POLITECNICO DI MILANO
Corso di Laurea Specialistica in Ingegneria Informatica
Dipartimento di Elettronica e Informazione

CATEGORIZZAZIONE DI UTENTI IPTV

VP-Lab
Laboratorio di Valutazione delle Prestazioni

Relatore: Prof. Paolo CREMONESI
Correlatore: Prof. Roberto TURRIN

Tesi di Laurea di:
Fabio GRANARA, matricola 711776

Anno Accademico 2009-2010

Alla mia famiglia

Sommario

Lo scopo di questa tesi è la definizione e l'analisi di algoritmi atti a dedurre il lifestyle di persone, per individuare categorie di utenti al fine di realizzare pubblicità mirata.
Il dominio è quello della televisione digitale, in particolare dell'IPTV, dove gli utenti vengono quotidianamente inondati da migliaia di messaggi pubblicitari. Nello specifico vengono utilizzate tecniche che permettono la creazione di un'associazione tra l'utente che visiona contenuti on-demand (ad esempio un film) e per i quali esprime un voto, ed il lifestyle, che incide pesantemente sul comportamento del consumatore nell'acquisto di prodotti. La categorizzazione degli utenti a seconda del proprio lifestyle, che in questa tesi è rappresentato dal sesso di un utente (UOMO o DONNA), è stata ottenuta attraverso due distinti approcci: l'utilizzo (i) di algoritmi di raccomandazione basati su tecniche di riduzione dimensionale e (ii) di regole d'associazione.
Di questi algoritmi verrà poi calcolata la correttezza grazie ad alcune metriche di accuratezza come la precision, la recall e graficamente con la curva di ROC, che vengono utilizzati per testarne rispettivamente la precisione, la completezza e la qualità. I risultati ottenuti nei vari test hanno mostrato una precision compresa tra 53.90% e 87.54%, mentre per la recall tra 24.56% e 82.20%.

Ringraziamenti

Desidero innanzitutto ringraziare il Professore Cremonesi, per avermi dato la possibilità di lavorare in un ambito di ricerca molto interessante ed innovativo, ed il Professor Turrin, che mi ha aiutato ad affrontare tutte le problematiche inerenti il lavoro di tesi ed a dirimere i miei dubbi durante la sua stesura. Inoltre vorrei esprimere una sincera gratitudine ai colleghi, che hanno condiviso con me i giorni di lavoro nel VP-Lab dispensando numerosi consigli, a tutte le persone che mi hanno sostenuto, ed in particolar modo alla mia famiglia, che mi ha permesso di affrontare il percorso universitario. Infine rivolgo un ringraziamento speciale a mio nonno.

Indice

Elenco delle tabelle

Elenco delle figure

Capitolo 1

Introduzione

Indice

In questo capitolo viene inquadrata l'area tematica, lo scopo e la struttura del lavoro di tesi.

1.1 Inquadramento generale e breve descrizione del lavoro

Lo scopo di questa tesi è la definizione e l'analisi di algoritmi atti a dedurre il lifestyle di persone, per individuare categorie di utenti al fine di realizzare pubblicità mirata. Attraverso questo metodo si cerca di evitare, quindi, l'allontanamento dallo schermo dell'utente durante le interruzioni pubblicitarie, raggiungendo l'attenzione del consumatore con prodotti e servizi di potenziale interesse.
Per dedurre il lifestyle di un consumatore è necessario conoscere alcuni dati che identificano l'utente in una particolare categoria, denominata indicatore di lifestyle, come ad esempio il sesso, l'età, l'occupazione, etc. In questo lavoro di tesi sono stati implementati sistemi che si basano su un indicatore specifico: il sesso dell'utente (UOMO o DONNA).

Per costruire tali sistemi si possono usare molte tecniche, ed in particolare sono state esplorate due alternative:

1. algoritmi di raccomandazione
2. regole d'associazione

Le metodologie dei sistemi di raccomandazione sono state utilizzate, in quanto sono varie le tematiche in comune con il lavoro di tesi, come:

- hanno lo stesso input, rappresentato dai voti (rating) che l'utente assegna ai film visionati
- creare pubblicità mirata può essere rapportato a fare raccomandazione, anche se non si arriva alla pubblicità mirata usando algoritmi di raccomandazione, ma ricavando il lifestyle di utenti a partire dai rating che gli utenti hanno espresso

Gli algoritmi descritti nel Capitolo 5 sono basati su un algoritmo di raccomandazione, a differenza di quelli del Capitolo 6 che si basano sulle regole d'associazione.

Negli ultimi anni sono stati creati i sistemi di raccomandazione essenzialmente per motivi di marketing; infatti vengono utilizzati per selezionare gli oggetti, detti item, da proporre ad un utente partendo da informazioni riguardanti le sue preferenze (esplicite od implicite). Gli item possono essere film, libri, vacanze, ristoranti o altro che possa risultare di reale interesse. Il funzionamento di questi sistemi si basa sull'analisi dei dati storici di un utente, come ad esempio i film per cui ha espresso un voto, oppure il numero di visualizzazioni. I sistemi di raccomandazione sono molto utilizzati dalle grandi aziende (come Amazon.com e iTunes), che vogliono consigliare l'acquisto di un prodotto, oppure un insieme di prodotti e/o servizi presenti nella loro offerta commerciale, ad un utente potenzialmente interessato. Tali sistemi si stanno espandendo in molti altri settori, perchè rappresentano un vantaggio sia per i produttori (in termini di vendite ed immagine) che per i consumatori. Proprio per questo motivo, i sistemi di raccomandazione sono diventati sempre più popolari nell'ultimo decennio sia all'interno della comunità scientifica sia in ambito industriale. Questo ha comportato una

serie sempre crescente di pubblicazioni che si occupano di questa tematica, proponendo nuove soluzioni.

Le regole d'associazione non utilizzano algoritmi di raccomandazione, ma cercano in ogni caso, attraverso varie fasi, di dedurre il lifestyle di utenti. In origine le regole d'associazione sono state utilizzate per rappresentare le regolarità di comportamento nell'acquisto di prodotti da parte di clienti all'interno di negozi. In questo lavoro di tesi sono state utilizzate, invece, per creare un'associazione tra i film visti da un utente ed il suo lifestyle.

In entrambi i casi (algoritmi di raccomandazione e regole d'associazione) è di fondamentale importanza, quindi, per l'implementazione di tali sistemi l'utilizzo di un input basato sulla lista di film visti dagli utenti ed i ralativi voti da loro assegnati. E' necessario un sistema televisivo che permetta all'utente di inviare informazioni al provider; in ambito televisivo esistono varie tecnologie: analogico, digitale, satellitare e IPTV. L'area tematica scelta in questo lavoro di tesi è quella dell'IPTV, perchè garantisce qualità, velocità e scambio di dati da parte del provider all'utente e viceversa. Inoltre in questo dominio sempre più spesso si hanno a disposizione i voti/preferenze degli utenti su contenuti on-demand , e, a partire da questi dati, si vorrebbe inquadrare un utente IPTV in uno specifico lifestyle, in modo da fornire questa informazione ad un provider di pubblicità.
Ottenere informazioni comporta la creazione di un profilo per ogni utente che ha visionato un film. Se si associano l'anagrafica, l'occupazione, la residenza, etc è possibile collocare l'utente in uno stile di vita, il lifestyle, che permette quindi la creazione di pubblicità mirata per quell'utente posizionato in un determinato segmento di consumatore.

Il dataset utilizzato come input per testare i sistemi implementati è Movielens, fornito dal laboratorio di ricerca GroupLens dell'University of Minnesota che si occupa di raccomandazione, il cui download è gratuito previa registrazione. Tramite questo dataset è possibile esaminare gli algoritmi implementati e calcolarne la correttezza attraverso alcuni parametri come la precision, la recall e la curva di ROC, che vengono utilizzati per testarne rispettivamente la precisione, la completezza e la qualità. I risultati ottenuti nei vari test hanno mostrato una precision compresa tra 53.90% e 87.54%, mentre per la recall tra 24.56% e 82.20%

1.2 Struttura della tesi

La tesi è strutturata nel modo seguente: nel *Capitolo 2* si descrive con maggior dettaglio il dominio in cui si colloca il lavoro di tesi. Nel *Capitolo 3* vengono mostrate le modalità con cui vengono affrontate le problematiche. Nel *Capitolo 4* è stato descritto il dataset utilizzato nella tesi e le metodologie di valutazione degli algoritmi implementati. La parte iniziale del *Capitolo 5* si occupa di proporre un algoritmo che si basa sulla raccomandazione ibrida, che utilizza tecniche di riduzione matriciale. Vengono poi illustrati con dello pseudocodice l'implementazione dell'algoritmo e descritte alcune varianti che sono state apportate. Infine vengono visualizzati i risultati. Invece nel *Capitolo 6* sono descritti ed illustrati gli altri tipi di algoritmi, quelli che si basano sulle regole d'associazione. Anche per questi vengono illustrati i risultati ottenuti. Infine nel *Capitolo 7* vengono riassunti gli scopi, riportate le conclusioni a cui si è giunti nel lavoro di tesi e si propongono possibili sviluppi futuri.

Capitolo 2

Targeted Advertising nell'IPTV

Indice

In questo capitolo viene riportato lo stato dell'arte del settore ed un inquadramento dell'area di ricerca orientato a presentare la problematica affrontata.

2.1 Tecnologia IPTV

L'IPTV (Internet Protocol Television) è un sistema di diffusione del servizio di televisione digitale attraverso l'utilizzo del protocollo IP ed infrastrutture di rete presenti sul territorio, che forniscono connessioni a banda larga. In particolare, questo servizio televisivo consente la visione da parte dell'utente di contenuti, con una qualità del servizio analoga a quella di altre piatta-

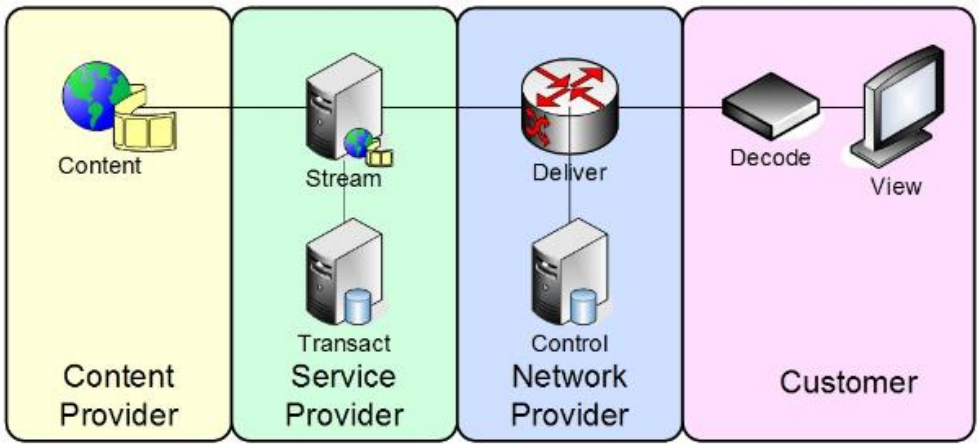

Figura 2.1: Architettura Tecnologia IPTV.

forme di tv digitale. L'IPTV solitamente offre due tipologie principali di contenuti: broadcast, distribuiti contemporaneamente a più utenti, e contenuti on-demand (VOD), dove il singolo utente ha la possibilità di fruire a pagamento o gratuitamente di un programma televisivo, di un film o di un qualsiasi altro contenuto presente in catalogo.

Nella maggior parte dei casi, l'IPTV viene offerto insieme ad altri servizi, come quello di telefonia o di accesso ad Internet (triple play).
Nella figura 2.1 [1] vengono mostrati i principali attori del servizio di IPTV:

- Content provider: che possiede i contenuti televisivi
- Service Provider: operatore che fornisce servizi di telecomunicazioni agli utenti attraverso contratti. Solitamente compra i contenuti del Content Provider per assemblarli in pacchetti di servizi.
- Network Provider: l'ente che gestisce l'infrastruttura di rete necessaria per la fornitura del servizio IPTV. Service Provider e Network Provider possono, e solitamente è così, essere la stessa società.
- End-user: l'utente finale che, grazie ad un ricevitore, la set-top-box, accede alla rete per usufruire di contenuti da essa forniti.

I vantaggi di un sistema IPTV sono:

- Personalizzazione: un servizio bidirezionale interattivo permette al fornitore di personalizzare il servizio in base all'utente, come ad esempio la creazione di una guida elettronica personalizzata (EPG)

- Integrazione: l'offerta voce, dati e video in un'unica soluzione consente al fornitore di offrire all'utente servizi che integrino i tre canali (web on tv, e-commerce dalla tv, etc), riducendo i costi e aumentando l'offerta

- Piattaforma on-demand: l'IPTV introduce un metodo per fruire dei contenuti televisivi, più flessibile rispetto alla Pay per View, dove il contenuto acquistato è vincolato ai palinsesti e agli orari imposti dai broadcast. Invece il video on-demand è privo di questo vincolo, in quanto può essere fruito, essere messo in pausa e rivisto in qualsiasi momento

- Interattività: il protocollo IP permette la comunicazione a due vie, cioè bidirezionale, tra l'utente finale e il Service Provider. Questo permette all'utente di accedere a vari servizi come giochi, scommesse, televoto, navigazione web, invio e ricezione dei messaggi. Anche altre tecnologie offrono questo tipo di servizi interattivi, ma con svantaggi come costo di connessione, lentezza della trasmissione dei dati ed occupazione della linea

2.2 Targeted Advertising

Al giorno d'oggi, i consumatori sono quotidianamente bombardati da centianaia di messaggi pubblicitari, di cui la maggior parte non sono di alcuno interesse per l'utente, che cerca in ogni modo di sfuggirne. Per quanto riguarda il campo televisivo, l'utente cerca di evitare la pubblicità trasmessa tra spezzoni del programma / film che sta visionando con lo zapping, l'allontanamento dalla televisione oppure tramite registrazione intelligente che permette di memorizzare su un dispositivo esterno solo il contenuto che si desidera, e non la pubblicità.

Lo scopo di questa tesi è proprio quello di evitare, o comunque diminuire, tale fuga del consumatore dalla pubblicità, grazie a strategie in grado

di raggiungere l'attenzione del consumatore. Il forte interesse che ha avuto Microsoft per l'acquisizione di Yahoo! è l'esempio più evidente relativo ai grandi interessi economici che girano intorno al mondo dell'advertising.

Il Targeted Advertising, che significa appunto pubblicità mirata, intende fornire una soluzione al disagio del consumatore inondato di pubblicità, ma anche alla necessità delle aziende di raggiungere solo i loro potenziali clienti, il loro target. Sviluppare tecniche in questo campo significa utilizzare le informazioni relative agli utenti allo scopo di associare le inserzioni pubblicitarie ai profili utenti corrispondenti al target dell'inserzione stessa.

Il media in cui è più semplice riconoscere l'implementazione di metodologie di targeted advertising è sicuramente Internet. La possibilità di raccogliere semplicemente (ad esempio attraverso i cookie) informazioni relative agli utenti per crearne il profilo e la disponibilità di tecnologie che consentono di analizzare i contenuti delle pagine web (es. tecniche di information retrieval) hanno reso possibile la realizzazione di tecnologie che permettono l'associazione di inserzioni pubblicitarie ai profili utenti e ai contenuti delle pagine web. In Figura 2.2 viene riportato uno screenshot, tratto da un sito Web, in cui è possibile notare la presenza di inserzioni pubblicitarie contestuali al contenuto della pagina web in cui sono collocate, cioè targeted advertising.

Più complessa, rispetto al caso del web, è l'implementazione di metodi di targeted advertising per l'altro media più popolare: la televisione. Nel caso della tv analogica è estremamente complesso, se non impossibile, personalizzare le interruzioni pubblicitarie sugli interessi degli utenti. Infatti non è possibile conoscere il profilo degli utenti che seguono una determinata trasmissione e proporre interruzioni pubblicitarie diverse, in quanto il segnale è trasmesso broadcast, cioè unico per tutti e, inoltre, la trasmissione è unilaterale, ossia dall'emittente all'utente, il quale può solo ricevere dati passivamente e non può inviarne. Nel corso degli anni sono nate agenzie (es. Auditel) il cui obiettivo è rilevare gli ascolti televisivi, raccogliendo i dati di fruizione dei vari canali da un campione della popolazione attraverso l'installazione di appositi apparecchi. Questo permette alle aziende interessate alla pubblicità di conoscere alcune informazioni relative agli ascolti e fruizioni dei vari canali e programmi, così da poter pianificare eventuali investimenti

Figura 2.2: Esempio di targeted advertising

pubblicitari. Tuttavia queste informazioni non consentono di implementare strategie di targeted advertising, ma si limitano, ad esempio, a consentire la deduzione di quali sono le fasce orarie più adatte per pubblicizzare un determinato prodotto.

Soluzioni che si avvicinano al concetto di targeted advertising vengono proposte negli ultimi tempi da alcune aziende statunitensi [23], dove la tecnologia della trasmissione televisiva permette di inviare contenuti differenti nelle diverse aree geografiche. Raccolti i dati campione relativi all'area geografica è possibile differenziare le interruzioni pubblicitarie. Ad esempio lo spot dello stesso oggetto, un potente fuoristrada, potrà presentare il prodotto che percorre una strada innevata oppure attraversa il deserto, a seconda che lo spettatore sia in Alaska o in Arizona. Un'altra differenziazione di target può avvenire sulla demografica comune di una certa area. All'interno di un'area che si deduce, dall'analisi su campioni, essere abitata da nuclei con alto reddito familiare si potranno, ad esempio, pubblicizzare prodotti di nicchia, il cui spot non sortirebbe effetti se proposto in un'area abitata da nuclei meno agiati e quindi sarebbe un inutile investimento per l'azienda del prodotto.

La tv interattiva, in particolare l'IPTV, costituisce il dominio su cui, potenzialmente, si possono applicare strategie di targeted advertising in senso stretto, in cui ad ogni utente possono essere inviate pubblicità differenti, a seconda del suo profilo. Questo grazie alla potenzialità interattiva del media, che consente la comunicazione a due vie tra emittente e utente, e all'infrastruttura del sistema, come illustrato nei paragrafi precedenti. Nel caso dell'IPTV, dominio di riferimento per questa tesi, l'architettura del sistema e l'utilizzo del protocollo IP abilitano la possibilità di inviare, senza difficoltà, dati diversi ad utenti diversi, come già avviene per i video on-demand.

Vista la recente introduzione dell'IPTV nel mercato televisivo, esistono ancora poche soluzioni proposte relative al targeted advertising

In [20] e [7] si propone un'architettura di targeted advertising per la tv interattiva, ovvero la tv digitale che utilizza un set-top box come decoder del segnale presso l'utente. La soluzione si basa sulla definizione di un profilo utente costituito da informazioni relative alla demografica e all'analisi del-

l'interazione tra utente e tv. Questi dati vengono analizzati periodicamente da un server centrale al fine di collocare ogni utente all'interno di cluster predefiniti, corrispondenti ai vari target delle agenzie pubblicitarie. Ottenute le regole associative tra utenti e inserzioni pubblicitarie, sui set-top box degli utenti vengono salvati i profili. Ad intervalli di tempo regolari tutte le inserzioni pubblicitarie, attraverso un apposito canale, vengono inviate ad ogni set-top box che svolge il compito di mostrare all'utente solo le inserzioni corrispondenti al profilo. Questo tipo di soluzione proposta incontra problemi di fattibilità, quali la modifica dell'architettura di sistema, per rendere possibile l'invio dei contenuti pubblicitari e dei set-top box con l'aggiunta della capacità di elaborazione dei profili utente. Inoltre, utilizzando i cluster, ovvero raggruppamenti di utenti, non si ottengono profili utenti accurati ma generalizzati. Infine, si rischia di violare la privacy dell'utente analizzando in maniera costante la sua interazione con il media; solo se l'utente autorizza la raccolta dei suoi dati, compresi quelli demografici, il sistema può proporre targeted advertising.

2.3 Sistemi di raccomandazione come Targeted Advertising

Vista l'assenza di una soluzione realizzabile e funzionale, questa tesi ha l'obiettivo di individuare un approccio innovativo al targeted advertising su piattaforme IPTV. Lo scopo è rendere possibile l'invio di contenuti pubblicitari basato sul profilo di interessi degli utenti, offrendo i vantaggi di un maggior ritorno di investimento per le imprese, che vedrebbero aumentare il raggiungimento del target desiderato, ed un maggior interesse degli utenti durante le interruzioni pubblicitarie, in seguito all'offerta di prodotti relativi al loro profilo. E' necessario quindi creare un sistema che funga da unione tra le varie pubblicità e gli utenti della popolazione, per la creazione di pubblicità mirata.
In questo lavoro di tesi sono stati utilizzati i sistemi di raccomandazione per creare questo tipo di associazione *utente - pubblicità*, come mostrato in

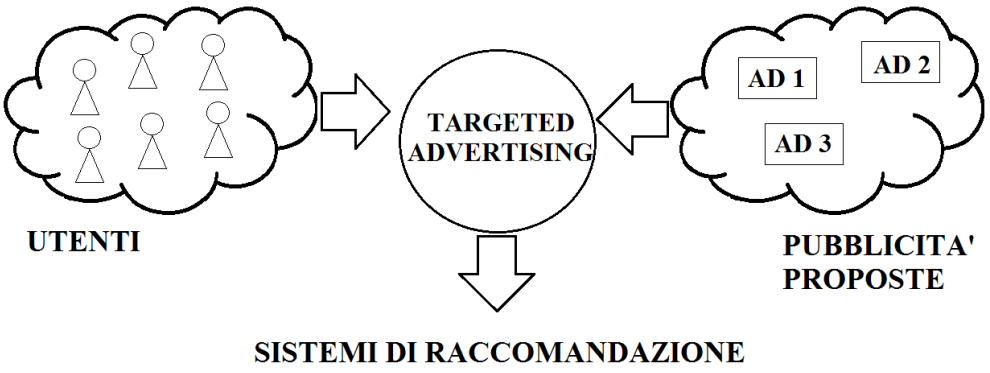

Figura 2.3: Sistemi di raccomandazione come Targeted Advertising

figura 2.3.

Prima di passare alla definizione dei sistemi di raccomandazione è importante sottolineare che in letteratura sono presenti varie tecniche che potrebbero essere applicate a questo tipo di problematica, ma la soluzione scelta in questa tesi è la più adatta all'obiettivo e per questo adottata come soluzione proposta, in quanto essa si basa su un processo d'inferenza del lifestyle degli utenti a partire da un campione di popolazione i cui interessi sono noti o quanto meno ricavabili.

2.4 Algoritmi di raccomandazione

I sistemi di raccomandazione sono costituiti da tecniche specifiche che mirano a suggerire elementi d'informazione, come musica, film, libri, e quant'altro che potrebbe essere d'interesse per l'utente. Si cerca, quindi, di suggerire ad un utente informazioni per lui nuove su un determinato argomento, cercando di prevedere il voto che quell'utente avrebbe dato ad un elemento che lui non aveva ancora preso in considerazione. In figura 2.4 viene semplificato il funzionamento di un sistema di raccomandazione, con relativi input ed output.

Negli ultimi tempi, i sistemi di raccomandazione hanno preso piede in

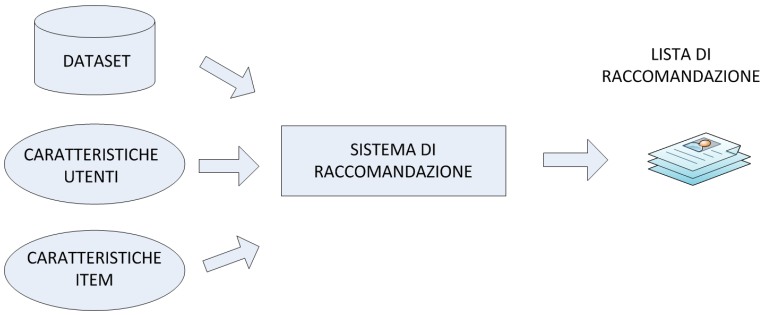

Figura 2.4: Schema semplificato di un sistema di raccomandazione

molti settori commerciali, come ad esempio:

- nell'e-commerce, dove i siti web memorizzano informazioni, attraverso i cookie, di prodotti di interesse per il cliente e ne consiglia altri correlati, o simili;
- siti di contenuti multimediali, che consigliano contenuti nuovi in base a ciò che è stato visionato nel passato recente da un utente;
- piattaforme che erogano servizi che permettono agli utenti di fruire, gratuitamente o a pagamento, di programmi televisivi, film (servizi on demand), come appunto l'IPTV.

I sistemi di raccomandazione sono stati sviluppati con l'idea di risolvere il problema dell'*Information and Product Overload* [12], che rappresenta la difficoltà di un utente nello scegliere i contenuti e i prodotti per lui interessanti, tra tutti quelli disponibili in un determinato contesto. L'uso dei sistemi di raccomandazione è diventato sempre più frequente, in quanto rappresenta un vantaggio sia per il cliente, che per il venditore. Infatti, il primo può scegliere con più facilità i contenuti d'interesse ed il secondo può incrementare le proprie vendite, offrendo prodotti che hanno maggiore possibilità di essere venduti.

Sono due le entità base che compaiono in un sistema di raccomandazione [30]:

- User: l'utente che riceve la raccomandazione
- Item: il prodotto, o l'insieme di prodotti, forniti da un servizio

Gli input di un sistema di raccomandazione appartengono ad uno delle seguenti categorie:

- Rating: esprime l'opinione di un utente su un item. I ratings sono normalmente forniti dall'utente e possono essere:
  - Espliciti: il sistema chiede all'utente di esprimere un voto sull'item fruito. La scala utilizzata per la votazione solitamente è da 1 a 5 (1 indica che l'item non è piaciuto affatto, mentre 5 che l'item è piaciuto molto). Può comparire anche il voto 0 che indica la non fruizione da parte dell'utente di quell'item.
  - Impliciti: il sistema non richiede all'utente un voto, ma cerca di capirlo dal suo comportamento. Ad esempio se l'utente ha acquistato un film e l'ha visto più volte, si ipotizza che è stato di suo gradimento.
- User: rappresenta le informazioni relative agli utenti, che possono essere dati anagrafici, piuttosto che il tipo d'occupazione. Questi dati sono difficili da ottenere e vengono richiesti esplicitamente agli utenti.
- Item: comprende tutti quei dati riferiti al prodotto. Ad esempio, se parliamo di un film, possono essere inclusi il genere, l'anno di produzione, gli attori, etc.

La rappresentazione di tali dati è molto semplice, ma comporta dei problemi di scalabilità, in quanto sono necessarie elevate risorse di memoria ed i tempi di esecuzione sono molto lunghi. Non è un aspetto positivo in quanto solitamente le raccomandazioni devono essere fatte a *real-time*.
Invece, l'output di un sistema di raccomandazione può essere di due tipi:

- Predizione: espressa come un valore numerico, che rappresenta la potenziale opinione di un utente su un item. La predizione solitamente viene espressa con una scala numerica. Questa forma di output è conosciuta anche come *Individual Scoring*.
- Raccomandazione: è espressa come una lista di N item che possono essere d'interesse per un utente. Questa forma di output è conosciuta anche come *Top-N Recommendation*, oppure *Ranked Scoring*.

Gli aspetti critici di un sistema di raccomandazione sono:

- Qualità della raccomandazione: gli utenti hanno bisogno di potersi fidare della raccomandazione ricevuta. Questo vuol dire che si deve cercare di proporre item che saranno effettivamente di gradimento per l'utente, e per farlo si può ad esempio provare a ridurre il rapporto tra il numero di item che non sono stati graditi ed il numero totale di item che sono stati raccomandati.

- Sparsità dei dati: spesso la quantità di informazione disponibile sugli item fruiti da ogni utente è molto ridotta rispetto al numero totale di item fruibili.

- Scalabilità: i sistemi di raccomandazione richiedono calcoli pesanti e la complessità computazionale cresce all'aumentare del numero di utenti e item. La scalabilità è quindi un aspetto critico per le prestazioni.

- Perdita di similarità transitiva: supponiamo che l'utente a sia molto simile all'utente b, ed inoltre che l'utente b sia molto simile all'utente c. Allora per la proprietà transitiva si ha che l'utente a è molto simile anche all'utente c. Può capitare però che gli utenti a e c non abbiano nel profilo item fruiti comuni, quindi essi non vengono considerati simili dal sistema di raccomandazione e l'informazione di similarità transitiva viene persa.

- Anomalie degli utenti: gli individui con gusti inusuali sono difficili da trattare, perchè non è possibile associarli ad alcun gruppo di utenti simili. Questo è anche conosciuto con il nome di *Problema di Gray Sheep*.

Inoltre in letteratura [8] esistono due diversi approcci che si possono seguire in un sistema di raccomandazione:

- Model-based: questi algoritmi [24] utilizzano l'insieme dei rating o degli item per generare un modello che viene poi utilizzato per effettuare raccomandazione. Si distinguono due fasi principali: la generazione del modello e la raccomandazione. Nella prima fase si genera il modello utilizzando un'elevata quantità di risorse e di tempo (fase *batch*), e

nella seconda fase si effettua la raccomandazione *real-time*. Quindi si riesce ad ottenere un sistema scalabile a *real-time*, dal momento che le operazioni computazionalmente complesse vengono effetturate nella fase di generazione del modello.

- Memory-based (o Heuristic-based): questi algoritmi [15] utilizzano delle euristiche per produrre delle predizioni o delle raccomandazioni sull'intero insieme di item o di fruizioni del sistema. Le operazioni devono essere effettuate tutti in un'unica fase, perchè nella computazione è necessario tenere in memoria tutti i dati relativi al dominio.

Ci sono varie tecniche per la creazione di sistemi di raccomandazione, ma si possono raggruppare in due gruppi in base a come vengono computate le raccomandazioni [2] :

- Content-Based Filtering (CBF): all'utente vengono raccomandati item simili a quelli che ha fruito in passato.
- Collaborative Filtering (CF): all'utente vengono raccomandati item fruiti da utenti con profili simili al suo.

Il funzionamento di queste tecniche di raccomandazione verrà spiegato con maggior dettaglio nei prossimi paragrafi.

2.4.1 Algoritmi Content-Based

Questi tipi di algoritmi raccomandano all'utente item simili a quelli da lui fruiti in passato. L'idea di base è la seguente: se nel passato ad un utente è piaciuto un oggetto, probabilmente in futuro gli piacerà un oggetto simile.

Il CBF ottiene le caratteristiche degli item (ad esempio per un film si valutano il regista, il genere, la lista degli attori, etc) e le confronta poi con i gusti dell'utente per generare la raccomandazione. Il profilo dell'utente viene costruito sulla base delle caratteristiche comuni a tutti gli item da lui fruiti. Ad esempio se un utente acquista più film di uno stesso regista, si può dedurre che i suoi gusti sono molto vicini a quel particolare regista; oppure, se ad un utente, in passato, è piaciuto il film *Il signore degli anelli - La compagnia dell'anello* allora molto probabilmente in futuro gli piaceranno film

come *Il signore degli anelli - Le due torri* oppure *Eragon*. Questo approccio consente di individuare quali caratteristiche dei contenuti interessano l'utente e ne provocano la scelta.

Gli algoritmi basati su contenuto hanno il grande vantaggio di poter raccomandare item che non sono mai stati fruiti, proprio perchè si basano sulle caratteristiche degli item per generare la raccomandazione. Tra gli svantaggi invece bisogna tenere in considerazione che le caratteristiche degli item devono essere in un formato automaticamente gestibile dal calcolatore, altrimenti devono essere inserite manualmente; inoltre un item con caratteristiche mai fruite dall'utente non verrà mai raccomandato. Infine non è possibile generare una raccomandazione per un utente nuovo, che non abbia ancora fruito alcun item, per questo stesso motivo un utente con un profilo poco informativo non riceverà una buona raccomandazione.

2.4.2 Algoritmi Collaborative Filtering

Ci focalizziamo maggiormente sugli algoritmi con approccio Collaborative Filtering, in quanto gli algoritmi sviluppati e analizzati in questo lavoro di tesi rientrano in questa categoria. A differenza dell'approccio Content-Based, in cui la raccomandazione si basa sulle caratteristiche di utenti e item, gli algoritmi collaborativi (CF) sfruttano le relazioni tra essi, ovvero in base alle fruizioni degli item da parte degli utenti. Da queste informazioni i sistemi di tipo CF cercano di trovare similarità tra profili degli utenti e le caratteristiche degli item per generare la raccomandazione.

L'approccio collaborativo in qualche modo simula il passaparola umano, ad esempio quando un utente non sa quale film noleggiare chiede consiglio agli amici con gusti simili. Allo stesso modo gli algoritmi collaborativi mirano a consigliare all'utente item fruiti da utenti a lui simili. Per determinare le similarità vengono utilizzate tecniche classiche che si basano sulla distanza tra i profili, come la Similarità del Coseno, il Coefficiente di Correlazione di Pearson e tecniche di riduzione dimensionale come la Singular Value Decomposition (SVD). Su quest'ultima si focalizza parte del nostro lavoro, come descritto nel Capitolo 4.
I CF presentano, però, anche alcuni limiti [4]:

- Problema dell'utente nuovo: un utente che non abbia ancora espresso un giudizio su alcun item non può essere associato a nessun altro utente.
- Problema dell'item nuovo: un item che non sia mai stato votato non può comparire nella lista di raccomandazione, in quanto non risulta simile a nessun altro item visto dall'utente.
- Sparsità dei dati: più informazioni si hanno, più accurata sarà la raccomandazione.
- Utenti con gusti particolari: non sono associabili ad altri utenti, quindi per essi la raccomandazione non sarà ottima.

Gli algoritmi collaborativi si possono ulteriormente classificare in Item-Based, User-Based e Riduzione dimensionale:

Item-based Gli algoritmi CF di tipo Item-based confrontano i profili degli item per cogliere delle similarità, e in base a queste effettuano la raccomandazione per l'utente. Supponiamo ad esempio che l'utente i abbia fruito gli item j e k; il sistema di raccomandazione andrà a cercare gli item più simili agli item j e k e li raccomanderà all'utente. In questo approccio è fondamentale la similarità tra gli item (due item sono simili se più utenti hanno espresso giudizi concordi relativamente ad essi), perchè in base ad essa il sistema intuisce quali sono gli item da raccomandare all'utente.

User-based Gli algoritmi CF di tipo User-based, a differenza dei precedenti, cercano le similarità tra gli utenti per generare la raccomandazione. Quello che si fa è cercare gli utenti con gusti simili. Supponiamo che l'utente i abbia un profilo molto simile all'utente j, allora l'algoritmo raccomanderà all'utente i gli item fruiti da j ma non da i.

Riduzione dimensionale può essere visto come una combinazione tre Item-based e User-based, poichè genera un modello che tiene in considerazione sia delle similarità tra item che tra user. Questo tipo di algoritmo verrà descritto più approfonditamente nel Capitolo 4

2.5 Regole d'associazione

Uno dei principali problemi che presentano gli algoritmi di raccomandazione è quello di lavorare su un grande numero di dati. Lavorare su un un dateset enorme che contiene spesso informazioni ridondanti rende necessario ridurre e filtrare i dati con cui si opera. Un metodo alternativo per la categorizzazione degli utenti è quello di creare regole d'associazione per inferire il *lifestyle* ad un utente della popolazione di cui si conoscono solo il rating espresso per i film visionati.

Dalla definizione presa in letteratura [22], la regola d'associazione è un popolare e molto usato metodo per scoprire le relazione tra diverse variabili in database molto grandi. Il problema dell'associazione delle regole è definito [3] in questo modo: preso $I = (i_1, i_2, \ldots, i_n)$ come l'insieme di n attributi binari chiamati *items* e preso $D = (t_1, t_2, \ldots, t_m)$ l'insieme di transazioni chiamato *database*, ogni transazione in D ha un'unico identificatore e contiene un sottoinsieme di items in I. Una regola è definita come un'implicazione della forma:

$$X \Rightarrow Y$$

dove $X, Y \subseteq I$ e $X \cap Y = \emptyset$.
Gli insiemi di item X e Y sono chiamati rispettivamente *antecedenti* e *conseguenti* della regola.

Per illustrare il concetto viene fornito un piccolo esempio derivante dal dominio dei supermercati: l'insieme di item è $I = (latte, pane, burro, birra)$. Un piccolo database contenente gli item è formato dalla tabella 2.1. dove 1

ID transazione	**latte**	**pane**	**burro**	**birra**
1	1	1	0	0
2	0	1	1	0
3	0	0	0	1
4	1	1	1	0
5	0	1	0	0

Tabella 2.1: Esempio di database per la creazione di regole d'associazione

identifica la presenza e 0 l'assenza dell'item nella transazione. Un esempio di regola per il supermercato potrebbe essere:

$$latte, pane \Rightarrow burro$$

Per selezionare interessanti regole dall'insieme di tutte le possibili regole, vengono introdotte alcuni valori soglia. Le più conosciute sono:

- supporto: $supp(X)$ di un itemset X è definito come la proporzione di transazione nel dataset che contiene l'itemset. Nell'esempio di prima, l'itemset $latte, pane, burro$ ha un supporto di 1 / 5 = 0.2, cioè questo occorre nel 20% di tutte le transazioni (1 su 5 transazioni)

- confidenza: $conf(X \Rightarrow Y) = supp(X \cup Y)/supp(X)$. Per esempio, la regola $latte, pane \Rightarrow burro$ ha una confidenza di 0.2 / 0.4 = 0.5 nel databease. Questo significa che per il 50% delle transazioni contenenti latte e pane, la regola è corretta.

Nel capitolo 6 verrà illustrato come queste regole d'associazione possono essere utili per il processo di raccomandazione del *lifestyle*.

Capitolo 3

Lifestyle

Indice

In questo capitolo vengono esposte con maggior dettaglio le componenti e le tecniche che sono state utilizzate nel lavoro di tesi.

3.1 Lifestyle

Come è stato esposto nel precedente capitolo, in questo lavoro di tesi si utilizzano tecniche di raccomandazione per poter ricavare informazioni sul profilo degli utenti. In particolare si vogliono conoscere le attività, gli interessi, i comportamenti, i dati demografici e tutte quelle informazioni che possano caratterizzare un utente all'interno di una popolazione. In sociologia il concetto di *lifestyle* è definito come *pattern in cui le persone vivono e spendono il loro tempo e i loro soldi* [17]. Il lifestyle di un individuo è influenzato da diversi fattori: esterni (cultura, stato sociale, famiglia,...) ed interni (personalità, memoria, sensibilità, attitudini,...).

Viene utilizzato il concetto di *lifestyle* perchè si riflette nei comportamenti delle persone e in particolare sul loro *consumption behavior* [14], cioè

il comportamento di un consumatore che, ogni giorno, acquista prodotti e servizi.

E' stato proposto nell'articolo [19] un approccio basato sulla nozione di *lifestyle* per aumentare l'accuratezza della predizione di algoritmi di raccomandazione, in condizioni di sparsità dei dati. E' stato inoltre dimostrato che, per un sistema di raccomandazione, avere come input sia le fruizioni dei programmi televisivi degli utenti, sia le informazioni demografiche rappresenta un ottimo, in quanto rappresentativo, indicatore del loro lifestyle. Per questi motivi, per il crescente uso in aree di marketing e per la facilità con cui si può applicare al settore interessato dell'IPTV, in questo tesi è stato scelto l'approccio del *lifestyle* come soluzione al problema della creazione di *targeted advertising*.

Nel prossimo paragrafo verrà mostrato un uso del *lifestyle* nel settore marketing.

3.1.1 VALS

VALS, acronimo di Values, Attitudes and Lifestyle, è un prodotto utilizzato, da parte di alcune aziende, per identificare il *consumer behavior* di un campione di utenti. E' stato sviluppato negli anni '70 dalla SRIC-BI (Stanford Research Institute Consulting Business Intelligence), che oggi ha assunto il nome di SBI (Stategic Business Insights) [33].
Proprio la SBI illustra i benefici del VALS, che si possono semplificare in:

- aiuta a comprendere i diversi stili di comunicazione del target di un'azienda
- crea profili ricchi di informazioni sui consumatori
- fa acquisire una nuova prospettiva per entrare nella testa degli utenti finali.

VALS, attraverso il questionario *VALS survey*, colloca gli utenti in uno degli otto segmenti di un framework. Ogni segmento identifica un diverso *lifestyle* ed è disposto, come illustrato in figura 3.1 [33], lungo due dimensioni. La dimensione verticale del VALS segmenta gli utenti in base alla

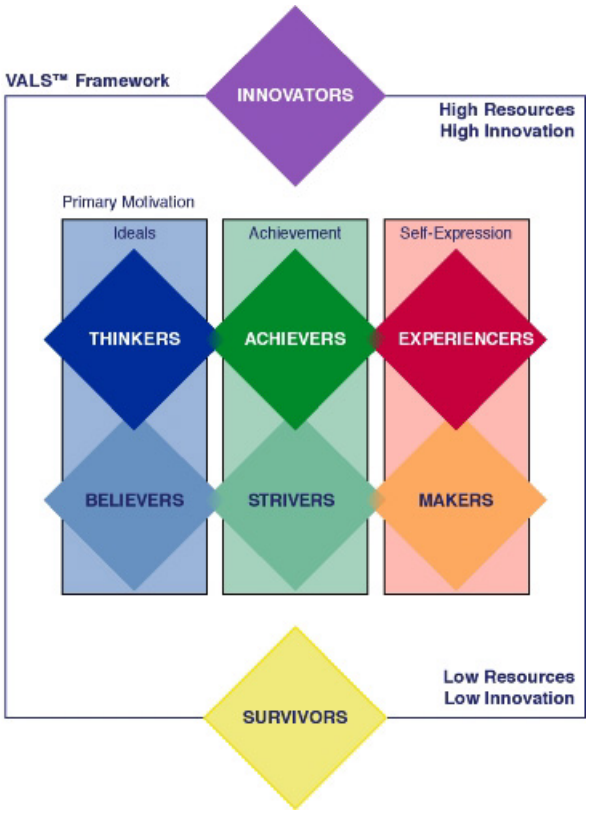

Figura 3.1: VALS Framework

loro innovatività e risorse (reddito, educazione, doti di leadership, ...); la dimensione orizzontale è relativa alle motivazioni principali che guidano il comportamento del consumatore. Nello specifico gli otto segmenti sono:

- *Innovators*: sono persone di successo, con alta autostima e molte risorse economiche, e non, a disposizione. Sono aperti alle nuove idee ed alla tecnologia. Gli *innovators* sono consumatori molto attivi ed i loro acquisiti riflettono gusti raffinati; coltivano interesse per prodotti e servizi di nicchia. Sono tra i leader affermati ed emergenti degli affari e del governo; i loro beni sono lo specchio del loro gusto per le belle cose della vita.

- *Thinkers*: motivati da ideali, sono persone mature e soddisfatte. Essi tendono ad essere educati e cercano attivamente le informazioni, prima di arrivare ad una decisione. Sono persone informate sugli eventi mon-

diali e nazionali ed attente alle opportunità mirate ad ampliare le loro conoscenze. I *Thinkers* hanno un moderato rispetto per le istituzioni, ma aperti a prendere in considerazione nuove idee. Anche se il reddito permetterebbe loro molte scelte, sono conservatori, consumatori pratici e guardano il valore dei prodotti che acquistano.

- *Achievers*: motivati dal desiderio di realizzarsi, aspirano a fare carriera e avere una famiglia. Infatti la vita sociale di questo gruppo di utenti è improntata sulla famiglia, lavoro e religione. Sono politicamente conservatori e rispettano l'autorità. Con molti desideri e bisogni, gli *Achievers* sono attivi sul mercato; danno grande importanza all'immagine e per questo ricercano prodotti di prestigio che li possano mettere in mostra. A causa della loro vita ricca d'impegni, sono spesso interessati ad una varietà di dispositivi per risparmiare tempo.

- *Experiencers*: giovani consumatori, entusiasti ed impulsivi. Essi cercano varietà d'eccitazione, lo stravagante ed il rischioso. La loro energia trova sbocco nello sport, nelle attività ricreative e sociali. Sono avidi consumatori e spendono la maggior parte del loro reddito in articoli di moda e divertimento.

- *Believers*: come i *Thinkers*, anche i *Belivers* sono motivati da ideali. Conservatori e convenzionali, il cui credo è basato sulla religione, la famiglia, la comunità e la nazione. Sono consumatori prevedibili, in quanto scelgono prodotti per la famiglia, e fedeli ai brand classici ed affermati.

- *Strivers*: sono persone alla moda e amanti del divertimento. Il denaro significa tutto per loro, ma non ne hanno abbastanza per raggiungere il successo. Acquistano prodotti che gli permettono di simulare uno status sociale superiore. Sono consumatori impulsivi, anche in base alle condizioni economiche del momento.

- *Makers*: persone pratiche, costruttive ed auto-sufficienti. Hanno grandi capacità ed energie per realizzare i propri progetti con successo.

Vivono in un contesto tradizionale di famiglia e lavoro e non sono particolarmente coinvolti per ciò che è al di fuori di tali ambiti. Non mostrano interesse nei beni materiali e nel lusso, perciò comprano solo prodotti essenziali.

- *Survivors*: hanno poche risorse a disposizione. Dovendosi concentrare sul soddisfacimento dei bisogni, piuttosto che dei propri desideri, i *Survivors* non mostrano una forte motivazione primaria. Sono consumatori cauti, e rappresentano un mercato molto modesto per la maggior parte di servizi e prodotti. Sono fedeli alle marche, soprattutto se i prodotti sono scontati.

Questi otto segmenti sono stati creati per suddividere gli adulti americani in cluster, cioè gruppi di unità (in questo caso persone) simili tra loro.

3.1.2 Eurisko

In questo paragrafo, viene mostrato in breve come in Italia è affrontata l'area tematica del *lifestyle*. In particolare, si fa riferimento all'Eurisko [32], ad oggi il più potente istituto nazionale operante nelle ricerche sul consumatore, che ha sviluppato *Sinottica*, un sistema integrato di informazioni sull'evoluzione socio-culturale, sul consumo e sull'esposizione ai mezzi di comunicazione degli Italiani.

Sinottica offre le seguenti tipologie di servizio:

- Analisi del posizionamento dei propri prodotti/marchi a fronte della concorrenza
- Progettazione del target primario per future attività di pianificazione media
- Analisi dell'esposizione ai mezzi dei target di interesse
- Controllo dell'evoluzione di prodotti competitivi, di nuove opportunità
- Analisi di scenario

3.1.3 Conclusioni

I lifestyle descrivono lo stile di vita degli individui, inteso come insieme di attività, interessi, comportamenti, demografica e quant'altro possa caratterizzarli.
I lifestyle dei consumatori si riflettono nel loro consumption behavior, ovvero la modalità con cui scelgono a acquistano prodotti e servizi. Per questo vengono spesso applicati nelle ricerche di marketing per definire e raggiungere i target. Anche nel settore dell'IPTV i lifestyle possono essere applicati per implementare strategie di targeted advertising, consapevoli, inoltre, che le fruizioni degli utenti sono indicatrici del loro lifestyle, informazione che verrà utilizzata nella soluzione proposta in questa tesi.

I prodotti e le strategie di individuazione e caratterizzazione dei lifestyle, che sono state presentate nelle pagine precedenti, soffrono di una limitazione: non sono direttamente applicabili alla raccomandazione di un prodotto o servizio. Con questo si intende che gli indicatori di lifestyle proposti sono troppo generici per applicarli direttamente all'interno di un sistema di raccomandazione. Si ricorda che l'obiettivo di questa tesi, in questa fase, è individuare un dominio che consenta l'associazione tra utenti e contenuti pubblicitari. Sistemi come VALS non consentono di raggiungere con efficacia questo obiettivo, in quanto il target di un prodotto non è facilmente esprimibile da uno dei segmenti del framework.
In conclusione, appurata la validità dei lifestyle come dominio di associazione tra utenti e contenuti pubblicitari, la soluzione proposta sarà indipendente da qualsiasi strategia di definizione dei lifestyle e quindi è possibile applicarla con qualunque tipo di lifestyle, purché vengano utilizzati degli indicatori adatti alla descrizione sia degli utenti, sia dei target dei prodotti e servizi da raccomandare.

3.2 Inferenza del Lifestyle

In questa tesi, si cerca di individuare il *lifestyle* di un campione di utenti per creare un dominio associativo tra la pubblicità e gli utenti stessi della piattaforma IPTV: si ipotizza, ragionevolmente, che ad ogni contenuto pub-

blicitario si possa associare il lifestyle target. Una volta noto il *lifestyle* degli utenti ed il target delle varie inserzioni pubblicitarie, basterà implementare delle regole associative per l'associazione utenti - contenuto pubblicitario, ottenendo cosi il targeted advertising.
Il punto centrale di questa tesi è quello di individuare ed associare dei *lifestyle* agli utenti.

La soluzione propone quindi una prima fase di raccolta di dati relativi ai *lifestyle* di un campione di utenti IPTV. Dimostrato che gli utenti esprimono, anche incosciamente, il loro lifestyle attraverso l'acquisto o la scelta di contenuti televisivi (che possono essere programmi, film, concerti o quant'altro), si inferisce il *lyfestyle* delle persone che non hanno risposto al questionario, in base alla similitudine che hanno questi individui con gli altri che invece l'hanno compilato. Per il calcolo delle similarità tra gli utenti, vengono proposti nei capitoli successivi alcuni algoritmi utili per il raggiungimento dello scopo finale.
Essendo necessario conoscere, in ogni caso, le informazioni relative al *lyfestyle* di un gruppo di persone, si è usato lo strumento del questionario, ottimale per lo scopo, in quanto:

- i questionari sono utilizzati moltissimo nel settore marketing per svolgere ricerche di mercato, come descritto nei paragrafi precedenti;
- è complesso e costoso raggiungere tutta la popolazione e, inoltre, difficilmente gli utenti sono disposti a rendere pubblici i loro interessi, gelosi della propria privacy. Da qui la necessità di un processo di inferenza sviluppato in modo tale che anche gli utenti che non dichiarano esplicitamente il proprio *lyfestyle* ne venga attributo uno;
- la compilazione del questionario, solitamente, per l'utente rappresenta solo una perdita di tempo. Per incentivare le persone, che vengono selezionate all'interno della popolazione per rispondere alle domande, si possono offrire alcune promozioni, come ad esempio la fruizione di qualche contenuto gratuitamente. Come detto nel punto precedente, per non rendere troppo oneroso la raccolta di dati, questa offerta può essere sostenuta dal gestore della piattaforma IPTV solo se limitata ad un numero limitato di utenti.

In figura 3.2 viene mostrato schematicamente quanto descritto fino ad'ora. In particolare, in questo lavoro di tesi il processo d'inferenza rappresenta

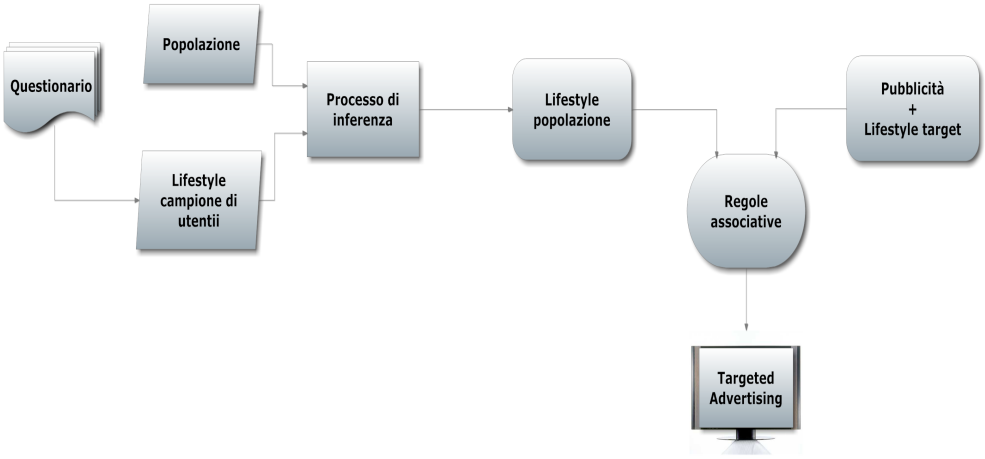

Figura 3.2: Schema generale della soluzione proposta

il punto centrale della soluzione. Gli altri aspetti come la preparazione del questionario, la modalità di raccolta delle informazioni e la realizzazione di regole associative tra i *lifestyle* degli utenti della popolazione e le inserzioni pubblicitarie non sono trattati in questa tesi, che vuole proporre una soluzione generale alla tematica affrontata.
Nel capitolo 4 viene presentato il dataset sui cui si sono testati gli algoritmi e gli strumenti per valutare validità e precisione dell'algoritmo stesso; invece nei capitoli 5 e 6 vengono descritte le soluzioni proposte e sviluppate per il processo d'inferenza.

Capitolo 4

Dataset e metodologia di valutazione

Indice

In questo capitolo viene presentato il dataset con cui sono stati effettuati i test e vengono mostrati gli strumenti utilizzati per poter valutare gli algoritmi implementati in questo lavoro di tesi.

4.1 Dataset Movielens

In questa sezione viene descritto il dataset utilizzato nella tesi, su cui sono stati applicati gli algoritmi proposti.
Come accennato nel precedente capitolo, l'esecuzione degli algoritmi proposti richiede che si conoscano i dati del campione, ossia che siano stati raccolti i questionari. Considerato il fatto che in questo lavoro i questionari non sono stati realizzati, si è utilizzato un dataset relativo a una piattaforma di video

on-demand che fornisce un elenco di utenti e, per ogni utente, la lista delle sue fruizioni di item ed alcune informazioni, che in questo prototipo rappresentano il lifestyle, simulando così di aver raccolto i dati del questionario. Inoltre, con questo dataset, non si conoscono solo i lifestyle del campione, ma di tutta la popolazione. Quindi sarà possibile confrontare i *lifestyle* inferiti attraverso gli algoritmi proposti con i *lifestyle* reali, applicando gli strumenti di valutazione elencati nelle prossime sezioni.

Il dataset MovieLens [31] è un dataset pubblicamente accessibile, creato dai componenti dell'Università del Minnesota (USA) che fanno parte del gruppo di ricerca GroupLens Research [26]. In particolare Movielens è il nome del sistema di raccomandazione gratuito realizzato da GroupLens per effettuare una ricerca nel campo dei sistemi di raccomandazione. GroupLens ad oggi mette a disposizione 3 versioni di dataset: la prima contiene 943 utenti e 1682 film per un totale di centomila fruizioni; la seconda contiene 6040 utenti e 3883 film per un totale di 1000209 ratings; la terza contiene 10681 di film, 71567 utenti e circa 10 milioni di ratings. Nei test effettuati è stata adottata la seconda versione, ovvero quella con circa un milione di ratings. I dataset Movielens, in particolare la versione da noi utilizzata, sono stati ampiamente usati in letteratura per i sistemi di raccomandazione, soprattutto per il fatto che non esistono utenti nell'insieme che abbiano visto meno di 20 items.

Nello specifico di questa tesi, si è potuto lavorare solo sul dataset Movielens, in quanto è l'unico, nel dominio dei video, a fornire informazioni personali sugli utenti che possono essere utilizzati per il lifestyle.

I dati forniti da Movielens rappresentano un dataset non molto sparso avendo densità del 4.26%. Il numero medio di fruizioni per utente è 165.6, le fruizioni medie per film sono 257.6 mentre il rating medio è 3.58. Si tratta inoltre di un dataset esplicito in cui la scala di rating varia da 1 a 5 e lo 0 indica che l'utente non ha fruito di quel film. Inoltre il numero degli utenti con sesso maschile è 4331, mentre quelli con sesso femminile è 1709.
Il dataset fornisce i seguenti dati:

- USERS: informazioni relative agli utenti presenti nel dataset. In particolare

- UserID: identificativo univoco di ogni utente;
- Gender: identifica il sesso dell'utente. E' rappresentato da 'M' se è maschio, oppure 'F' se è femmina;
- Age: denota la fascia d'età dell'utente. Si definiscono sette fasce:
  * utente con età minore di 18 anni;
  * età compresa tra 18 e 24 anni;
  * età compresa tra 25 e 34 anni;
  * età compresa tra 35 e 44 anni;
  * età compresa tra 45 e 49 anni;
  * età compresa tra 50 e 55 anni;
  * età maggiore di 55 anni;
- Occupation: indica l'occupazione lavorativa dell'utente. Sono presenti ventuno tipi:
  * other
  * academic/educator
  * artist
  * clerical/admin
  * college/grad student
  * customer service
  * doctor/health care
  * executive/managerial
  * farmer
  * homemaker
  * k-12 student
  * lawyer
  * programmer
  * retired
  * sales/marketing
  * scientist
  * self-employed

    * technician/engineer
    * unemployed
    * writer
  - ZipCode: è il codice postale del luogo di visione dell'utente

- MOVIES: riguarda informazioni relative agli item, film, presenti nel dataset. In particolare:
  - MoviesID: identificativo univoco del film nel dataset;
  - Title: Titolo del film
  - Genres: indica uno o più generi del film in questione.

- RATINGS: elenco delle votazioni date dagli utenti ai film. Il file in questione è costituito da:
  - UserID: identificativo univoco dell'utente;
  - MovieID: identificativo univoco del film a cui è stato assegnato un voto;
  - Rating: voto assegnato dall'utente al film. Si tratta di un rating esplicito espresso in scala da 1 a 5 (solo interi), dove 1 rappresenta il voto più basso e 5 quello più alto. Può essere presente anche un rating pari a 0 se l'utente non ha visionato il film;
  - Timestamp: riporta l'instante in cui il rating è stato espresso.

4.2 Strumenti di valutazione di un algoritmo

In questa sezione vengono presentati gli strumenti che sono stati utilizzati per valutare gli algoritmi proposti e testare così i risultati ottenuti.

Per poter, quindi, valutare un algoritmo che assegna un *lifestyle* ad ogni utente della popolazione grazie ad un processo d'inferenza , è necessario conoscere i dati reali, cioè i veri *lifestyle* delle persone. Solo in questo modo, confrontando i dati reali con quelli ottenuti, si riesce a verificare la validità dell'algoritmo sviluppato.
Il processo di inferenza dei *lifestyle* corrisponde alla produzione di una raccomandazione, come se ogni indicatore di *lifestyle* (età, lavoro, sesso,...)

fosse un item. Vista la somiglianza tra gli algoritmi proposti in questa tesi con quelli di raccomandazione proposti in letteratura, è possibile utilizzare gli stessi strumenti per valutarne i risultati e il funzionamento.

4.2.1 Metriche di classificazione dell'accuratezza

Le Classification Accuracy Metrics [10, 11] sono quegli strumenti che permettono di valutare l'efficacia di una predizione che distingue un contenuto di gradimento per l'utente da un altro di non gradimento. Attraverso queste metriche è possibile misurare la frequenza con cui un sistema di raccomandazione si comporta in maniera corretta od incorretta. Le Classification Accuracy Metrics sono applicabili a domini con rating binari, come nel caso in esame.

L'insieme dei dati analizzati dal sistema può quindi essere diviso in: contenuti rilevanti per l'utente (se sappiamo che è piaciuto all'utente), contenuti non rilevanti per l'utente (non è piaciuto all'utente), oppure né rilevanti né irrilevanti (non sappiamo con certezza se il contenuto è piaciuto o meno all'utente).
Questo strumento di valutazione classifica le raccomandazioni in:

- True positive (TP): il sistema raccomanda un contenuto di interesse per l'utente;
- True negative (TN): il sistema non raccomanda un contenuto che non è di interesse per l'utente;
- False negative (FN): il sistema non raccomanda un contenuto di interesse per l'utente;
- False positive (FP): il sistema raccomanda un contenuto non di interesse per l'utente.

Un contenuto viene considerato raccomandato dal sistema se inserito nella lista degli N-contenuti con raccomandazione più alta tra tutti i contenuti. In tabella 4.1 vengono mostrati schematicamente i parametri da utilizzare per classificare la raccomandazione.

	CONTENUTO RACCOMANDATO	**CONTENUTO NON RACCOMANDATO**
CONTENUTO RILEVANTE	TRUE POSITIVE (TP)	FALSE NEGATIVE (FN)
CONTENUTO NON RILEVANTE	FALSE POSITIVE (FP)	TRUE NEGATIVE (TN)

Tabella 4.1: Parametri per la classificazione della raccomandazione

Due sono le metriche più importanti, la *precision*, che può essere vista come una misura di esattezza o fedeltà, e la *recall*, misura di completezza.
Nell'Information Retrieval [4] (insieme di tecniche utilizzate per il recupero mirato dell'informazione in formato elettronico), la *precision* è definita come il numero di contenuti attinenti recuperati da una ricerca diviso il numero totale di contenuti recuperati dalla stessa ricerca, e la *recall*, chiamata anche *True Positive Rate* (TPR), è definita come il numero di contenuti attinenti recuperato da una ricerca diviso il numero totale di contenuti attinenti esistenti.
In un processo di classificazione statistica, la *precision* per una classe è il numero di veri positivi (il numero di contenuti etichettati correttamente come appartenenti alla classe) diviso il numero totale di elementi etichettati come appartenenti alla classe (la somma di veri positivi e falsi positivi, che sono oggetti etichettati erroneamente come appartenenti alla classe). La recall in questo contesto è definita come il numero di veri positivi diviso il numero totale di elementi che attualmente appartengono alla classe (per esempio la somma di veri positivi e falsi negativi, che sono contenuti che non sono stati etichettati come appartenenti alla classe, ma dovrebbero esserlo).
Nell'Information Retrieval, un valore di precision del 100% significa che ogni risultato recuperato da una ricerca è attinente, mentre un valore di recall pari al 100% significa che tutti i documenti attinenti sono stati recuperati dalla ricerca. Quindi le formule della *precision* e della *recall* sono:

$$precision = \frac{numero \quad di \quad contenuti \quad rilevanti \quad raccomandati}{numero \quad di \quad tutti \quad i \quad contenuti \quad raccomandati}$$

$$recall = \frac{numero \quad di \quad contenuti \quad rilevanti \quad raccomandati}{numero \quad di \quad tutti \quad i \quad contenuti \quad rilevanti}$$

Oppure schematicamente:

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

Per quanto riguarda i casi trattati in questa tesi, riferito quindi al domino dei *lifestyle*, le classificazioni si possono contestualizzare in questo modo:

- True positive (TP): il sistema assegna all'utente un indicatore di *lifestyle* che gli corrisponde;
- True negative (TN): il sistema non assegna all'utente un indicatore di *lifestyle* che non gli corrisponde;
- False negative (FN): il sistema non assegna all'utente un indicatore di *lifestyle* che gli corrisponde;
- False positive (FP): il sistema assegna all'utente un indicatore di *lifestyle* che non gli corrisponde;

4.2.2 Receiver operating characteristic

Le curve di ROC [28] (Receiver operating characteristic) costituiscono un'ulteriore metrica, in questo caso grafica, per la valutazione dei risultati prodotti dagli algoritmi mostrati in questa tesi.
L'utilizzo delle curve di ROC è relativamente comune per la valutazione dei più svariati test in molti settori. Lungo i due assi si possono rappresentare la *sensibilità* o True Positive Rate (TPR) e 1 - *specificità* o False Positive Rate (FPR). In altre parole si studiano i rapporti fra allarmi veri e falsi allarmi.
Il TPR è, come detto prima, la *recall*; mentre il FPR esprime il rapporto tra i contenuti erroneamente raccomandati all'utente e tutti i contenuti non rilevanti:

$$FPR = \frac{FP}{FP + TN}$$

Il test che si effettua attraverso l'analisi delle curve di ROC ha la capacità di separare propriamente la popolazione di utenti che hanno avuto una buona raccomandazione e utenti per cui la raccomandazione non è stata utile.

Questo si può fare, andando ad analizzare l'area sottesa dalla curva di ROC (Area Under Curve, AUC). La statistica di sintesi per valutare l'accuratezza di un modello predittivo è l'area sottesa alla curva, che permette di confrontare due modelli predittivi attraverso il confronto delle AUC.

4.2.3 Confidence analysis

Gli algoritmi proposti in questa tesi implementano un processo di inferenza dei lifestyle della popolazione, partendo da un campione di cui i lifestyle sono noti.
Come vedremo nel Capitolo 5, per lo sviluppo degli algoritmi viene utilizzata una matrice binaria *User-Lifestyle Matrix (ULM)*, ricavata dal dataset Movielens, in cui *ULM(i,j)*= 1 indica che l'indicatore di lifestyle j è stato associato all'utente i. Il processo d'inferenza, però, assegna valori non binari, che rappresentano il peso che viene dato dall'algoritmo a quel particolare indicatore di lifestyle. La scelta della soglia che determina se un coefficiente viene portato a 1 (si associa il lifestyle indicator all'utente) oppure a 0 (non si associa il lifestyle indicator all'utente) è rilevante ai fini della creazione della ULM binaria e , quindi, dei risultati definitivi degli algoritmi. Una soglia troppo bassa, ad esempio, provocherebbe una precision scadente e una recall alta; al contrario, una soglia troppo alta aumenterebbe la precision, ma la recall calerebbe drasticamente. L'utilizzo di lifestyle esclusivi, come nel caso proposto in questa tesi, può essere considerato nella fase di binarizzazione rendendola esclusiva, ossia impossibilitando l'assegnazione allo stesso utente di due o più lifestyle indicators tra loro esclusivi (es. si considera il vincolo che un utente non può essere sia maschio che femmina). Vista la rilevanza, ai fini dei risultati, della scelta della soglia di binarizzazione, si propone in questa tesi l'utilizzo di uno strumento, chiamato *confidence analysis*. Esso viene applicato successivamente al processo di inferenza con l'obiettivo di analizzare l'affidabilità dei risultati ottenibili con l'applicazione di diverse soglie di binarizzazione ad ognuno degli indicatori di lifestyle. Nel caso del lifestyle indicator UOMO, ad esempio, la confidence analysis permette di rispondere alla domanda: "Quanto si è sicuri che l'utente i sia maschio?",

ottenendo una valutazione del processo di inferenza (il cui compito, invece, è rispondere alla domanda "Quali utenti sono maschi?"). Per ogni indicatore di lifestyle vengono individuate diverse soglie, che conducono a diverse precision; si ricorda che, idealmente, maggiore è la soglia di binarizzazione, maggiore è la precision e minore è la recall. Si nota, inoltre, che applicando nella confidence analysis la stessa soglia del processo di inferenza, si ottiene la stessa precision, e quindi non si approfondisce l'analisi dei risultati. La confidence analysis prevede che la soglia si possa applicare:

- agli estimated values, ovvero ai coefficienti relativi a un lifestyle indicator. Se la soglia e maggiore del coefficiente, si attribuisce all'utente il lifestyle indicator, altrimenti no;
- alla differenza tra i due top coefficienti di una classe di lifestyle indicators. Se la differenza e maggiore della soglia si assegna all'utente il lifestyle indicator relativo al maggior coefficiente, altrimenti all'utente non si assegna alcun lifestyle indicator della classe.

La confidence analysis filtra i risultati ottenuti dal processo di inferenza, in quanto si applica una nuova soglia di binarizzazione alla matrice ULM non binaria. Si riassume l'esecuzione della confidence analysis su un lifestyle indicator j, con soglia s:

1. Estrazione degli utenti a cui il processo di inferenza ha assegnato il lifestyle indicator j;
2. Estrazione del coefficiente:
   - estrazione del coefficiente ($coeff$) non binari relativi al lifestyle indicator j, solo per gli utenti a cui esso e stato assegnato (utenti estratti al punto 1)

   oppure

   - calcolo della differenza ($diff$) tra il top coefficiente e il secondo coefficiente della classe a cui appartiene j, solo per gli utenti a cui esso e stato assegnato (utenti estratti al punto 1)

3. Per ogni utente, applicazione della soglia ai coefficienti estratti al punto 2:

   - Se $coeff > s$ mantieni l'assegnazione del lifestyle indicator j per l'utente, altrimenti cancella assegnazione;

   oppure

   - Se $diff > s$ mantieni l'assegnazione del lifestyle indicator j per l'utente, altrimenti cancella assegnazione.

   4) Calcolo della metrica precision sulla ULM ottenuta.

La confidence analysis, concludendo, oltre ad essere uno strumento di valutazione, può essere applicata per ottimizzare i risultati, filtrandoli per aumentare la precision. Questo aspetto viene applicato per proporre un algoritmo di individuazione della soglia di binarizzazione di ciascun lifestyle indicator, al fine di ottenere la precision desiderata su ognuno.

4.2.4 K-fold cross validation

Per gli algoritmi proposti nel prossimo capitolo è stato utilizzato un ulteriore strumento di valutazione: la tecnica del k-fold cross validation. Consiste nel partizionare il campione di utenti di cui si conosce il lifestyle in k parti uguali, una viene utilizzata come set di validazione per testare il modello (validation data), le rimanenti k-1 vengono applicate per costruire il modello (training data). In figura 4.1 viene mostrato un esempio con k = 4. Il processo di inferenza viene eseguito k volte, ognuna delle quali vede una k parte diversa come validation data, e le rimanenti vengono applicate per generare il modello. Il k-fold cross validation viene applicato negli algoritmi proposti nel Capitolo 5 per identificare, per ogni lifestyle indicator, n soglie di binarizzazione, in cui n è il numero di precision desiderate (es. se si vogliono calcolare le soglie per cui precision=60% e precision=80%, n è uguale a 2).

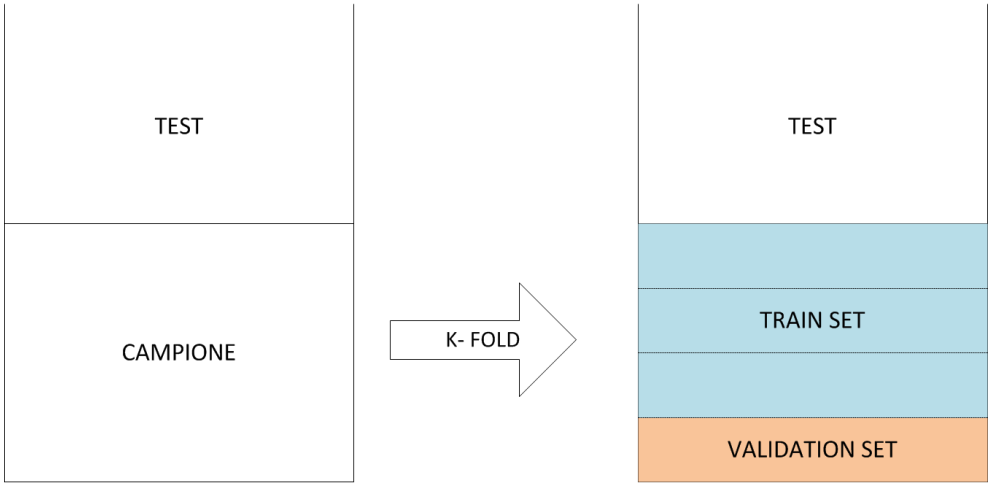

Figura 4.1: Esempio di K-fold validation

4.3 Ambiente di sviluppo

La maggior parte del lavoro sviluppato in questa tesi è stato realizzato con MATLAB. Questa scelta è stata motivata dal fatto che fosse necessario un ambiente particolarmente adatto al calcolo numerico ed all'analisi statistica. Se tra i vantaggi offerti da MATLAB ci sono una maggiore facilità nello sviluppare e analizzare algoritmi di calcolo numerico, allo stesso tempo esso presenta alcuni aspetti sfavorevoli. Lo svantaggio maggiore è sicuramente dato dalla limitata efficienza computazionale del compilatore interno, infatti fornendo un linguaggio di programmazione ad alto livello al contempo viene penalizzata l'efficienza in fase di esecuzione. Tutto ciò si traduce in una maggiore lentezza nell'esecuzione di calcoli particolarmente lunghi.

In questa tesi sono stati sviluppati algoritmi mirati alla ricerca del solo indicatore di *lifestyle* Uomo - Donna per due motivi principali:

- per la lentezza dei calcoli nell'eseguire i vari algoritmi;
- per verificare il comportamento dei vari algoritmi, per poi individuare miglioramenti o peggioramenti dei risultati.

Negli algoritmi presentati nel capitolo 5, sono stati utilizzati anche dei programmi sviluppati in C e C++.

Capitolo 5

Algoritmo basato su tecniche di raccomandazione

Indice

In questo capitolo verranno descritti i motivi della scelta dell'algoritmo di raccomandazione usato tra tutti quelli presenti in letteratura ed inoltre verranno illustrati i suoi concetti base con i relativi risultati.

5.1 Matrice URM

Il dataset Movielens è stato rappresentato da una matrice, detta User Rating Matrix (URM). La URM in Figura 5.1 è una matrice che ha sulle righe gli identificativi degli utenti e sulle colonne quelli degli item. Ogni cella rappresenta il rating espresso dall'utente per quel particolare item. La matrice URM è quindi costituita da m utenti ed n item ($m \times n$). Il rating $r_{i,j}$ dato

da un utente i ad un item j, con $i = 1, 2, ..., m$ e $j = 1, 2, ..., n$ è un numero reale in accordo con una scala di rating imposta dal sistema, ad esempio $r_{i,j} = 1, 2, 3, 4, 5$ se l'utente i ha visionato il film j, $r_{i,j} = 0$ se invece non l'ha visionato. Dalla matrice URM si generano le due sottomatrici essenziali

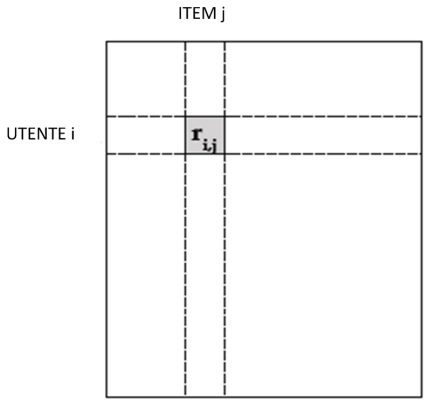

Figura 5.1: User Rating Matrix (URM)

per lo sviluppo di tutti gli algoritmi: CAMPIONE (training set) e TEST (test set). In tutti gli algoritmi sviluppati in questo lavoro di tesi è stato considerato che la matrice CAMPIONE fosse il 10% della matrice URM e TEST il restante 90%. Queste percentuali rappresentano il fatto che, come detto in precedenza, il questionario usato per conoscere le informazioni degli utenti viene rivolto solo ad una piccola parte di tutti gli utenti.
La matrice CAMPIONE delinea la parte di popolazione di cui si conosce l'indicatore di lifestyle del sesso (UOMO o DONNA) e a cui viene associata, quindi, la parte di ULM corrispondente; mentre, TEST identifica l'intera popolazione di utenti di cui si vuole conoscere il sesso.
Un esempio di suddivisione di tali matrici è mostrato in figura 5.2.
L'algoritmo viene eseguito, quindi, 10 volte ed in ognuna di essa viene considerata una parte diversa come training set. I risultati finali ottenuti saranno calcolati applicando la media ai risultati parziali, derivanti da ogni singolo ciclo.

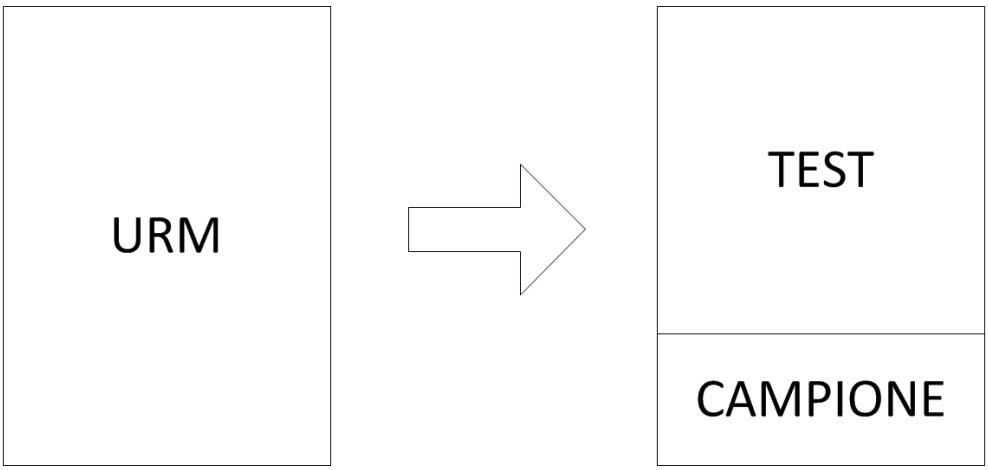

Figura 5.2: Partizionamento matrice URM

5.2 Algoritmi di raccomandazione ibrida

Come detto nel Capitolo 2, esistono due tipiche tecniche di raccomandazione: content-based e collaborative filtering. In letteratura [27] sono presenti molteplici metodi che combinano le due strategie (content e collabortive) in un unico approccio, formando così un sistema di raccomandazione ibrido, che utilizza i contenuti di caratteristiche integrative, al fine di migliorare l'accuratezza di previsione del tradizionale collaborative filtering.
Durante la generazione del modello, vengono utilizzate informazioni supplementari riguardanti l'utente che contribuiscono ad una descrizione più precisa dei dati, forniti nel caso in esame da Movielens, e prevedono di migliorare la precisione di predizione. Questo modello unificato di rating ed informazioni supplementari costituiscono il cuore di un sistema di *raccomandazione ibrido.*
L'iniettare caratteristiche addizionali di una fonte (come informazioni content) in un algoritmo designato a processare dati con una fonte diversa (come il tradizionale algoritmo di raccomandazione user-oriented o item-oriented) è noto come *Feature Combination* [9]. Ad esempio in [5] sono stati riportati gli esperimenti di questo algoritmo nel dominio di raccomandazione di film utilizzando sia le valutazioni degli utenti che le caratteristiche di contenuto. I risultati ottenuti risultano migliori di quelli di un approccio tipicamente

collaborativo.

Nel lavoro di tesi i *lifestyle* degli utenti sono in relazione agli item fruiti, ed è possibile costruire una matrice *URMplus*, da cui si genererà il modello per inferire i *lifestyle* della popolazione, come suggerito in [27]. La *URMplus* è cosi composta:

- sulle righe si riportano gli utenti del campione di popolazione scelto e di cui, quindi, si conosce il *lifestyle*;
- sulle colonne vengono riportati gli item e, in seguito, gli indicatori di *lifestyle*.

In pratica la *URMplus* si ottiene accoppiando la *URM* e la *ULM*, come mostrato nella figura 5.3. L'utilizzo della matrice *URM* produrrebbe la sola

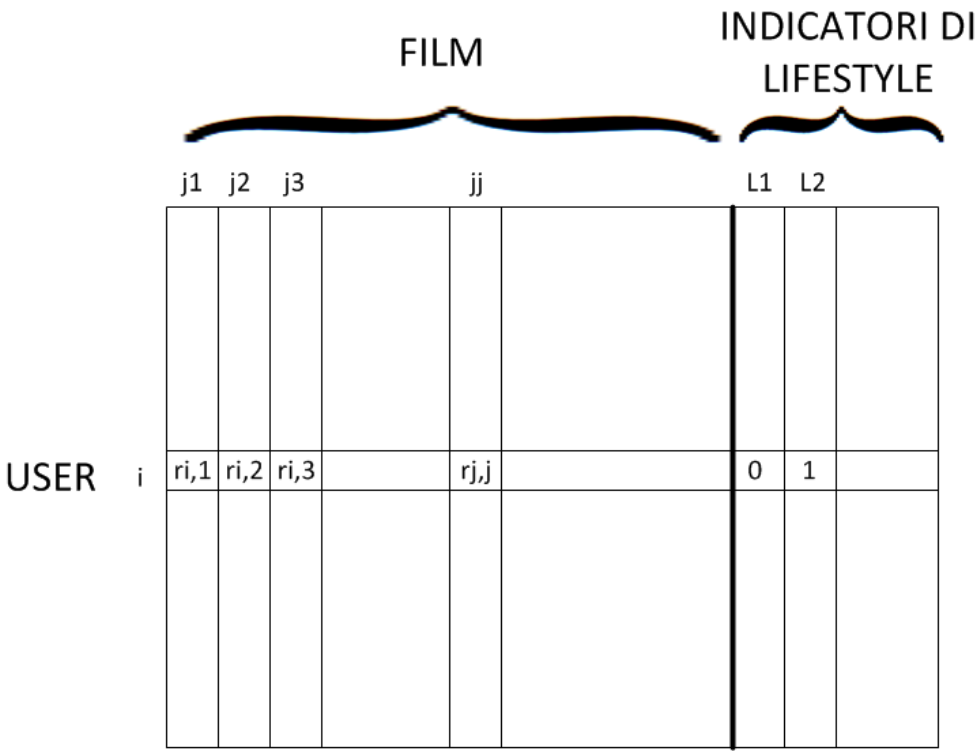

Figura 5.3: URMplus

raccomandazione di item, non utile ai fini del lavoro di tesi.

Per i motivi sopraelencati si rende necessario l'utilizzo di algoritmi di raccomandazione ibridi.

5.3 Scelta dell'algoritmo di raccomandazione

Una volta creata la matrice URMplus, che unisce informazioni collaborative e di contenuto, è possibile usare un qualsiasi algoritmo collaborativo. La scelta, quindi, viene effettuata in base alle caratteristiche e alla reversibilità dell'algoritmo rispetto al problema in questione. A tal proposito la Singular Value Decomposition (SVD) è stato ritenuto il migliore approccio al fine di generare un modello per effetture la raccomandazione, sfruttando proprietà di decomposizione matriciale.
Gli algoritmi basati sull'SVD sono i più applicati nei sistemi di raccomandazione per gli ottimi risultati che si ottengono e la relativa semplicità d'implementazione. L'SVD è stato proposto da [25] ed ampiamente usato per varie ragioni, tra le quali:

- lo spazio dimensionale del problema viene ridotto, in questo modo è possibile gestire più facilmente una grande quantità di dati;
- il rumore di fondo contenuto nei dati viene attenuato, come ad esempio il rumore causato da eventuali rating inseriti da un utente in modo casuale.

In algebra lineare la SVD è una particolare fattorizzazione basata sull'uso di autovalori e autovettori, utilizzata per approssimare la matrice originaria. Sia $A \in \mathbb{C}^{m \times n}$, allora esiste una fattorizzazione della stessa nella forma:
$A = U \cdot S \cdot V^*$
dove:

- U è una matrice unitaria di dimensioni $m \times r$
- S è una matrice diagonale di dimensioni $r \times r$
- V^* è la trasposta coniugata di una matrice unitaria di dimensioni $r \times n$

Gli elementi della diagonale di S sono detti valori singolari di A e hanno la proprietà di essere tutti quanti positivi, $s_i \geq 0 \quad \forall i$, e $s_1 \geq s_2 \geq \ldots \geq s_n$. Si

può dimostrare che il rango della matrice A è uguale a quello della matrice S. In particolare si osserva che il rango di S dipende dai valori singolari ed è proprio uguale al numero di valori singolari diversi da zero. Supponiamo di avere una matrice A con rango $rk(A) = r$, allora si ha che $s_1 \geq s_2 \geq \ldots \geq s_r > s_{r+1} = \ldots = s_n = 0$ e la decomposizione SVD di A è definita come:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix} =$$

$$= \underbrace{\begin{pmatrix} u_{11} & u_{12} & \ldots & u_{1r} \\ u_{21} & u_{22} & \ldots & u_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ u_{m1} & u_{m2} & \ldots & u_{mr} \end{pmatrix}}_{U} \cdot \underbrace{\begin{pmatrix} s_1 & 0 & \ldots & 0 \\ 0 & s_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & s_r \end{pmatrix}}_{S} \cdot \underbrace{\begin{pmatrix} v_{11} & v_{12} & \ldots & v_{1n} \\ v_{21} & v_{22} & \ldots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{r1} & v_{r2} & \ldots & v_{rn} \end{pmatrix}}_{V^*}$$

Il rango della matrice A, e di conseguenza della matrice singolare S, fornisce la dimensione effettiva delle tre matrici $U_{[m\times r]}, S_{[r\times r]}, V^*_{[r\times n]}$.
Le r colonne della matrice U e le r righe della matrice V^* rappresentano gli autovettori ortogonali associati agli r autovalori rispettivamente di $A \cdot A^T$ e $A^* \cdot A$.
In altre parole, le r colonne di U corrispondono ai valori singolari diversi da zero dello spazio delle colonne della matrice A e le r righe di V^* corrispondono ai valori singolari diversi da zero corrispondenti allo spazio delle righe della matrice A.
Inoltre U e V^* , essendo due matrici unitarie, godono della seguente proprietà: $U \cdot U^* = I \quad \text{e} \quad V \cdot V^* = I$
Gli algoritmi che utilizzano l'SVD sono costituiti da due fasi:

- Generazione del modello
- Generazione della raccomandazione

5.3.1 Generazione del modello

Come accade nella maggioranza degli algoritmi di raccomandazione, anche nel caso dell'SVD la generazione del modello è sicuramente la fase, all'interno del processo di raccomandazione, che ha maggiore complessità computazionale. Per generare il modello si parte dalla rappresentazione dei dati come matrice URM dei rating.
La generazione del modello coincide con la scomposizione algebrica della URM nelle matrici U, S, V. La matrice URM viene infatti scomposta nel seguente modo:

$$[U, S, V] = SVD(URM) \tag{5.1}$$

Se l'URM è una matrice di m righe per n colonne con rango minimo r, in S sono contenuti gli r autovalori non negativi della URM (ordinati in maniera decrescente), in U sono contenuti gli m autovettori relativi agli utenti e in V gli n autovettori relativi agli item. La decomposizione così prodotta ha dimensione r pari al rango minimo della matrice URM.
A questo punto si introduce una dimensione latente k tale che $k < rango(URM)$, in modo tale che la scomposizione diventi:

$$[U_k, S_k, V_k] = SVD(URM, k) \tag{5.2}$$

Con l'introduzione della dimensione latente k si riducono le dimensioni delle matrici U, S e V e allo stesso tempo si conferisce al modello la capacità di generalizzare sugli elementi che erano 0 nella matrice di partenza, oltre che di eliminare il rumore presente nei dati. Le matrici U_k, S_k, V_k rappresentano quindi il modello e verranno usate per generare la raccomandazione, come descritto nella seconda fase di generazione della raccomandazione.
C'è da aggiungere inoltre che l'SVD è una decomposizione matriciale nota in letteratura da tempo, è quindi stata ampiamente discussa ed ottimizzata. Questa decomposizione può essere infatti applicata anche ad URM di notevoli dimensioni. In particolare le versioni dell'algoritmo ottimizzate per matrici sparse riescono a decomporre una matrice in tempi proporzionali al numero di elementi non-zero (e non proporzionali alla dimensione reale della matrice).

5.3.2 Generazione della raccomandazione

A partire dalle matrici U_k e V_k create nella fase di generazione del modello, si generano le matrici $PseudoUser$ e $PseudoItem$ (tramite la moltiplicazione per $\sqrt{S_k}$). Definiamo quindi il vettore $pseudoUser(i)$ (riferito all'utente i-esimo) e la matrice $PseudoItem$ nel seguente modo:

$$pseudoUser(i)_k = U(i)_k * \sqrt{S_k} \tag{5.3}$$

$$PseudoUser_k = \underbrace{\begin{pmatrix} u_{11} & u_{12} & \dots & u_{1k} \\ u_{21} & u_{22} & \dots & u_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ u_{m1} & u_{m2} & \dots & u_{mk} \end{pmatrix}}_{U_k} \cdot \underbrace{\begin{pmatrix} \sqrt{s_1} & 0 & \dots & 0 \\ 0 & \sqrt{s_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sqrt{s_k} \end{pmatrix}}_{\sqrt{S_k}}$$

$$PseudoItem_k = \sqrt{S_k} * V_k \tag{5.4}$$

$$PseudoItem_k = \underbrace{\begin{pmatrix} \sqrt{s_1} & 0 & \dots & 0 \\ 0 & \sqrt{s_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sqrt{s_k} \end{pmatrix}}_{\sqrt{S_k}} \cdot \underbrace{\begin{pmatrix} v_{11} & v_{12} & \dots & v_{1n} \\ v_{21} & v_{22} & \dots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{k1} & v_{k2} & \dots & v_{kn} \end{pmatrix}}_{V_k^*}$$

- $pseudoUser(i)_k$ è il profilo dell'utente i-esimo proiettato nella dimensione latente k, è quindi un vettore di k elementi che si ottiene moltiplicando gli elementi del profilo utente i (riga i-esima della matrice U_k) per i k valori (le prime k righe/colonne) della matrice $sqrtS_k$.
- $PseudoItem_k$ è la matrice che contiene i profili proiettati di tutti i film. Si ottiene moltiplicando le prime k righe/colonne di S_k per le prime k righe della matrice $\sqrt{V_k}$.

La raccomandazione per l'utente i-esimo sarà quindi calcolata nel seguente modo:

$$recList(i) = pseudoUser(i)_k \cdot PseudoItem_k \tag{5.5}$$

L'SVD permette quindi di passare dal profilo dell'utente i-esimo (che è composto dai suoi ratings sui film visti) allo $pseudoUser(i)_k$ (che invece è un

vettore di k elementi); si può quindi dire che il profilo utente viene sintetizzato nelle k dimensioni dello $pseudoUser(i)_k$. Per i film si adotta lo stesso procedimento, i rating dati ad ogni film vengono ridotti anch'essi ad un vettore di k elementi. Si è quindi effettuata una *riduzione dimensionale* del problema.
A questo punto la raccomandazione all'utente i sul film j si ottiene moltiplicando elemento per elemento i vettori $pseudoUser(i)_k$ e $pseudoItem(j)_k$ e sommando il vettore così ottenuto.

$$recList(i) = \underbrace{\begin{pmatrix} pu_{i1} & pu_{i2} & \dots & pu_{ik} \end{pmatrix}}_{pseudoUser(i)_k} \cdot \underbrace{\begin{pmatrix} pi_{11} & pi_{12} & \dots & pi_{1n} \\ pi_{21} & pi_{22} & \dots & pi_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ pi_{k1} & pi_{k2} & \dots & pi_{kn} \end{pmatrix}}_{PseudoItem_k} \quad (5.6)$$

Occorre però modificare la costruzione del modello e la generazione della raccomandazione, poichè l'obiettivo, in questo lavoro di tesi, non è raccomandare item, ma *lifestyle*.
Nelle sezioni che seguono si descrivono l'algoritmo implementato, che utilizza l'SVD per il processo d'inferenza ed i suoi risultati.

5.4 Algoritmo proposto

L'obiettivo dell'algoritmo qui descritto è quello di individuare, per ciascun indicatore di lifestyle, la soglia di binarizzazione che consente di ottenere la precision desiderata sulla ULM inferita della popolazione. L'algoritmo contiene due parametri:

- K: numero di partizionamenti del campione
- C: valore da attribuire alle soglie che non sono state individuate nel test poichè la precision richiesta è troppo alta.

Gli algoritmi sviluppati e descritti in questo capitolo adoperano delle modiche sulla matrice *URMplus*. In particolare, tale matrice viene normalizzata,

come suggerito da [6] per ridurre i *global effects*, cioè le tendenze e attitudini comuni degli utenti nel valutare i film. La normalizzazione prevedere due comportamenti diversi a seconda della parte di matrice *URMplus* in questione:

- per la parte relativa alla matrice URM (contenente i rating) si normalizza sottraendo 3 ai valori diversi da 0;
- per la parte relativa alla matrice ULM (contenente gli indicatori di lifestyle) si normalizza assegnando 2 agli indicatori uguali a 1 , e -1 agli altri.

Il risultato di questo processo di normalizzazione è la matrice *URMplusNorm* su cui ci si basa per l'algoritmo in questione.
L'algoritmo proposto unisce la tecnica dell'SVD utilizzata nel processo d'inferenza, la confidence analysis che consente di individuare una soglia di binarizzazione che permette di raggiungere la precision richiesta per l'indicatore di lifestyle e il k-fold cross validation.
Di seguito viene proposto uno pseudocodice di questo algoritmo.

```
1  per ogni k:
2      VALIDATION = k-esima parte di CAMPIONE;
3      TRAIN = CAMPIONE - VALIDATION;
4      esecuzione del processo d'inferenza del lifestyle
           (TRAIN, VALIDATION);
5      per ogni precision n
6          si applica la confidence analysis per
               identificare la soglia che garantisce
               precision n;
7          se test positivo: salva soglia individuata;
8          se test negativo perchè la soglia è troppo
               alta
9              assegna soglia = C;
10     end
11 end
```

```
12 per ogni precision n:
13     calcola la soglia media tra le k soglie
           individuate
14 end
15 estrai soglia per precision richiesta
16 TRAIN = CAMPIONE;
17 TEST = popolazione che non ha risposto al questionario
18 esegui processo d'inferenza (TRAIN, TEST,
       soglia_individuata);
19 end
```

Riassumendo lo pseudocodice sopra mostrato, la matrice CAMPIONE viene divisa (riga 2 e 3 dello pseudocodice) in TRAIN e VALIDATION, come descritto nella tecnica del k-fold cross validation. In riga 4 viene eseguito il processo d'inferenza grazie all'algoritmo SVD, ottenendo per la precision richiesta una soglia di binarizzazione (riga 6).
Ogni soglia di binarizzazione (in totale k*n) viene individuata applicando la confidence analysis, come descritto nel capitolo precedente. Qualora non fosse possibile identificare una soglia, poichè la precision richiesta è troppo alta, l'algoritmo assegna come soglia un valore costante C (riga 9), valido per il calcolo della soglia media. In questo caso il test viene comunque considerato come test negativo. Si ritiene che una precision sia raggiungibile se la soglia relativa è stata individuata in almeno k/2 test. Il caso contrario indica che non è possibile garantire il raggiungimento della precision richiesta sull'indicatore di lifestyle corrente nella ULM della popolazione.
Riassumendo, il k-fold permette di ricavare n soglie di binarizzazione, ognuna ottenuta dalla media delle k soglie calcolate per ottenere la n-esima precision. La soglia di binarizzazione consente di ottenere la precision richiesta sull'indicatore di lifestyle corrente.
Selezionata la soglia ideale di ogni indicatore di lifestyle, ottenuta come media tra le k soglie calcolate per ricavare la precision richiesta, viene eseguito in riga 18 l'algoritmo di inferenza (l'SVD) in cui si usa il campione per generare il modello (train), che viene applicato sulla popolazione che non ha risposto al questionario.

In figura 5.4 è rappresentato un diagramma che sintetizza l'algoritmo, dove viene richiesta una certa precisione p. Dalla matrice CAMPIONE vengono ricavate le due soglie di binarizzazione (per l'indicatore di lifestyle UOMO e DONNA) e viene applicato il processo d'inferenza alla matrice TEST per ricavare poi precision e recall sia per UOMO che per DONNA.

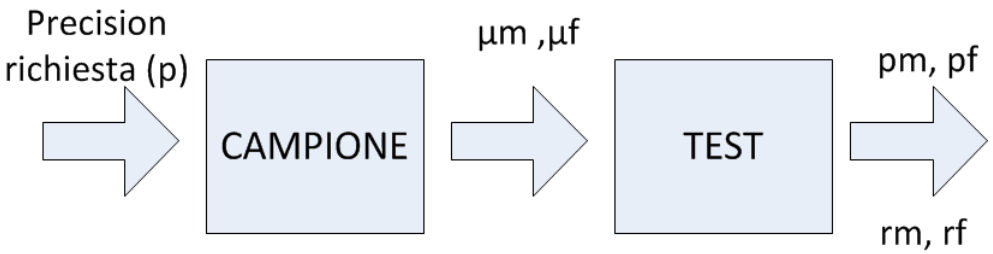

Figura 5.4: Diagramma dell'algoritmo con confidence analysis e k-fold cross validation

5.4.1 Varianti

In questo paragrafo vengono descritti altri algoritmi che sono stati implementati in maniera quasi identica all'algoritmo base mostrato sopra, ma con alcune modifiche. In particolare le varianti prese in considerazione sono:

- Variante 1: ULM a singola colonna con dataset non bilanciato
- Variante 2: ULM a due colonne con dataset bilanciato
- Variante 3: ULM a singola colonna con dataset bilanciato

Schematicamente gli algoritmi creati in questo capitolo possono essere schematizzati come in tabella 5.1.

Viene introdotto il concetto di dataset bilanciato che consiste nel ricavare, dal dataset, una matrice URM con un uguale numero di utenti di genere maschile e femminile. Le varianti che utilizzano un dataset bilanciato vogliono verificare se i risultati ottenuti da dataset con un maggior numero di utenti di genere maschile possano essere influenzati da questa disparità.

Le varianti 1 e 3 consistono nell'uso di una singola colonna per indicare il sesso degli utenti. Da questo ne deriva anche un cambiamento della

	ULM 1 colonna	**ULM 2 colonne**
Dataset non bilanciato	Variante 1	Caso base
Dataset bilanciato	Variante 3	Variante 2

Tabella 5.1: Varianti algoritmo di confidence analysis e k-fold cross validation

matrice degli indicatori di lifestyle (ULM). Nel dettaglio la matrice ULM diventa un vettore, dove viene indicato con 0 e con 1 rispettivamente il sesso UOMO e DONNA. Cambia anche la sua normalizzazione, all'indicatore di lifestyle 0 viene assegnato -2, invece a quello 1 il valore stabilito è 2. Le varianti che utilizzano la matrice ULM formata da una sola colonna sono state implementate per confrontare il diverso funzionamento dell'algoritmo SVD rispetto al caso base con due colonne.

5.5 Risultati

In questa sezione si presentano i risultati ottenuti grazie all'applicazione dell'algoritmo base e delle sue variante al dataset Movielens.
Nell'uso dell'algoritmo SVD il modello viene generato introducendo la dimensione latente, il cui valore deve essere impostato dall'utente al momento dell'avvio dell'algoritmo.
Come detto in precedenza i test sono stati effettuati solo sull'indicatore di lifestyle riguardante il genere sessuale: UOMO o DONNA. Questo algoritmo, come tutti gli altri sviluppati, applica una soglia di binarizzazione esclusiva, ovvero assegna all'utente un solo lifestyle (o UOMO o DONNA).

I test effettuati hanno utilizzato come valore dei parametri:

- Q (percentuale della matrice CAMPIONE rispetto all'URM) = 10
- C (soglia da attribuire alle soglie che non sono state individuate nel test poichè la precision richiesta è troppo alta) = massimo tra 1 e il massimo valore soglia individuato nei test positivi

- K (percentuale del TRAIN SET nel k-fold cross validation rispetto alla matrice CAMPIONE) = 75
- DM (dimensione latente) = 15

In questi test, viene scelto un valore basso di Q, che però crei una matrice CAMPIONE in modo tale da avere un sostanziale training set; la popolazione che non ha risposto al questionario è pari, quindi, al restante del 90% della matrice URM.
La scelta del valore della dimensione latente effettuata in questi test va ad influire sulla qualità del modello, e quindi dei risultati. Il valore adottato (15) è presente come dimensione latente in molti algoritmi che utilizzano l'SVD per la raccomandazione nell'area dell'IPTV ed offre ottimi risultati.
I test negativi sono dovuti ad uno dei seguenti motivi:

- la soglia non è stata individuata perchè la precision richiesta non è raggiungibile nel set di validazione usato. In questo caso, l'algoritmo assegna alla soglia il valore di C, come descritto nello pseudocodice precedente.
- l'algoritmo accetta soglie tali da raggiungere la precision richiesta con una tolleranza minima, individuata in questi test nel 2%. Se non è stato possibile individuare una precision in tale intervallo di tolleranza allora il caso viene identificato come test fallito

In ognuno dei quattro algoritmi implementati e descritti in questo capitolo vengono mostrate le Q soglie di binarizzazione ottenute per ogni indicatore di lifestyle. A differenza di quelli descritti nel capitolo successivo, per questi algoritmi non viene calcolata la curva di ROC, in quanto non viene fatta variare la soglia di binarizzazione, ma ne viene scelta una sola che riesca ad ottenere la precision richiesta. Inoltre vengono calcolati precision e recall di ogni algoritmo senza prendere in considerazione uno specifico indicatore. Il confronto dei risultati ottenuti avviene attraverso l'analisi di:

$$precision = \frac{precisionM * maschio + precisionF * femmina}{maschio + femmina} \tag{5.7}$$

$$recall = \frac{recallM * maschio + recallF * femmina}{maschio + femmina} \tag{5.8}$$

Nello specifico le variabili utilizzate sono:

- *precisionM*, *recallM* che sono la precision e la recall che si ottengono sull'indicatore di lifestyle UOMO
- *precisionF*, *recallF* che sono la precision e la recall che si ottengono sull'indicatore di lifestyle DONNA
- *maschio* e $femmina$ che sono rispettivamente il numero di uomini e quello delle donne a cui è stato assegnato un lifestyle (giusto o sbagliato).

Caso Base

Il caso base riguarda l'algoritmo che utilizza il dataset originario di Movielens, con le informazioni sugli indicatori dei lifestyle su doppia colonna. Di seguito sono riportate, in tabella 5.2 e 5.3, le soglie di binarizzazione individuate dall'algoritmo eseguendo l'algoritmo Q volte per gli indicatori UOMO e DONNA.

Il test in tabella 5.2 non fallisce mai e inoltre si riesce ad ottenere una preci-

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
q=1	0	0.2	1.3
q=2	0	0	1
q=3	0	0	1.2
q=4	0	0	1.3
q=5	0	0	1
q=6	0	0.15	0.45
q=7	0	0	0.85
q=8	0	0	0.99
q=9	0	0	0.72
q=10	0	0	1.23
Soglia media	0	0.035	1.004
Recall ottenuta	76.03%	33.82%	3.51%

Tabella 5.2: Soglie lifestyle per l'indicatore di lifestyle UOMO - Caso base

sion dell'80%, con una soglia di binarizzazione pari a 0, cioè senza applicarla, avendo anche un'ottima recall.

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
q=1	0	1.5	1.5
q=2	1	Fail	Fail
q=3	Fail	Fail	Fail
q=4	0.3	Fail	Fail
q=5	Fail	Fail	Fail
q=6	Fail	Fail	Fail
q=7	0.64	Fail	Fail
q=8	0.5	Fail	Fail
q=9	0	0.97	1.5
q=10	0	0.6	1.12
Soglia media	0.694	Fail	Fail
Recall ottenuta	13.50%	Fail	Fail

Tabella 5.3: Soglie lifestyle per l'indicatore di lifestyle DONNA - Caso base

L'algoritmo nei test riportati in tabella 5.3 utilizza $C = 1.5$. In questo caso i risultati ottenuti nei test con precision richiesta al 90% e 100% falliscono in molti casi e perciò le precision non sono raggiungibili. Questi risultati sono dovuti, come descritto nel capitolo precedente, al numero di utenti di genere maschile di gran lunga maggiore rispetto a quelli di genere femminile (4331 uomini contro 1709 donne) nel dataset Movielens.
Per ogni precision richiesta (80%, 90% e 100%) nella tabella 5.4 vengono mostrati i valori di precision e recall del caso generale, senza cioè calcolarli rispetto ad un singolo indicatore di lifestyle.
I risultati di tabella 5.4 sono influenzati dai numerosi test falliti per quanto riguarda l'indicatore DONNA; ma, per la precision richiesta dell'80% si riescono ad ottenere ottimi risultati di precision e recall.

	Precision richiesta 80%	Precision richiesta 90%	Precision richiesta 100%
Precision ottenuta	87.32%	Fail	Fail
Recall ottenuta	71.94%	Fail	Fail

Tabella 5.4: Precision e recall - Caso base

Variante 1

La variante 1 utilizza sempre il dataset originale di Movielens, ma viene modificata la matrice ULM che contiene gli indicatori di lifestyle indicanti il sesso degli utenti. Tale matrice viene trasformata in un vettore che, quindi, su un'unica colonna contiene tutte le informazioni del sesso degli utenti.
Nelle tabelle 5.5 e 5.6 vengono visualizzate le soglie di binarizzazione individuate dall'algoritmo eseguendo il k-fold cross validation e la confidence analysis per gli indicatori UOMO e DONNA.
I risultati ottenuti in questi casi sono simili al caso base precedente, ma con valori di soglia di binarizzazione più alta per ottenere la precision richiesta. Inoltre nel caso DONNA diminuiscono sensibilmente i test falliti. In tabella 5.6 l'algoritmo utilizza $C = 1.43$.
I risultati ottenuti nei test con Precision richiesta al 100% falliscono in tutti i k casi e perciò la precision non è raggiungibile.
Per ogni precision richiesta (80%, 90% e 100%) nella tabella 5.7 vengono mostrati i valori di precision e recall del caso generale ottenuti da questa variante di algoritmo
In particolare per il caso di precision richiesta al 80% si ottengono ottimi risultati, invece, per la precision al 90%, la precision ottenuta è inferiore, anche se di poco, a quella richiesta.

Variante 2

L'algoritmo indicato come variante 2, è uguale al Caso base, ma utilizza un dataset modificato; viene utilizzato un dataset con un uguale numero di uomini e donne. Viene quindi analizzata la matrice URM originale e siccome il numero di utenti maschili è maggiore di quelli femminili, vengono eliminati,

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
q=1	0	0.1	Fail
q=2	0.13	0.45	Fail
q=3	0	0.05	Fail
q=4	0.02	0.18	Fail
q=5	0	0.04	Fail
q=6	0	0.12	Fail
q=7	0	0.17	Fail
q=8	0.01	0.13	Fail
q=9	0.08	0.24	Fail
q=10	0.09	0.26	Fail
Soglia media	0.033	0.187	Fail
Recall ottenuta	76.84%	47.21%	Fail

Tabella 5.5: Soglie lifestyle per l'indicatore di lifestyle UOMO - Variante 1

casualmente, un numero di utenti UOMO tali da creare un dataset bilanciato sul genere sessuale. Gli utenti DONNA vengono tutti inseriti nella nuova matrice. In seguito, alla matrice URM e alla sua associata ULM viene applicato un mescolamento casuale delle righe della matrice, tale da rendere le k parti della matrice statisticamente simili. Infatti, dopo l'eliminazione di utenti maschili, si possono avere molte righe che rappresentano utenti dello stesso genere sessuale e questo apporterebbe problematiche per l'utilizzo del k-fold cross validation.

Anche in questo caso vengono riportate due tabelle, 5.8 e 5.9, che visualizzano le soglie di binarizzazione individuate dall'algoritmo eseguendo il k-fold cross validation e la confidence analysis per gli indicatori UOMO e DONNA.

I risultati ottenuti evidenziano, rispetto al caso base, una diminuzione di prestazioni dei risultati riferiti solo all'indicatore UOMO e un conseguente aumento per quello DONNA. I test con precision richiesta al 90% falliscono solo per l'indicatore DONNA, mentre per quella al 100% falliscono sia per UOMO che per DONNA in tutti i Q casi.

	Precision richiesta 80%	Precision richiesta 90%	Precision richiesta 100%
q=1	0.23	0.31	Fail
q=2	1.20	1.43	Fail
q=3	Fail	Fail	Fail
q=4	0.2	0.23	Fail
q=5	0.14	0.22	Fail
q=6	0.4	0.66	Fail
q=7	Fail	Fail	Fail
q=8	0.74	0.74	Fail
q=9	0.14	0.25	Fail
q=10	0.43	0.43	Fail
Soglia media	0.634	0.713	Fail
Recall ottenuta	15.47%	7.93%	Fail

Tabella 5.6: Soglie lifestyle per l'indicatore di lifestyle DONNA - Variante 1

Per ogni precision richiesta (80%, 90% e 100%) nella tabella 5.10 vengono mostrati i valori di precision e recall. I risultati ottenuti, relativamente al caso di precision richiesta all'80%, mostrano un'ottima precision, ma una recall bassa rispetto ai casi precedenti.

Variante 3

L'algoritmo denominato variante 3 utilizza, come nel caso precedente, il dataset bilanciato sul genere sessuale di Movielens. Viene inoltre utilizzata, come nella variante 1, il vettore ULM e non la matrice, dove cioè in una singola colonna vengono inserite le informazioni sul genere sessuale degli utenti.
Anche per quest'ultimo algoritmo descritto nel capitolo, vengono calcolate ed illustrate nelle tabelle 5.11 e 5.12 le soglie di binarizzazione individuate dall'esecuzione del k-fold cross validation e confidence analysis per gli indi-

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
Precision ottenuta	83.12%	87.17%	Fail
Recall ottenuta	74.59%	46.13%	Fail

Tabella 5.7: Precision e recall - Variante 1

catori UOMO e DONNA.

L'algoritmo, in tabella 5.11, utilizza $C = 1.3$. Invece in tabella 5.12 si utilizza un valore di C pari a 1.22. I risultati ottenuti in queste tabelle, rispetto a quelli rispettivi ottenuti nel caso Variante 1, mostrano una soglia media ben più alta, ma una recall più equilibrata tra i due indicatori di lifestyle. Per ultimo per ogni precision richiesta (80%, 90% e 100%) nella tabella 5.13 vengono mostrati i valori di precision e recall compelessivi. Nel primo caso si riesce ad ottenere un'ottima precision, a differenza della recall; mentre i risultati del secondo caso, quello con precision richiesta 90%, sono influenzati dai test falliti per l'indicatore DONNA.

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
q=1	0.4	1.2	Fail
q=2	0.35	1	Fail
q=3	0.47	1.3	Fail
q=4	0.2	1.3	Fail
q=5	0.22	1	Fail
q=6	0.15	0.45	Fail
q=7	0.62	0.75	Fail
q=8	0.58	1.29	Fail
q=9	0.12	0.77	Fail
q=10	0.65	1.3	Fail
Soglia media	0.376	1.036	Fail
Recall ottenuta	47.16%	23.12%	Fail

Tabella 5.8: Soglie lifestyle per l'indicatore di lifestyle UOMO - Variante 2

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
q=1	0	Fail	Fail
q=2	0.4	Fail	Fail
q=3	0.26	Fail	Fail
q=4	0.3	Fail	Fail
q=5	0.11	Fail	Fail
q=6	0.13	Fail	Fail
q=7	0.23	Fail	Fail
q=8	0.3	Fail	Fail
q=9	0.09	0.97	Fail
q=10	0	0.99	Fail
Soglia media	0.182	Fail	Fail
Recall ottenuta	39.12%	Fail	Fail

Tabella 5.9: Soglie lifestyle per l'indicatore di lifestyle DONNA - Variante 2

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
Precision ottenuta	85.45%	Fail	Fail
Recall ottenuta	43.14%	Fail	Fail

Tabella 5.10: Precision e recall - Variante 2

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
q=1	0.50	1.2	Fail
q=2	0.66	1	Fail
q=3	Fail	Fail	Fail
q=4	0.73	1.3	Fail
q=5	0.28	1	Fail
q=6	0.55	0.55	Fail
q=7	0.72	0.75	Fail
q=8	Fail	Fail	Fail
q=9	0.47	0.77	Fail
q=10	0.54	1.3	Fail
Soglia media	0.705	1.047	Fail
Recall ottenuta	56.97%	14.82%	Fail

Tabella 5.11: Soglie lifestyle per l'indicatore di lifestyle UOMO - Variante 3

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
q=1	Fail	Fail	Fail
q=2	0.99	1.22	Fail
q=3	0.56	0.56	Fail
q=4	0.22	0.63	Fail
q=5	0.20	0.93	Fail
q=6	0.64	Fail	Fail
q=7	Fail	Fail	Fail
q=8	0.96	Fail	Fail
q=9	0.53	Fail	Fail
q=10	0.70	Fail	Fail
Soglia media	1.37	Fail	Fail
Recall ottenuta	25.12%	Fail	Fail

Tabella 5.12: Soglie lifestyle per l'indicatore di lifestyle DONNA - Variante 3

	Precision richiesta 80%	**Precision richiesta 90%**	**Precision richiesta 100%**
Precision ottenuta	87.54%	Fail	Fail
Recall ottenuta	41.04%	Fail	Fail

Tabella 5.13: Precision e recall - Variante 3

Capitolo 6

Algoritmi basati su regole d'associazione

Indice

In questo capitolo vengono descritti algoritmi basati sulle regole d'associazione descritte nel Capitolo 2. In particolare vengono mostrate due metodologie di sviluppo: algoritmi che basano il processo d'inferenza sulle regole d'associazione come illustrato nella maggior parte dei testi di letteratura (sezione 6.2), ed algoritmi che utilizzano un approccio alternativo per la creazione di regole d'associazione, chiamate in questa tesi pseudo regole d'associazione (sezione 6.3). Sia i primi algoritmi che i secondi sono entrambi finalizzati al raggiungimento del medesimo scopo: dedurre il *lifetyle* degli

utenti IPTV, per categorizzarli successivamente.

6.1 Dominio d'associazione

Lo scopo degli algoritmi presentati nelle prossime sezioni è quello di creare un insieme di regole d'associazione a partire da un insieme di addestramento (training set), per inferire il *lifestyle* dell'insieme di persone di cui non si hanno tali informazioni (test set).
Le regole sono formate, come detto nel paragrafo 2.5, da un antecedente X e un conseguente Y, cioè del tipo

$$X \Rightarrow Y$$

Contestualizzando le regole d'associazione al lavoro di tesi, si divide la matrice *URMplus* in TEST e CAMPIONE, come svolto nei precedenti algoritmi, e si ricavano le regole dalla sottomatrice CAMPIONE, adottate poi in un processo d'inferenza sulla matrice TEST.

Per la creazione di regole, possiamo avere due politiche di approccio:

- Regole d'associazione per item: ad esempio,

  $$Item_1, Item_2, Item_3 \Rightarrow UOMO$$

  indica che una persona, a cui piacciono i film con ID $1, 2$ e 3, viene considerata di sesso maschile.

- Regole d'associazione per genere: ad esempio,

  $$Genere_1, Genere_2 \Rightarrow DONNA$$

  indica che una persona, a cui piacciono i generi di film con ID 1 e 2, viene etichettata come persona di sesso femminile.

Si vuole, cioè, creare regole che abbiano come antecedente un insieme di item o di generi e come conseguente il *lifestyle*, riferito esclusivamente in questa tesi al genere sessuale: *UOMO* o *DONNA*.
Come per l'algoritmo precedente, anche negli algoritmi che usano le regole d'associazione, la matrice, che rappresenta il dataset, viene divisa in Q parti

uguali. Delle Q parti, una (il training set) viene utilizzata per la creazione delle regole d'associazione e le altre Q-1 rappresentano l'insieme di utenti della popolazione di cui non si conoscono informazioni sul *lifestyle*. Il processo viene eseguito Q volte, ognuna delle quali vede una Q parte diversa come training data e le rimanenti come test set.

Nelle successive sezioni verranno mostrate tecniche diverse per la creazione di regole, ma tutte sono state implementate in uno dei due modi sopracitati (per item o per genere).

6.2 Regole d'associazione

Prima della creazione vera e propria delle regole d'associazione è necessario apportare delle modifiche al dataset utilizzato.
In particolare, è stato utilizzato un dateset bilanciato tra il numero di utenti maschili e quelli femminili, in quanto la *URM* originale presenta un maggior numero di utenti con *lifestyle* UOMO, piuttosto che DONNA. Questo accorgimento è stato adoperato per evitare la possibile creazione di molte regole con conseguente UOMO, le quali possono così falsare il processo d'inferenza. Inoltre vengono presi in considerazione solo film, visti dagli utenti, che hanno avuto una forte valutazione positiva. Questo è possibile andando a cancellare, dalla lista di film visti da ogni singolo utente presente nella matrice URM, quelli che hanno avuto un voto basso (nel caso in esame, sono stati eliminati i film con rating minore di 4, in una scala da 1 a 5). I rating con valore 4 e 5, invece, vengono posti uguale ad 1, creando cosi una nuova matrice URM binaria, che viene rinominata in *URMtop*. Come per l'algoritmo precedente, anche negli algoritmi che presentano le regole d'associazione è stata utilizzata una percentuale del 10% dell'intera matrice *URM* per creare la sottomatrice CAMPIONE.

Per queste regole d'associazione, dal database (la matrice URM) vengono prima cercati gli insiemi di item (gli itemset) più frequenti e poi da questi create le regole d'associazione. Per queste due fasi vengono utilizzati programmi già esistenti in letteratura. In particolare, per quanto riguarda la creazione di una lista di itemset frequenti, viene usato il programma *Linear time Closed itemset Miner version 2* (LCM2) [29], scritto in linguaggio C.

Un item è frequente se incluso in almeno uno specificato numero di transazioni; cioè vengono cercati gli item, o insiemi di item, che sono presenti con un numero maggiore di una certa soglia, definita come il supporto minimo imposto dall'utente (nel caso in esame pari a 50). Un esempio di output potrebbe essere:

$$5(132)$$

che indica che l'item 5 ha supporto 132.
Per la creazione di regole dalla lista di itemset frequenti è stato utilizzato il programma *Rules* [13] scritto in C++, a cui viene passato come input il file generato da LCM2. Anche per questo caso si possono filtrare le regole da prendere in considerazione, che devono superare un altro valore di soglia, la confidenza minima (nel caso in esame pari a 0, cioè vengono prese in considerazione tutte le regole). Un esempio di output del programma *Rules* è:

$$15 \Rightarrow M(112, 0.632768)$$

La regola indica che la persona che visiona l'item 15 viene considerata UOMO. La regola ha supporto 112 e confidenza 0.632768.
Il programma Rules è stato modificato in quanto poteva creare anche regole del tipo:

$$UOMO \Rightarrow Item_1, Item_2$$

ma per il lavoro di tesi è essenziale che solo il conseguente della regola contenga il genere sessuale dell'utente, ad esempio se ad un utente piacciono gli item i_1, i_2 viene assegnato il genere $Uomo$ e non il contrario.

Come detto nella sezione precedente, sono state usate due strategie per la creazione delle regole d'associazione: per item o per genere, che comporteranno alcune modifiche alla matrice URM, descritte successivamente.
Per la creazione di regole d'associazione sul dominio degli item, l'algoritmo è così sviluppato:

1. La matrice *URMtop* viene divisa, tramite MATLAB, in TEST e CAMPIONE, con le percentuali prima menzionate;

2. la sottomatrice CAMPIONE viene accoppiata con la matrice del lifestyle, la ULM, contenente informazioni sul sesso degli utenti di tale matrice, formandone una nuova;

3. quest'ultima matrice rappresenta l'input del programma *LCM2*, che genera così un file contenente gli itemset più frequenti;

4. da questo file si generano le regole d'associazione, tramite il programma *Rules*. L'output viene analizzato e portato in MATLAB sotto forma di matrice;

5. viene scelto in base alle politiche che vedremo nei due prossimi sottoparagrafi quale sesso (UOMO o DONNA) inferire ad ogni utente della matrice TEST (che rappresenta, in questo caso, la popolazione di cui non si conosce il *lifestyle*);

6. viene applicato lo strumento della confidence analysis anche per questi algoritmi: al variare di una soglia λ viene o meno confermato la scelta del sesso dell'utente nel processo d'inferenza. Vengono cioè filtrati i risultati ottenuti nel punto precedente;

7. vengono calcolati i parametri utili al calcolo della *precision*, della *recall* e della curva di *ROC* dell'algoritmo sviluppato.

Il parametro λ è usato per osservare come variano la *precision*, la *recall* e la curva di *ROC*, al variare dello stesso λ. Qui di seguito viene mostrato, tramite la figura 6.1, uno schema riassuntivo dei passaggi chiavi dell'algoritmo.

Per quanto riguarda la creazione di regole d'associazione sul dominio dei generi dei film, la matrice *URMtop* viene moificata in *UGMtop* (*User-Genre Matrix*, dove sulle righe sono presenti gli utenti del dataset Movielens e sulle colonne i generi dei film preferiti da quell'utente.
Per definire il genere o l'insieme di generi preferiti da un utente, si usa il "principio di Pareto" o "legge 80/20" [18], che è sintetizzabile nell'affermazione: la maggior parte degli effetti è dovuta ad un numero ristretto di cause. I valori 80% e 20% sono ottenuti mediante osservazioni empiriche e sono solo indicativi, ma è interessante notare come numerosi fenomeni abbiano una distribuzione statistica in linea con questi valori. Questo principio

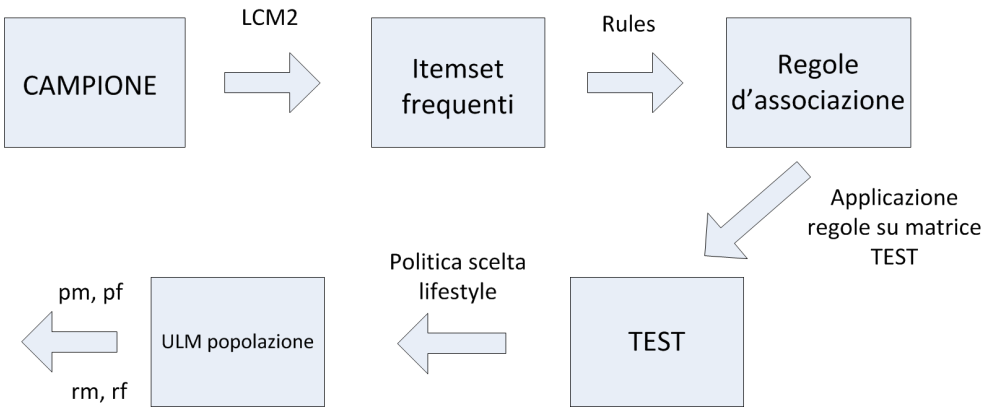

Figura 6.1: Diagramma dell'algoritmo con regole d'associazione

può avere diverse applicazioni pratiche in diversi settori, ad esempio: in economia: l'80% delle ricchezze è in mano al 20% della popolazione, oppure in informatica dove l'80% del tempo di esecuzione è impiegato solo dal 20% delle istruzioni di un programma.
Nel caso in esame, per ogni utente della matrice *URMtop* viene analizzata la lista di film visionati, e da ognuno di questi vengono estrapolati i generi (noti dal database Movielens). Di questi vengono presi solo i generi preferiti come descritto dal principio di Pareto, tenendo in considerazione solo quelli che rappresentano l'80% delle visualizzazioni tra tutti i generi. L'algoritmo poi procede come mostrato per le regole d'associazione con dominio i singoli film.

Nella prossima sezione viene mostrata una delle due politiche utilizzate per la scelta dell'assegnazione del *lifestyle* SESSO agli utenti della matrice TEST; quindi, si analizzeranno con maggior dettaglio i punti 5 e 6 della precedente enumerazione dello sviluppo dell'algoritmo.

6.2.1 Selezione regola singola

Dopo aver creato le regole d'associazione per item o per genere, sono state implementate due possibili politiche di scelta del *lifestyle* per gli utenti.
In questo paragrafo, si darà importanza ad una sola tra le regole create,

quella migliore, cioè quella con confidenza maggiore. Si ricorda che la creazione delle regole è affidato al programma *Rules*, che genera un output, dove per ogni regola vengono riporti due valori chiavi, il supporto e la confidenza. La politica in esame si comporta nel seguente modo:

- per ogni utente i della matrice TEST viene stilata una lista dei film o generi preferiti (a seconda se vogliamo utilizzare regole d'associazione per item o genere), chiamata $PREF_i$. Un film o un genere è considerato preferito dall'utente i se è presente nella matrice *URMtop* o *UGMtop* rispettivamente;

- viene analizzata ogni singola regola e se l'insieme dei dati che compongono l'antecedente è presente nella lista $PREF_i$, tale regola viene inserita in un'altra lista *REGOLE*, contenente tutte le regole potenzialmente utilizzabile per l'utente i;

- dopo aver analizzato tutte le regole precedentemente create, si cerca quella con confidenza maggiore nella lista *REGOLE*. In caso di valore di confidenza uguale per due o più regole, si andrà a confrontare il supporto e verrà presa quella con supporto maggiore.

- il conseguente (che è un valore del tipo: UOMO o DONNA) della regola scelta nel punto precendente diventa così il potenziale indicatore del *lifestyle* SESSO dell'utente i.

- viene, poi, applicato lo strumento della confidence analysis come detto in precedenza. Se la confidenza della regola scelta in precedenza è maggiore di una soglia λ, allora vengono confermati i risultati del punto precedente, altrimenti non viene assegnato alcun indicatore di *lifestyle*.

6.2.2 Selezione combinazione di regole

In questo secondo paragrafo viene mostrata la seconda politica utilizzata per la scelta dell'indicatore di *lifestyle* SESSO.

A differenza dell'algoritmo precedente, non viene data importanza alla

singola regola con confidenza maggiore, ma si sommano, in base al conseguente, i pesi di tutte le regole contenute nella lista *PREF* per poi decidere se assegnare l'indicatore UOMO o DONNA ad ogni utente della popolazione. I passaggi della politica di selezione di combinazione di regole sono:

- per ogni utente i della matrice TEST viene stilata la lista *PREFi*, nel modo descritto prima;
- viene creata anche la lista *REGOLE* contenente tutte le regole potenzialmente utilizzabile per l'utente i;
- vengono inizializzate due variabili: *maschio*, dove si sommano tutte le confidenze delle regole, contenute in *REGOLE*, che hanno conseguente la variabile UOMO, e $femmina$ con la somma delle confidenze di tutte le regole con conseguente la variabile DONNA. Nel caso di valori uguali di queste due variabili, si confrontano le regole nello stesso modo, ma rispetto al supporto.
- Se $maschio > femmina$, allora verrà assegnato all'utente i, l'indicatore di *lifestyle* UOMO, altrimenti DONNA.
- viene applicato anche in questo caso la confidence analysis. Se la differenza tra le due variabili *maschio* e $femmina$, in valore assoluto, è maggiore del parametro λ, allora viene confermato il risultato precedente, altrimenti non viene assegnato l'indicatore di *lifestyle* all'utente i.

Nel paragrafo seguente vengono mostrati i risultati ottenuti sia dalla politica di selezione della singola regola, che quella di combinazione di regole.

6.2.3 Risultati

Di questi algoritmi, come per quelli precedenti, vengono calcolati sia la precision e la recall al variare della soglia di binarizzazione e sia la curva di ROC; ognuno di essi viene calcolato relativamente al lifestyle UOMO, DONNA e nel caso senza nessuno indicatore particolare (caso generale). Per confrontare i risultati ottenuti per i singoli indicatori di lifestyle viene creata una

tabella riassuntiva, invece per il caso generale i valori ottenuti vengono visualizzati in un grafico.
Per quanto riguarda la soglia di binarizzazione λ si cerca di far variare il parametro in modo da trovare circa 10 valori, per una migliore valutazione dell'algoritmo. La scelta dei valori è empirica e varia da algoritmo ad algoritmo, in base alla politica utilizzata (singola regola o combinazione di regole).
Inoltre vengono visualizzati tramite tabelle e figure i valori ottenuti per la creazione della curva di ROC. Anche in questo caso le tabelle riporteranno i valori analitici utilizzati per la curva di ROC per il singolo indicatore UOMO o DONNA e graficamente verrà mostrata il grafico del caso generale, senza distinzione tra i due sessi. In ogni grafico la curva di ROC viene confrontata con un'ipotetica curva di ROC di un algoritmo casuale, dove cioè viene assegnato, senza alcuna politica, ad ogni utente della popolazione un indicatore di lifestyle in modo casuale. Tanto maggiore sarà la curva di ROC dell'algoritmo proposto rispetto a quella dell'algoritmo casuale ($TPR \gg FPR$), tanto maggiore sarà la qualità dell'algoritmo. Un valore di confronto tra i vari algoritmi, per quanto riguarda questo metodo valutativo, viene individuato nell'AUC, calcolata secondo [16].
Tutti i valori sono espressi in percentuali.
Gli algoritmi verranno confrontati attraverso i valori di precision e recall calcolati come nelle formule 5.7 e 5.8 del Capitolo 5 e specificità:

$$specificita = \frac{specificitaM * maschio + specificitaF * femmina}{maschio + femmina}$$

dove

- $specificitaM$ è la specificità che si ottiene sull'indicatore di lifestyle UOMO
- $specificitaF$ è la specificità che si ottiene sull'indicatore di lifestyle DONNA
- $maschio$ e $femmina$ sono rispettivamente il numero di uomini e quello delle donne a cui è stato assegnato un lifestyle (corretto od incorretto).

Valori di precision ritenuti validi devono essere al di sopra del 50%, che rappresenta l'ipotetica precision ottenuta da un algoritmo di assegnazione casuale del lifestyle. Per quanto riguarda la recall, tanto i suoi valori sono alti quanto grande sarà la percentuale di utenti considerati per la precision corrispondente.
Si ricorda che per costruire la curva di ROC sono necessarie le variabili: specificità e recall. Infatti l'asse delle ascisse di questa curva è formata dai valori della recall, mentre quella delle ordinate dal TPR ($= 1 - specificita$). Gli algoritmi sviluppati con la tecnica d'inferenza delle regole d'associazione sono quattro:

Regole d'associazione per item con selezione della singola regola

In tabella 6.1 e 6.2 vengono mostrati i risultati ottenuti da questo algoritmo, per l'indicatore UOMO e DONNA, al variare di λ tra 0.4 e 0.72 con un incremento di 0.4, ottenendo cosi 9 soglie di binarizzazione. Empiricamente sono state analizzate le regole d'associazione ed è stato visto che non ne esistono con confidenza minore di 0.4 o maggiore di 0.72. Si può notare da queste due tabelle che all'aumentare della soglia di binarizzazione, la recall cala drasticamente, mentre la precision, per l'UOMO cresce, per la DONNA dimunuisce anche se di poco. Inoltre si nota che per i valori di soglia pari a 0.68 e 0.72, l'algoritmo non produce risultati, perchè non sono state create regole d'associazione con quei valori di confidenza dal programma *Rules*.

La figura 6.2 mostra, tramite un grafico, i valori della precision e della recall del caso generale. I risultati ottenuti mostrano valori di precision non soddisfacenti, in quanto poco superiori al 50%.

Invece la figura 6.3 mostra, la curva di ROC del caso generale. L'AUC riferita a questo grafico è: 473.2491.

Regole d'associazione per genere con selezione della singola regola

Anche per questo caso vengono mostrati i risultati ottenuti al variare di λ tra 0.4 e 0.72, con un incremento di 0.4. Come detto in precedenza questo algoritmo è simile al precedente solo che si occupa di creare regole d'asso-

Lambda	Precision	Recall	TPR
0.40	50.32	85.19	83.89
0.44	50.32	85.19	82.89
0.48	50.32	85.19	82.89
0.52	49.96	84.54	81.92
0.56	51.39	79.52	75.19
0.60	53.80	67.14	58.17
0.64	57.25	39.34	30.81
0.68	0	0	0
0.72	0	0	0

Tabella 6.1: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore UOMO dall'algoritmo con regole d'associazione per item con selezione della singola regola

ciazione per genere.
In tabella 6.3 e 6.4 vengono mostrati i risultati ottenuti da questo algoritmo, per l'indicatore UOMO e DONNA per le variabili precision, recall e TPR. Non sono state create regole con confidenza alta (superiore al valore di λ di 0.60) e perciò non si riesce a calcolare la qualità dell'algoritmo per quei valori. I valori ottenuti in questo caso sono paragonabili a quelli del primo caso, ma con una qualità leggermente più alta.

La figura 6.4 mostra la precision e la recall tenendo in considerazione entrambi gli indicatori di lifestyle ed i risultati mostrano un piccolo miglioramento dei valori di tali variabili rispetto al caso precedente. La precision è poco al di sotto del 60%, mentre la recall si mantiene abbastanza stabile intorno all'80%, prima di toccare valori pari a 0, per mancanza di regole d'associazione con confidenza alta.

Invece la figura 6.5 mostra la curva di ROC dell'algoritmo. L'AUC è: 351.2782.

Lambda	Precision	Recall	TPR
0.40	64.45	14.84	12.84
0.44	64.45	14.84	12.84
0.48	64.45	14.84	12.84
0.52	60.92	14.26	12.43
0.66	60.45	12.27	11.19
0.60	59.43	8.20	8.23
0.64	60.72	4.79	4.27
0.68	0	0	0
0.72	0	0	0

Tabella 6.2: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore DONNA dall'algoritmo con regole d'associazione per item con selezione della singola regola

Regole d'associazione per item con selezione di combinazione di regole

In questo algoritmo è stato modificato il range di valori di λ, in quanto viene scelto come indicatore di lifestyle quello per cui la somma delle confidenze delle regole con conseguente l'indicatore stesso è maggiore. In pratica la soglia di binarizzazione viene applicata a:

$$| \sum confidenza_M - \sum confidenza_F |$$

e, quindi, λ è stato fatto variare tra 0 (non viene applicata nessuna soglia) e 5, con un incremento di 0.5 In tabella 6.5 e 6.6 vengono mostrati i risultati ottenuti da questo algoritmo, per l'indicatore UOMO e DONNA. I valori di precision e recall sono simili ai casi precedenti ed anche in questo tipo di algoritmo si nota che al crescere della soglia di binarizzazione diminuisce la precision per il caso DONNA.

Invece la figura 6.6 mostra la precision e la recall tenendo in considerazione entrambi gli indicatori di lifestyle. Per il caso con $lambda = 0$, la precision ottenuta è bassissima, ed anche per $lambda = 5$ rimane comunque al di sotto del 60%.

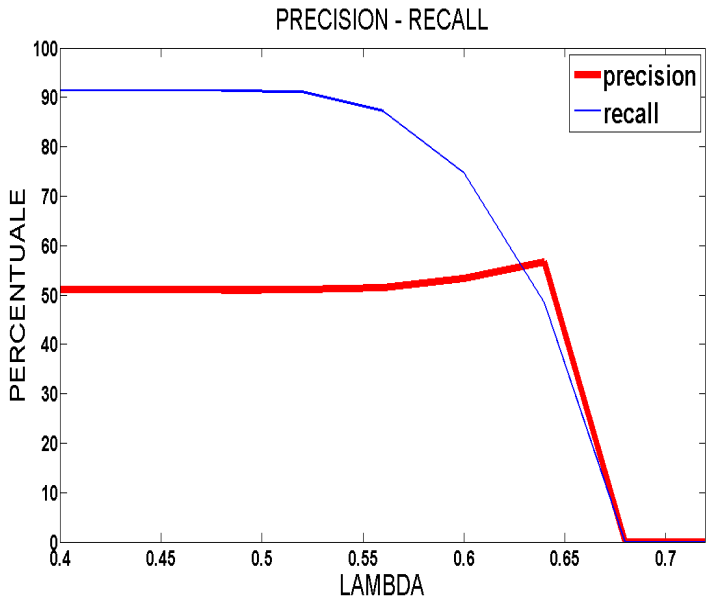

Figura 6.2: Precision e recall dell'algoritmo con regole d'associazione per item con selezione della singola regola

Il grafico in figura 6.7 mostra la curva di ROC del caso generale, quello cioè senza prendere in considerazione alcun indicatore in particolare. L'AUC riferita a questo grafico è: 496.6923.

Regole d'associazione per genere con selezione di combinazione di regole

Come per il caso precedente, anche in quest'algoritmo è stato utilizzato un λ che varia tra 0 e 5 con un incremento di 0.5. In tabella 6.7 e 6.8 vengono mostrati i risultati ottenuti da questo algoritmo, per l'indicatore UOMO e DONNA. In entrambi i casi, si riesce a superare il 60% di precision, ma con recall bassa (intorno al 30%).

La figura 6.8 mostra la precision e la recall tenendo in considerazione entrambi gli indicatori di lifestyle. Il risultato più significante del grafico è quello per $lambda = 5$, dove si ottiene un precision poco superiore al 60% e

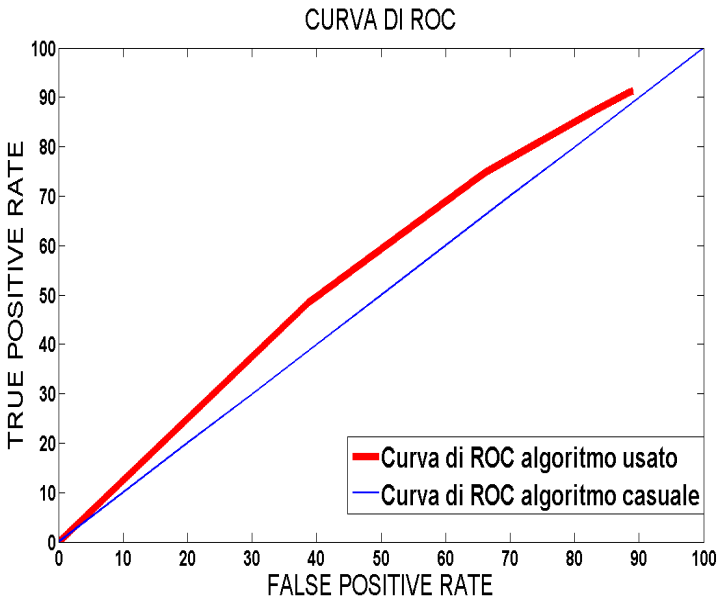

Figura 6.3: Curva di ROC dell'algoritmo con regole d'associazione per item con selezione della singola regola

una recall intorno al 45%.

Invece la figura 6.9 mostra la curva di ROC di questo algoritmo del caso generale, con AUC di 230.7675.

Lambda	Precision	Recall	TPR
0.40	54.60	84.12	72.63
0.44	54.60	84.12	72.63
0.48	54.60	84.13	72.63
0.52	54.72	83.86	71.99
0.56	55.36	79.46	66.49
0.60	58.50	31.47	22.22
0.64	0	0	0
0.68	0	0	0
0.72	0	0	0

Tabella 6.3: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore UOMO dall'algoritmo con regole d'associazione per genere con selezione della singola regola

6.3 Pseudo regole d'associazione

Nel lavoro [21] viene presentato un approccio alternativo per la classificazione, chiamato *associative classification* usato per la creazione di regole d'associazione che, in alcuni casi, raggiungono una maggiore accuratezza nella classificazione rispetto ai metodi tradizionali. Il concetto base dell'*associative classification* è quello di creare un grandissimo numero di regole, che però porta ad un alto livello di processamento.
I tradizionali classificatori di regole sono sostanzialmente più veloci, ma la loro accuratezza potrebbe non essere alta. Partendo dal lavoro descritto in precedenza, nell'articolo [34] viene proposto un nuovo metodo di classificazione: *Classification based on Predictive Association Rules, CPAR*, che combina i vantaggi dei metodi tradizionali di creazione di regole d'associazione e dell'*associative classification* ed adotta un algoritmo per generare le regole direttamente dai dati di training.
Per capire meglio l'approccio proposto è utile una descrizione dell'articolo [34] da cui è stata presa ispirazione per la creazione delle pesudo regole d'associazione.

Lambda	Precision	Recall	TPR
0.40	68.29	27.36	15.87
0.44	68.29	27.36	15.87
0.48	68.29	27.36	15.87
0.52	67.98	26.65	15.70
0.56	64.91	18.20	11.43
0.60	54.98	6.85	5.49
0.64	0	0	0
0.68	0	0	0
0.72	0	0	0

Tabella 6.4: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore DONNA dall'algoritmo con regole d'associazione per genere con selezione della singola regola

6.3.1 "Classification based on Predictive Association Rules" [34]

Il metodo del CPAR, introdotto da Xiaoxin Yin e Jiawei Han, integra le caratteristiche della classificazione associativa, ma rispetto a quest'ultima ha i seguenti vantaggi:

- CPAR genera un più basso numero di regole di alta qualità direttamente dal database
- ogni regola viene analizzata per verificare se già creata

Queste caratteristiche dovrebbero permettere di raggiungere alti valori di efficienza dal punto di vista dei tempi ed ottenere un'ottima accuratezza nella predizione.
Una volta generata la lista di regole, CPAR sceglie, per il processo d'inferenza, solo quelle il cui precedente soddisfa il contenuto in esame, per poi selezionare quella con confidenza maggiore.

Il concetto che la creazione di regole avvenisse direttamente dal dataset, senza cioè la formazione di insiemi di item frequenti, ha dato l'idea alla creazione degli algoritmi descritti dalla prossima sottosezione e denominati pseudo regole d'associazione.

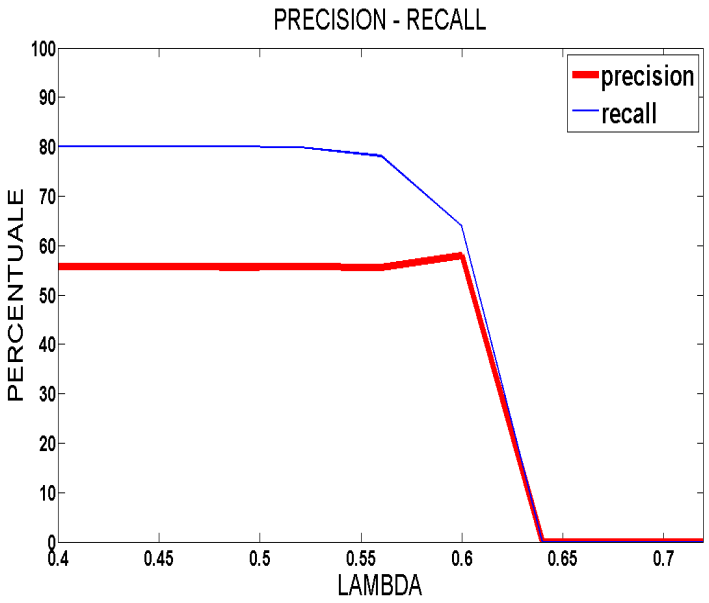

Figura 6.4: Precision e recall dell'algoritmo con regole d'associazione per genere con selezione della singola regola

La metodologia del CPAR è stata presa in considerazione e riadattata al lavoro di tesi, creando cioè regole d'associazione direttamente dalla sottomatrice CAMPIONE senza ricavare precedentemente l'insieme degli item più frequenti, per dedurre il lifestyle degli utenti di TEST. Ad esempio se si ipotizza che una riga della della matrice CAMPIONE mostra un utente che ha visionato i film:"Dracula" (con ID 12), "Rain man" (con ID 25) e "Dead man" (con ID 78) ed è maschio, allora l'algoritmo genererà direttamente una regola del tipo:

$$12, 25, 78 \rightarrow UOMO$$

Sono stati ridefinte le variabili chiave per la scelta delle regole d'associazione: *support* e *confidence*. In particolare per ogni utente i della matrice TEST verrà assegnato un valore di support e confidence ad ogni regola k derivata dalla matrice CAMPIONE. Se i ha visionato un film contenuto in k, allora si incrementa la variabile *support* riferita a quella regola. Il valore di *confidence*, invece, è il rapporto tra support e il numero di film visionati dall'utente i. Seguendo queste linee guida, sono stati sviluppati due algorit-

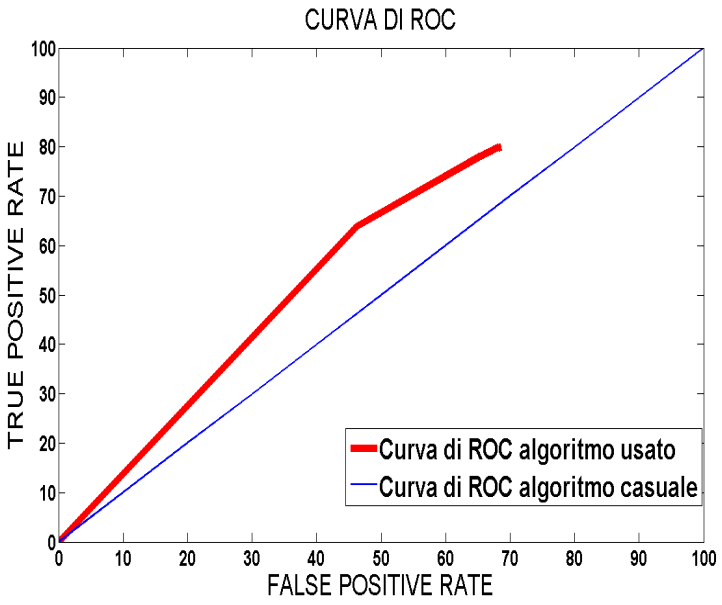

Figura 6.5: Curva di ROC dell'algoritmo con regole d'associazione per genere con selezione della singola regola

mi: il primo è basato sulla creazione di regole d'associazione con dominio sui film del dataset Movielens, mentre il secondo sull'insieme dei generi dei film.

6.3.2 Associazione tramite item

L'algoritmo descritto in questo paragrafo è molto simile alle regole d'associazione viste precedentemente, ma si differenzia per il modo in cui genera la lista di regole. In particolare, in una prima fase, per ogni utente dalla matrice CAMPIONE viene creata una regola d'associazione. Nella seconda fase dell'algoritmo vengono usate le tecniche adoperate nell'algoritmo con regole d'associazione con selezione di combinazione di regole.
I punti chiavi dell'algoritmo in questione sono:

- dalla matrice *URM* si ricava la matrice binaria *URMtop*, dove vengono presi in considerazione solo i film che sono stati molto apprezzati

Lambda	Precision	Recall	TPR
0	50.99	87.13	85.87
1	50.82	78.73	73.66
2	53.26	69.44	61.24
3	54.20	61.69	52.73
4	55.09	56.22	46.53
5	55.88	51.85	41.70
6	56.49	47.81	37.67
7	57.29	43.63	33.59
8	57.99	39.78	29.91
9	58.40	36.40	27.03
10	58.88	33.33	24.32

Tabella 6.5: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore UOMO dall'algoritmo con regole d'associazione per item con selezione di combinazione di regole

dall'utente. Infatti i rating dei film minori o uguali a 3, sono stati settati a 0 e quelli con rating maggiore a 3.

- vengono creati dalla matrice *URMtop* le sottomatrici TEST e CAMPIONE sempre con le stesse percentuali degli altri algoritmi (la matrice CAMPIONE è il 10% di *URMtop*, TEST il restante 90%).
- dalla matrice CAMPIONE viene creato l'insieme delle regole;
- per ogni utente i della matrice TEST viene stilata la lista *PREFi*, contenente i film preferiti dell'utente;
- viene creata, come in precedenza, la lista *REGOLE* contenente tutte le regole potenzialmente utilizzabili per l'utente i;
- vengono inizializzate due variabili *maschio*, dove si sommano tutte le confidenze delle regole, presenti in *REGOLE*, che hanno conseguente la variabile UOMO. Analogamente $femmina$ rappresenta la somma delle confidenze di tutte le regole di *REGOLE* con conseguente la

Lambda	Precision	Recall	TPR
0	66.40	15.12	12.86
1	66.53	11.45	10.32
2	59.11	8.39	8.45
3	49.44	7.25	7.57
4	47.67	6.47	6.96
5	46.67	5.78	6.21
6	46.75	5.11	5.70
7	46.61	4.65	5.22
8	45.86	4.11	4.75
9	45.27	3.68	4.36
10	44.66	3.27	3.97

Tabella 6.6: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore DONNA dall'algoritmo con regole d'associazione per item con selezione di combinazione di regole

variabile DONNA. Nel caso di valori uguali di queste due variabili, si confrontano le regole nello stesso modo, ma rispetto al supporto.

- Se $maschio > femmina$, allora verrà assegnato all'utente i, l'indicatore di *lifestyle* UOMO, altrimenti DONNA.

- i risultati vengono filtrati con il metodo della confidence analysis. Se la differenza tra le due variabili $maschio$ e $femmina$, in valore assoluto, è maggiore del parametro λ, allora viene confermato il risultato precedente, altrimenti non viene assegnato l'indicatore di *lifestyle* all'utente i.

Nel prossimo paragrafo viene mostrato un ulteriore algoritmo implementato in questo lavoro di tesi, che utilizza le pseudo regole d'associazione sul dominio dei generi dei film del dataset utilizzato.

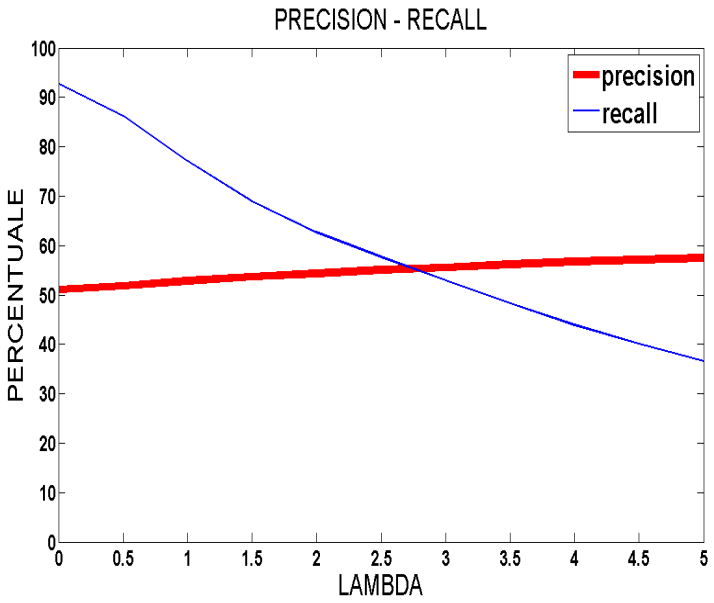

Figura 6.6: Precision e recall dell'algoritmo con regole d'associazione per item con selezione di combinazione di regole

6.3.3 Associazione tramite genere

Anche in questo algoritmo si ricavano le regole d'associazione direttamente dalla matrice CAMPIONE, senza il passaggio intermedio della creazione degli insieme di item (in questo caso i generi) frequenti.
Quest'ultimo algoritmo implementato in questo lavoro di tesi si può suddividere in tre fasi. Nella prima, si associa ad ogni genere di film un sesso, determinando se questo è visionato maggiormente da uomini o donne, per poi etichettarlo come genere maschile o femminile. Nella seconda fase si analizzano, per ogni utente della matrice TEST, i film per ricavarne il genere preferito. Nella terza ed ultima fase viene inferito il lifestyle in base al sesso associato al suo genere preferito nella prima fase.
Schematicamente quest'ultimo algoritmo può essere descritto con maggiore dettaglio nel modo seguente:

- come per tutti gli algoritmi presentati in questo capitolo, anche qui viene utilizzata la matrice *URMtop* per analizzare solo i film che sono piaciuti maggiormente all'utente;

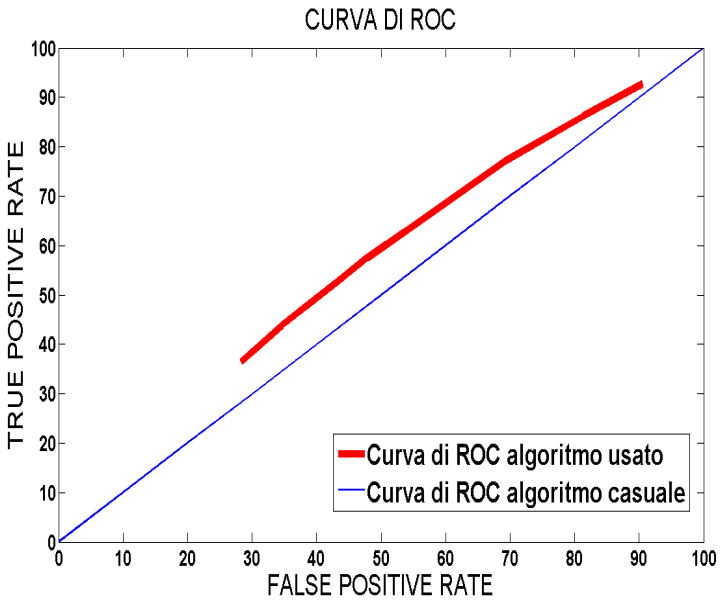

Figura 6.7: Curva di ROC dell'algoritmo con regole d'associazione per item con selezione di combinazione di regole

- dalla matrice *URMtop* vengono creati le sottomatrici TEST e CAMPIONE con le identiche percentuali degli altri algoritmi;
- per ogni genere di film vengono create due variabili $genereM$ e $genereF$
- per ogni utente i della matrice CAMPIONE vengono calcolati i generi preferiti (sempre secondo il principio di Pareto) e inseriti nella lista $PREF_i$;
- vengono incrementate le variabili $genereM$ e $genereF$ dei generi preferiti di i, a seconda del loro sesso: UOMO o DONNA
- dopo aver esaminato tutti gli utenti di CAMPIONE viene assegnato ad ogni genere un sesso: UOMO se $genereM > genereF$, DONNA se $genereF > genereM$.
- si calcolano i generi dei film più visti da ogni utente di TEST
- se vi è una maggioranza di 'generi maschili', allora quell'utente viene assegnato alla categoria UOMO, altrimenti a quella DONNA.

Lambda	Precision	Recall	TPR
0	55.84	79.92	64.92
1	57.66	68.51	52.10
2	58.67	61.27	45.43
3	58.58	58.68	42.49
4	60.25	50.68	34.85
5	60.63	49.72	33.85
6	60.73	40.43	27.07
7	61.72	36.08	23.35
8	61.83	27.93	17.90
9	61.80	26.92	17.25
10	61.83	26.08	16.64

Tabella 6.7: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore UOMO dall'algoritmo con regole d'associazione per genere con selezione di combinazione di regole

- anche in questo caso i risultati vengono filtrati con lo strumento della confidence analysis. Se la differenza del numero di visioni di generi maschili e quello di generi femminili, in valore assoluto, è maggiore del parametro λ, allora viene confermato il risultato precedente, altrimenti non viene assegnato l'indicatore di *lifestyle* a quel particolare utente.

6.3.4 Risultati

Con questi algoritmi si utilizza la politica di scelta dell'indicatore di lifestyle in base alla somma delle confidenze delle regole d'associazione, come per i due algoritmi descritti nella sezione 6.2.2. La differenza è che, in questo algoritmo, vengono create un maggior numero di regole d'associazione e quindi la somma di tutte le confidenze con conseguente UOMO oppure DONNA avrà valori maggiori rispetto agli algoritmi precedenti. Proprio per quest'aspetto, la soglia di binarizzazione λ è stata fatta variare tra 0 e 10, con un incremento di 1.

Lambda	Precision	Recall	TPR
0	66.45	35.07	20.07
1	67.49	30.53	17.47
2	68.01	25.93	14.28
3	67.65	24.53	13.65
4	72.37	15.34	8.58
5	54.77	6.62	5.36
6	53.04	5.83	5.05
7	52.84	5.68	4.96
8	52.87	5.55	4.85
9	52.08	5.00	4.50
10	51.96	4.94	4.47

Tabella 6.8: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore DONNA dall'algoritmo con regole d'associazione per genere con selezione di combinazione di regole

Gli algoritmi sviluppati sono due: regole con precedente formato da un insime di item oppure di generi.

Pseudo regole d'associazione per item

In tabella 6.9 e 6.10 vengono mostrati i risultati ottenuti da questo algoritmo, per l'indicatore UOMO e DONNA. Come viene mostrato da queste tabelle, con le pseudo regole d'associazione per item si riescono ad ottenere valori alti di precision a discapito di una recall molto bassa.

La figura 6.10 mostra la precision e la recall tenendo in considerazione entrambi gli indicatori di lifestyle. Il grafico rispecchia i risultati ottenuti con i singoli lifestyle, con precision e recall che aumentano e diminuiscono rispettivamente con grande velocità.

Invece la figura 6.11 mostra la curva di ROC di questo algoritmo, sempre confrontandola con la curva di ROC di un algoritmo con assegnazione dei lifestyle casuale. Tale curva ha un AUC di 234.1378.

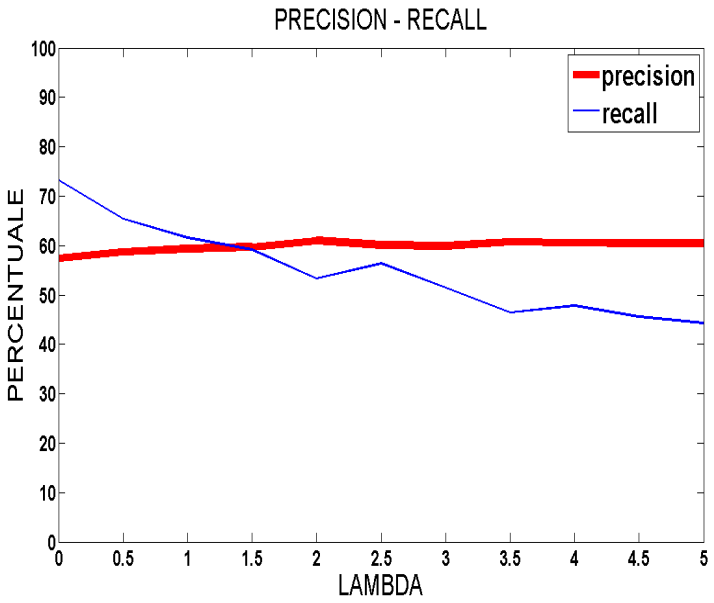

Figura 6.8: Precision e recall dell'algoritmo con regole d'associazione per genere con selezione di combinazione di regole

Pseudo regole d'associazione per genere

Come per il caso sopra, anche in quest'algoritmo è stato utilizzato un λ che varia tra 1 e 10 con un incremento di 1. In tabella 6.11 e 6.12 vengono mostrati i risultati ottenuti da questo algoritmo, per l'indicatore UOMO e DONNA. I risultati migliori, in questo caso, si ottengono senza applicare nessuna soglia di binarizzazione, cioè con $lambda = 0$.

La figura 6.12 mostra la precision e la recall tenendo in considerazione entrambi gli indicatori di lifestyle. La precision si mantiene stabile all'aumentare di lambda , a differenza della recall che raggiunge valori limite dal 75% al 35%.

Per ultimo, la figura 6.13 illustra la curva di ROC di quest'ultimo algoritmo, con AUC di 97.7080.

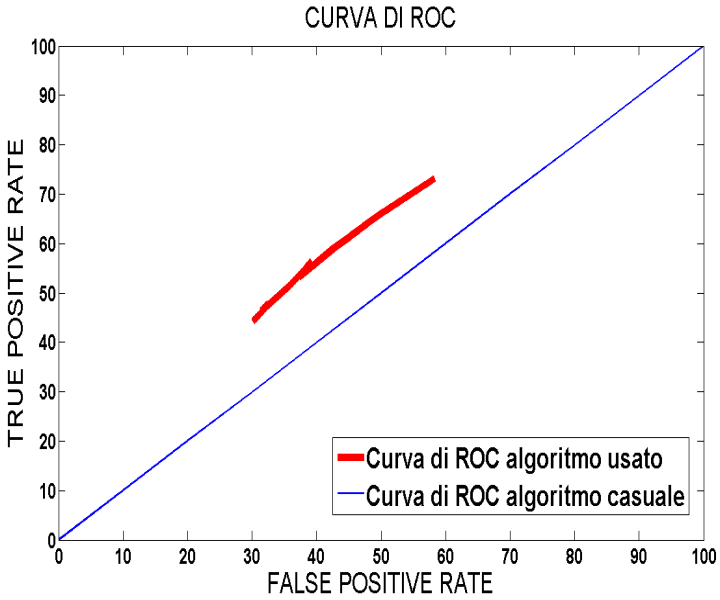

Figura 6.9: Curva di ROC dell'algoritmo con regole d'associazione per genere con selezione di combinazione di regole

Lambda	Precision	Recall	TPR
0	58.85	61.03	45.60
1	63.00	43.73	28.45
2	65.21	29.94	18.46
3	63.79	21.26	12.54
4	66.07	15.50	8.60
5	71.34	11.13	5.93
6	69.29	7.91	4.14
7	74.27	5.70	2.86
8	78.04	4.14	2.04
9	79.11	3.01	1.47
10	81.95	2.24	1.05

Tabella 6.9: Tabella riassuntiva dei valori di precision, recall E TPR ottenuti per l'indicatore UOMO dall'algoritmo con pseudo regole d'associazione per item

Lambda	Precision	Recall	TPR
0	65.91	44.39	28.90
1	73.79	27.22	15.63
2	77.05	16.23	9.19
3	80.77	10.18	5.90
4	77.62	6.22	3.80
5	73.95	4.06	2.66
6	81.75	2.68	1.71
7	87.20	1.78	1.21
8	84.22	1.20	0.87
9	75.18	0.91	0.63
10	77.95	0.72	0.46

Tabella 6.10: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore DONNA dall'algoritmo con pseudo regole d'associazione per item

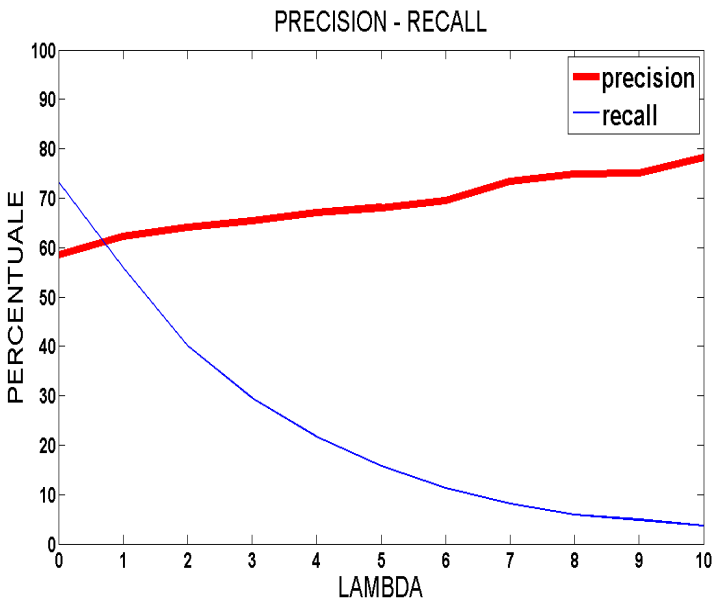

Figura 6.10: Precision e recall dell'algoritmo con pseudo regole d'associazione per item

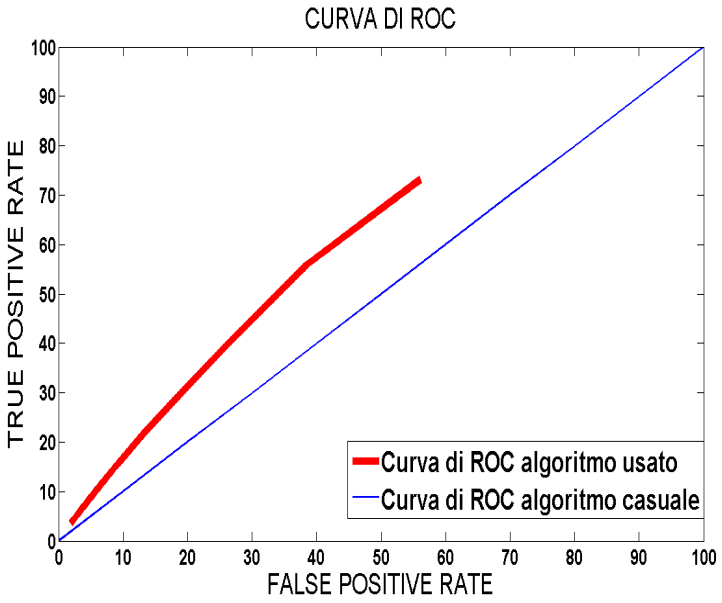

Figura 6.11: Curva di ROC dell'algoritmo con pseudo regole d'associazione per item

Lambda	Precision	Recall	TPR
0	53.41	71.36	65.45
1	53.99	65.16	58.86
2	54.16	58.60	52.69
3	54.59	53.04	47.13
4	54.58	48.08	42.75
5	54.69	43.71	38.54
6	54.89	40.34	35.11
7	55.61	37.16	32.17
8	55.21	34.26	29.70
9	55.11	31.83	27.55
10	55.07	29.65	25.53

Tabella 6.11: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore UOMO dall'algoritmo con pseudo regole d'associazione per genere

Lambda	Precision	Recall	TPR
0	64.00	34.54	28.63
1	63.72	27.86	23.00
2	64.11	23.06	18.93
3	66.74	19.60	16.17
4	64.24	16.93	13.96
5	66.58	14.92	12.31
6	66.52	13.11	10.88
7	66.52	11.75	9.75
8	68.77	10.53	8.72
9	68.53	9.57	7.93
10	64.57	8.70	7.21

Tabella 6.12: Tabella riassuntiva dei valori di precision, recall e TPR ottenuti per l'indicatore DONNA dall'algoritmo con pseudo regole d'associazione per genere

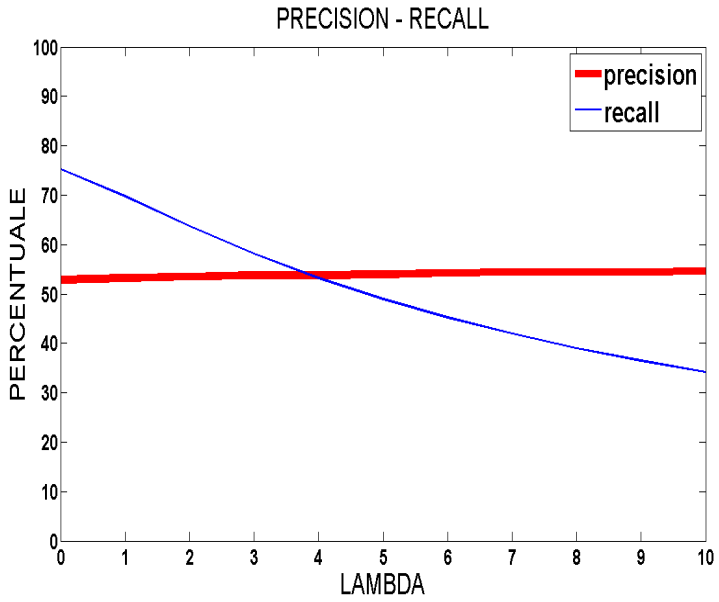

Figura 6.12: Precision e recall dell'algoritmo con pseudo regole d'associazione per genere

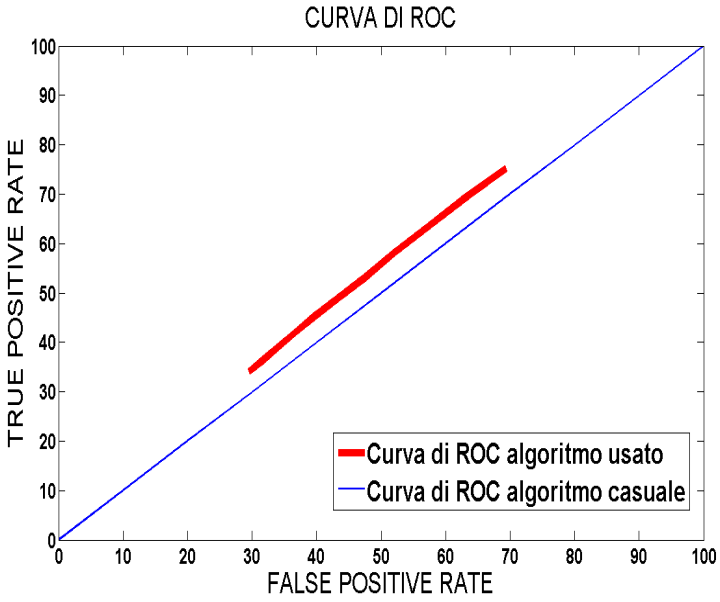

Figura 6.13: Curva di ROC dell'algoritmo con pseudo regole d'associazione per genere

Capitolo 7

Conclusioni e sviluppi futuri

Prima di trarre le conclusioni di questo lavoro di tesi, viene riportata in tabella 7.1 l'intervallo di confidenza di tutti gli algoritmi implementati e descritti (Caso base, Variante 1, Variante 2 e Variante 3 sono gli algoritmi del Capitolo 5, mentre gli altri del Capitolo 6). In particolare vengono mostrati valor medio (μ), il 25° (q_{25}) ed il 75° (q_{75}) percentile della *precision* e della *recall*.

I risultati degli algoritmi Caso Base e Variante 1 sono stati applicati al dataset originale di Movielens (URM), con un numero maggiore di utenti maschili rispetto a quelli femminili, mentre tutti gli altri hanno utilizzato un dataset bilanciato (URM bilanciata), dove è stata stabilita una parità numerica per i due sessi. I valori dei primi due algoritmi sono influenzati dai risultati ottenuti dall'indicatore di lifestyle UOMO e per questo motivo mostrano valori così alti di precision e recall; ma per poter effettuare un equo confronto di risultati e dare, così, uguale importanza ad entrambi gli indicatori di lifestyle, le conclusioni vengono tratte solo per quelle implementazioni che utilizzano l'URM bilanciata.

Si ricorda che la precision rappresenta la percentuale di successo dell'algoritmo nell'assegnare un lifestyle ad un utente, mentre la recall indica la percentuale di utenti a cui è stato assegnato un lifestyle. Un buon compromesso tra i valori di queste due variabili rappresenterebbe l'algoritmo migliore.

Dalla tabella si può notare che gli algoritmi implementati con le regole d'as-

		Precision			Recall		
Training	**Algoritmo**	μ	q_{25}	q_{75}	μ	q_{25}	q_{75}
URM	Caso Base	87.32	85.65	87.75	71.32	68.76	74.35
	Variante 1	83.12	82.42	84.09	74.59	69.12	76.43
URM BILANCIATA	Variante 2	85.45	84.01	87.00	43.14	36.34	44.26
	Variante 3	87.54	87.04	87.97	41.04	36.68	43.81
	Associazione per item singola regola	52.21	51.01	52.87	82.20	77.86	91.36
	Associazione per genere singola regola	56.01	55.65	55.67	77.02	78.10	80.08
	Associazione per item combinazione di regole	54.70	53.05	56.64	60.65	44.99	75.11
	Associazione per genere combinazione di regole	59.84	59.43	60.52	54.95	46.77	60.95
	Pseudo associazione per item	68.74	64.41	74.50	24.56	06.45	37.43
	Pseudo associazione per genere	53.90	53.52	54.37	51.44	39.72	62.29

Tabella 7.1: Intervallo di confidenza per la precisione e recall per tutti gli algoritmi

sociazione (standard e pseudo) ottengono valori medi di precision più bassi rispetto a quelli sviluppati con algoritmi di raccomandazione, ma con una recall più alta. In tabella 7.1 sono evidenziati in grigio i valori per cui la differenza tra il 25esimo ed il 75esimo percentile è superiore al 10%. La conclusione che si può trarre riguarda la soglia di binarizzazione (λ) che, applicata agli algoritmi attraverso il metodo della confidence analysis, influenza considerevolmente i risultati. Maggiore è la differenza in valore assoluto tra i due valori del percentile calcolati e maggiore è il peso che ha avuto λ nell'algoritmo.
In linea generale i valori di recall risentono maggiormente del parametro λ, rispetto a quelli della precision.

La figura 7.1, che sintetizza in un grafico i valori di precision e recall ottenuti dagli algoritmi con dataset bilanciato, è stata divisa in quattro quadranti. Il migliore ai fini dei risultati è quello in alto a destra, in quanto raggruppa valori alti per entrambe le variabili. Gli algoritmi che gli si avvicinano di più sono Variante 2 e Variante 3, che utilizzano sistemi di raccomandazione ibrida per la categorizzazione di utenti IPTV.

Come sviluppi futuri del lavoro presentato si potrà estendere l'anali-

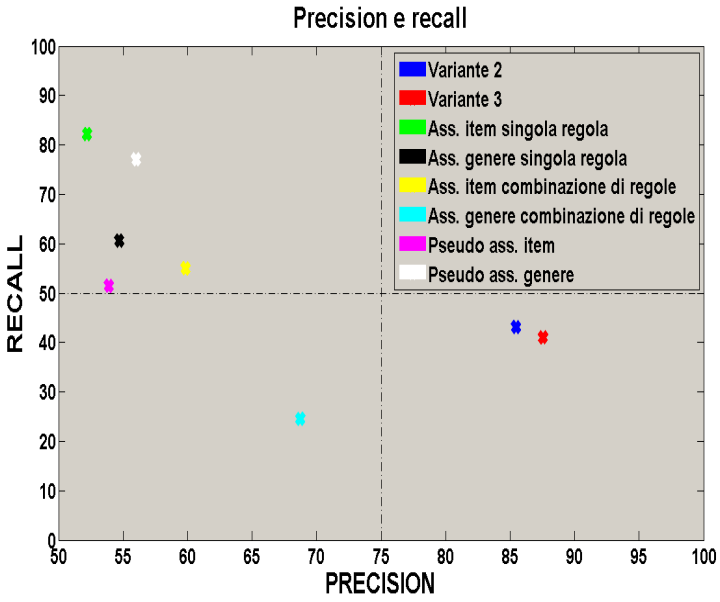

Figura 7.1: Valori di precision e recall

si ad ulteriori indicatori di lifestyle, come l'occupazione e l'età nel caso di Movilens. Inoltre potranno essere utilizzati nuovi dataset, contenenti informazioni riguardanti il lifestyle degli utenti.

Bibliografia

[1] *ITU TSB IPTV Consultation meeting (Doc. Iptv018e and 20e).*

[2] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, 17(6):734–749, 2005.

[3] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22:207–216, June 1993.

[4] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

[5] Chumki Basu, Haym Hirsh, and William Cohen. Recommendation as classification: Using social and content-based information in recommendation. In *In Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 714–720. AAAI Press, 1998.

[6] Robert M. Bell and Yehuda Koren. Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, pages 43–52, Washington, DC, USA, 2007. IEEE Computer Society.

[7] Theodoros Bozios, Georgios Lekakos, and Victoria Skoularidou. Advanced techniques for personalized advertising in a digital tv environment: The imedia system. In *In Proceedings of the eBusiness and eWork Conference*, 2001.

[8] John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. pages 43–52. Morgan Kaufmann, 1998.

[9] Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, editors. *The Adaptive Web: Methods and Strategies of Web Personalization*. Lecture Notes in Computer Science. Springer, Berlin, June 2007.

[10] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 39–46, New York, NY, USA, 2010. ACM.

[11] Paolo Cremonesi, Roberto Turrin, Eugenio Lentini, and Matteo Matteucci. An evaluation methodology for collaborative recommender systems. In *Proceedings of the 2008 International Conference on Automated solutions for Cross Media Content and Multi-channel Distribution*, pages 224–231, Washington, DC, USA, 2008. IEEE Computer Society.

[12] Yae Dai, HongWu Ye, and SongJie Gong. Personalized recommendation algorithm using user demography information. In *Proceedings of the 2009 Second International Workshop on Knowledge Discovery and Data Mining*, pages 100–103, Washington, DC, USA, 2009. IEEE Computer Society.

[13] Sito del Professore Bart Goethals: sviluppatore del programma Rules. *http://adrem.ua.ac.be/ goethals/.*

[14] Kenneth A Coney Delbert Hawkins Roger Best Kenneth Coney Delbert I Hawkins, Roger J Best. *Consumer Behavior: Building Marketing Strategy.* 1998.

[15] Joaquin Delgado and Naohiro Ishii. Memory-based weighted-majority prediction for recommender systems, 1999.

[16] Tom Fawcett. Roc graphs: Notes and practical considerations for researchers. Technical Report HPL-2003-4.

[17] G. Gunter and A. Furnham. *Consumer Profiles: An Introduction to Psychographics*. Routledge London, 1992.

[18] Joeseph M. Juran. *Quality Control Handbook*. 1951.

[19] George Lekakos. *Personalized advertising services through hybrid recommendation methods: the case of digital interactive television.*

[20] Georgios Lekakos, Dimitris Papakiriakopoulos, and Kostas Chorianopoulos. An integrated approach to interactive and personalized tv advertising. In *In L. Ardissono and Y. Faihe (eds.), Proceedings of the 2001 Workshop on Personalization in Future TV*, 2001.

[21] Bing Liu, Wynne Hsu, and Yiming Ma. Integrating classification and association rule mining. pages 80–86, 1998.

[22] G. Piatetsky-Shapiro. Discovery, analysis and presentation of strong rules. In G. Piatetsky-Shapiro and W. J. Frawley, editors, *Knowledge Discovery in Databases*, pages 229–248. AAAI Press, 1991.

[23] Sezione pubblicitaria di Comcast Cable: società di servizi telefonici e di internet a banda larga. *http://www.comcastspotlight.com.*

[24] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Incremental singular value decomposition algorithms for highly scalable recommender systems. In *Fifth International Conference on Computer and Information Science*, pages 27–28, 2002.

[25] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John T. Riedl. Application of dimensionality reduction in recommender system – a case study. In *IN ACM WEBKDD WORKSHOP*, 2000.

[26] Laboratorio di ricerca del dipartimento di Computer Science Sito web di GroupLens and Engineering della University of Minnesota. *http://www.grouplens.org/.*

[27] Stephan Spiegel, Jérôme Kunegis, and Fang Li. Hydra: a hybrid recommender system [cross-linked rating and content information]. In

Proceeding of the 1st ACM international workshop on Complex networks meet information & knowledge management, CNIKM '09, pages 75–80, New York, NY, USA, 2009. ACM.

[28] Julian Tilbury, Peter Van Eetvelt, Jonathan Garibaldi, John Curnow, and Emmanuel Ifeachor. Receiver operating characteristic analysis for intelligent medical systems - a new approach for finding non-parametric confidence intervals, 2000.

[29] Takeaki Uno, Masashi Kiyomi, and Hiroki Arimura. Lcm ver. 2: Efficient mining algorithms for frequent/closed/maximal itemsets. In *FIMI*, 2004.

[30] E. Vozalis and K. G. Margaritis. Analysis of recommender systems' algorithms. In *The 6th Hellenic European Conference on Computer Mathematics & its Applications (HERCMA), Athens, Greece*, 2003.

[31] Sito web Dataset MovieLens. *http://www.movielens.org*.

[32] Sito web del gruppo GfK Eurisko. *http://www.gfk.com/gfk-eurisko/*.

[33] Sito web di Values and Lifestyles Psychographic Segmentation. *http://www.strategicbusinessinsights.com/*.

[34] Xiaoxin Yin and Jiawei Han. Cpar: Classification based on predictive association rules, 2003.