Introduzione

L’attività di tirocinio è stata svolta internamente all’azienda per cui lavoro ***Fincons Group S.p.A.*** su di un suo cliente assicurativo.

Ho utilizzato l’applicativo del cliente per emettere delle polizze assicurative, *in ambiente di test*. L’applicativo risulta essere lento, anonimo e di poco impatto.

Le informazioni disponibili per il cliente che si sta analizzando nella pagina utilizzata per l’avvio dell’emissione sono scarse e di poca utilità per ottenere ulteriori informazioni riguardo il è necessario cambiare pagina e caricare la pagina del profilo cliente.

Successivamente l’utente deve ricominciare il processo emissivo; il processo emissivo può essere avviato solo se per il cliente sono stati acquisiti almeno i dati fondamentali. Se il profilo del cliente non è stato completato ricevo un messaggio dopo aver selezionato i beni che voglio assicurare e clicco su tasto *procedi* per avviare effettivamente l’emissione, questo comportamento potrebbe causare una perdita di tempo da parte dall’agente, in quanto bisogna tornare sulla pagina del profilo cliente, aggiornare il profilo con i dati mancanti e successivamente proseguire con l’emissione ricominciando il processo.

Durante i test sul processo emissivo ho cercato di individuare quali potessero essere i dati minimi dei clienti che fossero utili per l’addestramento del sistema di raccomandazione. In seguito ai test ed analisi eseguite i dati individuati sono stati:

* Sesso.
* Data di nascita.
* Stato civile.
* Indicazione se il coniuge è a carico.
* Numero figli.
* Indicazione del numero figli a carico.
* Tipologia di nucleo familiare.
* Professione.
* Indicazione se si tratta di un libero professionista o di un lavoratore dipendente.
* Reddito *(RAL)*.
* Provincia di residenza.

L’utilizzo di alcuni dati potrebbe essere considerato discriminatorio però in questo caso si sta analizzando il dominio applicativo per costruire un sistema personalizzato quindi le informazioni selezionate sono utili per la costruzione del profilo cliente.

Per la costruzione del sistema di raccomandazione è utile avere come informazione le polizze sottoscritte dal cliente quindi alle caratteristiche del cliente indicate verrà aggiunto il codice o il nome della polizza o delle polizze sottoscritte.

Dopo aver acquisito i dati ho iniziato i test per la costruzione del sistema di raccomandazione, un sistema di raccomandazione esegue le predizioni basandosi su di una struttura dati chiamata *matrice users-items*.

La matrice users-items è costituita dagli users, *in questo caso i clienti*, presenti sulle righe e dagli items, *in questo caso le polizze*, presenti sulle colonne in corrispondenza della coppia *user-item* viene salvato il rating, *la valutazione dell’utente generico per l’item generico*.

Nel dominio applicativo che si sta analizzando non esiste una valutazione esplicita da parte dell’utente *(cliente)* in merito al prodotto/servizio offerto quindi possiamo ipotizzare una gestione binaria del rating, *1 – se la polizza è stata sottoscritta quindi una valutazione positiva mentre 0 – se la polizza non è stata sottoscritta quindi una valutazione negativa*.

Durante l’esecuzione del primo test la matrice users-items è stata popolata solo con le valutazioni positive inserendo solo le coppie *user-item* con 1, eseguendo il training del modello con questa configurazione il modello ottenuto è stato considerato inconsistente, in quanto ha ottenuto uno score dello 0.5 *(50 %)* non era in grado di effettuare previsioni.

Il secondo test ha previsto la creazione di una matrice users-items che prendesse in considerazione tutte le possibili coppie *cliente-polizza* dove per ogni cella se il cliente avesse sottoscritto la polizza veniva inserito *“1”* altrimenti veniva inserito *“0”*, eseguendo il training del modello con questa seconda configurazione il modello ottenuto è stato considerato inconsistente, in quanto continuava ad avere uno score dello 0.5 *(50 %)* circa, l’unico valore in grado di predire era la polizza R.C.A. con una probabilità del 100 %. Questo comportamento del modello di raccomandazione è legato alla distribuzione della matrice users-items, in quanto è una matrice sparsa e molto sbilanciata per quanto riguarda le polizze R.C.A, la tabella mostra la distribuzione delle polizze.

|  |  |
| --- | --- |
| **Nome Polizza** | **Numero Clienti** |
| R.C.A. | 574047 |
| INFORTUNI | 64399 |
| MULTIGARANZIA ABITAZIONE | 2415 |
| MALATTIA | 768 |
| R.C. DIVERSI | 78 |
| A.R.D. | 22 |
| GLOBALE FABBRICATI | 8 |
| INCENDIO/FURTO | 4 |
| TUTELA GIUDIZIARIA | 2 |

Tabella con Distribuzione dei dati

Grafico che mostra la distribuzione delle polizze acquistate dai clienti

Per risolvere il problema dello stato della matrice sparsa dopo avere eseguito delle ricerche nella letteratura riguardo i *recommender system*, ho trovato l’implementazione di un framework per sistema di raccomandazione applicato ad un dataset di films.

Per valorizzare le celle della matrice users-items non valorizzate ho definito un algoritmo di classificazione, il quale prevede la prossima polizza che l’utente andrà a sottoscrivere e lo score della predizione viene utilizzato per valorizzare la cella della matrice users-items.

Sono stati necessari alcuni test prima di trovare la configurazione migliore per il modello di classificazione.

Sul prima modello creato ho definito le proprietà del dataset illustrato nella tabella:

|  |  |
| --- | --- |
| **Nome Campo** | **Ruolo Campo** |
| Sesso | Caratteristica cliente - *Feature* |
| Data di nascita | Caratteristica cliente - *Feature* |
| Stato civile | Caratteristica cliente - *Feature* |
| Coniuge è a carico | Caratteristica cliente - *Feature* |
| Numero figli | Caratteristica cliente - *Feature* |
| Numero figli a carico | Caratteristica cliente - *Feature* |
| Tipologia di nucleo familiare | Caratteristica cliente - *Feature* |
| Professione | Caratteristica cliente - *Feature* |
| Libero Professionista/Lavoratore Dipendente | Caratteristica cliente - *Feature* |
| Reddito *(RAL)* | Caratteristica cliente - *Feature* |
| Provincia di residenza | Caratteristica cliente – *Feature* |
| Identificativo Polizza | Valore di predizione – *Label* |

Tabella composizione Dataset

Le predizioni ottenute dal modello di classificazione così configurato formano una matrice users-items molto simile alla matrice users-items iniziale quindi il modello di classificazione creato non va bene.

Il dataset per l’algoritmo di classificazione così configurato presenta un altro problema nel caso venga acquisito un nuovo cliente è molto difficile individuare all’interno del dataset un cliente che abbia le stesse caratteristiche del cliente che il sistema ha acquisito; per questo motivo ho applicato al dataset delle funzioni di trasformazione del dato in modo da rendere il dataset delle features cliente più generico. Nella tabella vengono mostrate le features del nuovo dataset:

|  |  |
| --- | --- |
| **Nome Campo** | **Ruolo Campo** |
| Sesso | Caratteristica cliente - *Feature* |
| Data di nascita | Caratteristica cliente - *Feature* |
| Stato civile | Caratteristica cliente - *Feature* |
| Single | Caratteristica cliente - *Feature* |
| Coniuge è a carico | Caratteristica cliente - *Feature* |
| Numero figli | Caratteristica cliente - *Feature* |
| Numero figli a carico | Caratteristica cliente - *Feature* |
| Tipologia di nucleo familiare | Caratteristica cliente - *Feature* |
| Professione | Caratteristica cliente - *Feature* |
| Libero Professionista/Lavoratore Dipendente | Caratteristica cliente - *Feature* |
| Freelancer | Caratteristica cliente - *Feature* |
| Fascia reddito | Caratteristica cliente - *Feature* |
| Provincia di residenza | Caratteristica cliente – *Feature* |
| Regione di residenza | Caratteristica cliente - *Feature* |
| Identificativo Polizza | Valore di predizione – *Label* |

Tabella composizione Dataset

Oltre alle caratteristiche vere e proprie del cliente per rendere utilizzabile il classificatore sono state aggiunte come ulteriori feature alcune caratteristiche della polizza sottoscritta, anche questa fase è avvenuta per passi successivi.

* Il primo passo ha previsto l’aggiunta del codice polizza questa caratteristica ha permesso di ottenere un classificatore che produceva delle predizioni più omogenee ma per le polizze sottoscritte con minore frequenza il modello continuava a non effettuare la predizione.
* Il secondo passo ha previsto l’aggiunta del nome polizza questa caratteristica ha permesso di ottenere un classificatore che produceva delle predizioni più omogenee andando a predire in alcuni casi anche i valori per le polizze sottoscritte con la frequenza minima.
* Il terzo passo ha previsto l’aggiunta della descrizione polizza questa caratteristica ha permesso di ottenere un classificatore che produceva le predizioni anche per le polizze sottoscritte con frequenza minore ma non riusciva a predire le polizze con la frequenza minima.

Dai test eseguiti la configurazione migliore per il classificatore risulta essere quella presentata in tabella:

|  |  |
| --- | --- |
| **Nome Campo** | **Ruolo Campo** |
| Sesso | Caratteristica cliente - *Feature* |
| Data di nascita | Caratteristica cliente - *Feature* |
| Stato civile | Caratteristica cliente - *Feature* |
| Single | Caratteristica cliente - *Feature* |
| Coniuge è a carico | Caratteristica cliente - *Feature* |
| Numero figli | Caratteristica cliente - *Feature* |
| Numero figli a carico | Caratteristica cliente - *Feature* |
| Tipologia di nucleo familiare | Caratteristica cliente - *Feature* |
| Professione | Caratteristica cliente - *Feature* |
| Libero Professionista/Lavoratore Dipendente | Caratteristica cliente - *Feature* |
| Freelancer | Caratteristica cliente - *Feature* |
| Fascia reddito | Caratteristica cliente - *Feature* |
| Provincia di residenza | Caratteristica cliente – *Feature* |
| Regione di residenza | Caratteristica cliente - *Feature* |
| Codice polizza | Caratteristica cliente – *Feature* |
| Nome polizza | Caratteristica cliente – *Feature* |
| Identificativo polizza | Valore di predizione – *Label* |

Tabella composizione Dataset

Dopo aver eseguito l’addestramento del modello ho provveduto a popolare la matrice users-items con lo score delle predizioni calcolato per ogni coppia *cliente-polizza*.

Una volta popolata la matrice users-items ho provveduto ad eseguire l’addestramento del sistema di raccomandazione.

Il ciclo descritto è stato eseguito per le varie configurazioni fino a quando non sono riuscito a determinare la configurazione con le prestazioni migliori. I passi seguiti si possono riassumere:

1. Acquisizione dei dati dallo storage *(file .csv)*.
2. Salvataggio dei dati sulla tabella del database utilizzata come dataset.
3. Normalizzazione delle caratteristiche cliente.
4. Aggiunta caratteristiche polizza al dataset come caratteristiche cliente aggiuntive.
5. Addestramento modello di classificazione.
6. Valutazione modello di classificazione.
7. Popolamento matrice users-items.
8. Addestramento sistema di raccomandazione.
9. Valutazione sistema di raccomandazione.