Introduzione al problema

Il progetto presentato prevede l’analisi del sistema emissivo polizze di un cliente assicurativo di **Fincons Group S.p.A.**, lavorando sul sistema in questione ho notato delle peculiarità del sistema che andrebbero migliorate.

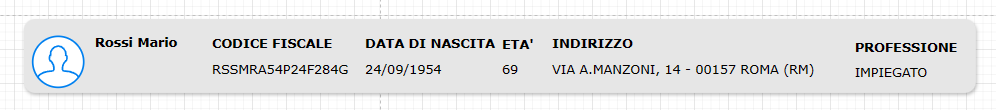
L’attuale sistema emissivo delle polizze assicurative risulta essere farraginoso, di poco impatto e anonimo. L’immagine mostra la pagina utilizzata dagli agenti per l’emissione delle polizze.

Immagine che contiene testo, schermata, software, Icona del computer

Descrizione generata automaticamente

Pagina iniziale emissione polizze

L’agente assicurativo non ha a disposizione una panoramica esauriente del cliente che sta gestendo, per visualizzare il profilo del cliente deve aprire la pagina del profilo completo, senza avere la possibilità di visualizzare contemporaneamente le sue informazioni e le coperture assicurative che può proporgli. Nell’immagine che segue vengono mostrati i dati del cliente a disposizione dell’agente nella pagina di emissione.



Dati Cliente

L’agente assicurativo visualizza le coperture assicurative che è possibile vendere al cliente suddivise per tipologia prodotto. Le coperture assicurative raggruppate secondo questo criterio non sono personalizzate; quindi, l’agente non ha alcun supporto riguardo le coperture assicurative che sarebbe opportuno proporre al cliente. Se le coperture assicurative di una tipologia sono numerose non è possibile visualizzarle tutte contemporaneamente nella stessa schermata. Nell’immagine vengono mostrate le coperture assicurative presenti in pagina secondo il raggruppamento descritto.

Immagine che contiene testo, schermata, Policromia, software

Descrizione generata automaticamente

Elenco coperture assicurative

La pagina descritta è stata realizzata utilizzando in parte tecnologie legacy, quindi ogni volta che l’utente interagisce con la pagina essa viene ricaricata, senza garantire una buona interazione fornendo all’utente un’esperienza di navigazione limitata e deludente.

Dall’analisi eseguita sono emersi le problematiche riguardo l’utilizzo dell’applicativo:

* Durante l’emissione delle polizze assicurative, è possibile visualizzare solo una minima parte dei dati relativi al profilo cliente.
* Le coperture assicurative sono visualizzate indipendentemente dal cliente selezionato.
* Le coperture assicurative sono raggruppate in base alla tipologia di prodotto e non in base all’interesse del cliente.
* L’applicazione fornisce una scarsa interattività ed un’esperienza utente deludente.

Analisi funzionale

Per risolvere i problemi descritti nel capitolo precedente si propone la realizzazione di una nuova dashboard.

La sua progettazione ha previsto di porre al centro il cliente ed è rivolta agli agenti assicurativi. Sono state previste tre sezioni:

* Profilo assicurativo del cliente
* Sezione coperture assicurative
* Sezione beni selezionati

Il cruscotto progettato avrà l’aspetto mostrato nell’immagine:

Immagine che contiene testo, schermata, software, numero

Descrizione generata automaticamente

Cruscotto cliente

Profilo assicurativo

Il profilo del cliente è stato pensato per rendere disponibili all’agente sempre le informazioni fondamentali, su richiesta le altre informazioni. L’agente avrà subito a disposizione un grafico riassuntivo delle coperture assicurative, il nominativo, ed i suoi recapiti principali. L’immagine mostra un esempio appena descritto.

Immagine che contiene testo, schermata, software, Carattere

Descrizione generata automaticamente

Profilo assicurativo del cliente

Nella schermata visualizzata sono mostrati:

* Il nominativo del cliente nell’intestazione della pagina.
* Sulla sinistra un grafico riassuntivo delle coperture attive, *in questo caso due coperture per i veicoli e quattro per la famiglia.*
* Sulla destra i dati essenziali del cliente, *nominativo, indirizzo principale o di residenza, il telefono principale, l’indirizzo email principale.*

Cliccando sul pulsante *espandi* presente in basso a destra verrà visualizzata la il profilo completo del cliente. È prevista una scheda per ogni tipologia di dato *anagrafica, indirizzo, recapiti,* professione.

Immagine che contiene testo, numero, Carattere, schermata

Descrizione generata automaticamente

Profilo completo cliente

Cliccando sul pulsante *espandi* presente a destra nella per la sezione *coperture assicurative* verrà visualizzato un report riassuntivo contenente le polizze stipulate dal cliente. L’immagine mostra un esempio del report appena descritto:

Immagine che contiene testo, schermata, software, Pagina Web

Descrizione generata automaticamente

Coperture assicurative

Il report mostra tre polizze:

* Polizza Auto
* Polizza Animale Domestico
* Polizza Casa

Per ogni copertura viene mostrato:

* Identificativo polizza
* Tipologia polizza
* Dettaglio bene assicurato
* Data stipulazione contratto
* Data scadenza contratto
* Importo premio totale

Le sezioni descritte permettono all’agente di avere una panoramica della situazione assicurativa del cliente.

Le sezioni che verranno descritte forniranno all’agente un supporto relativo a nuove polizze da proporre al cliente oppure quelle da rinnovare.

La prima sezione contiene le coperture assicurative che vengono proposte direttamente al cliente in base alla sua storia assicurativa ed alla storia assicurativa di clienti simili.

La seconda sezione contiene le coperture assicurative più vendute rispetto ad una statistica di vendita eseguita sull’anno precedente, le coperture assicurative visualizzate sono state escluse quelle proposte direttamente al cliente *“raccomandate”*. In questo modo ogni cliente avrà una visualizzazione differente.

La terza sezione contiene le coperture assicurative escluse dalle sezioni precedentemente descritte.

Le sezioni descritte contengono delle *“card”* ogni card mostra i dettagli della copertura assicurativa.

Immagine che contiene testo, Carattere, linea, schermata

Descrizione generata automaticamente

Card relative alle coperture assicurative

Nell’immagine vediamo due card relative a due coperture assicurative, le quali contengono rispettivamente le informazioni:

* Icona del prodotto.
* Nome del prodotto.
* Categoria o linea di vendita del prodotto.
* Per l’icona, il nome e la categoria del prodotto è stato utilizzato il colore definito per la categoria.
* Descrizione breve del prodotto.
* Spunta *(check)* che permette di selezionare/deselezionare il prodotto.

In fondo alla pagina è presente un pannello dove verranno visualizzati dei segnaposto relativi alle coperture selezionate. Lo sfondo dei segnaposto corrisponde al colore della categoria a cui appartiene la polizza selezionata.

Immagine che contiene testo, schermata, Carattere, Marchio

Descrizione generata automaticamente

Coperture selezionate

I sistemi di raccomandazione

L’obbiettivo di un sistema di raccomandazione[[1]](#endnote-1) consiste nel generare suggerimenti significativi per l’utente del sistema che possano essere di suo interesse. *I libri suggeriti da Amazon oppure il film su Netflix, rappresentano un chiaro esempio del funzionamento di un sistema di raccomandazione*. L’architettura del sistema dipende dal dominio applicativo, dai dati disponibili e dalle modalità con le quali l’utente interagisce con l’applicativo. *Su Netflix dopo aver visto un film è possibile esprimere una valutazione di gradimento*. Le valutazioni che esprime l’utente vengono memorizzate all’interno di una matrice *users-items*.

Immagine che contiene numero, schermata, parole crociate, calendario

Descrizione generata automaticamente

Matrice Users-Items

La matrice users-items rappresenta le preferenze degli utenti *(n in questo caso 5)* per gli items del sistema *(m in questo caso 6)*. Ogni cella **rij** *con i che va da 1 a 5 e j che va da 1 a 6* indica la valutazione dell’utente *(i)* per l’item *(j)*. La matrice users-items è una matrice sparsa, l’obbiettivo di un sistema di raccomandazione è calcolare la valutazione di un generico utente per gli items sui quali non ha espresso alcuna valutazione.

Ci sono diversi approcci ai sistemi di raccomandazione che si possono classificare in:

* Content based recommending: nei sistemi basati sul contenuto gli *items* che vengono suggeriti all’utente hanno un contenuto simile rispetto a quelli che in passato l’utente ha dato una valutazione positiva oppure ha acquistato.
* Collaborative Filtering Methods: nei sistemi collaborativi gli *items* suggeriti all’*utente* sono basati sulle valutazioni complessive degli utenti.
* Hybrid approaches: nei metodi ibridi vengono combinate le strategie viste sia per l’approccio basato sul contenuto che quelle basate sui sistemi collaborativi.

Immagine che contiene testo, schermata, linea, Carattere

Descrizione generata automaticamente

Tipologia sistemi di raccomandazione

Sistemi basati sul contenuto

I sistemi di raccomandazione basati sul contenuto prevedono la creazione di una base di conoscenza creata in base agli elementi che si vogliono proporre al cliente.

Un esempio di base di conoscenza potrebbe essere definito:

* Base di conoscenza videoteca
  + Titolo film
  + Descrizione trama
  + Cast
  + Regista
  + Genere

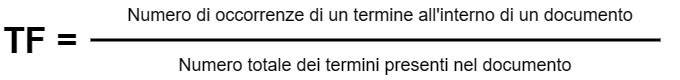
I campi proposto sono un sotto insieme, ogni campo a sua volta può essere collegato ad altre basi di conoscenza. Il genere potrebbe essere descritto:

* Base di conoscenza genere
  + Titolo genere
  + Descrizione genere
  + Generi collegati
  + Registi che seguono il genere
  + Film del genere

La base di conoscenza creata è un grafo, dove i nodi rappresentano le entità ed le associazioni rappresentano la relazione che intercorre tra le entità collegate.

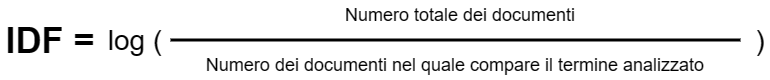
Dopo aver creato e popolato la base di conoscenza bisogna misurare la similarità tra gli elementi; essa viene calcolata utilizzando il **TF-IDF[[2]](#endnote-2)**, Il TF-IDF è costituito dal *TF - Term Frequency* e dall’*IDF - Inverse Document Frequency*.

Il **Term Frequency** di un termine viene calcolato come il rapporto tra *“il numero di volte che quel termine compare all’interno di un documento”* ed *“il numero totale dei termini presenti all’interno del documento”*.



Formula per il calcolo di TF – *Term Frequency*

L’**Inverse Document Frequency** di un termine viene calcolato come il *logaritmo* del rapporto tra *“il numero totale dei documenti”* ed *“il numero di documenti nei quali compare il termine analizzato”*.



Formula per il calcolo di IDF – Inverse Document Frequency

Il **TF-IDF** viene calcolato come il prodotto degli elementi presentati.

Immagine che contiene testo, Carattere, bianco, Elementi grafici

Descrizione generata automaticamente

Formula per il calcolo di TF-IDF

L’utilizzo del TF-IDF permette di non dover eliminare le *“stop words*[[3]](#endnote-3)*”* in quanto il TF-IDF calcolato per esse risulta 0 e quindi non rilevante.

Le raccomandazioni calcolate utilizzando l’approccio *Content Based Methods* prevedono di calcolare la similarità tra le coperture sottoscritte dal cliente e le coperture presenti all’interno della base di conoscenza.

Per calcolare la similarità è necessario rappresentare la base di conoscenza in forma numerica; per forma numerica s’intende creare un vettore relativo ad ogni copertura assicurativa presente nella base di conoscenza. Ogni vettore è costituito dalle caratteristiche individuate della copertura assicurativa rappresentata utilizzando gli **embeddings**[[4]](#endnote-4), la similarità viene misurata calcolando il coseno[[5]](#endnote-5) dell’angolo creato fra l’embedding relativo alla copertura assicurativa sottoscritta dal cliente e quelle presente nella base di conoscenza, se il risultato è “1” rappresenta la massima similarità mentre se è “0” rappresenta la minima similarità.

Sistemi collaborativi

I sistemi collaborativi lavorano sulla collezione delle preferenze sugli items indicate dall’utente per un determinato dominio applicativo, esplorando le similarità presenti fra le valutazioni esistenti. *Per fare un confronto nella vita reale i sistemi collaborativi si basano sul concetto “intuitivo” di passaparola, es. una persona chiede ai propri amici o conoscenti cosa pensano dell’ultimo film uscito al cinema, oppure l’indicazione su quale ristorante scegliere.*

I sistemi collaborativi possono essere suddivisi in due ulteriori categorie:

* Metodo basato sul vicinato *(neighborhood based)* chiamato anche metodo basato sulla memoria *(memory based method)*.
* Metodo basato sul modello.

Immagine che contiene testo, schermata, linea, Carattere

Descrizione generata automaticamente

Suddivisione – sistemi collaborativi

### Sistemi memory Based (neighborhood based)

Le tecniche dei sistemi basati sul vicinato prevedono di calcolare la similarità dell’utente attivo con un sottoinsieme di utenti; una media pesata delle loro valutazioni viene utilizzata per calcolare la predizione da eseguire per l’utente attivo. I passaggi di questo approccio possono essere schematizzati nell’algoritmo:

1. Assegnare un peso a tutti gli utenti rispettando la similarità con l’utente attivo.
2. Seleziona k-utenti che hanno la similarità maggiore con l’utente attivo, *comunemente chiamati vicinato (neighborhood)*.
3. Calcolare la predizione attraverso una media ponderata tra le valutazioni del vicinato.

Per quanto riguarda il passaggio (1), si definisce *wa,u* la misura di similarità tra l’utente attivo *a* ed un utente generico *u*. La formula più diffusa per il calcolo della similarità è il *coefficiente di correlazione di Pearson*.

Immagine che contiene testo, Carattere, linea, bianco

Descrizione generata automaticamente

Formula per il calcolo del Coefficiente di correlazione di Pearson tra utente *a* ed utente *u*

Gli elementi presenti nel coefficiente di correlazione di Pearson sono identificati:

* I: rappresenta l’insieme delle preferenze definite dagli utenti.
* ru,i: è la valutazione dell’utente *u* relativo all’item *i*.
* **r**u: indica la media delle valutazioni eseguite dall’utente *u*.

Nel passaggio (3) le predizioni vengono calcolate come una media ponderata dello scarto relativo alle valutazioni dei vicini rispetto alla media delle loro valutazioni.

Immagine che contiene testo, Carattere, linea, bianco

Descrizione generata automaticamente

Formula per il calcolo dele predizioni

La formula relativa alle predizioni è definita come segue:

* pa,i: è la previsione per l’utente *a* relativa all’elemento *i*.
* wa,u: è la similarità tra l’utente *a* e l’utente *u*.
* K: è il vicinato, formato dagli utenti simili.

La similarità calcolata attraverso il coefficiente di correlazione di Pearson misura la distanza tra due variabili linearmente dipendenti. Ci sono diverse alternative per il calcolo della similarità, anche se studi empirici[[6]](#endnote-6) hanno dimostrato che il coefficiente di correlazione di Pearson offre prestazioni migliori.

L’approccio basato sul vicinato risulta essere poco scalabile in quanto la ricerca di utenti simili risulta un’elevata complessità computazionale. Linden[[7]](#endnote-7) propone un approccio *collaborativo item-to-item*, in questo approccio vengono accoppiati gli item simili tra loro partendo dagli item valutati dall’utente. Seguendo questo approccio il coefficiente di correlazione di Pearson viene modificato ottenendo:

Immagine che contiene testo, Carattere, linea, bianco

Descrizione generata automaticamente

Formula del Coefficiente di correlazione di Pearson relativo all’approccio *item to item*

Gli elementi presenti in questa versione del coefficiente di correlazione di Pearson sono:

* U: rappresenta l’insieme degli utenti che hanno valutato sia l’item *i* che l’item *j*.
* ru,i: è la valutazione dell’utente *u* relativo all’item *i*.
* ri: è la media delle valutazioni degli utenti rispetto all’item *i*.

La predizione della valutazione sull’item *i* da parte dell’utente attivo si ottiene eseguendo una media ponderata:

Immagine che contiene Carattere, testo, bianco, linea

Descrizione generata automaticamente

Formula per il calcolo delle predizioni per l’approccio collaborativo *item to item*

### Sistemi model based

I sistemi basati sul modello forniscono le raccomandazioni in base alle statistiche sulle preferenze degli utenti. Un prima approccio[[8]](#endnote-8) ai sistemi collaborativi suggerisce di trattare il problema come se fosse un problema di classificazione, quindi viene costruito un classificatore per ogni utente attivo rappresentando gli elementi *(items)* come vettori di features[[9]](#footnote-1) e le valutazioni dell’utente come le labels[[10]](#footnote-2).

L’approccio che viene maggiormente utilizzato fa uso dei *fattori latenti e della fattorizzazione di matrici*. Le tecniche di questo approccio si basano su nozioni statistiche che calcolano la similarità fra gli utenti, o fra gli elementi. Il modello a fattori latenti presume che la similarità tra utenti ed items sia indotta da fattori nascosti presenti nella struttura dati. *Es. un utente che valuta un film in modo positivo, probabilmente valuterà in modo positivo film che appartengono allo stesso genere o dello stesso regista.*

Nella fattorizzazione delle matrici sono presenti un’ampia serie di tecniche utilizzate con successo che prevedono la rappresentazione simultanea degli utenti e degli item come un insieme di vettori di features sconosciute *(vettori colonna)* *wu, hi ∈ ℜk* , lungo *k* dimensioni latenti *(nascoste)*. Queste feature attraverso il prodotto *wuThi* approssimano la valutazione dell’utente *u* relativa all’elemento *i*, *ru,i* rispettando la funzione di costo *(loss function)[[11]](#endnote-9)*.

La scelta predefinita per la funzione di costo è la *squared loss*, l’obbiettivo è minimizzare la funzione

Immagine che contiene testo, Carattere, bianco, calligrafia

Descrizione generata automaticamente

Formula Squared Loss Function

Gli elementi presenti nella *squared loss function* sono definiti:

* W = [w1…wn]T è una matrice di rango *n \* k*.
* H = [h1…hm] è una matrice di rango m \* k.
* L identifica l’insieme di coppie *user-item* per cui il *rating[[12]](#footnote-3)* è noto.

Nei sistemi reali la matrice *R n \* m* contenente le valutazioni degli utenti sugli elementi sarà una matrice sparsa di grandi dimensioni, quindi si può ridurre il problema fattorizzando la matrice *R* utilizzando la decomposizione a valori singolari troncata *(truncated SVD)[[13]](#endnote-10)* di *R*. Utilizzando questo metodo *R* equivale al prodotto delle matrici U, D, VT *(la trasposta di V)*, *R = UDVT*. Successivamente si impostano le matrici *W* e *H*, rispettivamente. *W* è il risultato del prodotto fra le prime *k* colonne di U *(Uk)* e la radice quadrata della matrice D troncata *(Dk1/2)*, mentre *H* è il risultato del prodotto tra la radice quadrata della matrice D troncata *(Dk1/2)* e le prime *k* righe della trasposta della matrice V *(*VkT*)*, dove *Uk, Dk, VkT* contengono i *k* triplete singolari maggiori di R.

Sistemi ibridi

I sistemi ibridi utilizzano tecniche che appartengono sia ai sistemi collaborativi che ai sistemi basati sul contenuto e calcolano le raccomandazioni usando entrambi gli approcci. Le implementazioni dei sistemi ibridi prevedono il calcolo delle predizioni sia con l’approccio basato sul contenuto che utilizzando l’approccio collaborativo separatamente e poi successivamente vengono combinate le previsioni ottenute da entrambi gli approcci.

Valutazione di un sistema

Per valutare un sistema di apprendimento automatico, vengono utilizzate delle metriche di valutazione, le metriche principalmente utilizzate sui sistemi di raccomandazione sono il *supporto* e la *confidenza*.

Considerazioni in ambito assicurativo

L’approccio descritto risulta valido in ambito *e-commerce, streaming on-line* ma non si può applicare in ambito assicurativo, in quanto ci sono delle peculiarità[[14]](#endnote-11) del dominio applicativo troppo complesse per essere rappresentate nella matrice users-items; pertanto è necessario utilizzare degli accorgimenti.

Le peculiarità dell’ambito assicurativo che si possono individuare sono:

* **Cardinalità ridotta**: l’offerta assicurativa può contenere un numero massimo di 100 prodotti rispetto all’offerta delle piattaforme e-commerce *(es. milioni di libri presenti su Amazon)* o delle piattaforme di streaming on-line *(es. migliaia di film e serie tv presenti su Netflix)* l’offerta assicurativa risulta limitata.
* **Affidabilità**: l’acquisto di un libro, la visione di un film o l’acquisto di altri prodotti di uso comune che vengono acquistati giornalmente o settimanalmente. Le coperture assicurative vengono modificate raramente pertanto è necessario avere un elevato livello di affidabilità da parte del cliente. L’errato acquisto di un libro o di un film creerà al cliente un danno minimo, sia in termini economici che in quantità di tempo. Sbagliare a sottoscrivere l’assicurazione causerà un notevole danno al cliente, inoltre per modificare il contratto è necessario eseguire una variazione *(voltura)*.
* **Vincoli**: quando si sottoscrive un’assicurazione ci sono diversi vincoli che bisogna rispettare, *es. bonus per le assicurazioni auto se non sono stati commessi sinistri negli ultimi anni, bonus se si sta assicurando la prima casa, limite d’età per le assicurazioni sulla salute, limiti legati a patologie per le assicurazioni sulla salute.*

1. **Recommender Systems**, Prem Melville and Vikas Sindhwani [↑](#endnote-ref-1)
2. [**TF-IDF — Term Frequency-Inverse Document Frequency – LearnDataSci**](https://www.learndatasci.com/glossary/tf-idf-term-frequency-inverse-document-frequency/), Fatih Karabiber, [↑](#endnote-ref-2)
3. [**Stop word - Wikipedia**](https://en.wikipedia.org/wiki/Stop_word), Wikipedia [↑](#endnote-ref-3)
4. [**Getting Started With Embeddings (huggingface.co)**](https://huggingface.co/blog/getting-started-with-embeddings), Omar Espejel [↑](#endnote-ref-4)
5. [**Coseno - Wikipedia**](https://it.wikipedia.org/wiki/Coseno), Wikipedia [↑](#endnote-ref-5)
6. John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Madison, WI, July 1998 [↑](#endnote-ref-6)
7. Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing,

   7(1):76–80, 2003. [↑](#endnote-ref-7)
8. Daniel Billsus and Michael J. Pazzani. Learning collaborative information filters. In Proceedings of the Fifteenth International Conference on Machine Learning (ICML-98), pages 46–54, Madison, WI, 1998. Morgan Kaufmann [↑](#endnote-ref-8)
9. Feature: le caratteristiche che identificano un elemento *(item)* [↑](#footnote-ref-1)
10. Label: l’etichetta valore su cui eseguire la predizione [↑](#footnote-ref-2)
11. [Loss Function](https://en.wikipedia.org/wiki/Loss_function) [↑](#endnote-ref-9)
12. Rating: valutazione di un utente su di un elemento [↑](#footnote-ref-3)
13. [Decomposizione a valori singolari (SVD)](https://it.wikipedia.org/wiki/Decomposizione_ai_valori_singolari) [↑](#endnote-ref-10)
14. **A Recommendation System For Car Insurance**, Laurent Lesage, Madalina Deaconu, Antoine Lejay, Jorge Augusto Meira, Geoffrey Nichil, Radu State [↑](#endnote-ref-11)