona.
$Pr(\mathcal{R}_i = k|\mathcal{A}' = a_u)$ è la probabilità che l'articolo i riceva una valutazione k da un utente le cui preferenze sono le stesse di un utente u. Nel caso in cui le preferenze dell'utente non siano presenti, si approssima con $Pr(\mathcal{R}_i = k)$, prendendo in considerazione criteri generali di accettazione.
Infine $Pr(\mathcal{R}_{ui} = k|\{\mathcal{R}_{vi} = r_{vi} : \forall v \in U(i) \cap N(u)\})$ è la probabilità che l'utente u dia una valutazione k all'articolo i date le valutazioni degli amici sullo stesso articolo. Questa è la parte in cui l'influenza degli amici viene considerata. Il sistema deve apprendere la correlazione presente fra i due collegamenti.
Nelle sezioni successive sarà dimostrato come collegamenti vicini siano più propensi a dare valutazioni simili rispetto a collegamenti lontani. Per questo motivo la funzione di valutazione viene ridotta al risultato degli utenti, con un errore ε applicato.

$$\mathcal{R}_{ui} = \mathcal{R}_{vi} + \varepsilon_{uv}, \quad i \in I(u) \cap I(v), v \in N(u) \cap U(i)$$

La valutazione risulta quindi dipendente dallo scostamento delle preferenze fra due amici.

Per esempio, prendendo in considerazione la correlazione delle preferenze di due utenti u e v, come riportato in tabella 3.1, si ottiene che $\mathcal{R}_{uI_6} = 4$ risulta 40% e $\mathcal{R}_{uI_6} = 3$ risulta 60%.

	u	v
I_1	5	5
I_2	3	4
I_3	4	4
I_4	2	3
I_5	4	5
I_6	?	4

Tabella 3.1: Esempio di immediate friend inference.

Nel caso in cui un utente abbia più amici che hanno valutato lo stesso articolo allora il risultato è la produttoria normalizzata, tenente conto dell'errore, delle preferenze fra ogni coppia di amici

$$Pr(\mathcal{R}_{ui} = k | \{\mathcal{R}_{vi} = r_{vi} : \forall v \in U(i) \cap N(u)\}) = \frac{1}{Z} \prod_{v} \frac{1}{Z_v}(k - r_{vi})$$

dove Z_v è la costante di normalizzazione con l'utente v.

3.1.2 Distant Friend Inference

Considerando l'enorme mole di articoli che possono essere valutati, non è sempre detto che un utente abbia un amico che espresso una preferenza verso un determinato prodotto. Per risolvere questo problema si è introdotto il concetto di *distant friend inference*.
Considerando due amici u e v ed un articolo i per cui si cerca di predire la valutazione di $\mathcal{R}_{ui}$ sapendo che non esiste $\mathcal{R}_{vi}$, si cerca un amico w di v che ha espresso tale preferenza calcolando in maniera iterativa la immediate friend inference prima su v e poi su u.

La *classificazione iterativa* è una tecnica di approssimazione per classificare entità relazionali. Essa si basa sul fatto che le entità siano collegate fra loro. Stimare la classificazione di un'entità spesso dipende dalle stime della classificazione dei vicini. A differenza del tradizionale *data mining*[12] che le considera indipendenti ed uniformemente distribuite, classificandole una per una, l'approccio iterativo classifica tutte le entità simultaneamente, in quanto correlate. Questo tipo di approccio è stato utilizzato per classificare, con un ragionevole tasso di successo, profili di compagnie, documenti ipertestuali e email.[10]
L'algoritmo originale prediceva la valutazione di tutti gli utenti contemporaneamente. Tuttavia considerando l'enorme quantità di utenti presenti sui social network, si utilizza un sottoinsieme di utenti $\mathcal{N}$ che include gli utenti in esame per l'oggetto i e i loro amici stretti. Ad ogni iterazione si genera un ordinamento casuale degli utenti $\mathcal{O}$. Per ogni utente $u \in \mathcal{O}$, se u non ha un amico che fa parte dell'insieme di utenti che hanno una preferenza o una stima di preferenza per l'articolo $U(i)$ allora si salta, altrimenti si effettua una immediate friend inference.

3.2 Social Influence e Previsioni per Gruppi

L'approccio proposto nella sezione precedente migliora le performance del sistema, tuttavia esso non tiene in considerazione l'influenza sociale degli amici di un utente, che varia da persona a persona. Inoltre i recommender system sono generalmente utilizzati per consigliare articoli ad un singolo utente. Ci sono, però, alcune decisioni di acquisto che non dipendono dal singolo individuo ma dal gruppo sociale di cui fa parte. Alcuni esempi sono la scelta di un ristorante o la scelta di un film da vedere durante un'uscita collettiva. A differenze delle decisioni per il singolo utente, le preferenze di

[12]Insieme di tecniche e metodologie che hanno per oggetto l'estrazione di informazioni utili da grandi quantità di dati (es. database, datawarehouse ecc...), attraverso metodi automatici o semi-automatici e l'utilizzo scientifico, aziendale/industriale o operativo delle stesse.

un gruppo non dipendono solo dai gusti degli utenti che lo compongono ma anche dalle opinioni di altri utenti.[11]
In questo paradigma si propongo due approcci per risolvere questi problemi: *social contagion*, che ha trovato riscontri positivi nell'ambito del viral marketing, e *social influence network theory*, per modellare la formazione di opinioni di un gruppo di utenti sotto influenze interpersonali.[12]
Tipicamente lo scenario che si presenta è composta da una lista di m utenti $\mathcal{U} = \{u_1, u_2, \ldots, u_m\}$ e una lista di n articoli $\mathcal{I} = \{i_1, i_2, \ldots, i_n\}$. Ogni utente u_i possiede una lista di articoli I_u dove l'utente ha espresso preferenze o il valore della preferenza può essere ricavato. Questi dati formano una matrice $m \times n$ di valutazioni. Un social network può essere definito come un grafo indiretto $\mathcal{G} = (\mathcal{U}, \mathcal{E})$ dove $\mathcal{U}$ è l'insieme dei nodi, ovvero degli utenti, e $\mathcal{E}$ è l'insieme degli archi, ovvero le relazioni sociali fra utenti. Definisce un insieme dei vicini $N(u)$, ovvero gli amici dell'utente u, tale per cui dati due utenti $u, v \in \mathcal{U}$, si ha che $v \in N(u), u \in N(v)$.

3.2.1 Social Contagion Model

Il flusso di informazioni in un social network può essere visto come la diffusione di un'epidemia. Si utilizza un modello *linear threshold social contagion*[13] per effettuare previsioni sulla base della natura della diffusione e delle preferenze dell'utente. Verranno approfonditi due approcci: un *Simple Binary Rating Case*, in cui le preferenze sono definite in termini di "mi piace" e "non mi piace", e un *General Case*.

Simple Binary Rating Case In questo caso si possono avere 3 stati presenti: *like*, *dislike* e *inactive*. Un nodo v è influenzato da ogni vicino w in base ad un fattore di influenza $b_{v,w} \in [0, 1]$ tale che $\sum_{w \in N(v)} b_{v,w} = 1$. Questo valore dipende dalla fiducia e dalla frequenza di comunicazioni interpersonali. Nel caso in cui non fossimo a conoscenza di queste informazioni, si può par-

tizionare in maniera casuale l'influenza in base ai vicini, in modo da simulare la casualità nella cascata del flusso di informazioni. Il processo poi procede scegliendo una soglia $\theta_{v,i} \in [0,1]$ per ogni utente v e articolo i, che rappresenta la frazione pesata dei vicini che devono diventare attivi(esprimere una preferenza), affinché l'utente v diventi attivo a sua volta. Definito questo e un insieme iniziale di nodi A_i, ovvero gli utenti con una preferenza per l'articolo i, l'algoritmo inizia in maniera discreta ad intervalli. Ad ogni intervallo di tempo il nodo diventa attivo se la frazione di vicini attivi supera il valore di soglia:

$$\left| \sum_{w \in N(v)} b_{vw} \cdot S_{w,i} \right| \geq \theta_{v,i}$$

dove $S_{w,i}$ è lo stato del nodo w riguardo l'articolo i.

$$\theta_{v,i} = \begin{cases} P^{cf}_{v,i}, & \text{se } P^{cf}_{v,i} \cdot \left(\sum_{w \in N(v)} b_{vw} \cdot S_{w,i} \right) < 0 \\ min\{P^{cf}_{v,i}, 1 - P^{cf}_{v,i}\}, & \text{altrimenti} \end{cases}$$

Dove $P^{cf}_{v,i}$ è la previsione del collaborative filtering per l'utente v sull'articolo i.
Si può evincere come ricevere troppi pareri controversi possa portare il nodo a non attivarsi in nessuno stato. Inoltre le opinioni simili alle aspettative dell'utente sono più semplici da accettare rispetto a quelle contrastanti.
Lo stato del nuovo nodo v è deciso da

$$S_{v,i} = sign \left(\sum_{w \in N(v)} b_{vw} \cdot S_{w,i} \right)$$

Questo potrebbe, tuttavia, rendere attivi i nodi nei passi successivi potenzialmente causando un comportamento uniforme a casacata di *like* e *unlike*.
Il processo continua fino a che non sono più possibili nuove attivazioni.
Per l'utente u si ha quindi

$$P_{u,i}^{si} = \begin{cases} S_{u,i} & \text{se } u \text{ è attivo} \\ \text{inactive} & \text{altrimenti} \end{cases}$$

Se l'intervallo di tempo in cui bisogna prendere una scelta commerciale è relativamente ristretto, allora l'unica influenza rilevante è quella degli immediati vicini dell'utente.
Tutto questo può essere visto come un *network cooperative game*[13] avente tre possibili stati: *like*, *dislike* e *inactive*. Visto l'*homophily effect*[14], se due nodi sono connessi, possiedono un incentivo ad avere la stessa opinione.

	like	*dislike*	*inactive*
1	a_{vw}, a_{vw}	$-a_{vw}, -a_{vw}$	$-f(P_{v,i}^{cf}, 1), 0$
-1	$-a_{vw}, -a_{vw}$	a_{vw}, a_{vw}	$-f(P_{v,i}^{cf}, -1), 0$
0	$0, -f(P_{v,i}^{cf}, 1)$	$0, -f(P_{v,i}^{cf}, -1)$	0,0

Tabella 3.2: Esempio di Three-strategy Cooperative Game.

Ogni nodo gioca diverse istanze di questo gioco con tutti i suoi vicini simultaneamente. $a_{vw} > 0$ rappresenta il *payoff* che il nodo v riceve se si coordina con il nodo w, mentre $f(P_{v,i}^{cf}, \cdot)$ rappresenta la penalità che il nodo v riceve se decide di attuare una strategia di attivazione essendo il suo vicino *inactive*. Il risultato del gioco è la somma di tutti i payoff dei giochi individuali che sono rispettivamente:

- $\text{Payoff}_1 = \sum_{w \in N(v)} \left(a_{vw} S_{w,i} - f(P_{v,i}^{cf}, 1) 1_{\{S_{w,i}=0\}} \right)$
- $\text{Payoff}_{-1} = \sum_{w \in N(v)} \left(-a_{vw} S_{w,i} - f(P_{v,i}^{cf}, -1) 1_{\{S_{w,i}=0\}} \right)$

[13]Nella teoria dei giochi, un **cooperative game** (o **coalitional game**) è un gioco con competizione tra gruppi di giocatori ("coalizioni") con la possibilità di applicare un comportamento cooperativo.

[14]Nello studio delle reti, afferma che nodi con attributi simili sono più propensi a collegarsi rispetto a nodi con attributi contrastanti.

- $\text{Payoff}_0 = 0$

Se il nodo v sceglie la strategia che massimizza il payoff e $f(P^{cf}_{v,i}, \cdot)$ è inversamente proporzionale al numero di nodi inattivi allora la soluzione combacia con il *social contagion model*.

General Case Estendendo il caso ad un sistema di valutazione che va da 1 a n, si denota $\mathcal{R} = \{1, 2, \ldots, n\}$ come l'insieme delle possibili valutazioni e $\mathcal{S} = \{\pm 1, \pm 2, \ldots, \pm\lfloor \frac{n}{2} \rfloor\}$ come l'insieme degli stati attivi di differente valore. *active-0* $\in \mathcal{S}$ se n è dispari. Si definisce la funzione $f_r : \mathcal{R} \to \mathcal{S}$

$$f_r(r) = r - \bar{r}$$

dove

$$\bar{r} = \begin{cases} \lceil \frac{n}{2} \rceil & \text{se } n \text{ è dispari} \\ \frac{n}{2} & \text{se } n \text{ è pari e } r > \frac{n}{2} \\ \frac{n}{2} + 1 & \text{se } n \text{ è pari e } r \leq \frac{n}{2} \end{cases}$$

Ad ogni intervallo di tempo si ha che

$$S_{v,i} = \begin{cases} \text{inactive-0} & se \left| \sum_{\substack{w \in N(v) \\ |S_{w,i}| = S}} b_{vw} \cdot sign(S_{w,i}) \right| < \theta_{v,i} \\ S \cdot sign \left(\sum_{\substack{w \in N(v) \\ |S_{w,i}| = S}} b_{vw} \cdot S_{w,i} \right) & \text{altrimenti} \end{cases}$$

In questo caso si sta implicitamente assumendo che lo stato medio *active-0* non fornisca alcuna informazione rilevante e che quindi non sia in grado di influenzare la valutazione del nodo. La previsione su un utente u risulta quindi essere

$$P^{si}_{u,i} = \begin{cases} f^{-1}(S_{u,i}) & \text{se } u \text{ è attivo} \\ \textit{inactive-0} & \text{altrimenti} \end{cases}$$

3.2.2 Social Influence Model

Questo modello serve per predire le preferenze di un gruppo di utenti. Esso differisce dalla semplice aggregazione delle previsioni per ogni utente. La *social influence theory*[14, 15] descrive come una rete di influenze interpersonali entrano nel processo di formazione dell'opinione di un utente. Dati un gruppo di n utenti, un articolo i ad un istante di tempo t, si ha che

$$P_i^{(t)} = AWP_i^{(t-1)} + (I - A)P_i^{(1)}$$

con $P_i^{(t)}$ che è un vettore lungo n di opinioni degli utenti per l'articolo i al tempo t. $W = [w_{uv}]$ è una matrice $n \times n$ di influenze interpersonali con $0 < w_{uv} \leq 1, \sum_j^n w_{uv} = 1$ e $A = diag(\alpha_{11}, \alpha_{22}, \ldots, \alpha_{nn})$ è una matrice diagonale $n \times n$ di utenti, la cui opinione dell'articolo i è influenzabile, con $0 < \alpha_{uu} \leq 1$.
Quando $A = 0$ significa che nessun membro del gruppo è disposto a cambiare la propria opinione in base ai gusti degli altri membri. Questa è l'assunzione base che viene fata dai recommender system tradizionali. Tuttavia, in un gruppo, gli individui tendono a coordinare le proprie decisioni anche senza raggiungere un consenso.[16] Considerando questo processo di convergenza che porta ad un equilibrio, l'equazione diventa

$$P_i^{(\infty)} = AWP_i^{(\infty)} + (I - A)P_i^{(1)}$$

se $I - AW$ è una matrice non singolare[15], allora

$$P_i^{(\infty)} = VP_i^{(1)}$$

dove

$$V = (I - AW)^{-1}(I - A)$$

[15]Una matrice quadrata è detta *non singolare* o *regolare* se il suo determinante è diverso da 0 e quindi esiste la sua inversa.

3.3 Similarity Metrics nei Social Network

Nei social network le communities sono una componente fondamentale. Come già trattato in precedenza, le raccomandazioni per le communities sono state luogo di forte interesse. La precisione di queste raccomandazioni, inoltre, risulta molto più alta se gli utenti condividono molti interessi. Al giorno d'oggi è possibile identificarne miliardi all'interno di questi sistemi. Non tutte le communities, però, sono adatte per ottenere buoni risultati. Per poter identificare quelle ottimali occorre quindi utilizzare una metrica che possa stimare il grado di similarità degli interessi degli utenti al suo interno. A differenza di quelle già presenti, che si focalizzano sul grado di similarità fra due utenti, questa deve essere in grado di valutare un gruppo più ampio. Al tempo stesso, per trovare un'applicazione nel mondo reale, deve essere efficiente da calcolare, in modo che possa in breve tempo analizzare l'enorme quantità di dati presenti.
Per questo motivo è stato introdotto il *community similarity degree.*[17] Questo indicatore assume valori reali in un intervallo $[0, 1]$, dove 0 indica che la community non condivide alcun interesse e 1 se gli utenti presentano esattamente gli stessi interessi.

3.3.1 Community Similarity Degree

Si definisca *fan* un utente che esprime gradimento verso un determinato interesse. L'aggregazione di tutti gli interessi degli utenti forma un'insieme di interessi della community.
Questa metrica deve essere in grado di valutare le seguenti intuizioni:

- Se tutti gli utenti della community possiedono gli stessi interessi allora il grado di similarità è massimo.

- Se due utenti qualunque non hanno almeno un interesse in comune, il grado di similarità è minimo.

- Se esiste un solo interesse distinto nella community, allora più fan possiede quell'interesse, maggiore sarà il grado di similarità.
- Se esiste un certo numero di interessi distinti nella community, la similiratà dovrebbe essere tanto elevata, quanto maggiore è il numero totale di fan per ogni interesse distinto.

Si definisce formalmente una community come $c = \{\mathcal{U}_c, \mathcal{R}_c\}$, dove $\mathcal{U}_c$ rappresenta l'insieme degli utenti appartenenti alla community e $\mathcal{R}_c$ l'insieme degli interessi degli utenti. Si definisce il numero utenti come $\mathcal{N}_u(c)$ e il numero totale di interessi come $\mathcal{N}_r(c)$. Per ogni interesse distinto $r \in \mathcal{R}_c$, definiamo il numero di fan per quell'interesse come $p(r)$. Infine si definisce il peso della community come somma di tutti i fan di ogni interesse distinto $W(c) = \sum_{r \in \mathcal{R}_c} p(r)$. Seguendo le assunzioni fatte prima si ottengono i seguenti criteri:

- In base al **primo** punto si ha che $CSD(c) = 1, W(c) = \sum_{r \in \mathcal{R}_c} p(r) = \sum_{r \in \mathcal{R}_c} \mathcal{N}_u(c) = \mathcal{N}_r(c) \times \mathcal{N}_u(c)$.
- In base al **secondo** punto si ha che $CSD(c) = 0, W(c) = \sum_{r \in \mathcal{R}_c} p(r) = \sum_{r \in \mathcal{R}_c} 1 = \mathcal{N}_r(c)$.
- In base al **terzo** e **quarto** punto, date due communities c_1, c_2 con lo stesso numero di utenti $\mathcal{N}_u(c_1) = \mathcal{N}_u(c_2) = \mathcal{N}_u$ e lo stesso numero di interessi distinti $\mathcal{N}_r(c_1) = \mathcal{N}_r(c_2) = \mathcal{N}_r$, allora la community con il peso maggiore è quella con il CSD maggiore $1 \geq CSD(c_1) > CSD(c_2) \geq 0$, $\mathcal{N}_r \times \mathcal{N}_u \geq W(c_1) > W(c_2) \geq \mathcal{N}_r$.

Definiti i criteri di cui sopra si ha che

$$CSD(c) = \frac{W(c) - \mathcal{N}_r(c)}{\mathcal{N}_u(c) \times \mathcal{N}_r(c) - \mathcal{N}_r(c)} = \frac{W(c)/\mathcal{N}_r(c) - 1}{\mathcal{N}_u(c) - 1}$$

Si nota quindi come il CSD definisce la similarità basandosi sul rapporto fra la popolarità media degli interessi($W(c)/\mathcal{N}_r(c)$) e il numero totale di membri

nella community($\mathcal{N}_u(c)$), variando il suo valore fra 0 e 1.
Si può notare, inoltre, come la complessità nel calcolo sia pari a $O(n)$, dove n è il numero di utenti presenti nella community.

Parte II

Analisi e Valutazione

Capitolo 4

Casi di Studio

In questo capitolo verranno effettuati degli esperimenti empirici, per ogni modello e metrica descritti nel capitolo precedente, su insiemi di dati di varia natura e dimensione,

4.1 Social Network-based Recommender System

Per analizzare questo modello in base a dati reali, essi sono stati generati prendendo quelli presenti nella rete sociale `Yelp`.[16]
Le relazioni sociali, in questa piattaforma, sono rappresentate dalle amicizie che risultano essere di mutua accettazione. In particolare si è scelto di recuperare i dati relativi ai ristoranti. Per questo sono stati recuperati i dati dell'area di Los Angeles, recuperando 4 152 record. Da questi sono state, inoltre, recuperati tutti i profili degli utenti che hanno lasciato una recensione, ottenendo 9 414 utenti. Da questi utenti si è ricostruita la social network, dove due nodi sono collegati se gli gli utenti possiedono una relazione di amicizia. Vista la natura delle relazioni in `Yelp` la struttura che è stata generata è un grafo indiretto.

[16]Yelp.com è un servizio di ricerca locale di recensioni degli utenti. (https://en.wikipedia.org/wiki/Yelp)

Dopo uno studio preliminare, si è rilevato che il numero totale di recensioni ammonta a 55 801. Quindi si ha che in media un utente esegue 5.93 recensioni e ogni ristorante possiede una media di 13.44 recensioni. Inoltre ogni utente possiede in media 8.18 amici.

4.1.1 Correlazione delle Recensioni per gli Immediate Friends

La prima cosa che si vuole scoprire qual'è la probabilità che se un utente recensisce un ristorante anche uno fra i suoi *immediate friends* lo abbia recensito a sua volta.
Contando, per ogni utente, la percentuale di ristoranti recensiti anche da almeno un suo amico, si ottiene il valore medio di 18.6%. Come verifica si calcola la stessa probabilità assumendo che gli utenti effettuino recensioni in maniera casuale ed indipendente. Presupponendo in un social network la presenza di n utenti, ognuno dei quali possiede q amici stretti ed un ristorante con m recensori(tra cui lo stesso utente), la probabilità che almeno un amico stretto compaia nella lista dei recensori è

$$1 - \frac{\binom{n-q-1}{m-1}}{\binom{n-1}{m-1}}$$

Calcolando questo valore per ogni utente ed ogni ristorante da lui recensito si ottiene una probabilità media pari al 3.7%, che comparata a quella trovata in precedenza, dimostra come gli amici stretti non recensiscano in maniera casuale i ristoranti.
Estendendo questo caso anche agli amici più lontani, ad al massimo due *hops*[17] di distanza, che hanno recensito lo stesso ristorante si ottiene una probabilità pari al 45.2%. Questo serve anche per gestire il caso in cui nessun amico stretto abbia recensito il ristorante, ma un amico di un amico si. Senza

[17]Con *hop* si intende la distanza da un nodo ad un altro all'interno di un grafo. Tipicamente rappresenta il numero di archi da attraversare per raggiungere il nodo A partendo dal nodo B.

questo caso preso in considerazione si sarebbero potute effettuare previsione per una piccola parte del bacino di utenza. Se effettuiamo lo stesso calcolo di probabilità in maniera casuale, si ottiene una probabilità pari al 34.2%, confermando la teoria secondo cui utenti collegati fra loro tendono a recensire gli stessi ristoranti.
Infine, per confermare ulteriormente la tendenza, si calcola il numero medio di ristoranti co-recensiti fra due qualsiasi amici stretti e due qualsiasi utenti si ottengono rispettivamente 0.85 e 0.03.

4.1.2 Correlazione dei Punteggi per gli Immediate Friends

Al fine di verificare che amici stretti assegnano punteggi simili, si è calcolata la differenza di valutazione media, per lo stesso ristorante, tra recensori che sono amici e recensori che non sono amici. I risultati ottenuti sono:

- Una differenza media di valutazione di 0.88 con una deviazione standard pari a 0.89 per gli utenti **amici**.
- Una differenza media di valutazione di 1.05 con una deviazione standard pari a 0.98 per utenti **non amici**.

Questi risultati mostrano come la tendenza degli utenti di assegnare punteggi simili agli amici sia confermata.

4.2 Community Similarity Degree

In questa sezione verrà introdotto il set di dati su cui è stato effettuato l'esperimento e quattro differenti tipologie di community che sono state create partendo da esso.

4.2.1 Descrizione dei Dati

Le informazioni utilizzate per l'esperimento sono state recuperate dal social network `Facebook`, partendo da un utente e recuperando le informazioni

del profilo, gli interessi, suddivisi in cinque categorie distinte, e la lista degli amici. Da qui si è effettuato lo stesso procedimento per ogni amico e così via. Le categorie prese in considerazione sono *programmi televisivi, libri, musica, film e giochi*.
In totale sono stati recuperati 208 634 utenti e 542 597 interessi distinti dalle categorie sopra citate. In media un utente presenta 11 interessi distinti; il 12% degli utenti possiede un solo interesse mentre il 5% ne possiede più di 100. Sono stati recuperati utenti da 150 nazioni e circa 9 000 città. Gli utenti sono collegati ad amici in un intervallo che va da qualche decina a 5 000. L'85% degli interessi hanno meno di 10 fan, 10 000 ne possiedo circa 100 e 1 000 circa un migliaio di fan. I 100 interessi più popolari contano circa 8 000 fan ognuno.

4.2.2 Descrizione delle Community

Sono state costruite quattro differenti tipi di communities in modo da poter valutare l'efficienza e la correttezza della metrica nella selezione della community più adatta.

Community Friend-based Questo tipo di communities sono formate da un utente e tutti i suoi amici che hanno almeno un interesse. Sono state rilevate 865 communities *friend-based*. Esse sono quelle tipicamente usate dai recommender system nei social network.
Come si potrà notare successivamente, la dimensione della community ha un forte impatto sul CSD. Per questo motivo si è cercato di uniformare tutte le tipologie di communities in modo tale che avessero la stessa distribuzione di questa(Figura 4.1).

Community Random-based Come si intuisce dal nome, esse sono communities costruite prendendo n utenti casuali dal set di dati, seguendo la distribuzione sopra citata.

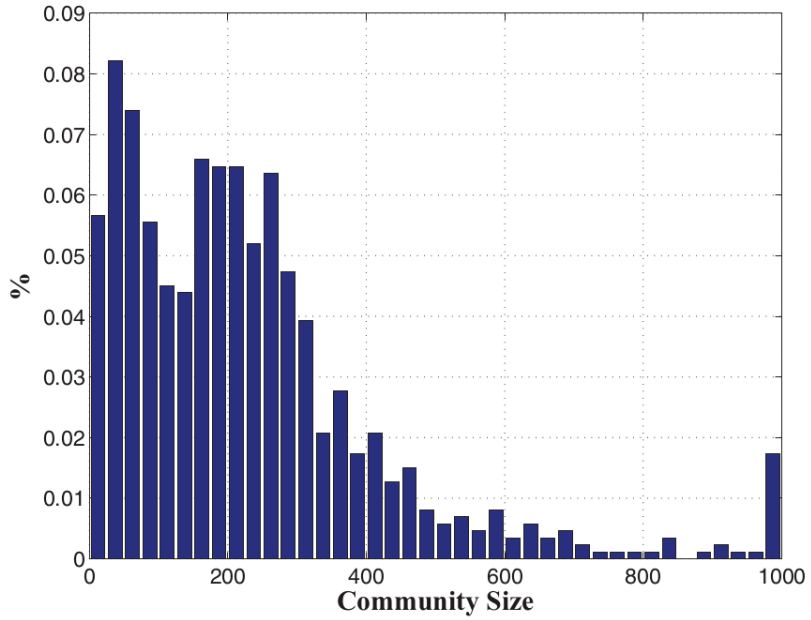

Figura 4.1: Distribuzione della dimensione delle communities rilevate dal set di dati.

Community Interest-based Queste communities sono composte da utenti che hanno, almeno, un interesse in comune. L'interesse su cui è basata la costruzione è stato scelto in maniera casuale, sempre rispettando il fatto che ci fossero almeno n fans.

Community Location-based Questo tipo di community sono composte da persone che si trovano nella stessa città. Allo stesso modo di prima sono state costruite 865 communities, rispettando la distribuzione sopra citata.

4.2.3 Caratterizzazione del CSD

Al fine di studiare la caratterizzazione del CSD, sono stati effettuati degli esperimenti sulle communities friend-based. In modo da valutare la sua variazione in base al numero degli utenti ($\mathcal{N}_u(c)$), al numero degli interessi ($\mathcal{N}_r(c)$) e al peso della community (W(c)).

Influenza del numero di utenti Al fine di studiare questo effetto sono state raggruppate le communities di dimensione simile in contenitori. Per effettuare questo raggruppamento sono stati utilizzati l'*equal width binning* e l'*equal frequency binning.*[20]
Nell'equal width binning si considerano dieci contenitori $bin(b)$ che includono tutte le communities la cui dimensione è compresa nell'intervallo $((b-1)*100, b*100]$ con $b \in [1,9]$. Il valore $b=1$ comprende communities con dimensione $[2,100]$ in quanto non è interessante prendere communities formate da una sola persona. Per $b=9$ si hanno tutte le communities con dimensione superiore a 900.
Nell'equal frequency binning, si generano sempre dieci contenitori, contenenti ognuno il 10% di tutte le communities.

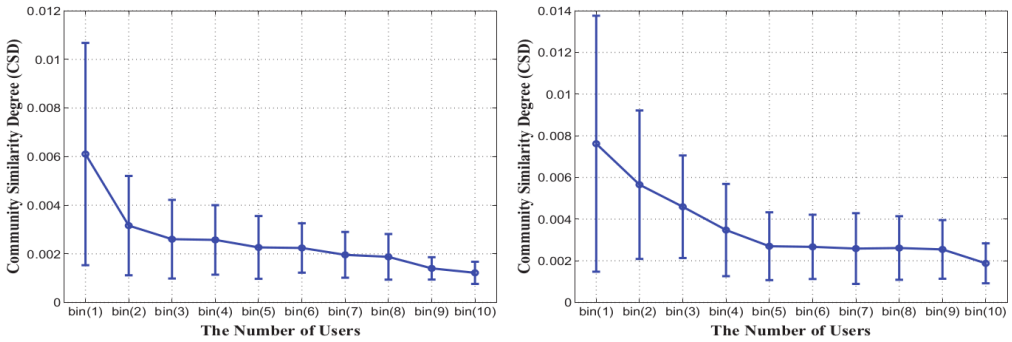

Figura 4.2: CSD e dimensione della Community in base al numero di utenti.

Come si può notare dalla figura 4.2, il valore del CSD diminuisce con l'aumentare della dimensione della community. Il crollo maggiore si ha passando dal $bin(1)$ al $bin(2)$. Un comportamento analogo si ha per entrambi i metodi di raggruppamento. Questo è dovuto al fatto che con l'aumentare del numero di utenti all'interno della community, aumenta anche l'insieme di interessi distinti. Le communities più grandi portano molta diversità, in termini di utenti ed interessi, che conduce ad un valore del CSD inferiore.

Influenza del numero di interessi In questo caso si ripete la metodologia precedentemente descritta applicandola al numero di interessi. Analogamente vengono creati dieci contenitori $bin(v)$ aventi un intervallo nel numero di interessi pari a $((b-1)*1\,000, b*1\,000]$ con $b \in [1,9]$ e in tranche del 10% per quanto riguarda il secondo grafico.

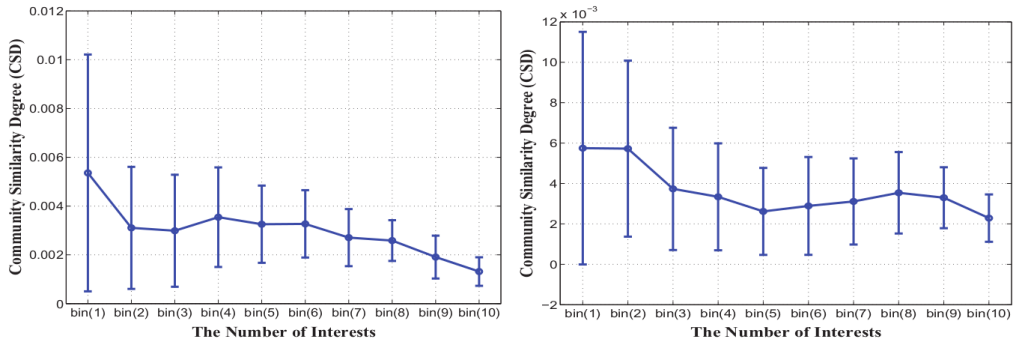

Figura 4.3: CSD e dimensione della Community in base al numero di interessi.

Come si può vedere dalla figura 4.3, il trend generale del CSD è discendente, tuttavia si può notare un aumento del CSD medio quando si passa da $bin(3)$ a $bin(4)$. Il cambiamento della deviazione standard, inoltre, mostra come la variabilità del CSD diminuisce all'aumentare del numero di interessi. Ci sono due aspetti fondamentali da prendere in considerazione in questo caso. Con l'aumentare del numero di interessi, aumenta la diversità. Questo porta, inevitabilmente, ad una riduzione del CSD. Al tempo stesso, questo aumento porta anche ad un aumento della popolarità di tutti gli interessi all'interno della community stessa. Questo può portare ad un aumento del CSD a sua volta

Influenza del peso della community Im maniera analoga alle precedenti si è studiato il comportamento al variare del *community weight*. Si è suddiviso il numero totale di communities in contenitori $bin(b)$ in modo tale che $((b-1)*2\,000, b*2\,000]$ con $b \in [1,9]$ e per le frequenze sempre in blocchi del 10%.

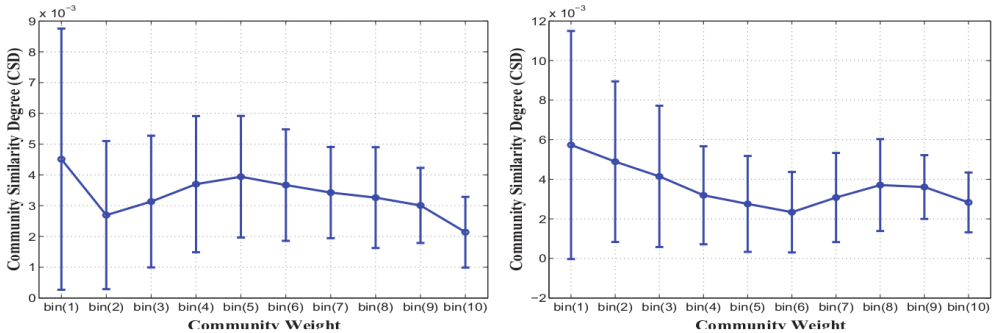

Figura 4.4: CSD e in relazione al peso della Community.

Prendendo in considerazione la definizio del CSD si nota come il comportamento del peso vari rispetto a quello del numero di utenti o del numero di interessi. Questo si può vedere analizzando la formula $\frac{W(c)/\mathcal{N}_r(c)-1}{\mathcal{N}_u(c)-1}$. Da qui si può notare come il valore del CSD sia direttamente proporzionale al peso della community $W(c)$.
Tuttavia, come riporta la figura 4.4, l'incremento nel trend del CSD non è individuabile all'aumentare del peso della community. La somiglianza con la figura 4.3, inoltre, porta a presupporre una correlazione fra il peso $W(c)$ e il numero di interessi $\mathcal{N}_r(c)$.

4.2.4 CSD per i Diversi Tipi di Communities

Si è studiata la *cumulative distribution function(CFD)*[18] per i diversi tipi di communities. Dalla figura 4.5, si nota come i valori del CSD nel mondo reale siano molto lontani dal massimo valore che esso può raggiungere. Tuttavia la interest-based community è quella che ha ottenuto le performance migliori.
In aggiunta si è studiato il CSD in base alla dimensione delle diverse communities (figura 4.6). In maniera simile agli approcci precedentemente descritti,

[18]In statistica, la *cumulative distribution function* di una variabile casuale X, valutata in x è la probabilità che X prenda valori minori o uguali a x.

si sono creati i dieci contenitori in base alla dimensione delle communities. Anche in questo caso la interest-based community rimane quella con i risultati migliori.
Da questo si è notato come questa tipologia di community risulta la più adatta ad effettuare l'esperimento.

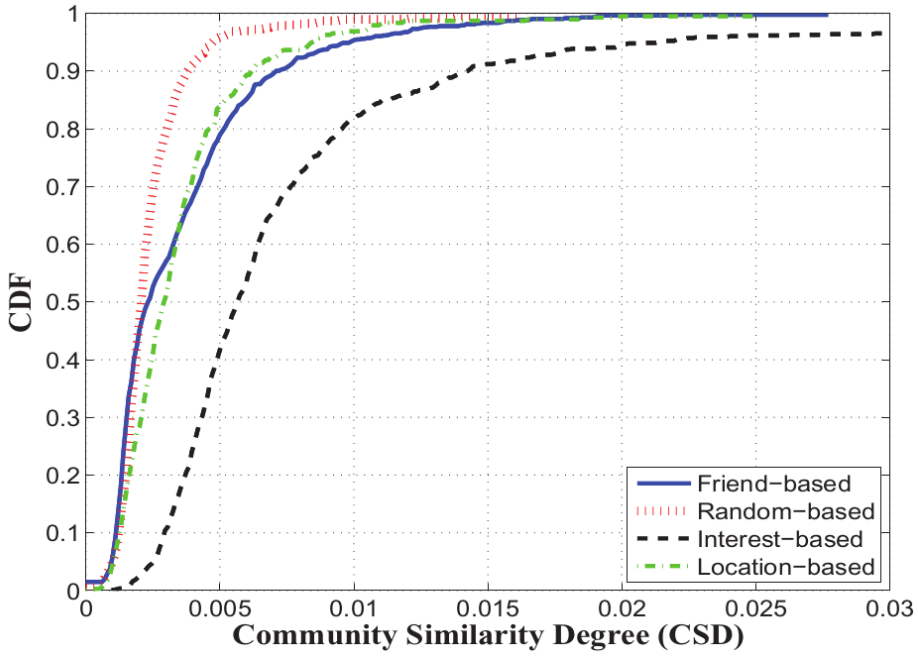

Figura 4.5: Cumulative Distribution Function del CSD per i quattro tipi di communities.

4.2.5 CSD per le Diverse Categorie di Interessi

Si è studiato l'andamento del CSD nelle interest-based communities per quanto riguarda le varie categorie di interessi. Per ogni categoria sono state prese circa 1 000 communities composte da 2 a 500 utenti.

Come si può notare dal grafico a sinistra nella figura 4.7 si è studiato il comportamento della *cumulative distribution function* per il CSD delle varie

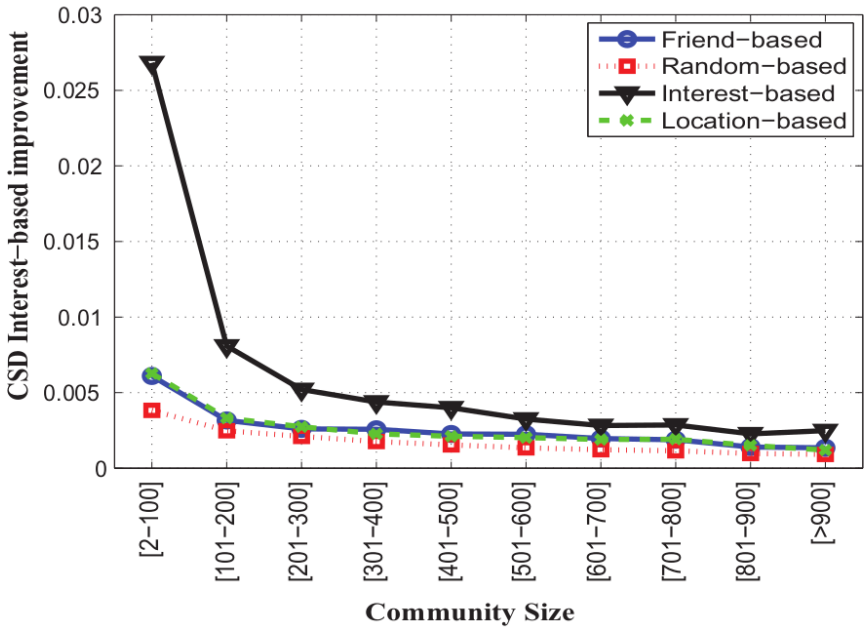

Figura 4.6: Cumulative Distribution Function del CSD per i quattro tipi di communities, in base alla dimensione.

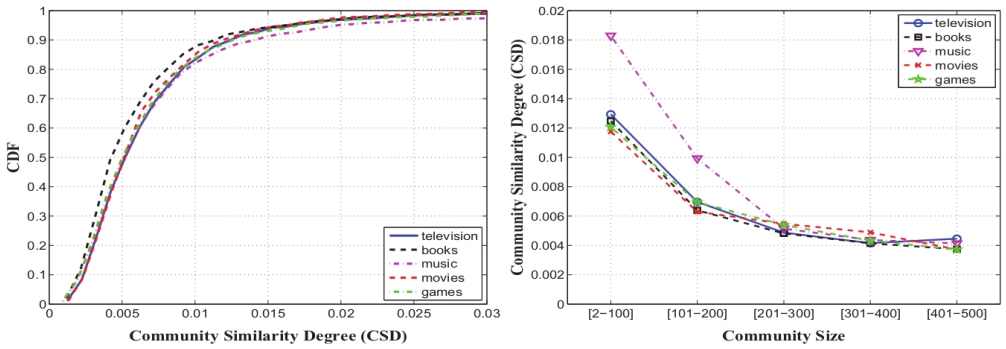

Figura 4.7: Cumulative Distribution Function del CSD e in base alla dimensione, per le categorie di interessi delle interest-based communities.

communities. Da qui si può notare come quasi tutte le categorie possiedono lo stesso comportamento. Nota particolare per l'ambito musicale in cui i valori sono risultati leggermente più alti.

Il grafico a sinistra nella figura 4.7, mostra il CSD medio per le diverse categorie di interessi in base alla dimensione delle communities. Per le communities più piccole(dimensione inferiore a 200) si ha un valore più alto del CSD per quanto riguarda la categoria "musica". Per le restanti communities tutte le categorie presentano il medesimo comportamento.

4.2.6 Osservazioni

Si è notato come la maggior parte delle communities emulate possieda un CSD molto basso. Questo valore indica che i membri, spesso, condividano solo una piccola parte di tutti gli interessi presenti. Inoltre è lecito supporre che, all'aumentare del numero di utenti e di differenti interessi, questo valore sia ancora più basso.

Capitolo 5

Analisi dei Risultati

In questo capitolo verranno analizzati i risultati ottenuti dagli esperimenti empirici e verranno effettuate alcune considerazioni.

5.1 Social Network-based Recommender System

Per valutare le performance del *SNRS* sul set di *Yelp* ci si è focalizzati sul problema della *prediction accuracy*,*data sparsity* e *cold-start*. Come attributo dell'articolo si è scelto il range di prezzo del ristorante. Siccome su Yelp non vi erano attributi dell'utente interessanti si è sostituita la formula $Pr(\mathcal{R}_i = k|\mathcal{A} = a_u)$ con $Pr(\mathcal{R}_i = k)$, quando si è stimata l'accettazione. Infine si è stabilita una soglia di tre ristoranti co-recensiti da due amici stretti per essere presi in considerazione.

5.1.1 Criteri di Comparazione

Al fine di valutare le performance sono stati utilizzati i seguenti metodi di comparazione.

Friend Average (FA) Rappresenta la valutazione media degli amici stretti di un utente sullo stesso articolo.

Weighted Friends (WVF) A differenza del FA, questa metrica considera che ogni amico stretto di un utente possieda un differente impatto (o peso) sull'utente stesso. Quindi la probabilità della previsione sulla valutazione dell'utente è proporzionale al peso accumulato in ogni valore

$$Pr(\mathcal{R}_{ui} = r_{ui} | \{\mathcal{R}_{vi} = r_{vi} : \forall v \in U(i) \cap N(i)\}) = \frac{1}{z} \sum_{v} w(u,v)\delta(k, r_{vi})$$

dove z è una costante di normalizzazione. $w(u,v)$ è il peso fra l'utente u e l'utente v. $\delta(k, r_{vi})$ è una funzione che ritorna 1 se e solo se $r_{vi} = k$, altrimenti ritorna 0. Essenzialmente questo metodo è molto simile al *relational-neighbor classifier* che si è dimostrato performante nel classificare insiemi di dati relazionali come citazioni e film.[18]

Naive Bayes (NB) I social network possono essere visti come reti Bayesiane. [19] In questo caso si è implementato un particolare tipo di rete Bayesiana detto classificatore *Naive Bayes*. In particolare esso assume che la valutazione di un utente influenzi le valutazioni degli amici stretti. Valutazioni che avvengono in maniera indipendente fra loro.
Data la valutazione degli amici stretti su un articolo i, si calcola la probabilità condizionata come segue

$$Pr(\mathcal{R}_{ui} = r_{ui} | \{\mathcal{R}_{vi} = r_{vi} : \forall v \in U(i) \cap N(i)\}) = \frac{1}{z} Pr(\mathcal{R}_u = k) \prod_{v} Pr(\mathcal{R}_v = r_{vi} | \mathcal{R}_u = k)$$

dove z è una costante di normalizzazione, $Pr(\mathcal{R}_u = k)$ è la distribuzione delle valutazioni definita in precedenza, $Pr(\mathcal{R}_v = r_{vi} | \mathcal{R}_u = k)$ è la probabilità condizionata che la valutazione un amico stretto v sia uguale a r_{vi} se la valutazione di u è uguale a k. Queste due probabilità sono state stimate contando le valutazioni delle coppie di amici stretti.

Collaborative Filtering (CF) In questo caso si è implementato l'algoritmo standard come descritto in precedenza con $k = 20$.

5.1.2 Accuratezza delle Previsioni e Coverage

L'accuratezza delle previsioni per questo esperimento è stata calcolata utilizzando il *Mean Absolute Error(MAE)*[19]

$$MAE = \frac{\sum_{u,i} |r_{ui} - r'_{ui}|}{l}$$

dove l è il numero di istanze testate. Minore è questo valore, maggiore è l'accuratezza.
Siccome SNRS, FA, WVF, e NB dipendono dalle valutazioni degli amici di un utente su un determinato articolo, nel caso in cui non ci sia nessun amico che abbia valutato lo stesso queste metriche non possono essere utilizzate. Similmente, anche il CF non può effettuare previsioni nel caso in cui non ci siano amici "simili" all'utente in esame. Per questi motivi si è introdotta la *coverage*. Questa metrica rappresenta la percentuale di istanze testate per cui il metodo ha potuto effettuare previsioni.
Come si nota dalla Tabella 5.1, la migliore performance è stata ottenuta dal SNRS in termini di MAE (0.718), mentre il CF ha ottenuto la peggiore (0.871). SNRS ha migliorato del 17.8% le prestazioni a fronte di una riduzione della coverage pari al 12.7%. Si può notare come generalmente la coverage non ha mai raggiunto quota 0.6. Questo è dovuto all'incompletezza dei dati per quanto riguarda tutti gli utenti. A differenza degli altri algoritmi, SNRS è in grado di effettuare una previsione anche dagli amici distanti(amici di amici) nel caso in cui non sia presente una valutazione da parte dell'amico stretto. Questo ha contribuito ad aumentare la coverage.

5.1.3 Data Sparsity

Al fine di valutare il comportamento dell'SNRS al variare della data sparsity, si è suddiviso casualmente il dataset in dieci gruppi. Di questi se ne selezionavano n e i restanti fungevano da training set. Il valore di n controlla la sparsity.

[19]Il MAE è una misura della differenza fra due variabili continue.

	MAE	COVERAGE
SNRS	**0.716**	0.482
FA	0.814	0.228
WVF	0.808	0.228
NB	0.756	0.237
CF	0.871	**0.552**

Tabella 5.1: MAE e Coverage per le varie metriche di valutazione.

Nella Figura 5.1 sono messi a confronto il MAE del SNRS e del CF, variando la percentuale del testing set da 10% a 70%. Vista la natura del dataset, la data sparsity rimane comunque elevata anche con un sottoinsieme piccolo dei dati presi in considerazione. Da qui si può notare come l'errore del SNRS è considerevolmente inferiore rispetto a quello di CF. Si può notare, inoltre, come l'accuratezza della previsione usando il CF sia molto affetta dalla data sparsity.
Nella Figura 5.2 viene confrontata la coverage di entrambi i metodi. Man mano che la data sparsity aumenta, si può notare come la coverage di entrambi diminuisce. Questo comportamento era atteso in base alle considerazioni effettuate prima, tuttavia il trend fornisce ulteriori informazioni sulla differenza di questi due metodi. CF è più performante in presenza di un grande training set, tuttavia la coverage del SNRS diminuisce più lentamente. Questo cambio nel trend è dovuto al fatto che quest'ultimo è in grado di sfruttare le valutazioni degli amici lontani per effettuare la previsione.

5.1.4 Cold-Start

Il *cold-start* è un caso estremo di data sparsity dove un nuovo utente non ha effettuato alcuna valutazione. In questo caso il CF non è in grado di effettuare nessuna previsione in quanto non è in grado di trovare utenti simili. Il SNRS non è in grado di effettuare la previsione solo se l'utente, oltre a non avere valutazioni, non possiede amici. Tuttavia in alcuni casi è possi-

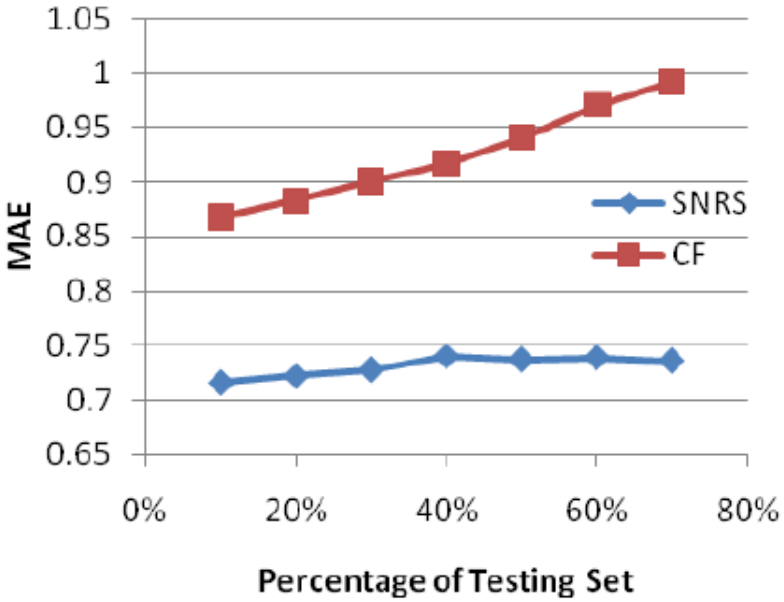

Figura 5.1: MAE per SNRS e CF in base alla data sparsity.

bile effettuare delle previsioni, senza conoscenza pregressa delle preferenze dell'utente, basandosi sulle preferenze dei suoi amici. Per verificare questo comportamento si è svolto un esperimento, basato su tre assunzioni:

1. Non vi è alcuna influenza da parte delle preferenze dell'utente utilizzando una distribuzione uniforme per $Pr(\mathcal{R}_u = k|\mathcal{A}' = a'_i)$.

2. Sono stati utilizzate direttamente le valutazioni degli amici stretti sull'articolo in esame $Pr(\{\mathcal{R}_{vi} = r_{vi} : \forall v \in U(i) \cap N(u)\})$.

3. Ad eccezione di delle preferenze dell'utente in esame, tutte le preferenze degli altri utenti sono note.

Effettuando questo esperimento per ogni utente, il MAE risultante è stato pari a 0.706 con una coverage pari a 0.691. Questi risultati dimostrano che anche in presenza di cold-start le performance del SNRS sono accettabili.

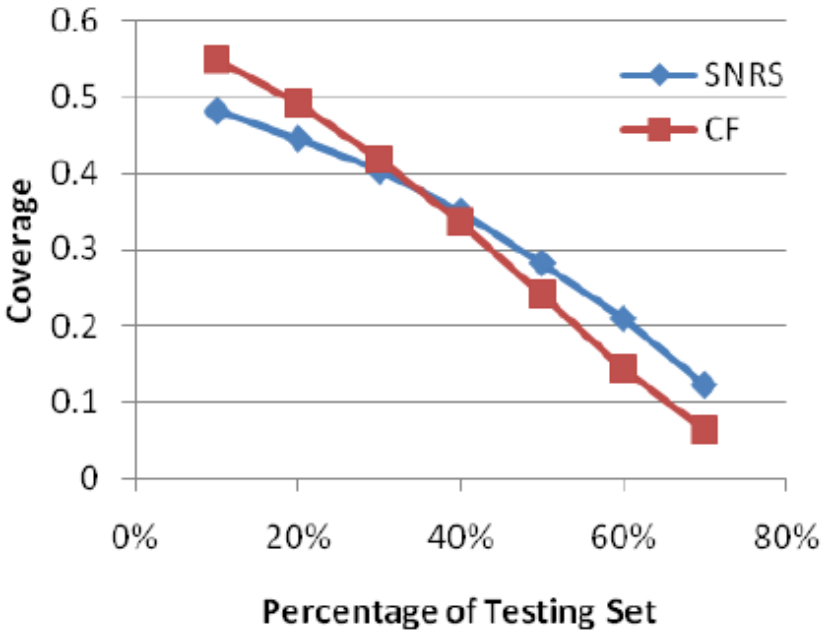

Figura 5.2: Coverage per SNRS e CF in base alla data sparsity.

5.1.5 Distant Friends

Al fine di verificare come l'impatto degli amici lontani influisca sull'accuratezza delle previsioni si è svolto un esperimento rispettivamente con e senza di essi. Come si può notare dalla tabella 5.2, senza l'ausilio degli amici lontani, il MAE si riduce ottenendo previsioni più accurate. Tuttavia la coverage dimezza permettendo di effettuare queste previsioni per circa la metà degli utenti.

	MAE	**COVERAGE**
Con Distant Friends	0.716	**0.482**
Senza Distant Friends	**0.682**	0.237

Tabella 5.2: MAE e Coverage con e senza Distant Friends Inference.

5.2 Community Similarity Degree

In questa sezione verrà valutata l'efficienza e la bontà della metrica proposta. Per fare ciò verrà emulato un recommender system per la community, per consigliare prodotti agli utenti delle communities. Verranno poi ordinate in base al valore del CSD. Ci si aspetta di avere una precisione maggiore per quelle communities il cui CSD è superiore. Al fine di valutare l'efficienza verrà calcolato il costo computazionale per il calcolo del CSD per una community.

5.2.1 Approccio e Metrica per le Recommendation

Si è implementato un recommendation system basato sull'idea di *rank aggregation e collaborative filtering.*[21] Questo approccio consiste in due passi principali: per primo, si calcola una lista di recommandation ranking per ogni utente della community; poi si aggregano le liste di tutti gli individui singoli, tramite euristiche predefinite, generando una ranking list per la community.
Per generare la lista individuale si è utilizzato, prima, l'approccio *item-based top-n recommendation algorithm*[22] per determinare quali articoli consigliare. In particolare si calcola la rilevanza fra due articoli r_i e r_j, in modo che essa sia alta se agli utenti piacciono entrambi. Successivamente, si utilizza il *Borda count aggregation method*[23] per generare le raccomandazioni individuali. Ogni lista individuale comprende gli articoli che hanno un'alta rilevanza con quelli specificati negli interessi del suo profilo. Lo stesso metodo viene utilizzato per aggregare tutti i risultati dei singoli e generare la lista per la community. Infine vengono consigliati i primi k articoli dalla lista a tutti i membri della community.
Al fine di valutare le performance del recommendation system per ogni community sono state definite l'*average precision* e la *mean average precision.* Data una community $\mathcal{U}_c$ e una lista di recommendation ranking, si assuma k elementi consigliati ad un utente $u \in \mathcal{U}_c$. Si definisce la precisione al grado k

per u come

$$P@K(u) = \frac{rel_k(u)}{k}$$

dove $rel_k(u)$ è il numero di articoli che u ha valututato positivamente nelle prime k raccomandazioni. Poi per ogni community si calcola la *average precision*

$$AP@K = \frac{1}{|\mathcal{U}_c|} \sum_{u \in \mathcal{U}_c} P@K(u) = \frac{1}{|\mathcal{U}_c|} \sum_{u \in \mathcal{U}_c} \frac{rel_k(u)}{k}$$

infine di definisce la *mean average precision* per un insieme di communities $\mathcal{C}'$ come

$$MAP@K = \frac{1}{|\mathcal{C}'|} \sum_{c \in \mathcal{C}'} AP@K(c) = \frac{1}{|\mathcal{C}'|} \sum_{c \in \mathcal{C}'} \frac{1}{|\mathcal{U}_c|} \sum_{u \in \mathcal{U}_c} \frac{rel_k(u)}{k}.$$

5.2.2 Valutazione

Al fine di valutare efficienza e bontà della metrica saranno svolti tre esperimenti:

1. Si utilizzerà il CSD per selezionare l'insieme di communities e verrà valutato se quelle con il CSD maggiore ottengono migliori performance per quanto riguarda il recommendation system.

2. In questo esperimento si verificherà se effettivamente se le communities *interest-based* ottengono una precisione maggiore rispetto alle altre tipologie.

3. In questo caso verrà calcolato il tempo di calcolo del CSD per valutarne l'efficienza.

Raccomandazione nelle communities tramite il CSD In questo esperimento si prendono in considerazione i primi 3,5 e 10 articoli da raccomandare ad ogni community, calcolando il corrispondente $AP@K$. Successivamente si genera una lista di raccomandazioni ordinata in base al CSD in ordine decrescente. Succesivamente si ragrruppano le varie communities della lista in blocchi e viene calcolato il $MAP@$Ķ per valutare la precisione degli articoli

consigliati.

Si utilizza $T_{\mathcal{C}[n_1-n_2]}$ per rappresentare l'insieme delle communities nelle prime posizioni da n_1 a n_2. Si utilizza $B_{\mathcal{C}[n_1-n_2]}$ per rappresentare l'insieme delle communities nelle prime posizioni da n_1 a n_2. Ci si aspetta che quelle che si trovano nelle prime posizioni ottengano performance migliori di quelle nelle ultime posizioni.

Communities	$T_{\mathcal{C}[1-100]}$	$T_{\mathcal{C}[101-200]}$	$T_{\mathcal{C}[201-400]}$	*Tutte*	$B_{\mathcal{C}[201-400]}$	$B_{\mathcal{C}[101-200]}$	$B_{\mathcal{C}[1-100]}$
CSD Medio	0.052	0.012	0.009	0.005	0.0013	0.0011	0.0007
MAP@3	0.112	0.089	0.083	0.047	0.022	0.018	0.014
MAP@5	0.110	0.086	0.079	0.045	0.02	0.019	0.014
MAP@10	0.100	0.086	0.078	0.048	0.023	0.021	0.017

Tabella 5.3: MAP@K per CSD

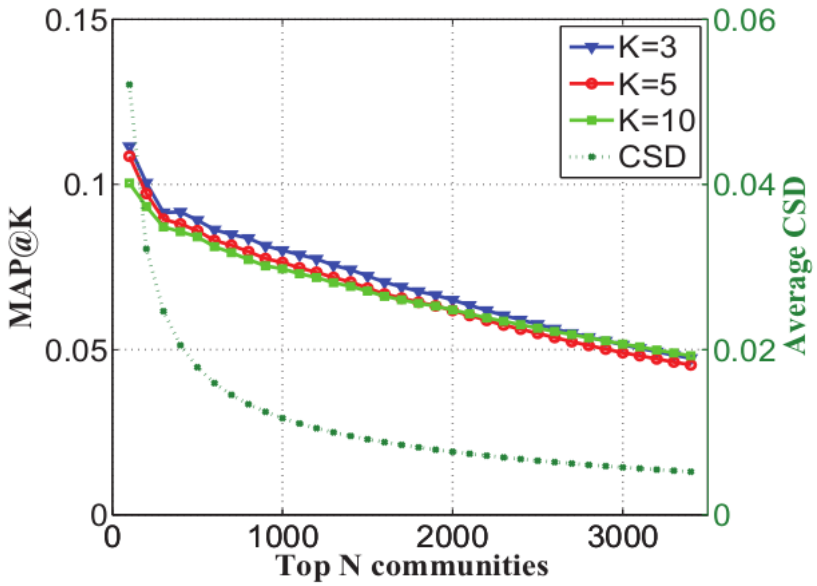

Figura 5.3: MAP@K delle top N communities nella ranking list.

Come si può notare dalla tabella 5.3, i risultati migliori si ottengono in $T_{\mathcal{C}[1-100]}$. Dalla figura 5.3, si nota come il $MAP@K$ si riduce all'aumentare

di N. All'aumentare di N, communities con un CSD inferiore vengono prese in considerazione, perciò anche il CSD medio delle migliori si riduce, provocando una riduzione della precisione delle raccomandazioni.
Prendendo la mediana[20] nella figura 5.4, si nota come selezionare le communities dalle prime posizioni porti maggiori benefici piuttosto che selezionarle casualmente o dalle ultime posizioni.
Questo esperimento ha dimostrato che le communities con un CSD elevato possono apportare un miglioramento delle performance nei recommender systems.

Raccomandazioni per differenti tipologie di communities In questo esperimento si sono valutate le performance del recommender system nelle varie tipologie di communities.

Communities	*Interest*	*Friend*	*Location*	*Random*
CSD Medio	0.0108	0.0036	0.0036	0.025
MAP@3	0.090	0.036	0.033	0.030
MAP@5	0.085	0.034	0.031	0.029
MAP@10	0.085	0.043	0.033	0.030

Tabella 5.4: MAP@K per i differenti tipi di communities.

Dalla tabella 5.4 dalla figura 5.5, si può notare come le interest-based communities siano più performanti rispetto a tutte le altre tipologie. Le *friend-based* non ottengono un buon risultato, come le precedenti, probabilmente a causa del fatto che un utente aggiunge amici conosciuti in ambienti diversi e in modi diversi, portando quindi a non condividere, necessariamente, gli stessi interessi.
Anche in questo caso si è dimostrato come un più alto CSD porti ad un miglioramento delle performance.

[20]In statistica, di una successione finita di valori, il valore intermedio fra gli estremi di tale successione.

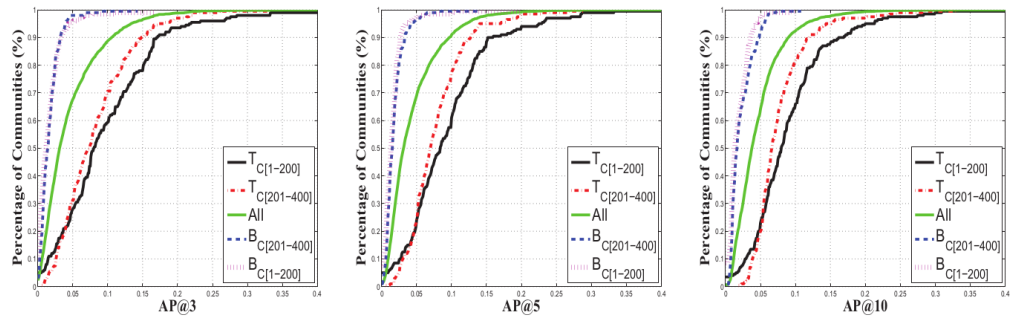

Figura 5.4: CDF of AP@K by CSD.

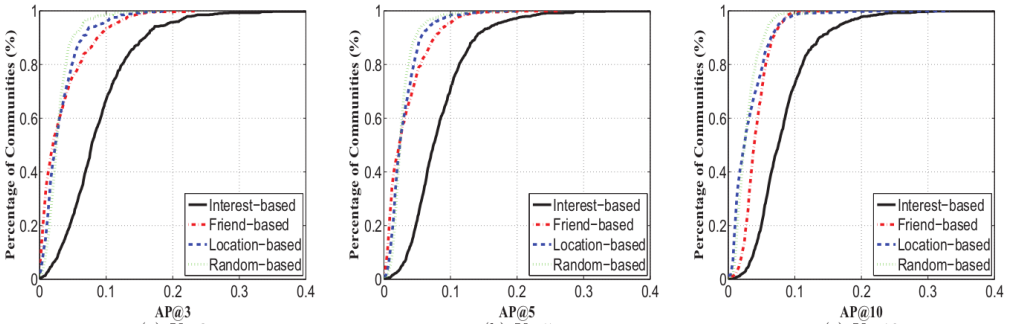

Figura 5.5: CDF of AP@K per i diversi tipi di communities.

Tempo di elaborazione per il CSD Al fine di valutare l'efficienza del CSD, si è registrato il tempo impiegato per effettuare il calcolo su tutte le tipologie di community precedentemente trattate. Per fare questo si è utilizzato un laptop di fascia media. Il tempo medio di calcolo è di circa 2.46ms per community. Questo implica che il calcolo per 1 milione di communities impiegherebbe circa 41 minuti. La figura 5.6 mostra il tempo di calcolo in relazione alla dimensione della community o al numero di interessi. Si può notare come ci vogliano meno di 10ms per calcolare il CSD di una community con 1 500 utenti o 20 000 interessi.

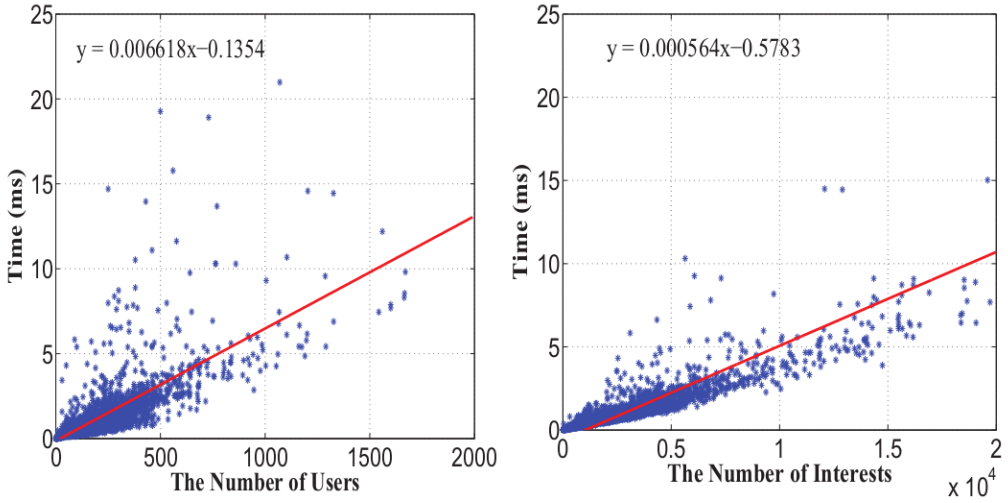

Figura 5.6: Tempo di elaborazione del CDS.

Conclusioni

In questa tesi si sono analizzati i correnti approcci utilizzati per lo sviluppo di un recommendation system e le problematiche da essi derivanti. Si è, inoltre, approfondito il mondo dei social network e come l'enorme quantità di dati generati dal loro utilizzo possa essere utilizzata per affinare le previsioni di questi sistemi e quindi la loro efficacia ed efficienza. Si è successivamente definita l'idea di raccomandazioni basata sulla community di cui un utente fa parte e non sul singolo. Infine si è definita una metrica utilizzabile per identificare quelle communities che sono maggiormente predisposte per l'utilizzo dei sistemi prima definiti.
Come si è visto nel Capitolo 2, la diffusione del mercato e-commerce è in continua crescita a livello mondiale. Come ogni mercato, anche questa tipologia, tuttavia, soffre delle stesse problematiche dei mercati canonici e in particolare della saturazione. Questo si può notare anche dai dati che mostrano come la crescita percentuale delle vendite sia in calo. Per questo motivo utilizzare pratiche di marketing mirate a massimizzare le vendite di prodotti è una tematica che, negli ultimi anni, sta prendendo sempre più piede. Questo ha portato a sperimentare i vecchi sistemi di recommendation verso nuovi orizzonti ed opportunità che, se sfruttate al meglio, possono portare benefici al mercato. Studiando i social network, e i dati generati da queste piattaforme, è stato possibile creare dei modelli che, a differenza dei vecchi approcci di content-based filtering e collaborative filtering, si basano sugli interessi, sulle relazioni sociali e sull'influenza che alcune figure hanno sulle scelte dell'utente.

Dalle analisi effettuate, si è sottolineato come gli utenti siano influenzabili dalle scelte degli amici stretti e che quindi esse possano essere prese in considerazione per le raccomandazioni di prodotti. Questo concetto è estendibile anche ad amici più lontani, sacrificando in parte l'accuratezza, ma ottenendo ugualmente dei risultati migliori ai sistemi canonici. Questo tipo di approccio non solo ha consentito di aumentare l'accuratezza delle previsioni, ma ha anche permesso di risolvere alcune problematiche legate al collaborative filtering canonico. Normalmente i recommender system vengono utilizzati per fornire consigli per utenti singoli. Tuttavia alcune scelte d'acquisto non sono imputabili al singolo utente ma, più che altro, alla comunità di cui fa parte. Un altro aspetto importante da prendere in considerazione è quello della privacy. Con l'entrata in vigore del GDPR, che regola la protezione dei dati personali dei cittadini dell'UE, sia dentro che fuori i confini territoriali, le norme riguardanti tale tematica sono diventate stringenti e l'utilizzo dei dati è diventato un terreno scottante in cui spesso le aziende scelgono di non incappare per evitare sanzioni. In questo modo l'utilizzo di questi sitemi applicati alle communities è diventato tema attuale, in quanto esse, oltre a considerare casi che non venivano trattati in precedenza, consentono una maggiore anonimizzazione dei dati personali degli utenti coinvolti nel processo. Questo approccio ha, però, portato a far fronte a nuove problematiche relative alla scelta della tipologia di communities più adatta a cui applicare tale modello. Per questo motivo è stato introdotto il Community Similarity Degree. Una metrica che permettesse di valutare analiticamente quanto una community sia omogenea e quindi quanto essa sia valida per essere presa in considerazione per questi sistemi. Questo indicatore ha mostrato come, a differenza di ciò che si pensava, le communities generate partendo dagli interessi di ogni singolo utente ottengano performance maggiori rispetto a quelle generate casualmente, basate sulla posizione degli utenti e basate sui collegamenti sociali (amicizie).

I risultati ottenuti dal SNRS e dal CSD possono sembrare, a prima vista, discostanti. Tuttavia la ragione che spiega il tutto è il fatto che spesso un

utente stringe amicizie in luoghi differenti (al lavoro, a scuola, nella vita di tutti i giorni, etc.). Risulta quindi naturale che gli interessi di tutti i suoi amici non siano omogenei e questo provoca delle previsioni errate da parte del sistema. Un'altra spiegazione sta nel fatto che se due amici hanno interessi simili per una determinata categoria di prodotti, non è detto che lo stesso interesse si ripresenti per un'altra categoria.
Fatte tutte queste premesse, i risultati ottenuti per quanto riguarda il Social Network-based Recommender System sono che le prestazioni del sistema sono migliori del collaborative filtering a discapito della coverage, ovvero la quantità di utenti su cui è possibile effettuare previsioni. Viene leggermente ridotta la Data Sparsity e in parte risolto il problema del Cold-Start se si estende l'algoritmo anche agli amici lontani, sacrificando un po' di precisione. Con il Community Similarity Degree è possibile identificare le communities per cui l'algoritmo di recommendation è più preciso, tuttavia i valori ottenuti sono risultati sempre bassi in confronto al valore massimo che esso poteva assumere.
Uno sviluppo futuro potrebbe essere quello di identificare le motivazioni per cui due utenti si sono collegati, in quei Social Network dove è possibile recuperare questa informazione, per poter identificare il tipo di relazione e quindi costruire una community basata oltre che sull'amicizia anche sul significato semantico intrinseco. A questo punto è possibile utilizzare il CSD per poter valutare la bontà della community creata e verificare che, effettivamente, essa sia più omogenea rispetto a quelle già trattate e più facilmente reperibili.
L'unica cosa certa è che il mercato dell'e-commerce e tutto l'indotto sono destinati a crescere in futuro. Perciò ogni azienda del settore che vuole stare al passo col mercato e non vuole farsi sorpassare dai concorrenti dovrebbe prevedere una parte degli investimenti per la ricerca e sviluppo in questo ambito in modo da poter guadagnare un vantaggio tattico da poter sfruttare in futuro. Ma in fondo parlare del futuro è come cercare di prevedere su quale faccia si poserà il dado che abbiamo appena lanciato. Possiamo calcolare con che probabilità tale evento avverrà ma non avremo mai la certezza

matematica. In fondo è proprio questo parte ignota, lasciata al caso, che ci ha spinto, ci spinge e continuerà a farlo in futuro, a conoscere ciò che non conoscevamo e ad inventare ciò che non credevamo possibile.

Bibliografia

[1] Gomez-Uribe, Carlos A.; Hunt, Neil (28 December 2015). "The Netflix Recommender System". ACM Transactions on Management Information Systems. 6 (4): 1–19.

[2] Aggarwal, Charu C. (2016). Recommender Systems: The Textbook. Springer. ISBN 9783319296579.

[3] Peter Brusilovsky (2007). The Adaptive Web. p. 325. ISBN 978-3-540-72078-2.

[4] Blanda, Stephanie (May 25, 2015). "Online Recommender Systems – How Does a Website Know What I Want?". American Mathematical Society. Retrieved October 31, 2016.

[5] Macedo AA, Pollettini JT, Baranauskas JA, Chaves JC (2016). "A Health Surveillance Software Framework to deliver information on preventive healthcare strategies". J Biomed Inform. 62: 159–70. doi:10.1016/j.jbi.2016.06.002. PMID 27318270

[6] Terveen, Loren; Hill, Will (2001). "Beyond Recommender Systems: Helping People Help Each Other". Addison-Wesley. p. 6. Retrieved 16 January 2012.

[7] Altman, N. S. (1992). "An introduction to kernel and nearest-neighbor nonparametric regression". The American Statistician. 46 (3): 175–185. doi:10.1080/00031305.1992.10475879. hdl:1813/31637.

[8] Jamming He; Wesley W. Chu. "A Social Network-Based Recommender System (SNRS)". Computer Science Departmente, University of California Los Angeles.

[9] D. Lowd e P. Domingos (2005). "Naive Bayes Model for Probability Estimation". Proceedings of the Twenty-Second Internation Conference on Machine Learning(ICML), ACM Press.

[10] J.Neville, D. Jensen (2000). "Iterative Classification in Relational Data". Proceedings of the Workshop of Learning Statistical Models from Relational Data at the Seventeeth National Conference on Artificial Intelligence (AAAI), pp. 13-20.

[11] S. Shang, P. Hui, S.R. Kulkarni, P.W. Cuff (2013). "Wisdom of the Crowd: Incorporating Social Influence in Recommendation Models". Department of electrical Engineering, Princeton University. airXiv: 1208.0782v2.

[12] D. Cosley, D. Huttenlocher, J. Kleinberg, X. Lan, S. Suri (2010). "Sequential Influence Models in Social Network". In Proc. 4th International AAAI Conference on Weblogs and Social Media.

[13] J. Kleinberg (2007). "Cascading Behavior in Networks: Algorithmic and Economic Issues". In Algorithmic Game Theory, Cambridge University Press.

[14] N.E. Friedkin, E.C. Jhonsen(1999). "Social Influence Networks and Opinion Change". Advance in Group Processes. vol. 16, pp.1-29.

[15] P. Hui, S. Buchegger(2009). "Groupthink and Peer Pressure: Social Influence in Online Social Network Groups". In Proceeding International Conference on Advances in Social Networks Analysis and Mining(ASONAM), Athens, Greece.

[16] D. Cosley, D. Huttenlocher, J. Kleinberg, X. Lan, S. Suri(2010). "Sequential Influence Models in Social Networks". In Proc. 4th Internationa AAAI Conference on Weblogs and Social Media.

[17] X. Han, L. Wang, R. Farahbakhsh, A. Cuevas, R. Cuevas, N. Crespi, L. He(2016). "CSD: A multi-user similarity metric for community recommendation in online social networks". Elsevier Ltd. doi:10.1016/j.eswa.2016.01.003.

[18] S. Macskassy, F. Provost (2003). "A Simple Relational Classifier". In Proceedings of the KDD-2003 Workshop on Multirelational Data Mining.

[19] J. He, W.W. Chu, Z. Liu (ISI 2006). "Inferring Private Information from Social Network". Proceedings of IEEE Intelligence and Security Informatics Conference.

[20] J. Dougherty, R. Kohavi, M. Sahami (1995). "Supervised and unsupervised discretization of continuous features". Machine learning: proceedings of the twelfth international conference: Vol. 12 (pp. 194–202).

[21] L. Baltrunas, T. Makcinskas, F. Ricci(2010). "Group recommendations with rank aggregation and collaborative filtering". In Proceedings of the ACM RECSYS (pp. 119–126). Barcelona, Spain: ACM.

[22] M. Deshpande, G. Karypis (2004). "Item-based top-n recommenda- tion algorithms". ACM Transactions on Information Systems, 22, 143–177.

[23] D. Coppersmith, L. K. Fleischer, A. Rurda (2010). "Ordering by weighted number of wins gives a good ranking for weighted tournaments". ACM Transactions on Algorithms, 6, 55:1–55:13.

Ringraziamenti

I ringraziamenti, può sembrare strano dirlo, ma risultano la parte più complessa che mi sia ritrovato a scrivere redigendo questa tesi. Sarei potuto essere banale e scontato. Sicuramente in questo modo sarebbe stato tutto più semplice. Ed è proprio da qui che intendo partire, perchè le banalità proprio non riesco a farmele piacere.
Il ringraziamento più grande va proprio a coloro che mi hanno cresciuto, che mi hanno accudito, che mi hanno sopportato e mi hanno sempre assecondato in ogni mia decisione. Mi hanno insegnato cosa significa il rispetto, cosa significa la dedizione, quali sono le cose importanti della vita e a non perderle mai di vista. Mi hanno insegnato a non arrendermi e a spingermi oltre ogni mio limite. Ad essere umile, cosa che forse non ho recepito fino in fondo. A credere nelle mie potenzialità e a puntare sempre al massimo. Grazie Papà, grazie Mamma, grazie Simo, grazie Nonno, grazie Nonna, grazie Zii.
Un ringraziamento particolare va anche al Prof.re Giovanni Rossi. Mi ha aiutato nello svolgimento di questa tesi, mi ha spronato e non ha mai smesso di credere che sarei arrivato fino a qui. Azzardo nel dire che, per quanto mi riguarda, c'è più di un semplice rapporto accademico, c'è anche e soprattutto un rapporto di amicizia e di profondo rispetto.
Un altro grande grazie va alla famiglia che ho acquisito. Parlo di tutte quelle persone che sono state, sono e saranno parte importante e fondamentale del percorso che mi ha portato fino a qua, la vita. A tutte quelle persone che ho perso, a quelle che ci sono ora e a quelle che incontrerò in futuro. Potrei mettermi qui ad elencarli ad uno ad uno ma come ho lasciato intendere le cose

scontate non mi piacciono. Grazie per essermi stato vicino nei momenti in cui più mi ero perso. Grazie per avermi fatto scoprire l'amicizia. Grazie per avermi fatto conoscere l'amore. Grazie per avermi fatto scoprire la passione per l'arte. Grazie per avermi strappato un sorriso. Grazie per esserti fidata di me. Grazie per non essertene mai andata. Grazie per aver condiviso tanto. Chiunque legga queste righe saprà riconoscersi nel posto giusto.
In ultimo un grazie va anche alla vita. Questo percorso difficile, frastagliato, pieno di cadute, di passioni, di sentimenti, di emozioni e di avvenimenti che reputavi impossibili. Grazie per tutte le volte che ho sofferto e per tutte le volte in cui mi sono sentito al settimo cielo. Grazie di tutto perché se oggi mi trovo qui, e affronto le cose in questa maniera, è principalmente grazie a te.