UNIVERSITÀ POLITECNICA DELLE MARCHE

FACOLTÀ DI ECONOMIA "GIORGIO FUÀ"

Corso di Laurea Magistrale in Data Science per l'Economia e le Imprese

Tecniche Avanzate di Analisi dei Dati per il Customer Service: Business Intelligence e NLP nel Settore della GDO

Advanced Data Analysis Techniques for Customer Service: Business Intelligence and NLP in the Retail Sector

Relatore:

Prof. Domenico Potena

Tesi di Laurea di:

Leandro Sciuto

A.A. 2023/2024

Indice

Introduzione

Nel panorama odierno della Grande Distribuzione Organizzata (GDO), la digitalizzazione dei processi ha generato una quantità sempre maggiore di dati preziosi. Questi dati, che spaziano dalla logistica alle abitudini di acquisto dei clienti, offrono alle aziende l'opportunità di estrarre conoscenze utili per ottimizzare le strategie e prendere decisioni informate. In questo contesto, gli strumenti di data analytics diventano essenziali per trasformare i dati grezzi in informazioni strategiche.

Magazzini Gabrielli S.p.A., un'azienda nella GDO con sede nelle Marche e Abruzzo, in fase di espansione verso il centro Italia, gestisce una vasta rete di superstore, supermercati e punti vendita in franchising. L'elevato numero di punti vendita ha portato l'azienda a esternalizzare il servizio clienti a un call center esterno. Per valutare l'efficacia di questo servizio, l'azienda ha richiesto un'analisi approfondita dei dati relativi alle attività di supporto (ticket) gestite dal call center.

L'obiettivo di questa tesi è analizzare i dati dei ticket per ottenere una panoramica completa sulla qualità del servizio clienti offerto.

Per raggiungere questi obiettivi, la tesi si avvarrà di tre metodologie analitiche complementari:

- **Business Intelligence (BI):** Verrà impiegata per effettuare un'analisi descrittiva dei dati dei ticket, al fine di monitorare a posteriori il servizio di assistenza tecnica erogato. Strumenti come Microsoft PowerBI saranno utilizzati per estrarre informazioni sulle caratteristiche dei ticket, come il tempo di risposta e il tipo di problema riscontrato.

- **Text Mining:** Attraverso tecniche di text mining sarà possibile analizzare i contenuti testuali dei ticket per identificare i temi ricorrenti. Queste informazioni saranno cruciali per la creazione di una nuova metodologia di valutazione del servizio, che vada oltre i semplici indicatori basati sui tempi di risposta e le classificazioni assegnate dagli utenti.

- **Natural Language Processing (NLP):** Attraverso l'uso di modelli di elaborazione del linguaggio naturale, Large Language Models in particolare, si analizzerà il contenuto dei ticket per tracciare un quadro sintetico ed esplicativo dei problemi riscontrati dai clienti. Questa analisi permetterà di

individuare temi e problematiche ricorrenti, fornendo una visione più approfondita delle esigenze e delle percezioni degli operatori.

In conclusione, questa tesi mira a fornire a Magazzini Gabrielli gli strumenti e le conoscenze necessarie per migliorare il servizio clienti, basandosi su un'analisi dettagliata e approfondita dei dati raccolti dal call center esterno. Le metodologie proposte permetteranno di valutare in modo oggettivo le performance attuali e di prendere decisioni strategiche informate per il futuro dell'azienda.

Capitolo 1

1.1 Collegamento Database e Operazioni Preliminari sul Dataset

La presente tesi si concentra sullo sviluppo e l'implementazione di un cruscotto interattivo utilizzando Power BI, uno strumento di Business Intelligence di Microsoft, e del suo collegamento diretto con il database IBM DB2 dell'azienda. Il cruscotto è stato progettato per migliorare la visualizzazione e l'analisi dei dati aziendali, facilitando decisioni basate su dati concreti e aggiornati.

In particolare, il cruscotto deve essere in grado di:

- Collegarsi in modo sicuro al database IBM DB2 tramite una connessione VPN aziendale.
- Recuperare e visualizzare dati rilevanti in tempo reale.
- Fornire una rappresentazione chiara e interattiva dei dati dei ticket, migliorando così l'efficacia delle operazioni di Magazzini Gabrielli.

La prima fase del progetto ha comportato una serie di passaggi tecnici e metodologici per garantire la corretta integrazione tra Power BI e il database IBM DB2. Di seguito, si descrivono in dettaglio i principali passaggi eseguiti.

Per garantire la sicurezza e la riservatezza dei dati durante il trasferimento, è stata stabilita una connessione VPN (Virtual Private Network) con l'azienda. Questo passaggio è stato cruciale per permettere a Power BI di accedere ai dati nel database IBM DB2 senza esporre le informazioni a potenziali rischi esterni.

Una volta stabilita la connessione VPN, il passo successivo è stato configurare Power BI per connettersi al database IBM DB2. Questo ha comportato la creazione di query specifiche per estrarre i dati necessari. La query principale utilizzata è stata la seguente:

```
SELECT *
FROM TICKET t
FULL OUTER JOIN LONGDESCRIPTION l
ON t.TICKETUID = l.LDKEY
WHERE t.reportdate >= '2021-01-01'
AND LDOWNERCOL = 'FR2CODE'
ORDER BY t.REPORTDATE ASC
```

Questa query ha permesso di recuperare tutti i ticket con una descrizione dettagliata, filtrando i dati a partire dal 1º gennaio 2021 e ordinandoli per data di segnalazione in ordine cronologico.

Dopo il caricamento dei dati, sono state svolte delle modifiche direttamente su Power BI tramite lo strumento di Power Query. In primis, è stato eseguito uno script Python per estrarre da una colonna chiamata “LDTEXT”, contenente la descrizione del problema riscontrato, tre categorie chiamate “levels” che l’azienda usa per dare una classificazione sommaria dei ticket risolti. La prima parte spiega l’oggetto del problema, la seconda parte la causa e la terza parte la soluzione proposta.

```
import pandas as pd
import numpy as np
import re

def extract_levels(text):
    if pd.isnull(text):
        return np.nan, np.nan, np.nan
    match = re.search(r'{{(.+?)-//-([^)]+?)-//-([^)]+?)}}', text)
    if match:
        return match.groups()
    else:
        return np.nan, np.nan, np.nan

dataset[['first_level', 'second_level', 'third_level']] =
dataset['LDTEXT'].apply(lambda x: pd.Series(extract_levels(x)))
```

Ad esempio, se all’interno della colonna ‘LDTEXT’ di un ticket vi fosse scritto “[...] {{SW-VISUALSTORE-//-SW-ERRORE DI PROGRAMMA F.E.-//-PROBLEMA NON RISCONTRATO}}”, verrebbero create tre

colonne “first level”, “second level” e “third level” con i seguenti rispettivi valori:

- First level: SW-VISUALSTORE
- Second level: SW-ERRORE DI PROGRAMMA F.E.
- Third level: PROBLEMA NON RISCONTRATO

A seguito di ciò, le operazioni sono state invece quelle di:

- Formattazione delle date con la seguente espressione:

```
= Table.TransformColumns(#"Rimosse colonne",{{"Value.REPORTDATE", each Date.From(DateTimeZone.From(_)), type date}, {"Value.STATUSDATE", each Date.From(DateTimeZone.From(_)), type date}, {"Value.AFFECTEDDATE", each Date.From(DateTimeZone.From(_)), type date}, {"Value.ACTUALSTART", each Date.From(DateTimeZone.From(_)), type date}, {"Value.ACTUALFINISH", each Date.From(DateTimeZone.From(_)), type date}, {"Value.CHANGEDATE", each Date.From(DateTimeZone.From(_)), type date},
```

- Rinominazione colonne Date Inizio e Fine:

```
= Table.RenameColumns(#"Data analizzata",{{"Value.ACTUALFINISH", "Data_Fine"}, {"Value.ACTUALSTART", "Data_Inizio"}})
```

- Creazione misure Durata_giorni e Durata_minuti:

```
Durata_giorni = DATEDIFF(Query2[Data_Inizio],
Query2[Data_Fine], DAY)
Durata  minuti = DATEDIFF(Query2[Data  Inizio],
```

Queste operazioni hanno permesso di strutturare i dati in modo adeguato alla successiva analisi, garantendo che fossero pronti per essere visualizzati e analizzati attraverso Power BI.

1.2 Dashboard Power BI e Analisi Descrittiva

Dopo aver completato l'integrazione tra Power BI e il database IBM DB2 e aver eseguito le operazioni preliminari sul dataset, il cruscotto interattivo ha iniziato a fornire preziose informazioni. Prima di parlare dei risultati però, al fine di una più chiara lettura, si ritiene opportuno fare il punto della situazione sul dataset a nostra disposizione tramite la Tabella 1.

Variabile	Tipo	Descrizione
Ticket ID	int	Identificativo univoco del ticket
Typestart	testo	Indica la modalità di apertura del ticket, ad esempio (“Email”, “App” o “Telefono”)
ReportedBy	testo	Indica l’operatore che ha aperto il ticket
DataInizio	data	Data di apertura del ticket
DataFine	data	Data di chiusura del ticket
Durata_giorni	float	Numero di giorni tra apertura e chiusura del ticket
Durata_minuti	float	Numero di minuti tra apertura e chiusura del ticket
LDTEXT	testo	Contiene i testi inseriti dagli utenti all’atto di apertura del ticket e le risposte degli operatori del servizio
Topic First Level	testo	Indica l’oggetto del problema riscontrato che ha portato all’apertura di un ticket
Topic Second Level	testo	Indica la causa del problema riscontrato che ha portato all’apertura di un ticket
Topic Third Level	testo	Indica la soluzione proposta

Tabella 1: Descrizione del dataset e delle sue variabili

Sarà quindi sulla base di queste variabili che verranno descritti i principali risultati ottenuti attraverso l'uso della dashboard di Power BI, focalizzandosi su diverse dimensioni analitiche dei dati dei ticket.

Questa, infatti, offre una vista d'insieme dei ticket gestiti, visualizzando metriche chiave come il numero totale di ticket, la loro distribuzione nel tempo e la suddivisione per categoria. Consente inoltre di identificare immediatamente i volumi di lavoro e le tendenze generali, come i periodi con picchi di richieste.

1.2.1 Canali di Comunicazione e Ticket per Utente

La prima dashboard presenta una panoramica sui canali di comunicazione utilizzati per l'apertura dei ticket. Dai dati emerge che l'e-mail è il canale di comunicazione più utilizzato, rappresentando il 69,82% dei ticket. Il telefono è il secondo canale più utilizzato, con il 28,61% dei ticket, mentre l'app è utilizzata solo per l'1,47% dei ticket.

L'alta percentuale di ticket aperti via e-mail suggerisce una preferenza per i canali digitali asincroni, che permettono agli utenti di segnalare problemi senza la necessità di un'interazione immediata. Il telefono, sebbene meno utilizzato rispetto all'email, rappresenta ancora un mezzo significativo per la comunicazione. L'uso minimo dell'app indica una bassa adozione di questo canale.

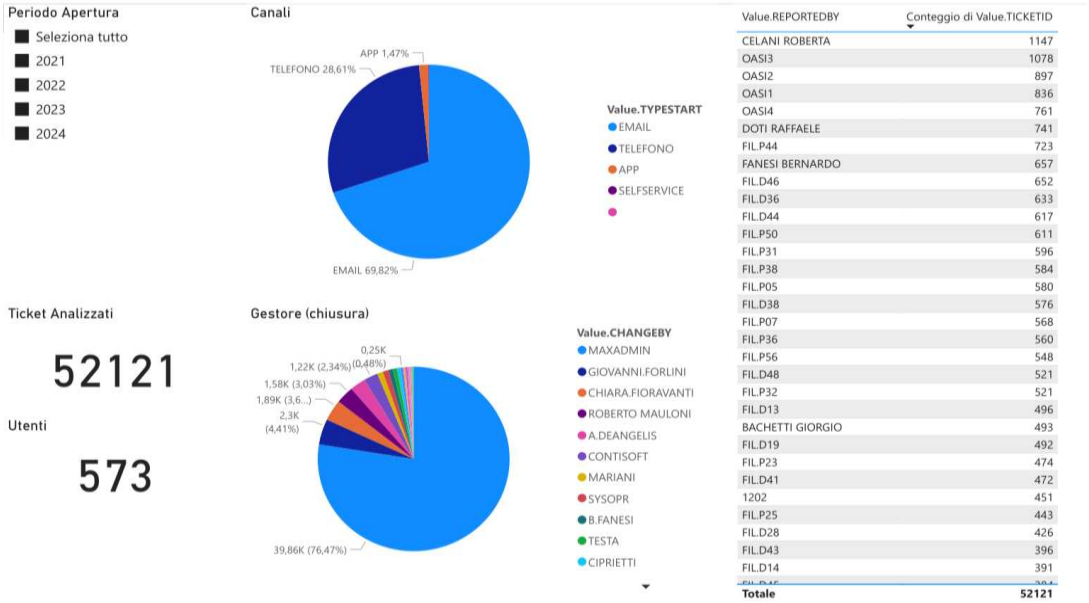

Figura 1.1: Prima pagina della dashboard

La dashboard fornisce una visione dettagliata del numero totale di ticket (52.121) e del numero di utenti che li hanno segnalati (573). Vengono evidenziati i principali reporter di ticket, con Celani Roberta in testa con 1.147 ticket, seguita da OASI3 con 1.078 ticket e OASI2 con 897 ticket.

Un numero ristretto di utenti contribuisce in modo significativo al volume totale dei ticket, suggerendo che ci potrebbero essere dipartimenti o ruoli con maggiori esigenze di supporto. La distribuzione dei ticket tra gli utenti indica che alcuni potrebbero beneficiare di formazione aggiuntiva o risorse specifiche per ridurre il numero di richieste di supporto.

1.2.3 Andamento Apertura Ticket

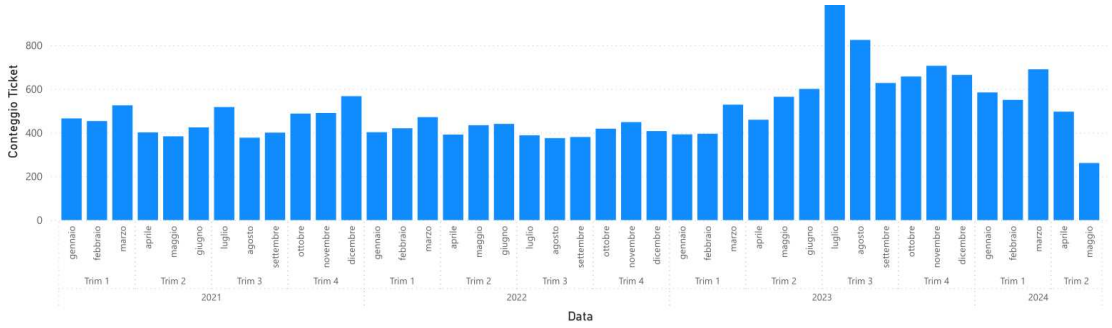

Figura 1.2: Andamento temporale di apertura dei ticket

La terza dashboard mostra un grafico temporale dell'apertura dei ticket dal gennaio 2021 agli inizi di maggio 2024, suddiviso per trimestri per un'analisi dettagliata della tendenza. L'andamento dei ticket può aiutare a identificare i periodi di maggiore pressione sul team di supporto e a pianificare le risorse di conseguenza.

Si osserva una variabilità nell'apertura dei ticket durante l'anno, con picchi e cali che potrebbero essere correlati a fattori stagionali o operativi specifici. In particolare, si può notare come i principali picchi siano stati in presenza di festività che generalmente portano ad un aumento degli acquisti, come nei mesi di novembre e dicembre. Più anomalo invece è l'ammontare di ticket aperti nei mesi di luglio ed agosto 2023, situazione che merita una maggiore attenzione.

1.2.4 Topic dei Ticket

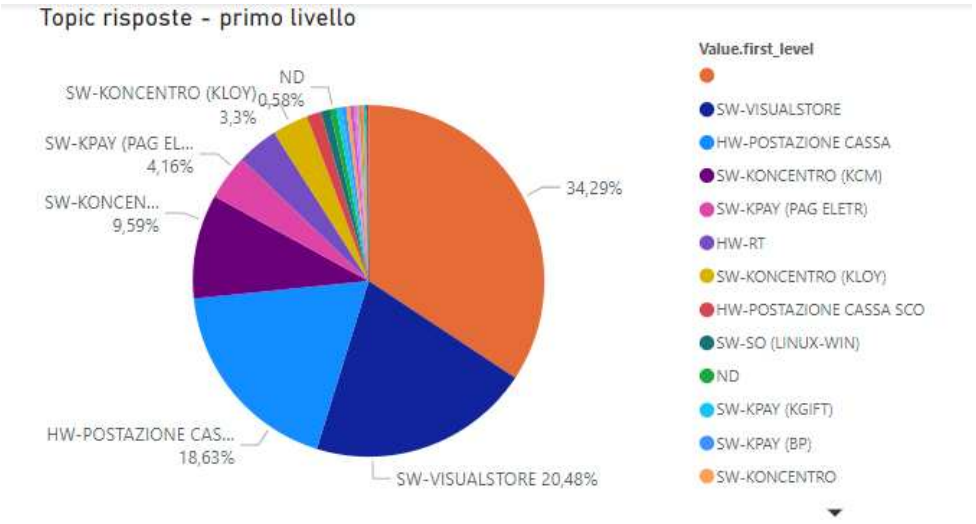

Figura 1.3: Primo livello dei Topic Risposte

La categoria “first level” nella Figura 1.3 elenca l’oggetto del problema riscontrato che ha portato all’apertura di un ticket, con SW-Visualstore al primo posto con 10.712 ticket, seguita da HW-Postazione Cassa con 9.743 ticket e SW-Koncentro (KCM) con 5.016 ticket.

Le categorie più frequenti, quindi, riguardano sia componenti hardware come la postazione cassa che software, con una leggera predominanza delle richieste dovute al gestionale Visualstore.

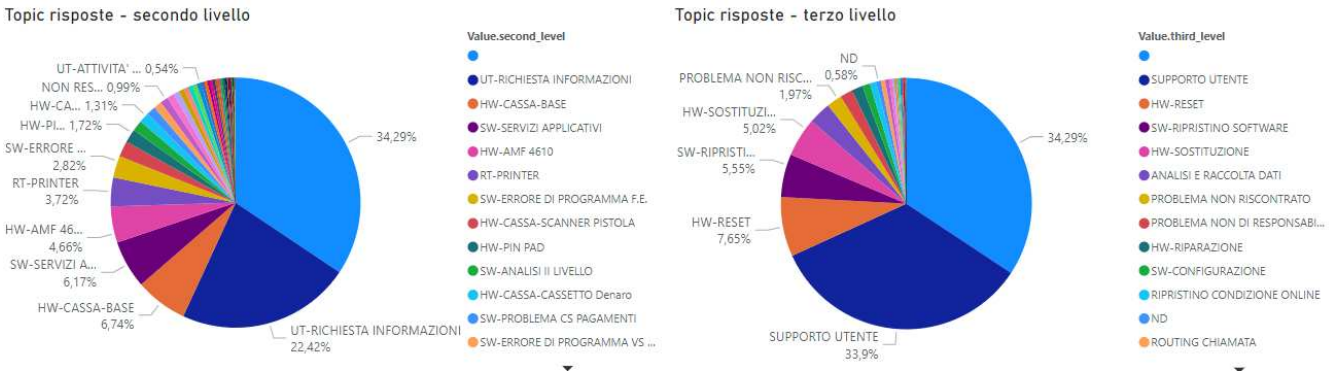

Figura 1.4: Secondo e Terzo livello dei Topic Risposte

Il secondo e il terzo livello del topic invece spiegano rispettivamente la causa del problema e la soluzione proposta. Si può notare come quindi la maggioranza dei ticket siano stati aperti a seguito di una richiesta da parte degli utenti e che la soluzione proposta sia stata di tipo “Supporto Utente”.

1.2.5 Distribuzione Temporale dei Ticket

Media di Durata_giorni	Deviazione standard di Durata_giorni	Media di Durata_minuti	Deviazione standard di Durata_minuti
2,82	15,46	4062,42	22258,78

Figura 1.5: Andamento temporale di apertura dei ticket

Altre informazioni utili a disposizione tramite la dashboard riguardano le tempistiche. La media della durata in giorni tra l’apertura e la chiusura di un ticket è di 2,82 giorni, con una deviazione standard di 15,46 giorni. In minuti, la media è di 4.062,42 minuti con una deviazione standard di 22.258,78

minuti. Questi dati suggeriscono che la maggior parte dei problemi venga risolta rapidamente, ma la deviazione standard elevata indica la presenza di alcuni ticket che richiedono tempi significativamente più lunghi.

La dashboard permette anche la visualizzazione dei tempi medi di chiusura dei ticket per le due categorie più preponderanti ovvero "HW-POSTAZIONE CASSA" e "SW-VISUALSTORE". In particolare, possiamo andare a confrontare le categorie second level e third level che fungono come ulteriore classificazione.

Nel caso dei topic del tipo "SW-VISUALSTORE", si può evincere come i ticket che al secondo livello mostrano l'etichetta "RILEVATO BUG", abbiano una durata di risoluzione media decisamente maggiore rispetto alla media totale, pari a 53,58 giorni.

Nei problemi del tipo "HW-POSTAZIONE CASSA" invece non vi sono casi limite come questo, suggerendo come i problemi software siano più complessi da gestire e risolvere rispetto a quelli hardware.

Capitolo 2

2.1.1 Operazioni preliminari in ambito di text mining

Il secondo obiettivo di questo progetto, ovvero l'analisi del testo presente nel campo LDTEXT del dataset, assume un ruolo cruciale per estrarre informazioni significative dai ticket di assistenza dei supermercati. Come già accennato nel capitolo precedente, questo campo racchiude i testi inseriti dagli utenti all'apertura dell'istanza e le relative risposte degli operatori e sono molteplici le problematiche che ostacolano un'analisi efficace del suo contenuto.

La principale sfida risiede nell'elevata presenza di simboli e sigle generate principalmente da quei canali di comunicazione come ad esempio EMAIL e APP. Questi testi, infatti, vengono salvati con un insieme di tag di formattazione propri della pagina web o del servizio di posta elettronica. Tale "rumore" eccessivo, se non eliminato, rischia di compromettere l'accuratezza delle tecniche di analisi.

Per ovviare a questa criticità, è stata implementata una fase di pre-processing dedicata alla colonna LDTEXT. Attraverso un'analisi manuale approfondita dei ticket, è stato definito un insieme di operazioni di pulizia volte a standardizzare i termini e a rimuovere i token irrilevanti.

```
import re
import pandas as pd
import numpy as np
from bs4 import BeautifulSoup
import unicodedata

def clean_ldtext(text):
    if pd.isnull(text):
        return np.nan

    # Rimozione dei tag HTML e CSS
    text = BeautifulSoup(text, "html.parser").get_text()

    # Rimozione degli hyperlinks e hashtags
    text = re.sub(r'http\S+|www.\S+|#\S+', ' ', text)

    # Rimozione dei simboli di formattazione, segni di punteggiatura e altri caratteri
speciali
    text = re.sub(r'[^\w\s]', ' ', text)
```

Figura 2.1: Codice - Snippet del codice per pulizia del testo

Il codice sviluppato per la pulizia del testo nel campo LDTEXT del dataset TICKET si avvale di diverse librerie Python, tra cui re, pandas, numpy, BeautifulSoup e unicodedata. La funzione principale, clean_ldtext, è progettata per normalizzare e depurare il testo da elementi superflui e rumorosi. Inizialmente, la funzione verifica se il testo è nullo e in tal caso restituisce un valore np.nan per gestire correttamente i valori mancanti. Successivamente, utilizza BeautifulSoup per rimuovere i tag HTML e CSS, estraendo il testo pulito. Per eliminare gli URL e gli hashtag, si applicano espressioni regolari che sostituiscono tali elementi con spazi vuoti. Un ulteriore passaggio con espressioni regolari rimuove simboli di

formattazione, segni di punteggiatura e altri caratteri speciali, lasciando solo caratteri alfanumerici e spazi.

```
# Conversione di tutti i caratteri in minuscolo
    text = text.lower()

    # Gestione delle parole con l'apostrofo
    text = re.sub(r'\b(l)(\')(\w+)', r'la\3', text)
    text = re.sub(r'\b(un)(\')(\w+)', r'uno\3', text)

    # Gestione delle lettere accentate
    text = unicodedata.normalize('NFKD', text).encode('ASCII',
'ignore').decode('utf-8')

    # Rimozione degli spazi multipli
    text = re.sub(r'\s+', ' ', text).strip()

    return text

# Utilizzo
df['LDTEXT'] = df['LDTEXT'].apply(clean_ldtext)
```

Figura 2.2: Codice - Snippet del codice per pulizia del testo

Il testo viene poi convertito in minuscolo per garantire l'uniformità dei dati. La gestione delle parole con l'apostrofo è effettuata tramite sostituzioni specifiche, trasformando contrazioni come "l'opzione" in "la opzione" e "un'opzione" in "un opzione".

Per trattare le lettere accentate, la funzione utilizza “unicodedata” per normalizzare i caratteri, convertendoli nelle loro versioni non accentate. Infine, gli spazi multipli sono ridotti a uno singolo, e vengono rimossi gli spazi iniziali e finali. La funzione clean_ldtext è quindi applicata a ogni

elemento della colonna LDTEXT del data frame tramite df['LDTEXT'].apply(clean_ldtext), garantendo così che i testi siano depurati e pronti per le successive fasi di analisi testuale.

2.1.2 Pulizia del testo

Come evidenziato nel capitolo precedente, la maggior parte delle segnalazioni avviene tramite e-mail, presentando un'ulteriore sfida significativa: i testi dei ticket inviati via e-mail contengono spesso informazioni aggiuntive come contatti del mittente e del destinatario, oggetti, espressioni di saluto e ringraziamento, firme in calce e, in alcuni casi, persino testi inoltrati da altre e-mail, ognuno con le proprie informazioni aggiuntive. Questi contenuti rappresentano un ostacolo notevole per l'efficacia delle tecniche di analisi del testo poiché le informazioni superflue possono compromettere l'accuratezza dei risultati e rendere più complessa l'individuazione dei contenuti rilevanti. Si è reso necessario quindi individuare una tecnica in grado di identificare e rimuovere in modo efficiente le parti di testo frequenti, come le firme in calce e altre sigle ricorrenti, in un corpus di documenti. Nella letteratura, questo processo è noto come "stopword removal" e nel nostro caso viene implementato utilizzando la tecnica degli N-grammi.

Questa tecnica consiste nella divisione del testo in sequenze contigue di “N” elementi consecutivi, chiamati appunto N-grammi, dove “N” rappresenta il parametro di modellazione. I tipi più comuni di N-grammi sono i bi-grammi (N = 2), i tri-grammi (N = 3) e i 4-grammi (N = 4). Questa tecnica permette di identificare e contare le occorrenze di determinate sequenze di parole o caratteri all'interno di un corpus di documenti, consentendo di individuare e rimuovere le porzioni di testo più frequenti e, presumibilmente, meno rilevanti per l'analisi.

L'approccio basato sugli N-grammi si presenta quindi come una soluzione promettente per affrontare la sfida della rimozione delle parti di testo superflue nei ticket di assistenza, preservando le informazioni utili e migliorando l'efficacia delle successive fasi di analisi del testo. Questo processo si articola in diverse fasi consecutive:

1. **Generazione degli N-grammi:** Il corpus di testo viene suddiviso in N-grammi, producendo un dizionario contenente tutti gli N-grammi presenti al suo interno e permettendo di analizzarne distribuzione e frequenza.
2. **Controllo della frequenza degli N-grammi:** Ogni N-gramma generato viene analizzato per determinare la frequenza di occorrenza all'interno del corpus, contando quante volte ogni specifico N-gramma appare nel testo.

3. **Confronto con la soglia minima (min_cut):** La frequenza di ciascun N-gramma viene confrontata con una soglia minima predefinita (min_cut). Gli N-grammi vengono suddivisi in due liste distinte:
    - Se la frequenza di un N-gramma è inferiore alla soglia, esso viene aggiunto alla lista "ngrams_to_ignore", considerato non rilevante e non ulteriormente processato.
    - Se la frequenza di un N-gramma è superiore o uguale alla soglia, esso viene inserito nella lista "ngrams_to_delete", considerato rumore e da rimuovere dal testo.
4. **Unione degli N-grammi da eliminare:** Gli N-grammi presenti nella lista "ngrams_to_delete" vengono uniti per formare una lista complessiva di frasi da eliminare, consentendo di rimuoverli in modo efficiente dal corpus di testo.
5. **Eliminazione delle frasi:** Le frasi identificate nella lista "sentence_to_delete" vengono rimosse dal corpus di testo originale, pulendo il testo e migliorandone la qualità per le successive analisi.

Attraverso questo processo è possibile identificare e rimuovere automaticamente le porzioni di testo più frequenti e meno informative, come le firme in calce, le sigle e altre informazioni superflue, preservando al

contempo le informazioni rilevanti per l'analisi dei ticket di assistenza. Ora procederemo con la presentazione dettagliata del codice.

2.1.3 Implementazione della Logica degli N-grams

Nel primo passo del processo di implementazione del codice per l'estrazione degli N-grams dai documenti, i dati di input sono memorizzati in un dataframe che contiene i testi dei ticket con il valore dell'attributo TYPESTART uguale a "EMAIL".

Si inizia definendo un dizionario vuoto, all_ngrams{}, il quale serve a tenere traccia di tutti gli N-grams estratti e della loro frequenza all'interno del dataframe. Il codice viene applicato iterativamente su un insieme di dataframes, dove ogni dataframe contiene i ticket inviati da un operatore specifico. Questa scelta si basa sull'ipotesi che la firma di un operatore sia costante in tutti i ticket inviati dallo stesso operatore. Di conseguenza, se un N-gramma appare nel 90-95% dei record di un dataframe, è molto probabile che si tratti di una firma o di un'altra sigla frequentemente utilizzata.

A livello implementativo, per ogni riga del dataframe relativo a un determinato operatore, si verifica che il valore dell'attributo LDTEXT, ovvero il testo dell'e-mail, sia una stringa. Successivamente, si divide l'intero testo del record in gruppi di N elementi, scegliendo nel nostro caso il valore quattro per N. Questo parametro è stato scelto per evitare di eliminare gruppi

di parole che potrebbero essere significativi; ad esempio, un gruppo come "cassa 8 bloccata" potrebbe essere ripetuto spesso e registrare un'alta frequenza, ma contiene informazioni rilevanti e non dovrebbe essere rimosso.

Per ogni 4-gramma estratto, se è già presente nel dizionario all_ngrams{}, si incrementa il valore associato di una unità. Altrimenti, si aggiunge il nuovo 4-gramma al dizionario con un valore iniziale pari a uno. Questo processo permette di costruire un inventario completo delle sequenze di parole più frequenti nei ticket degli operatori, che sarà poi utilizzato per ulteriori analisi e operazioni di pulizia del testo.

2.1.4 Unione degli N-grams

Il secondo step del processo prevede l'utilizzo del dizionario all_ngrams{} che contiene coppie chiave-valore dove la chiave rappresenta il 4-gramma, mentre il valore indica la frequenza di quel 4-gramma nel dataframe dei ticket inviati dall'operatore specifico tramite EMAIL. Questo dizionario include tutti i 4-grammi estratti dai testi dei ticket.

Per ogni N-gramma, se la sua percentuale di presenza nel dataframe dell'operatore supera una determinata soglia, 0.60 nel nostro caso, i quattro elementi che compongono l'N-gramma vengono aggiunti a una lista come stringhe separati da spazi. L'obiettivo è trasformare la struttura degli N-

grams in una lista di sequenze di parole, dove ogni sequenza è composta dai quattro elementi dell'N-gramma uniti da uno spazio.

Dato che la firma di un operatore aziendale può essere una sequenza di stringhe piuttosto lunga, è utile combinare le stringhe più frequenti. A questo scopo, si definisce la funzione unione_stringhe2(), che prende in input la lista n_grams_to_delete[] e combina le sequenze di testo basandosi su una determinata corrispondenza: gli ultimi tre elementi di una stringa devono corrispondere ai primi tre elementi della stringa successiva. La funzione utilizza un approccio iterativo e opera come segue:

1. Estrae il primo elemento dalla lista, ovvero la prima sequenza di quattro elementi (primo_Ngramma), poi estrae il secondo elemento (secondo_Ngramma).
2. Confronta gli ultimi tre elementi del primo con i primi tre elementi del secondo, cercando una corrispondenza.
3. Quando trova la corrispondenza, combina le due sequenze di elementi. Aggiunge la nuova sequenza di testo ottenuta alla lista di output risultato.
4. Dopo aver confrontato tutte le sequenze nella lista di input, la funzione pulisce la lista di output iterando sulla lista e verificando se ciascun elemento è una sottostringa di un altro elemento; in tal caso lo esclude dalla lista, eliminando così le sottostringhe duplicate.

2.1.5 Eliminazione degli N-grams

Il terzo e ultimo step del processo consiste nell'eliminazione delle sequenze di parole più frequenti, che sono state ottenute dalla fusione degli N-grams. Per comprendere meglio, si considera un esempio di output della lista n_grams_to_delete[]:

```
['roberta celani digital marketing direzione marketing magazzini gabrielli s p a contrada monticelli 63100 ascoli piceno tel 0736 4061 fax 0736 406308 logo la', 'soggetta a direzione e coordinamento di f g holding s p a societa soggetta a direzione', 'la societa adotta il codice']
```

Figura 2.3: Esempio output della lista n_grams_to_delete

Come si può osservare, gli elementi della lista sono lunghe sequenze di stringhe simili a una classica firma apposta da un operatore aziendale.

Il passo successivo è l'eliminazione delle "frasi" costituite dalla fusione degli N-grams più frequenti. Il codice itera per ogni elemento della lista n_grams_to_delete[], individua la posizione dell'elemento all'interno del testo associato al record nel dataframe e lo sostituisce con uno spazio vuoto.

Questi tre step del processo di pulizia del testo vengono applicati in maniera iterativa a ogni dataframe contenente i ticket associati a un operatore. Un ciclo itera sulla lista dei nomi degli operatori e, per ogni nome nella lista, definisce il dataframe e applica i codici sopra mostrati.

2.2 Generazione di Sentence Embedding

La seconda fase del progetto si concentra sull'adozione di una tecnica efficiente per il clustering dei ticket aziendali con l'obiettivo di trovare dei gruppi di ticket che riguardano problematiche simili che chiameremo macro-ticket. L'idea è che trovando insiemi di ticket che trattano lo stesso problema e che vengono aperti in rapida successione, si abbia indicazione di come la risoluzione dei singoli ticket non abbia risolto il problema alla radice. La tecnica scelta è il Sentence Embedding che consente di rappresentare semanticamente una frase o una sequenza di parole in uno spazio vettoriale. Questo processo genera un "embedding", cioè un vettore numerico associato alla frase, che permette di misurare la similarità semantica tra frasi confrontando i rispettivi vettori.

Per ottenere questa conversione delle sequenze di testo da stringhe a vettori, viene utilizzato l'Universal Sentence Encoder (USE), un modello di deep learning sviluppato da Google (Cer et al. 2018). L'USE è pre-addestrato per numerosi compiti di NLP e produce rappresentazioni vettoriali dense che

permettono di calcolare la similarità semantica tra frasi e il raggruppamento di documenti. Il modello decide automaticamente il numero di features, rappresentando tutte le frasi con lo stesso numero di elementi per consentire la rappresentazione in uno spazio di dimensione N.

Le caratteristiche principali dell'USE sono:

- **Encoding Universale:** essendo addestrato su una vasta quantità di dati testuali in lingua naturale, è capace di gestire una grande varietà di frasi e contesti.
- **Embedding Semantico:** produce rappresentazioni vettoriali delle frasi dense e ciò fa in modo che frasi semanticamente simili abbiano vettori simili. La similarità tra frasi può essere calcolata usando misure come la similarità del coseno tra i vettori.
- **Pronto all'uso:** Distribuito tramite TensorFlow Hub, l'USE può essere facilmente caricato e utilizzato direttamente senza la necessità di addestrare un modello NLP da zero.

Per implementare il modello si carica il modulo TensorFlow Hub (TF-Hub) pre-addestrato. Dopo averlo caricato, abbiamo costruito un dizionario sentence_embeddings{} dove ogni record è composto da una coppia chiave-valore in cui la chiave è il codice identificativo del ticket (TICKETID) e il valore è il vettore numerico associato al testo del ticket, ottenuto dall'applicazione del modello. Questo approccio permette di rappresentare

semanticamente i testi dei ticket, facilitando il successivo compito di clustering basato sulla similarità semantica delle frasi.

2.3 Estrazione dei Macro-Ticket

Nella fase precedente è stato generato il dizionario sentence_embeddings{} contenente le rappresentazioni vettoriali dei testi dei ticket nel dataset.

Per segmentare i ticket si utilizza una misura della similarità tra vettori, scegliendo la similarità del coseno per la sua capacità di catturare la similarità relativa tra vettori senza essere influenzata dalla loro lunghezza. La similarità del coseno misura l'angolo tra due vettori nello spazio vettoriale e restituisce un valore compreso tra -1 e 1:

- 1 indica vettori perfettamente simili
- 0 indica vettori ortogonali
- -1 indica vettori diametralmente opposti

La formula per calcolare la similarità del coseno tra due vettori A e B è:

$$CosineSimilarity(A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|}$$

Dove:

- A·B rappresenta il prodotto scalare tra i vettori A e B

- ‖A‖ e ‖B‖ rappresentano le norme euclidee dei vettori A e B rispettivamente

A questo punto si può descrivere nel dettaglio il funzionamento del codice. In primis si analizzano tutti i ticket aperti da un determinato operatore, identificando i ticket simili basati su condizioni di similarità del coseno predefinite e raggruppandoli in macro-ticket. Il processo è iterativo e include i seguenti passaggi:

1. **Definizione del DataFrame:** Filtra il dataset iniziale per includere solo i ticket associati a ciascun operatore.
2. **Aggiunta dei Sentence Embedding:** Aggiunge una colonna "EMBEDDING" al DataFrame, memorizzando i vettori numerici associati ai testi dei ticket.
3. **Definizione dei Parametri:**
    - MAX_DAYS: Numero massimo di giorni che definisce la finestra temporale entro la quale i ticket sono considerati per il raggruppamento.
    - MIN_SIM: Soglia minima di similarità del coseno per determinare se due ticket sono simili.

4. **Analisi e Raggruppamento:**
    - Esegue un ciclo for per selezionare il ticket di riferimento e un ciclo while per selezionare i ticket successivi.
    - Verifica se la differenza tra le date di apertura dei ticket è minore di MAX_DAYS e calcola la similarità tra gli embedding.
    - Se la similarità supera MIN_SIM, aggiunge il ticket simile a una lista di raggruppamento.
5. **Unione delle Liste:** Dopo l'iterazione unisce le liste di ticket che contengono almeno un elemento comune.
6. **Assegnazione delle Etichette:** Assegna un'etichetta unica a ciascun gruppo di ticket simili e memorizza l'informazione dell'etichetta nel DataFrame iniziale.

Questo processo viene ripetuto per ogni operatore garantendo un'identificazione efficace dei macro-ticket, ovvero i nostri gruppi di problemi ricorrenti e non risolti. Per raggiungere gli obiettivi dello studio, sono stati effettuati numerosi test variando i parametri MAX_DAYS e MIN_SIM. Nello specifico MAX_DAYS è stato configurato per 5, 10 e 15 giorni, mentre per MIN_SIM sono stati scelti i valori 0.90, 0.95, 0.97, 0.98 e 0.99. È fondamentale sottolineare che gli ultimi due valori di MIN_SIM richiedono un grado di somiglianza estremamente elevato tra i testi, al punto

che devono contenere quasi esattamente le stesse parole. Tuttavia, questa scelta è stata fatta per assicurare una analisi completa.

Nella sezione successiva verranno presentati i risultati ottenuti dall'implementazione del codice di clustering, esaminando diverse combinazioni dei valori dei due parametri.

2.4 Risultati

Le analisi sono suddivise in tre sezioni, ciascuna dedicata a uno dei tre valori del parametro che determina l'intervallo temporale per il confronto tra i testi dei ticket, ovvero 5, 10 e 15 giorni.

2.4.1 Finestra temporale di 5 giorni

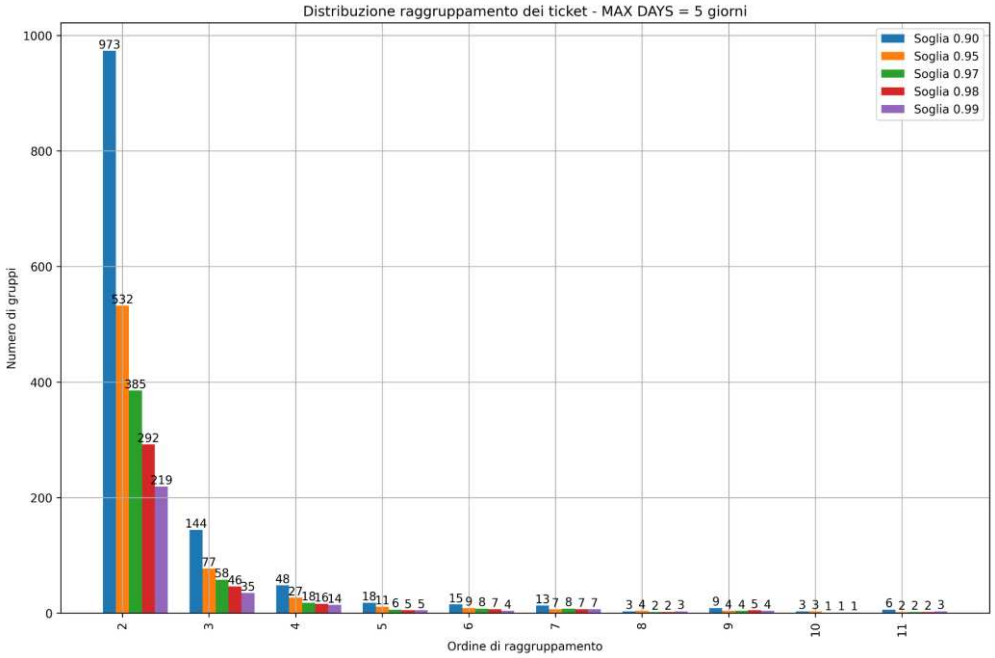

Figura 2.4: Istogramma - Macro-ticket in finestra temporale 5 giorni

L’istogramma presentato sopra illustra il numero di ticket considerati semanticamente simili, determinati attraverso la similitudine dei vettori di sentence embedding associati ai ticket stessi, per la finestra temporale di cinque giorni. L'istogramma mostra due tipi di confronto relativi ai "macro-ticket" estratti: il primo riguarda la variazione dei risultati in funzione del parametro della soglia di similarità, il secondo evidenzia la quantità di ticket inclusi in ogni gruppo.

L'istogramma evidenzia che, con una soglia di similarità fissata a 0.90, indicata con una barra blu, si ottengono 973 gruppi contenenti ciascuno due ticket, 144 gruppi con tre ticket, 48 gruppi con quattro ticket, e così via. È evidente che, mantenendo costante il valore della soglia di similarità, un aumento del numero di ticket per gruppo comporta una diminuzione del numero di macro-ticket di quella dimensione.

Per approfondire la distribuzione dei gruppi di ticket (macro-ticket) in base alla soglia di similarità, vengono utilizzate due misure statistiche:

- Media: calcolata per il numero di gruppi formati per ciascun valore di soglia di similarità. Questa misura offre una indicazione del numero medio di ticket nei gruppi e permette di confrontare l'impatto della soglia di similarità sulla composizione dei gruppi formati.
- Deviazione standard: accompagna la media per comprendere la distribuzione e la variabilità dei dati. Una deviazione standard elevata

indica una maggiore dispersione dei dati rispetto alla media, mentre una più bassa indica una dispersione minore. Questo aiuta a valutare quanto la media sia rappresentativa dei dati.

Considerando solo i gruppi con più di un ticket, i risultati di queste misure sono riportati nella seguente tabella, fornendo una panoramica quantitativa della distribuzione dei macro-ticket in relazione alla soglia di similarità.

Misura	**Soglia 0.90**	**Soglia 0.95**	**Soglia 0.97**	**Soglia 0.98**	**Soglia 0.99**
Media	2.6990	2.7747	2.8586	2.9924	3.1705
Deviazione Standard	3.0834	3.0121	3.3776	3.0682	3.3667
Minimo	2	2	2	2	2
Massimo	77	58	58	42	42

Tabella 2.1: Analisi del numero di ticket nei macro-ticket

Analizzando i dati emerge che indipendentemente dalla soglia di similarità utilizzata, esiste sempre almeno un gruppo composto da due ticket, rappresentando così il numero minimo di elementi per gruppo. Al contrario, il numero massimo di elementi per gruppo varia in funzione della soglia: si passa da 77 elementi con una soglia di 0.90 a 42 elementi con una soglia di

0.99. Ciò implica che un aumento della soglia di similarità porta a condizioni di raggruppamento più rigide, riducendo il numero di elementi nei gruppi.

Per quanto riguarda la media del numero di ticket per macro-ticket, i valori variano tra 2.6990 e 3.1705. L'incremento della media per i macro-ticket formati con soglie di similarità più alte si spiega con la significativa riduzione del numero di gruppi da due ticket. Ad esempio, con una soglia di 0.90 si formano 973 gruppi da due ticket, mentre con una soglia di 0.99 tale numero scende a 219. Al contempo però la variabilità del numero di ticket (deviazione standard) rimane elevata ma non mostra una tendenza chiara come quella della media.

Un'altra analisi svolta riguarda il calcolo dei tempi di chiusura dei macro-ticket. Considerando come giorno iniziale la data di apertura del primo ticket all'interno del macro-ticket e come giorno finale la data dell'ultimo, per le cinque soglie si ottengono i seguenti risultati:

Misura	**Soglia 0.90**	**Soglia 0.95**	**Soglia 0.97**	**Soglia 0.98**	**Soglia 0.99**
Media	1.3924	1.2965	1.3864	1.5685	1.7606
Deviazione Standard	2.5134	2.5184	2.6884	2.9939	3.2794
Minimo	0	0	0	0	0

25%	0	0	0	0	0
50%	0	0	0	0	0
75%	2	2	2	2	2
Massimo	35	27	23	23	23

Tabella 2.2: Analisi dei tempi di risoluzione dei macro-ticket a 5gg

La media del tempo trascorso tra i ticket all'interno dei gruppi, espressa in giorni, aumenta con la soglia di similarità. La deviazione standard, anch'essa espressa in giorni, misura la dispersione dei tempi tra i ticket nei gruppi e aumenta con soglie di similarità più alte. I valori minimi, al 25%, ed al 50% rimangono costanti a 0 giorni per tutte le soglie di similarità, indicando che in molti casi i ticket nei gruppi sono registrati nello stesso giorno. Tuttavia, il valore massimo diminuisce con soglie più alte, riflettendo che i gruppi con ticket distanziati nel tempo diventano meno comuni. Si nota infine come nel 75% dei casi i macro-ticket non superino i due giorni di durata.

Nonostante l'analisi delle misure statistiche, risulta arduo derivare una regola matematica precisa per determinare il valore ottimale della soglia di similarità. I numeri calcolati evidenziano semplicemente che all'aumentare

del valore della metrica il filtraggio dei testi dei ticket diventa più restrittivo, causando una riduzione nella dimensione dei gruppi. Questa contrazione è più evidente per i gruppi composti da due elementi, dove le differenze tra le diverse situazioni sono più marcate. Pertanto, per definire quale sia il valore più appropriato del parametro MIN_SIM, è necessario esaminare direttamente i testi dei macro-ticket estratti. Di conseguenza, nella sezione successiva, verrà effettuato un confronto dei macro-ticket ottenuti utilizzando una finestra temporale scorrevole di 5 giorni con valori di minima somiglianza pari a 0.90, 0.95 e 0.99, al fine di valutare l'impatto sulla qualità dei risultati. In particolare, verranno presi in esame i ticket afferenti ai topic “first level” estratti nel primo capitolo come HW-CASSA e SW-VISUALSTORE.

2.4.1.1 Analisi Topic HW-CASSA e SW-VISUALSTORE

A titolo esemplificativo si è scelto di partire con il macro-ticket FIL.P36_346 che, nel caso della soglia di similarità fissata a 0.90, è formato dai seguenti cinque ticket inoltrati tutti il 16/11/2023 a distanza di qualche ora:

Etichetta	Ticket ID	Topic	Testo
	102354	HW-POSTAZIONE CASSA	buongiorno le stampanti delle casse n 2 3 8 6 stampano male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
	102374	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 2 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536

FIL.P36_346	102375	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 3 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
	102376	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 8 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
	102377	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 6 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536

Tabella 2.3: FIL.P36 - Testi Macro-Ticket MAX_DAYS=5 e MIN_SIM=0.90

Si può notare come il contenuto sia effettivamente molto simile e come a cambiare sia solo il numero della cassa che presenta problemi. Risulta pertanto sensato che questi ticket siano stati etichettati sotto la stessa etichetta.

Innalzando la soglia di similarità a 0.95, si nota una perdita nella capacità di generalizzazione, con i cinque ticket che appaiono semanticamente quasi identici suddivisi in due gruppi distinti.

Etichetta	Ticket ID	Topic	Testo
FIL.P36_348	102354	HW-POSTAZIONE CASSA	buongiorno le stampanti delle casse n 2 3 8 6 stampano male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
FIL.P36_349	102374	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 2 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
	102375	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 3 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
	102376	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 8 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
	102377	HW-POSTAZIONE	buongiorno la stampante della cassa n 6 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe

		CASSA	verticali sono sbiaditi grazie panda536

Tabella 2.4: FIL.P36 - Testi Macro-Ticket MAX_DAYS=5 e MIN_SIM=0.95

Con una soglia di 0.99, infine, ogni ticket viene etichettato individualmente.

Etichetta	Ticket ID	Topic	Testo
FIL.P36_354	102354	HW-POSTAZIONE CASSA	buongiorno le stampanti delle casse n 2 3 8 6 stampano male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
FIL.P36_355	102374	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 2 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
FIL.P36_356	102375	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 3 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
FIL.P36_357	102376	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 8 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536
FIL.P36_358	102377	HW-POSTAZIONE CASSA	buongiorno la stampante della cassa n 6 stampa male gli scontrini le scritte e i codici finali sullo scontrino non si leggono bene soprattutto su alcune righe verticali sono sbiaditi grazie panda536

Tabella 2.5: FIL.P36 - Testi Macro-Ticket MAX_DAYS=5 e MIN_SIM=0.99

In conclusione, per il contesto analizzato, una soglia di similarità pari a 0.90 rappresenta il valore più adatto per la formazione dei macro-ticket. Questo permette di aggregare in modo efficiente i ticket simili, migliorando la gestione dei problemi e ottimizzando i tempi di risoluzione. Tale valore consente di mantenere un equilibrio tra la necessità di generalizzare e quella di dettagliare i singoli casi, risultando nella soluzione più pratica per l'analisi e la risoluzione dei ticket di assistenza.

2.4.2 Finestra temporale di 10 giorni

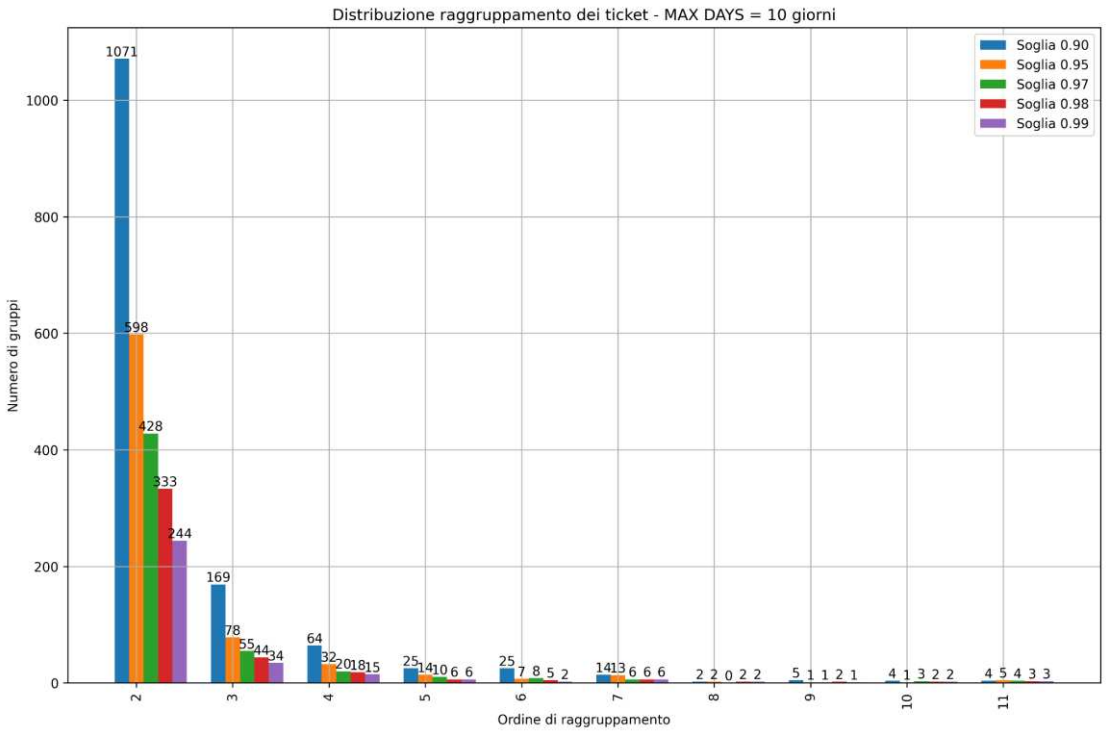

Figura 2.5: Istogramma - Macro-ticket in finestra temporale 10 giorni

Il grafico evidenzia che, con una soglia di similarità fissata a 0.90, per quanto riguarda la finestra temporale di 10 giorni, si ottengono 1071 gruppi contenenti ciascuno due ticket, 169 gruppi con tre ticket, 64 gruppi con quattro ticket, e così via discorrendo. Mantenendo costante anche qui il valore della soglia di similarità, un aumento del numero di ticket per gruppo comporta una diminuzione del numero di macro-ticket di quella dimensione. L'andamento delle misure è simile a quello osservato per la finestra di cinque giorni. La media del numero di ticket per macro-ticket aumenta

progressivamente al crescere della soglia di similarità, variando tra 2.8285 con soglia 0.90 e 3.2948 con soglia 0.99. Allo stesso tempo, la deviazione standard rimane elevata ma non mostra una tendenza chiara.

Il numero minimo di ticket per macro-ticket rimane due indipendentemente dalla soglia. Il numero massimo invece diminuisce da 84 a 52 ticket al crescere della soglia, indicando che soglie più restrittive portano a macro-ticket di dimensioni inferiori.

Misura	**Soglia 0.90**	**Soglia 0.95**	**Soglia 0.97**	**Soglia 0.98**	**Soglia 0.99**
Media	2,8285	2,8605	2,9672	3,0759	3,2948
Deviazione Standard	3,7672	3,6745	4,1749	3,9777	4,4442
Minimo	2	2	2	2	2
Massimo	84	59	59	52	52

Tabella 2.6: Analisi del numero di ticket nei macro-ticket a 10gg

La media dei tempi di risoluzione oscilla tra 2.8 e 3.4 giorni circa, con un valore leggermente superiore alle soglie più alte. La deviazione standard è elevata, indicando un'ampia variabilità nei tempi di risoluzione.

Per tutte le soglie, il 25% dei macro-ticket viene risolto lo stesso giorno, mentre la mediana è di un giorno. Il 75% dei casi rientra in una finestra di

quattro o cinque giorni. Il tempo massimo registrato di 132 giorni non varia al variare della soglia di similarità.

In sintesi, soglie di similarità più restrittive non sembrano influenzare significativamente i tempi medi di risoluzione, che rimangono nell'ordine di pochi giorni per la maggior parte dei casi.

Misura	**Soglia 0.90**	**Soglia 0.95**	**Soglia 0.97**	**Soglia 0.98**	**Soglia 0.99**
Media	3,2270	2,8644	2,9361	3,1908	3,4346
Deviazione Standard	5,9961	6,3566	7,0158	7,6587	8,6053
Minimo	0	0	0	0	0
25%	0	0	0	0	0
50%	1	1	1	1	1
75%	5	4	4	5	5
Massimo	132	132	132	132	132

Tabella 2.7: Analisi dei tempi di risoluzione dei macro-ticket a 10gg

2.4.3 Finestra temporale di 15 giorni

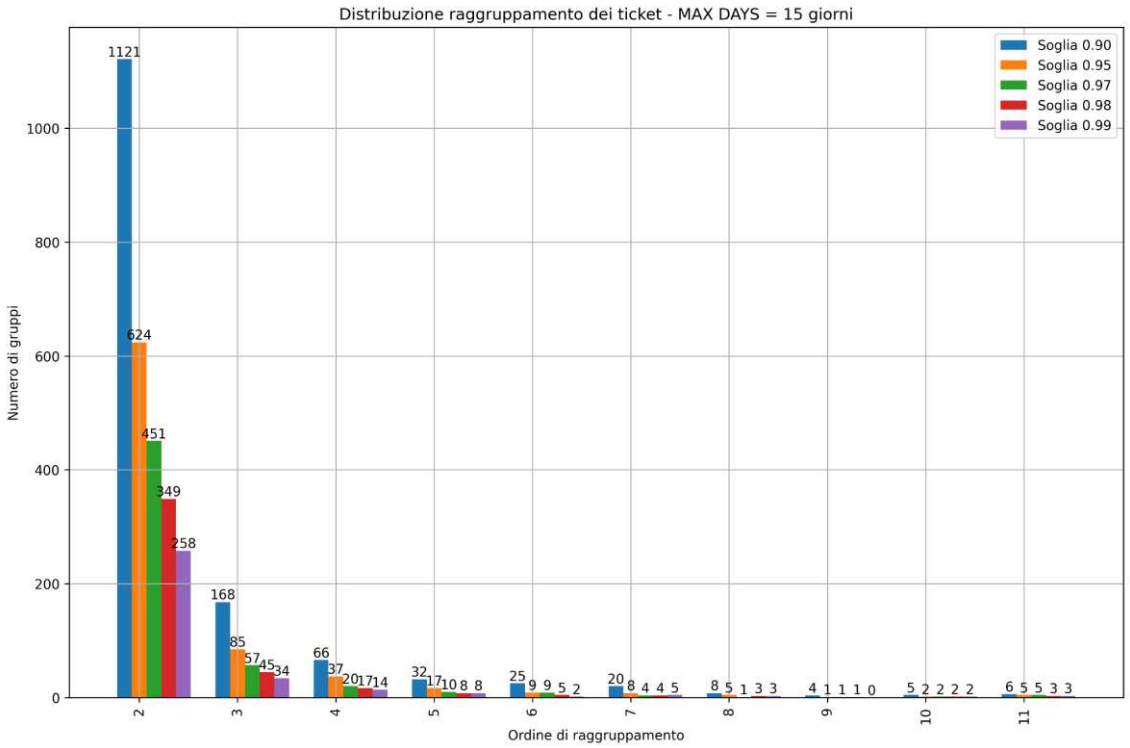

Figura 2.6: Istogramma - Macro-ticket in finestra temporale 15 giorni

Anche l'istogramma per la finestra temporale di 15 giorni evidenzia il numero di ticket considerati semanticamente simili determinati attraverso la similitudine dei vettori di sentence embedding. Come nelle finestre temporali di 5 e 10 giorni, esso mostra la variazione dei risultati in funzione del parametro della soglia di similarità e la quantità di ticket inclusi in ogni gruppo.

Con una soglia di similarità fissata a 0.90, si ottengono 1121 gruppi contenenti ciascuno due ticket, 168 gruppi con tre ticket, 66 gruppi con quattro ticket, e via discorrendo. Anche qui, mantenendo costante il valore

della soglia di similarità, un aumento del numero di ticket per gruppo comporta una diminuzione del numero di macro-ticket di quella dimensione.

Misura	**Soglia 0.90**	**Soglia 0.95**	**Soglia 0.97**	**Soglia 0.98**	**Soglia 0.99**
Media	2.9134	2.8999	2.9878	3.1042	3.3149
Deviazione Standard	4.0720	3.9551	4.5339	4.4954	4.7501
Minimo	2	2	2	2	2
Massimo	84	59	59	52	52

Tabella 2.8: Analisi del numero di ticket nei macro-ticket a 15gg

Analizzando i dati della tabella, si osserva che il numero medio di ticket per macro-ticket aumenta leggermente al crescere della soglia di similarità, variando tra 2.9134 con soglia 0.90 e 3.3149 con soglia 0.99. La deviazione standard, che misura la dispersione nel numero di ticket per gruppo, varia da 4.0720 a 4.7501, indicando una variabilità relativamente contenuta ma presente. Il numero minimo di ticket per macro-ticket rimane costante a 2 indipendentemente dalla soglia, mentre il numero massimo diminuisce da 84 a 52 ticket al crescere della soglia, suggerendo che soglie più restrittive portano a macro-ticket di dimensioni inferiori.

La media dei tempi di risoluzione varia tra 4.0802 e 4.9113 giorni, con un incremento leggero alle soglie più alte. La deviazione standard, che misura la dispersione dei tempi di risoluzione, aumenta con soglie di similarità più alte, indicando una maggiore variabilità nei tempi di chiusura dei macro-ticket e risulta essere la più elevata tra le nostre finestre temporali. I valori minimi, così come i percentili al 25%, rimangono costanti a 0 giorni, riflettendo che anche qui in molti casi i ticket nei gruppi sono registrati e risolti nello stesso giorno, mentre il valore della mediana è di 1 giorno. Il 75% dei casi rientra in una finestra di sei o sette giorni. Il valore massimo di 132 giorni rimane invariato per tutte le soglie, suggerendo che alcuni gruppi contengono ticket molto distanziati nel tempo.

Misura	**Soglia 0.90**	**Soglia 0.95**	**Soglia 0.97**	**Soglia 0.98**	**Soglia 0.99**
Media	4.9113	4.2398	4.0802	4.3037	4.6967
Deviazione Standard	8.5752	8.5335	9.2201	9.6656	10.7866
Minimo	0	0	0	0	0
25%	0	0	0	0	0
50%	1	1	1	1	1

75%	7	6	5	6	6
Massimo	132	132	132	132	132

Tabella 2.9: Analisi dei tempi di risoluzione dei macro-ticket a 15gg

In conclusione, l'analisi delle finestre temporali di 5, 10 e 15 giorni ha rivelato che la finestra di 5 giorni offre i valori più uniformi e meno temporalmente dispersi, rendendola la scelta ottimale per l'aggregazione dei ticket simili. La media del tempo di risoluzione per la finestra di 5 giorni varia tra 1.39 e 1.76 giorni, con una deviazione standard compresa tra 2.51 e 3.27 giorni. Questi valori indicano che la dispersione dei tempi di risoluzione è contenuta, assicurando una gestione più coerente e prevedibile dei ticket.

A confronto, nella finestra di 10 giorni, la media dei tempi di risoluzione oscilla tra 2.86 e 3.43 giorni, con una deviazione standard significativamente più alta, tra 5.99 e 8.60 giorni. Anche la finestra di 15 giorni mostra una maggiore dispersione, con una deviazione standard che varia tra 8.75 e 10.78 giorni. Inoltre, l'analisi della distribuzione dei macro-ticket per la finestra di cinque giorni mostra che, mantenendo costante il valore della soglia di similarità, un aumento del numero di ticket per gruppo comporta una diminuzione del numero di macro-ticket di quella dimensione. Ad esempio,

con una soglia di similarità di 0.90, si ottengono 973 gruppi di due ticket e solo 48 gruppi di quattro ticket. Questa tendenza è meno pronunciata nelle finestre di 10 e 15 giorni, dove la distribuzione dei ticket è meno uniforme e mostra una maggiore variabilità.

In sintesi, la finestra temporale di cinque giorni permette di aggregare in modo più efficace i ticket simili, migliorando la gestione dei problemi e ottimizzando i tempi di risoluzione. La stabilità e la bassa variabilità dei dati supportano la scelta di questa finestra come la più efficiente per l'analisi della risoluzione dei ticket di assistenza.

Nel seguente capitolo, continueremo ad analizzare il testo dei ticket aziendali utilizzando un Large Language Model (LLM) open source. L'obiettivo è determinare se questo genere di modelli possano offrire un metodo ancora più efficiente e preciso per raggruppare i ticket dell'azienda. L'utilizzo di LLM promette di migliorare l'accuratezza nell'identificazione delle somiglianze semantiche tra i ticket, permettendo una gestione ancora più ottimizzata e tempestiva dei problemi segnalati. Questa fase successiva dell'analisi rappresenta un passo importante verso l'adozione di tecnologie avanzate volte al continuo miglioramento dei processi aziendali.

Capitolo 3

In questo capitolo utilizzeremo dei Large Language Models (LLM) per analizzare nel dettaglio il contenuto dei ticket di assistenza dell'azienda. Prima di addentrarci nel processo di analisi, tuttavia, è utile introdurre cosa siano e come funzionino i modelli linguistici di grandi dimensioni, al fine di una maggiore completezza.

3.1 Introduzione ai Modelli Linguistici di Grandi Dimensioni

3.1.1 Definizione di LLM

I modelli linguistici di grandi dimensioni (LLM) sono algoritmi di intelligenza artificiale avanzati che utilizzano reti neurali profonde per comprendere e generare linguaggio naturale. Questi modelli, che sono addestrati su enormi volumi di testo, sono in grado di catturare le complessità sintattiche, semantiche e contestuali del linguaggio umano. Nel contesto dell'elaborazione del linguaggio naturale (NLP), gli LLM sono fondamentali perché consentono di automatizzare e migliorare una vasta gamma di applicazioni, tra cui la traduzione automatica, il riassunto di testi, l'analisi del sentiment e la risposta automatica alle domande. La loro capacità di generare testo coerente e contestualmente appropriato rappresenta un significativo avanzamento rispetto al passato.

3.1.2 Storia e Sviluppo

Lo sviluppo degli LLM può essere tracciato attraverso diverse fasi chiave nella storia del machine learning e dell'informatica. I primi modelli di NLP erano basati su regole ben specifiche che richiedevano una codifica manuale delle regole grammaticali. Successivamente, sono stati introdotti i modelli basati su N-grammi, che utilizzavano la probabilità di occorrenza di sequenze di parole per prevedere il testo. Una svolta è arrivata con l'introduzione delle reti neurali ricorrenti (RNN) e, in particolare, delle Long Short-Term Memory (LSTM), che hanno permesso di gestire le dipendenze a lungo termine nel testo.

Il progresso più significativo, tuttavia, è avvenuto con l'introduzione dell'architettura Transformer proposta da Vaswani et al. nel 2017. Questo approccio ha rivoluzionato l'NLP grazie all'attenzione multi-testa e alla capacità di essere parallelizzato, migliorando notevolmente l'efficienza e le prestazioni di tecnologie simili. Successivamente, modelli come BERT (Bidirectional Encoder Representations from Transformers) e GPT (Generative Pre-trained Transformer) hanno dimostrato capacità straordinarie in vari compiti di NLP, spingendo i confini di ciò che è possibile con i modelli linguistici.

3.1.3 Architettura e Principi di Funzionamento

L'architettura degli Large Language Models è principalmente basata sul modello Transformer introdotto da Vaswani et al. nel 2017. Come accennato prima, questo modello ha rivoluzionato il campo dell'elaborazione del linguaggio naturale (NLP) grazie alla sua capacità di gestire contesti lunghi e complessi in modo parallelo, rendendo il processo di addestramento più efficiente rispetto alle precedenti architetture sequenziali come le RNN (Reti Neurali Ricorrenti) e le LSTM (Long Short-Term Memory).

L'architettura del Transformer è composta da due parti principali: l'encoder e il decoder. Entrambi sono costituiti da una serie di blocchi di trasformazione identici impilati uno sopra l'altro, ciascuno composto da due sottostrati principali:

- **Meccanismo di Attenzione:** Permette al modello di pesare l'importanza delle diverse parole nel contesto di una frase, migliorando la comprensione del contesto globale. Il tipo di attenzione usato è chiamato “self-attention” (auto-attenzione), che consente di considerare tutte le parole della sequenza di input contemporaneamente.

- **Reti Neurali Feed-Forward:** Applicano una trasformazione non lineare alle rappresentazioni generate dal meccanismo di attenzione, per catturare relazioni complesse tra le parole.

Nel dettaglio i meccanismi di attenzione possono essere di due tipi:

- Self-Attention: Questo meccanismo calcola una rappresentazione per ogni parola nella sequenza di input, pesando tutte le altre parole in base alla loro rilevanza. Questo permette al modello di capire quali parole del contesto sono più importanti per la parola corrente.
- Multi-Head Attention: È una variante del self-attention che applica il meccanismo di attenzione più volte in parallelo con diversi set di pesi. Ogni "testa" di attenzione cattura diversi aspetti delle relazioni tra le parole, migliorando la capacità del modello di comprendere pattern complessi.

Per gestire input di testo, i Transformers utilizzano diverse tecniche di rappresentazione che lavorano sinergicamente con i meccanismi di attenzione:

- Word Embeddings: Ogni parola viene convertita in un vettore che ne rappresenta il significato in uno spazio vettoriale continuo.

- Subword Embeddings: Le parole vengono suddivise in unità, subword (caratteri o n-grammi di caratteri), permettendo di gestire parole fuori vocabolario e migliorare la cattura di similarità morfologiche e semantiche.
- Positional Encodings: Poiché il modello Transformer non è sequenziale, ha bisogno di un modo per incorporare la posizione delle parole. I Positional Encodings aggiungono informazioni sulla posizione di ogni parola nella sequenza, permettendo al modello di distinguere tra parole in posizioni diverse.
- Segment Embeddings: In alcuni modelli la sequenza di input può essere divisa in segmenti (es. frasi o paragrafi). I segment embeddings indicano a quale segmento appartiene ciascun token.

Gli LLM sono addestrati inizialmente usando apprendimento non supervisionato, dove il modello impara i pattern e le relazioni tra le parole in grandi corpora di testo chiamati "corpus". Successivamente, vengono perfezionati (fine-tuning) su compiti specifici con dataset più piccoli e annotati. Il fine-tuning implica solitamente il congelamento dei pesi del modello precedentemente addestrato e l'addestramento solo dei livelli specifici del compito, per ottimizzare le prestazioni su attività particolari come la traduzione automatica o la sintesi di testo.

3.2 Analisi della colonna ‘LDTEXT’ tramite LLM

Avendo delineato, seppur sommariamente, il funzionamento degli LLM, possiamo adesso passare a trattare l’utilizzo di questo genere di modelli nel nostro caso specifico. La procedura di analisi della colonna 'LDTEXT' è stata eseguita attraverso una serie di passaggi tecnici mirati a estrarre e classificare i problemi più frequenti riportati nei ticket di assistenza.

3.2.1 Identificazione degli Argomenti Principali tramite LLM

Il dataset completo è stato inizialmente filtrato e diviso per estrarre le osservazioni relative alle due categorie della variabile “first level” più frequenti viste nel Capitolo 1: HW-POSTAZIONE CASSA e SW-VISUALSTORE.

Per ciascuno di questi subset è stata estratta la colonna 'LDTEXT', che contiene le descrizioni testuali dei problemi riportati. Inizialmente l’idea era quella di far scorrere ad un LLM open source scaricato in locale, Hermes-2-Pro-Mistral-7B-GGUF, il contenuto dei ticket per ciascun subset e di fargli restituire una lista contenente gli argomenti più frequenti nelle segnalazioni. In una seconda iterazione, i topic della lista sarebbero stati usati per classificare i ticket presi singolarmente, assegnando a ciascuno la corrispondente etichetta. Questo procedimento, tuttavia, non è stato possibile seguirlo così per come lo si è appena descritto in quanto il modello Mistral-

7B presenta una “finestra di contesto” non abbastanza ampia per gestire con coerenza il testo preso per intero della colonna LDTEXT. La finestra di contesto per un modello linguistico di grandi dimensioni (LLM) si riferisce alla quantità massima di testo che il modello può considerare in qualsiasi momento quando genera una risposta. Questo include sia il prompt fornito dall'utente sia il testo generato dal modello.

Dopo aver valutato negativamente l’uso a pagamento dei modelli GPT di OpenAI tramite API, per individuare la lista dei topic più frequenti nelle segnalazioni, è stato utilizzato ChatGPT nella sua versione più recente <<4o>>. Gli è stato chiesto di identificare i principali temi relativi ai problemi riportati nei due subset tramite il seguente prompt:

“Stai analizzando dei ticket di assistenza di una azienda operante nella Grande Distribuzione Organizzata dei supermercati. Voglio che tu mi fornisca una lista con gli argomenti più frequenti analizzando il testo manualmente:”

Gli argomenti trovati per HW-POSTAZIONE CASSA includono:

- Blocchi delle casse: "cassa n1 bloccata", "cassa 4 si è spenta", "cassa 7 non si avvia il programma di vendita".
- Problemi con la stampante: "stampante della cassa 10 non funziona", "printer di cassa 1 si continua a bloccare".
- Scanner non funzionante: "pistola scanner della cassa 16 non funziona", "cassa 1 non funziona scanner di cassa".

- Problemi con il POS: "POS della cassa 8 non funziona", "POS rimane fermo su operazione in corso".
- Problemi con il touch screen: "cassa 4 sui tasti touch screen non sempre risponde", "cassa 2 touch non funziona bene".
- Problemi con il cassetto della cassa: "il cassetto della cassa del box assistenza non si apre", "cassetto portadenaro cassa 2 non si apre automaticamente".
- Problemi con la chiusura del conto e lo scontrino: "scontrino con chiusura pendente", "cassa 2 non fa uscire lo scontrino".
- Problemi con il monitor: "monitor della cassa 14 si spegne", "cassa 4 display si spegne e rimane nero".
- Errori fiscali: "errore fiscale scontrino cassa 3", "cassa 3 errore fiscale 0xffff".
- Altri problemi vari: "problema alla cassa 12 per tappeto strappato", "cassa 3 ha emesso del fumo".

Per SW-VISUALSTORE, gli argomenti principali individuati sono invece:

- Problemi di chiusura fiscale: "cassa 1 non ha effettuato la chiusura", "non riesce a chiudere la prima nota".
- Errore di input o lettura: "la cassa da errore input", "codice sacchetto non viene letto".

- Blocchi delle casse: "cassa 18 scollegata", "tutte le casse bloccate in modalità test".
- Problemi con gli sconti e le promozioni: "sconti del 10% per i dipendenti non applicati", "promo natale e giocattoli non passano alle casse".
- Problemi con il programma di vendita: "programma di vendita si è chiuso", "casse non caricano programma di vendita".
- Problemi con la schermata iniziale: "cassa bloccata su schermata Windows", "tutte le casse su schermata nera".
- Problemi con la validazione dei dati: "non riesce a validare l'ammontare e le chiusure fiscali".
- Problemi di sincronizzazione e aggiornamento: "non aggiornati gli incassi", "differenze nelle dichiarazioni finanziarie".
- Problemi con le transazioni: "transazione non letta", "trx elettronica non annullata".
- Richieste di modifiche e aggiornamenti: "modifica dell'intestazione su scontrino", "riattivare i cluster".

3.2.3 Classificazione dei Dataset con un LLM Open Source

Per la classificazione vera e propria, invece, si è potuto utilizzare la nostra prima scelta di LLM open source, Hermes-2-Pro-Mistral-7B-GGUF, per classificare i dataset in base agli argomenti identificati da ChatGPT (Jiang et al. 2023). Questa operazione è stata eseguita utilizzando LMStudio, un ambiente di lavoro per l'implementazione e la gestione dei modelli di linguaggio in locale. Il modello è quindi stato scaricato e utilizzato su un server dell'università.

Il codice con il prompt utilizzato per la classificazione del dataset HW-CASSA è il seguente:

```
def generate_summary(text):
    client = OpenAI(base_url="http://localhost:1234/v1", api_key="lm-studio")
    response = client.completions.create(
        model="model-identifier",
        prompt=(f"Stai analizzando dei ticket di assistenza di una azienda operante nella Grande Distribuzione Organizzata dei supermercati. Sintetizza il contenuto del testo in meno di 10 parole specificando l'oggetto del problema tra i seguenti argomenti: 'Blocchi delle casse', 'Problemi con la stampante', 'Scanner non funzionante', 'Problemi con il POS', 'Problemi con il touch screen', 'Problemi con il cassetto della cassa', 'Problemi con la chiusura del conto e lo scontrino', 'Problemi con il monitor', 'Errori fiscali' o 'Altro':\n{text}\n\nProblema:"),
        temperature=0.2,
        max_tokens=16,
        n=1,
        stop=None )
    summary = response.choices[0].text.strip()
    return summary
```

Figura 3.1: Codice - Snippet del codice per la classificazione tramite LLM

La funzione generate_summary() utilizza un modello di linguaggio naturale per generare un riassunto conciso del testo dei ticket, specificando l'oggetto del problema. Il parametro "**prompt"** è il testo che viene passato al modello per generare la risposta. In questo caso il prompt istruisce il modello a sintetizzare il contenuto del testo del ticket in meno di dieci parole, specificando l'oggetto del problema tra le categorie elencate. Il text del ticket viene inserito all'interno del prompt per contestualizzare la richiesta.

Il parametro "**temperature"** controlla la creatività della risposta generata. Un valore più basso, come 0.2, rende la risposta più deterministica e focalizzata, riducendo la variabilità delle risposte generate. In "max_tokens" viene definito il numero massimo di token (parole e simboli) che il modello può generare nella risposta. Il limite è fissato a 16 token per garantire che la risposta sia breve e concisa. Infine, il parametro "n" indica il numero di risposte da generare, mentre "stop" specifica una sequenza di token che, se incontrata, interrompe la generazione del testo. In questo esempio lo stop è impostato a None, quindi la generazione continua fino al raggiungimento del limite di token o alla fine della risposta.

Per la classificazione del dataset SW-VISUALSTORE è stato usato un prompt simile ma con le categorie ad esso associate.

Sulla base delle classificazioni ottenute, sono stati selezionati i principali macro-argomenti per ciascun dataset. Questo è stato fatto in funzione di

capire se con un solo livello di dettaglio si perdessero informazioni più specifiche a discapito di una classificazione più generica. Nel caso di HW-POSTAZIONE CASSA è stato scelto “Blocchi delle casse” in quanto argomento più frequente al 49,31%, mentre per SW-VISUALSTORE sono stati selezionati i primi tre macro-argomenti ovvero “Problemi con il programma di vendita”, “Problemi con la validazione dei dati” e “Problemi di chiusura fiscale”. A questo punto è stata riproposta l'operazione iniziale con ChatGPT per ottenere una lista di argomenti specifici da usare come base per una classificazione ad un livello di dettaglio più approfondito.

Il modello Hermes-2-Pro-Mistral-7B-GGUF è stato utilizzato nuovamente per assegnare le micro-categorie all'interno degli argomenti estratti, i quali così si presentano:

- **Blocchi delle casse**
    - Cassa bloccata durante la transazione
    - Cassa bloccata al login
    - Cassa bloccata su schermate specifiche
    - Cassa bloccata durante la chiusura
    - Cassa bloccata con errori di sistema
    - Problemi con il POS
    - Problemi con la pistola scanner
    - Problemi con la stampante
    - Errore di sistema
    - Problemi con il cassetto portadenaro
    - Cassa offline

- **Problemi di chiusura fiscale**
    - Differenze tra totale finanziario e totale fiscale
    - Chiusure pendenti
    - Errore di input o bloccaggio delle casse
    - Azzeramento fiscale non emesso
    - Problemi con i report di Visual Store
    - Richieste di modifica dati fiscali (come IVA e intestazioni scontrini)
    - Annullamenti e modifiche delle transazioni

- **Problemi con il programma di vendita**.
    - Errore di input durante la scansione del codice a barre
    - Sconti non applicati
    - Crash e blocchi del programma di vendita
    - Chiusure disallineate o non corrette
    - Problemi con specifici prodotti
    - Problemi con i buoni sconto
    - Problemi di lettura dei codici a barre
    - Problemi con l'allineamento dei fondi cassa
    - Problemi con la funzione di arrotondamento
    - Problemi di sincronizzazione dei dati.

- **Problemi con la validazione dei dati**
    - Problemi di chiusura fiscale
    - Blocco delle casse
    - Errori di validazione
    - Assenza di dati o incongruenze
    - Problemi con scontrini
    - Buoni e coupon non validati
    - Errore nei resi
    - Malfunzionamenti vari

I risultati dell'analisi svolta a seguito delle procedure appena descritte saranno mostrati, in dettaglio, all'interno dei prossimi paragrafi.

3.3 Analisi dei risultati Primo Argomento

3.3.1 HW-POSTAZIONE CASSA

L'analisi dei dati relativi alla categoria principale "HW-POSTAZIONE CASSA" offre una panoramica dettagliata delle problematiche riscontrate e dei tempi medi di risoluzione associati a ciascun argomento. L'argomento più frequente, come mostrato nell'immagine successiva, è quello di "Blocchi delle casse", con un totale di 4695 ticket, seguita da "Scanner non funzionante" (1437) e "Problemi con il POS" (1285).

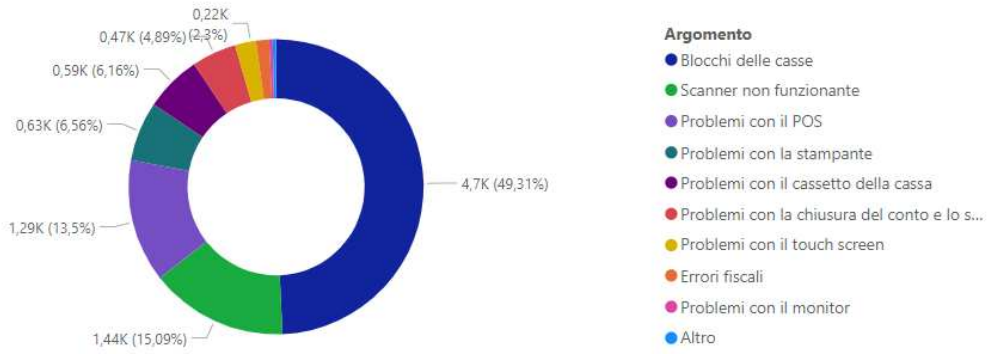

Figura 3.2: Argomenti dei ticket in HW-Postazione Cassa

Per quanto riguarda i tempi medi di risoluzione per l'intera categoria "HW-POSTAZIONE CASSA", abbiamo una media di circa 0.78 giorni (1123 minuti), con una deviazione standard di 2.42 giorni (3479 minuti). Questi dati suggeriscono che la maggior parte dei problemi hardware venga risolta relativamente rapidamente, anche se ci sono casi eccezionali che richiedono

tempi molto più lunghi. Un'analisi più dettagliata dei tempi medi di risoluzione per ciascun argomento assegnato dal Large Language Model (LLM) rivela tuttavia interessanti differenze. Ad esempio, gli argomenti con i tempi medi più bassi sono "Problemi con la chiusura del conto e lo scontrino" (0.26 giorni) e "Problemi con il POS" (0.56 giorni), mentre quelli con i tempi medi più alti sono "Problemi con il cassetto della cassa" (1.46 giorni), "Scanner non funzionante" (1.26 giorni) e "Altro" (1.03 giorni). Le deviazioni standard dei tempi di risoluzione per ciascun argomento sono generalmente elevate, indicando una variabilità significativa nei tempi di risoluzione per problemi simili. Quasi tutte le problematiche, ad eccezione di “Problemi con il cassetto della cassa”, vengono risolte nel 75% dei casi nel giro di un giorno come mostrato nel grafico.

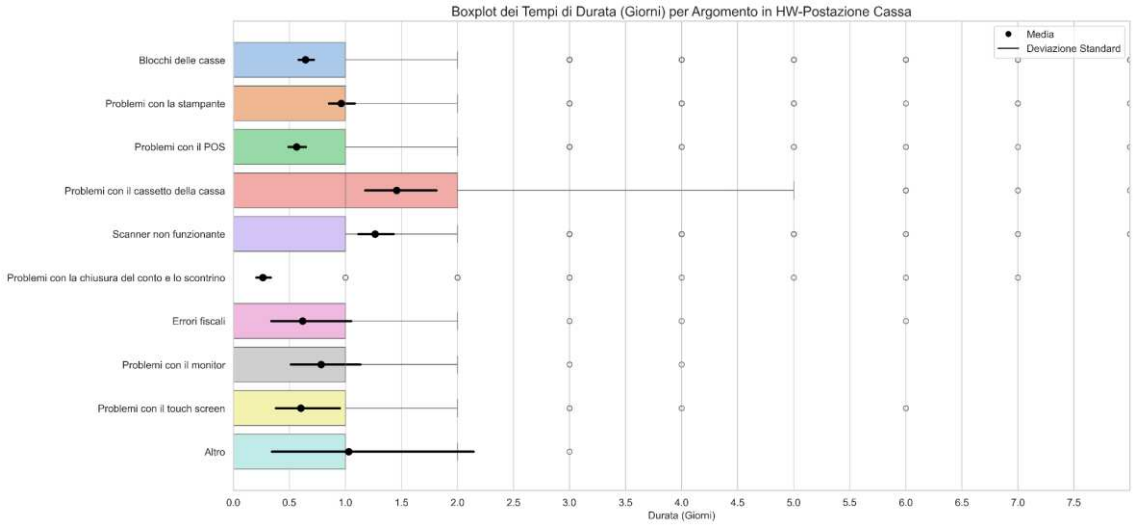

Figura 3.3: Boxplot della durata dei ticket in HW-Postazione Cassa

Un’altra analisi svolta è stata quella di osservare la concordanza tra le classificazioni dell'azienda ("second level") e quelle del Large Language Model ("Argomento") per la categoria principale "HW-POSTAZIONE CASSA", facendo emergere ulteriori punti chiave. Per fare ciò è stata creata una matrice di confusione come quella in figura.

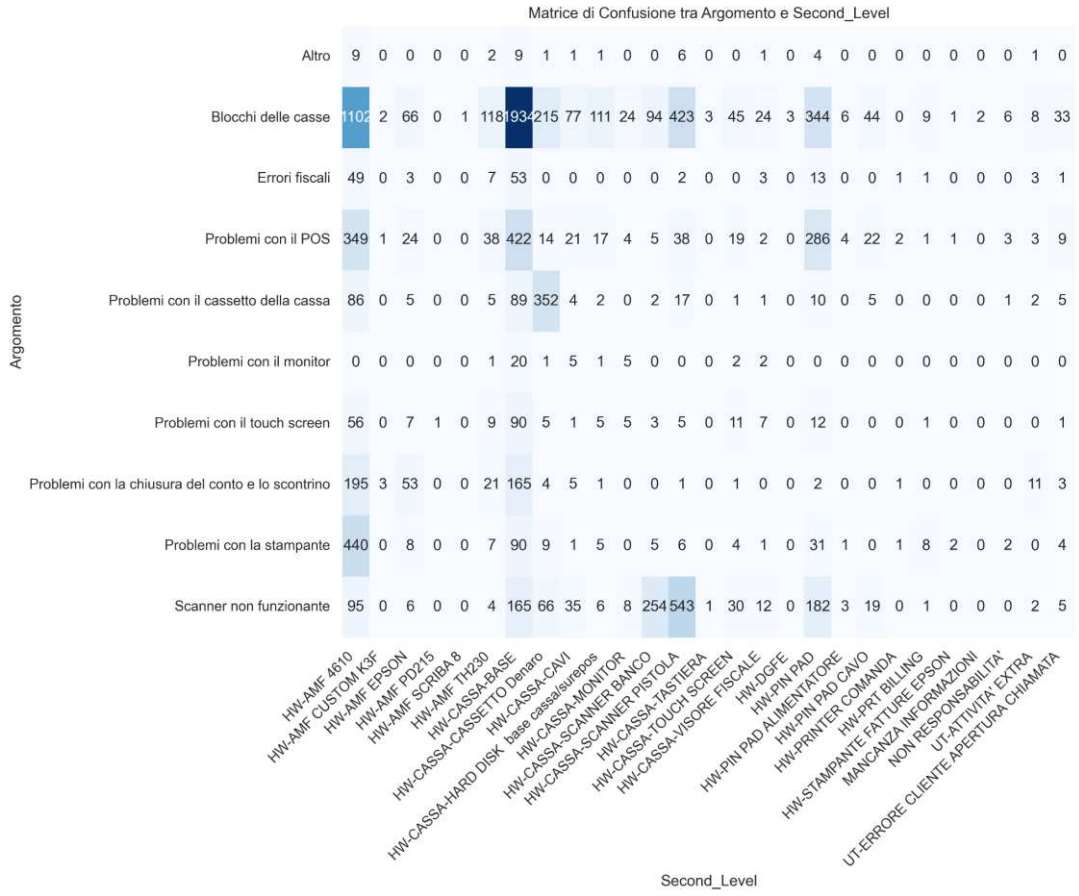

Figura 3.4: Matrice di confusione tra Argomento e Second Level

Per "Blocchi delle casse" si osserva una buona corrispondenza tra le classificazioni dell'azienda e quelle del modello LLM con 1934 ticket classificati come "HW-CASSA-BASE" e 1102 ticket classificati come

“HW-AMF 4610”, suggerendo un'identificazione corretta dei problemi generali con le casse da entrambi i sistemi.

Un altro esempio lo abbiamo per "Scanner non funzionante" e “Problemi con il POS”, per i quali il modello LLM sembrerebbe essere in grado di riconoscere correttamente il problema nella maggior parte dei casi.

Al contrario, per "Problemi con il monitor", il modello LLM ha identificato solo 20 ticket rispetto ai 46 ticket classificati dall'azienda come "HW-CASSA-MONITOR".

Per ultimo, è stato creato un grafico a nastro per visualizzare l’andamento delle problematiche avute dal 2021 ad oggi in cui si può notare come “Blocchi delle casse” sia stato l’argomento più comune ininterrottamente, con “Problemi con il POS” e “Scanner non funzionante” che si sono invece intervallati.

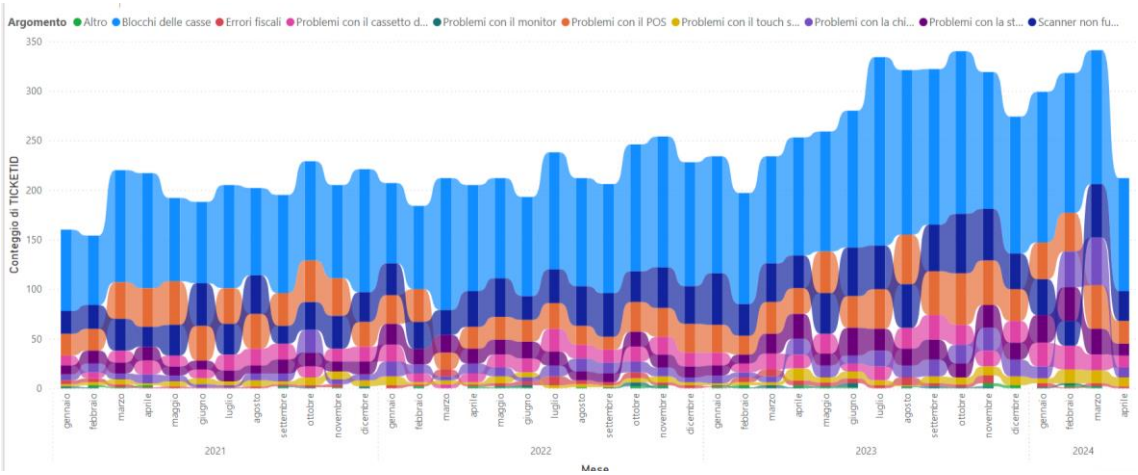

Figura 3.5: Andamento dei numeri di ticket per Argomento

3.3.2 SW-VISUALSTORE

Passando all'analisi dei risultati relativi al first level "SW-VISUALSTORE", possiamo notare come tra gli argomenti più frequenti vi sia "Problemi con il programma di vendita" che rappresenta il 26,24% dei ticket, per un totale di 2809, seguito da "Problemi con la validazione dei dati" (2514) e "Problemi di chiusura fiscale" (1409).

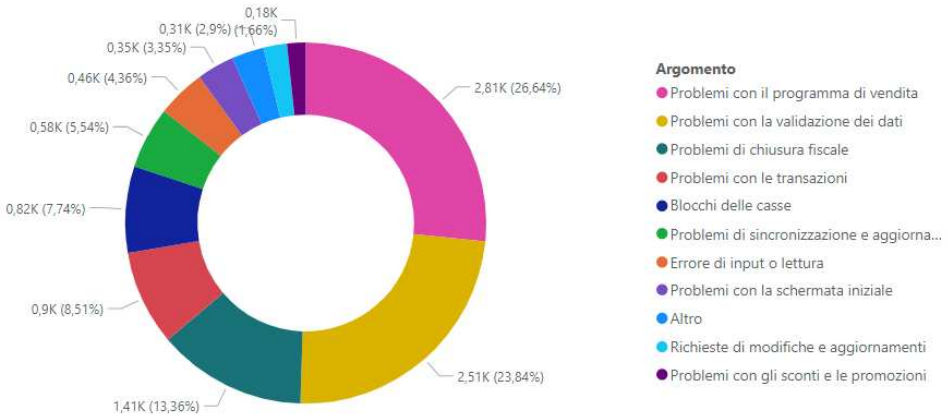

Figura 3.6: Argomenti dei ticket in SW-Visualstore

Il tempo medio di risoluzione per il first level "SW-VISUALSTORE" è di circa 3.15 giorni, equivalente a 4538 minuti, con una deviazione standard di 17.09 giorni (24610 minuti). Questi dati non solo evidenziano una significativa variabilità nei tempi di risoluzione, con alcuni ticket che richiedono un tempo estremamente più lungo rispetto alla media, ma denotano anche una lentezza maggiore nella risoluzione rispetto ai problemi HW analizzati in precedenza.

Anche in questo caso sono stati analizzati nel dettaglio i tempi medi di risoluzione per ciascun argomento. Si può notare come la maggiore variabilità nei tempi di chiusura si ripercuota anche sui singoli argomenti, causando una maggiore dispersione non solo tra i valori medi, ma anche tra quelli mediani.

Ci sono argomenti come quelli riguardanti “Problemi con gli sconti e le promozioni” che nel 75% dei casi impiegano fino a 3 giorni per essere risolti e che in media vengono chiusi in 5.84 giorni, con una deviazione standard però elevata, pari a 27.41.

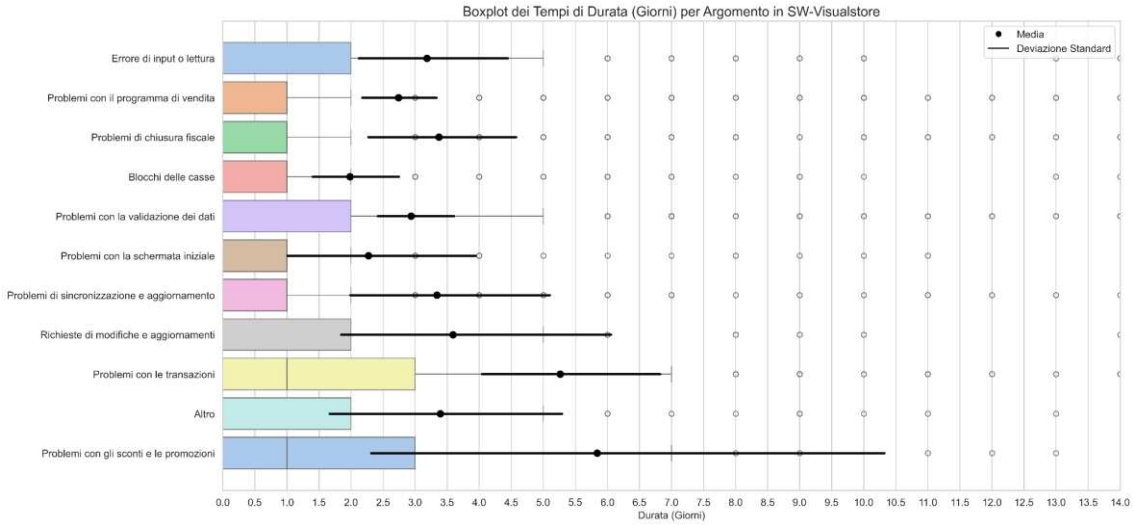

Figura 3.7: Boxplot della durata dei ticket in SW-Visualstore

Contrariamente con quanto visto per la matrice di confusione tra le classificazioni dell'azienda ("second level") e quelle del modello LLM ("Argomento") nel caso dei ticket “HW-Postazione Cassa”, analizzare la concordanza tra le due classificazioni ed identificare eventuali discrepanze o

pattern, non è stato immediato. Un second level molto frequente sotto la classificazione “SW-Visualstore” è infatti il generico “UT-Richiesta Informazioni”, sotto il quale ricadono la maggior parte degli argomenti estratti dal large language model. Si possono tuttavia notare, tolta questa categoria, certe concordanze come quelle tra “Problemi con il programma di vendita” e “SW-Servizi Applicativi” e “SW-Errore di programma F.E.”.

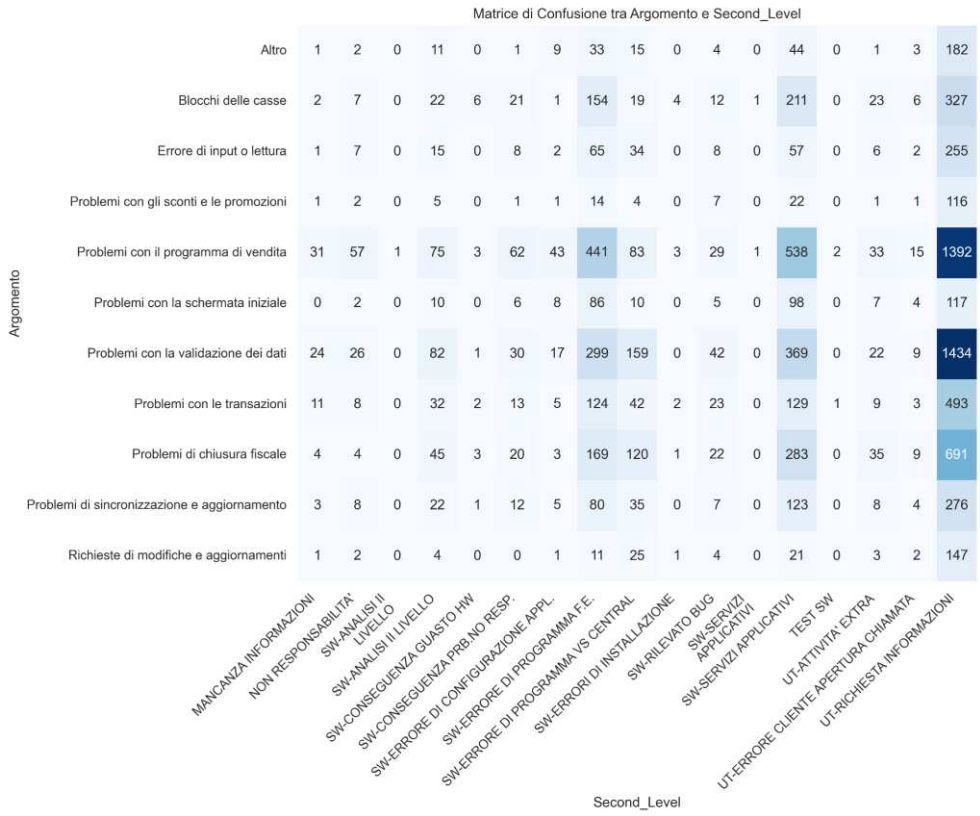

Figura 3.8: Matrice di confusione tra Argomento e Second Level

Infine, a differenza degli argomenti di HW-Postazione Cassa, quelli di SW-Visualstore mostrano un andamento temporale più vario come mostrato in figura dal seguente grafico a nastro.

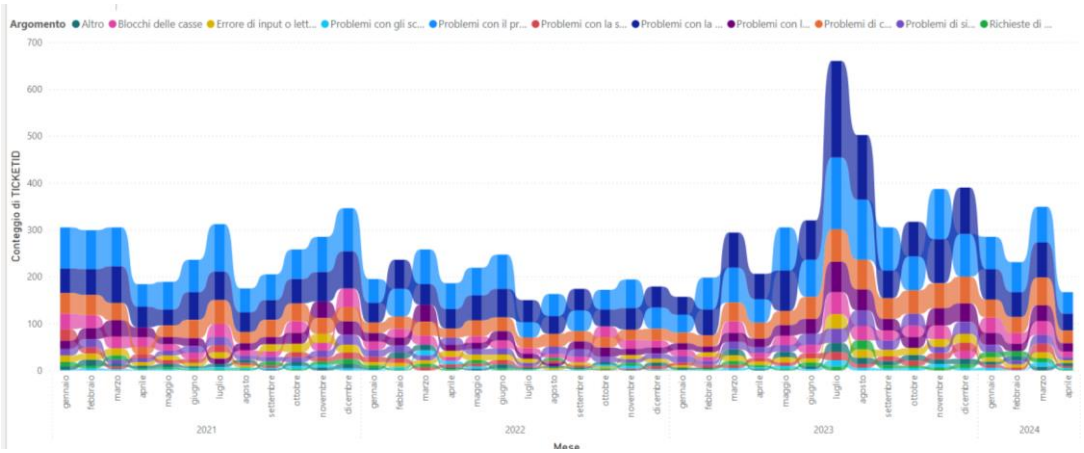

Figura 3.9: Andamento dei numeri di ticket per Argomento

3.4 Analisi dei risultati: Secondo Argomento

3.4.1 HW-POSTAZIONE CASSA: Blocco delle casse

Analizzando nel dettaglio l'argomento "Blocco delle casse", il sotto argomento più frequente, come mostrato nell'immagine successiva, è quello di "Cassa bloccata durante la transazione", per un totale di 3576 ticket su 4695.

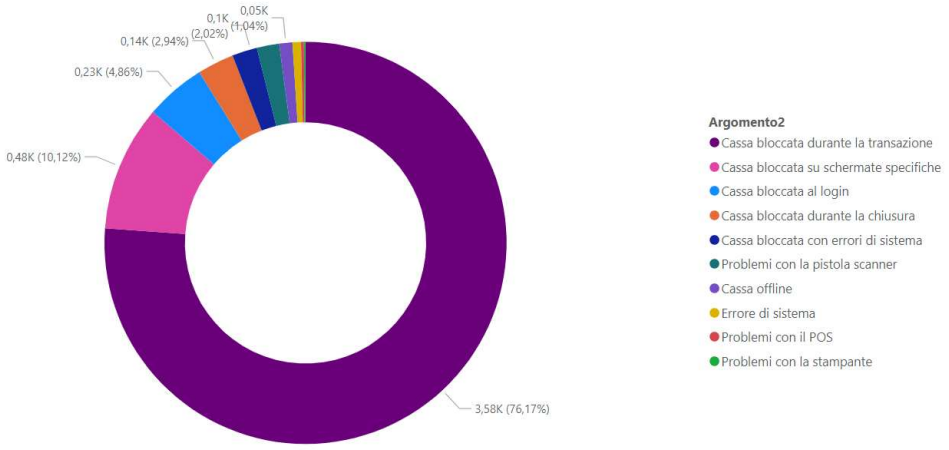

Figura 3.10: Argomenti dei ticket in Blocco delle casse

L'analisi dei tempi medi di risoluzione delle sottocategorie dell'argomento "Blocco delle casse" mostra come al proprio interno la distribuzione dei tempi dei sotto argomenti, ad eccezione di “Cassa bloccata durante la transazione” sia più eterogenea e variabile rispetto al macro topic preso nella sua interezza.

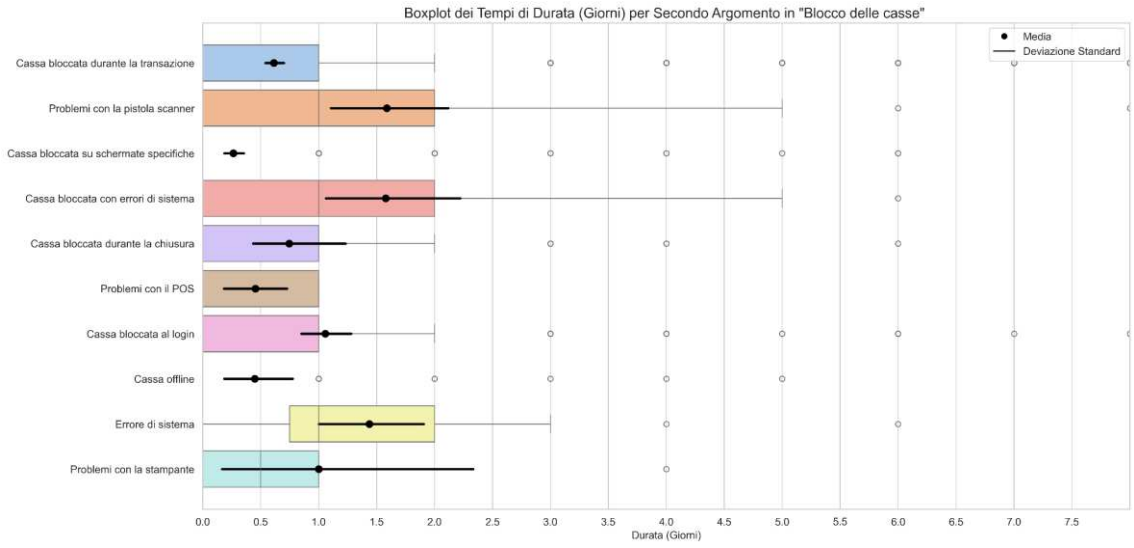

Figura 3.11: Boxplot della durata dei ticket in “Blocco delle casse”

La sottocategoria con il tempo medio di risoluzione più alto, seppur di poco, è "Problemi con la pistola scanner" con 1.58 giorni (circa 38 ore). Questa categoria presenta anche una deviazione standard elevata (2.79 giorni), indicando una variabilità significativa nei tempi di risoluzione.

È interessante notare come la maggior parte delle sottocategorie presenti un valore del terzo quartile (75° percentile) di 1 o 2 giorni, indicando che la maggior parte dei problemi venga risolta entro questo intervallo di tempo.

Calcolata la matrice di confusione tra le classificazioni dell'azienda e le nuove del LLM, notiamo come per "Cassa bloccata al login", la corrispondenza tra la classificazione del Second Level e quella del modello LLM mostra una buona identificazione, con una prevalenza di ticket classificati correttamente come HW-CASSA-BASE. Discorso simile si può fare per "Cassa bloccata su schermate specifiche" e "Problemi con il POS con la pistola scanner" sebbene ci siano maggiori discrepanze.

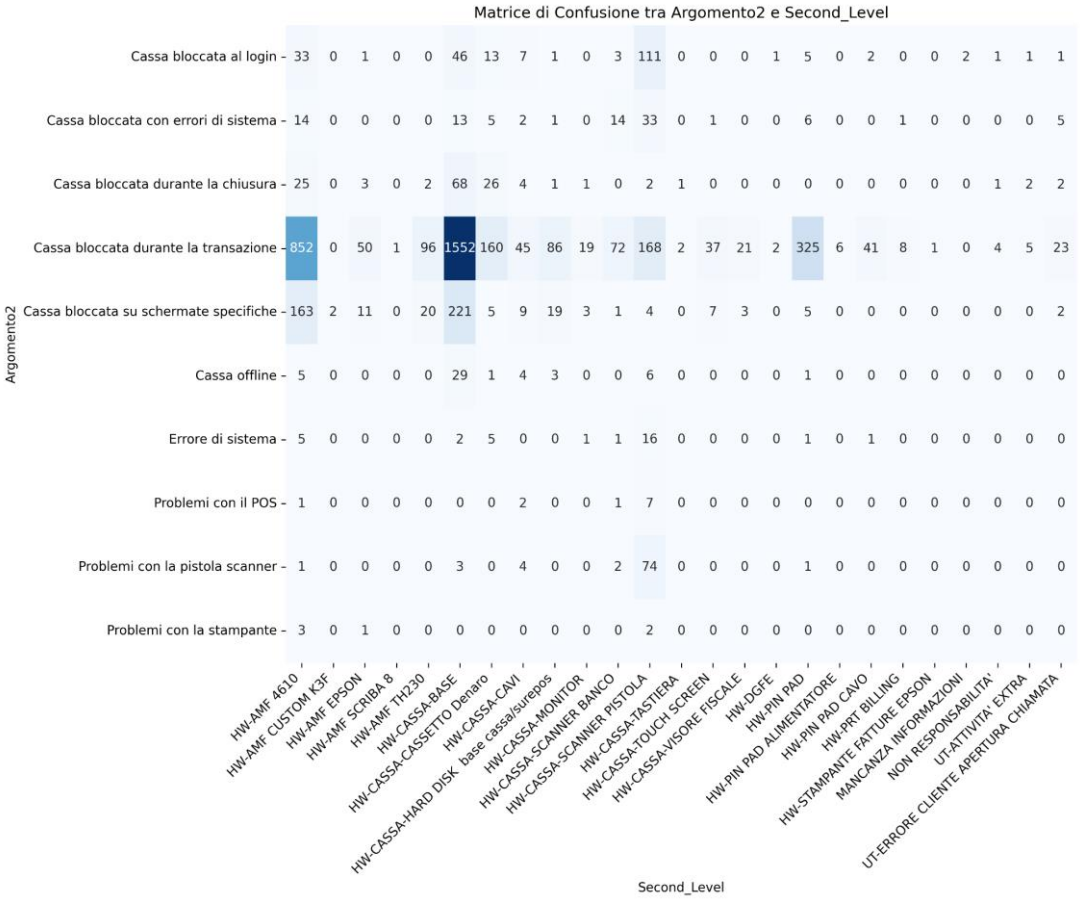

Figura 3.12: Matrice di confusione tra Argomento2 e Second Level

L'analisi della matrice di confusione tra il secondo livello di dettaglio del LLM e le soluzioni proposte dall'azienda (Terzo Livello) rivela invece un quadro chiaro delle corrispondenze tra i tipi di problemi identificati e le soluzioni applicate.

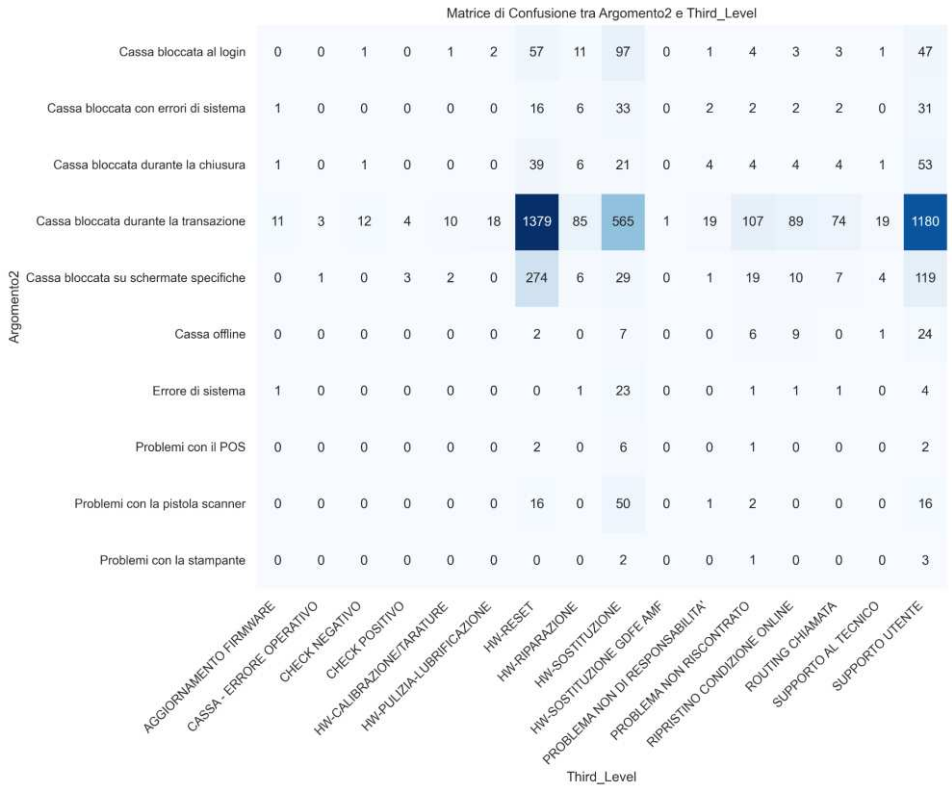

Figura 3.13: Matrice di confusione tra Argomento2 e Third Level

In generale, l'analisi indica che, mentre il reset hardware è una soluzione rapida e frequente, la sostituzione di componenti hardware è necessaria per problemi più gravi e persistenti. Altra categoria ricorrente è quella del Supporto Utente.

3.4.2 SW-VISUALSTORE: Problemi con il programma di vendita

Passando all'analisi dei risultati relativi all’argomento "Problemi con il programma di vendita", possiamo notare una distribuzione più bilanciata tra gli argomenti rispetto a quelli afferenti a “Blocco delle casse”. Le tre

categorie maggioritarie sono "Errore di input durante la scansione del codice a barre" che rappresenta il 31,01%, con un totale di 871 ticket, seguito da "Chiusure disallineate o non corrette" (714) e "Sconti non applicati" (656).

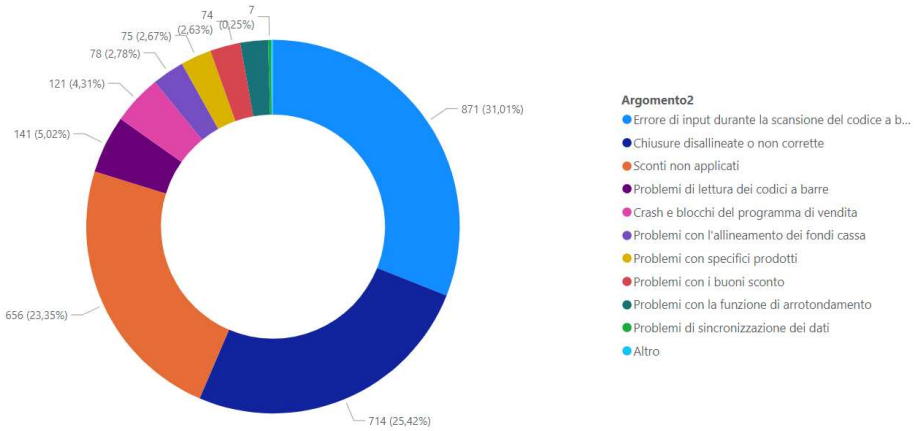

Figura 3.14: Argomento dei ticket in Problemi con il programma di vendita

In questo caso, l'analisi dei tempi medi di risoluzione delle sottocategorie, oltre a denotare un'elevata variabilità come in “Blocco delle casse”, mostra anche tempi di risoluzioni medi e mediani più elevati.

"Chiusure disallineate o non corrette" richiede in media 1.75 giorni, ma con una notevole variabilità (deviazione standard pari 7.2). Più lunga e variabile risulta la gestione di "Crash e blocchi del programma di vendita" con un tempo medio di 4.51 giorni e una deviazione standard di 23.6. Problematiche come "Errore di input durante la scansione del codice a barre" vengono generalmente risolte in media in 2.42 giorni, ma con una grande variabilità (deviazione standard di 15.7 giorni). La sottocategoria che registra il tempo

di risoluzione più lungo è "Problemi di lettura dei codici a barre" con una media di 8 giorni e una deviazione standard di 37. In generale, emergono risoluzioni più veloci per problematiche circoscritte e specifiche come "Chiusure disallineate o non corrette" rispetto a casistiche più generiche quali "Crash e blocchi del programma di vendita".

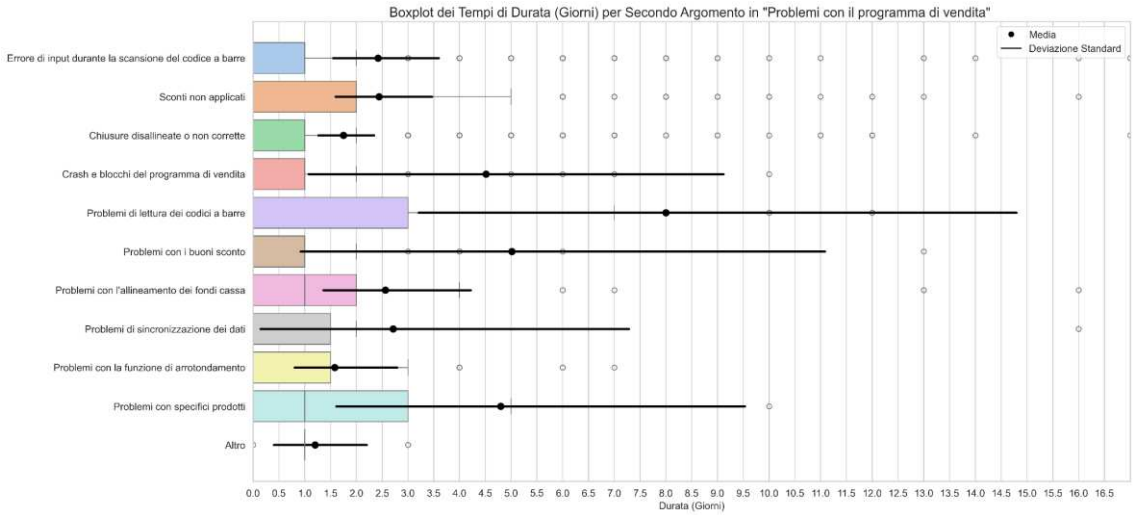

Figura 3.15: Boxplot della durata dei ticket in “Programma di vendita”

Sulla falsariga di quanto visto finora, abbiamo calcolato le due matrici di confusione per mettere a confronto i nostri argomenti assegnati dal LLM e quelli dell’azienda. Anche in questo caso, similmente a quanto già visto per gli argomenti dei problemi in SW-Visualstore, il second level più frequente è il generico “UT-Richiesta Informazioni”, sotto il quale ricadono la maggior parte degli argomenti estratti dal large language model. Si può tuttavia notare come la concordanza con “SW-Servizi Applicativi” e “SW-Errore di

programma F.E.” sia maggiore. Ad esempio, “Errore di input durante la scansione del codice a barre” mostra sì una prevalenza di soluzioni classificate come "UT-RICHIESTA INFORMAZIONI" (380 casi) ma con le categorie "SW-SERVIZI APPLICATIVI" (181 casi) e SW-ERRORE DI PROGRAMMA F.E. (155) non troppo distanti.

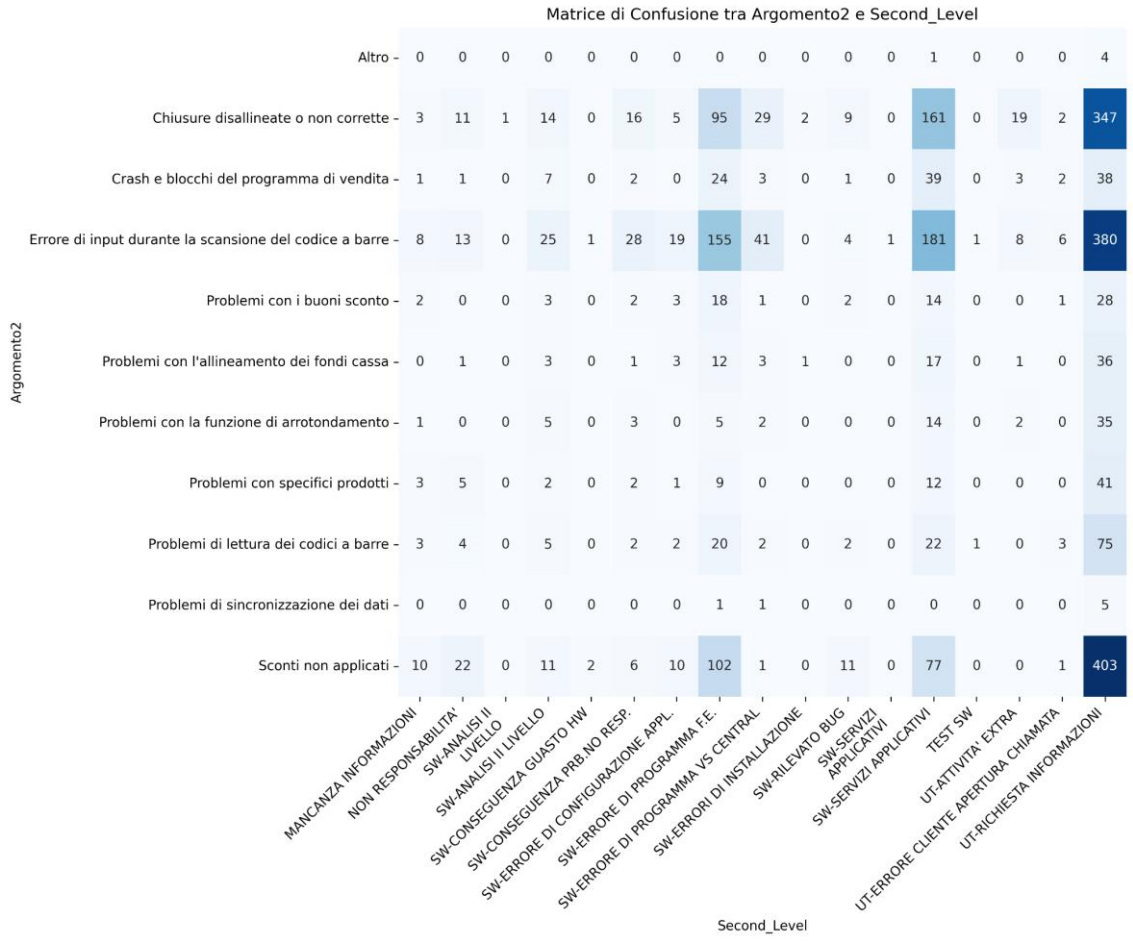

Figura 3.16: Matrice di confusione tra Argomento2 e Second Level

Il confronto con la variabile "Third Level" evidenzia ulteriormente le corrispondenze tra i problemi identificati e le soluzioni applicate dall'azienda. Per "Chiusure disallineate o non corrette", le soluzioni

maggiormente applicate sono "SUPPORTO UTENTE" (461 casi) e "SW-RIPRISTINO SOFTWARE" (82 casi). "Crash e blocchi del programma di vendita" vede una predominanza di "SUPPORTO UTENTE" (56 casi) e "SW-RIPRISTINO SOFTWARE" (29 casi). Risultati che portano a pensare che la scomposizione in categorie più dettagliate sia stata coerente con il contenuto del ticket.

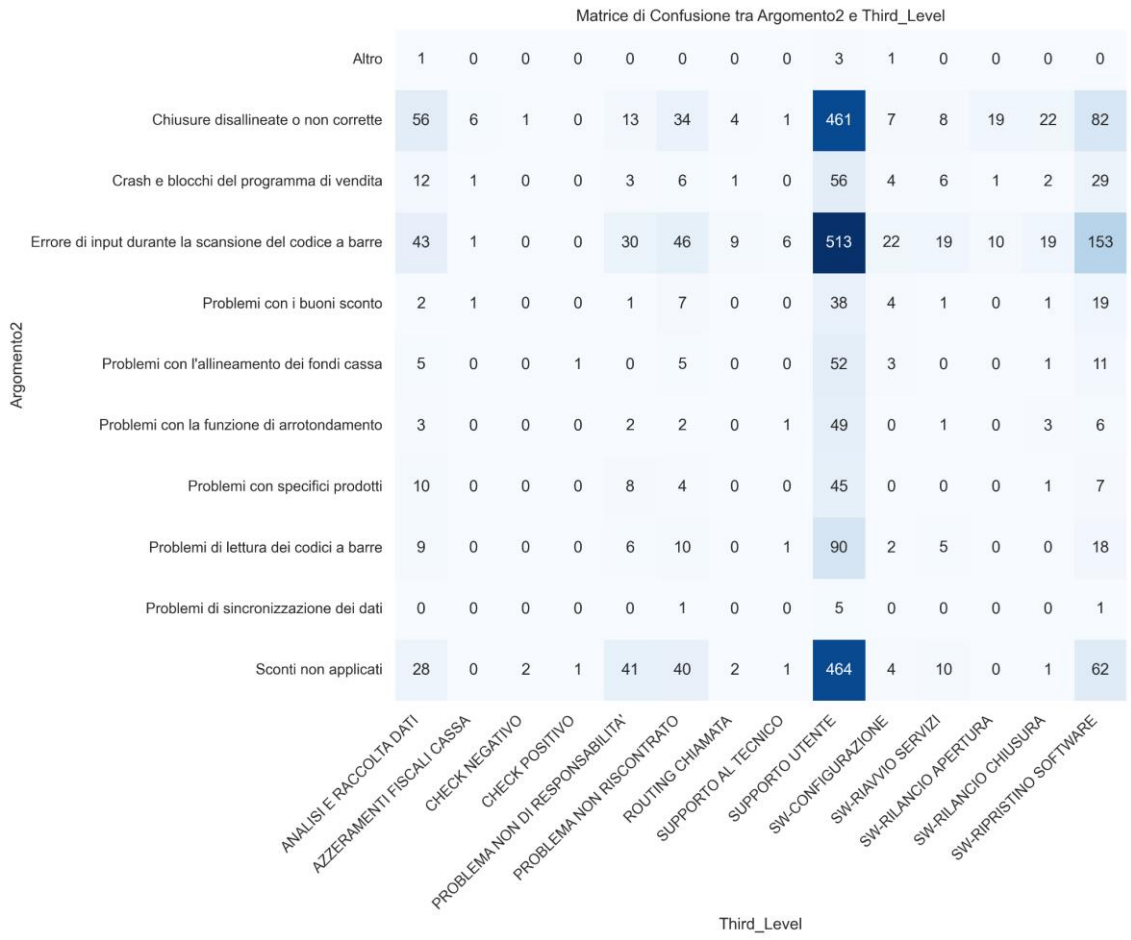

Figura 3.17: Matrice di confusione tra Argomento2 e Third Level

3.4.3 SW-VISUALSTORE: Problemi con la validazione dei dati

Tra “Problemi con la validazione dei dati” le tre categorie maggioritarie sono "Errori di validazione" che rappresenta il 40,33%, seguito da "Assenze di dati o incongruenze" (28,4%) e "Blocco delle casse" (16,95%).

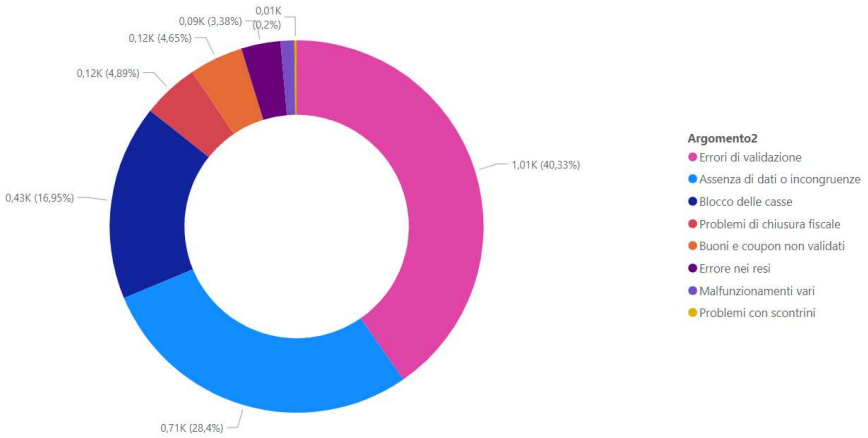

Figura 3.18: Argomento dei ticket in Problemi con validazione dati

I tempi di chiusura variano significativamente tra i secondi argomenti. L'analisi dei dati evidenzia che "Malfunzionamenti vari" ha una durata media più alta (13.97 giorni), suggerendo problemi più complessi o difficili da risolvere. Al contrario, "Problemi con scontrini" ha la durata media e la deviazione standard più bassa (2 giorni per entrambe), indicando risoluzioni più rapide. "Errori nei resi" e "Problemi di chiusura fiscale" mostrano una notevole variazione nei tempi di chiusura, con due elevate deviazioni standard rispettivamente di 11.08 e 28.02, riflettendo la variabilità nella complessità dei problemi affrontati.

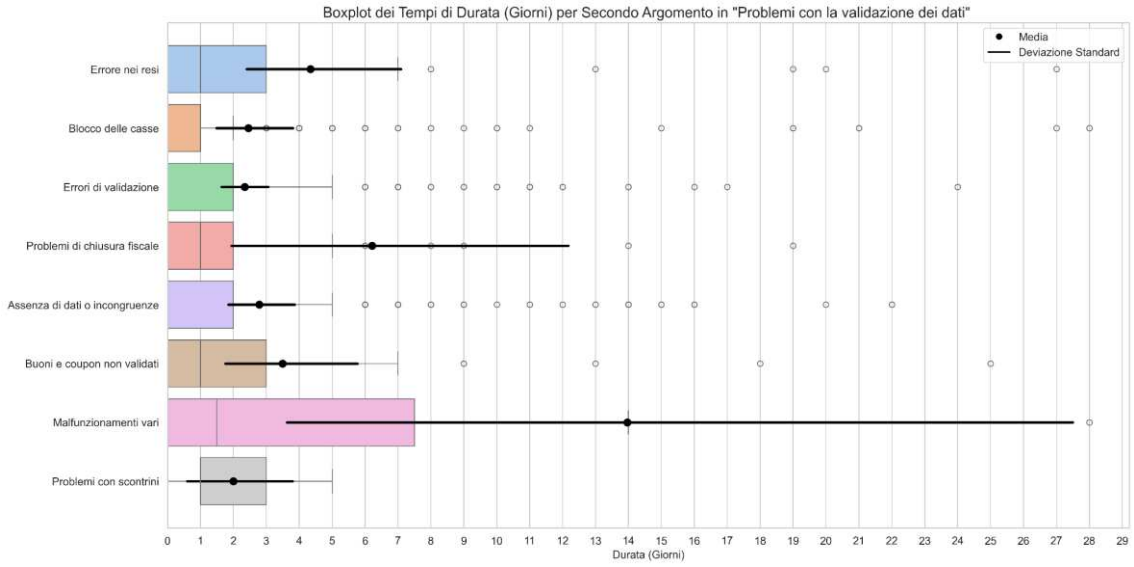

Figura 3.18: Boxplot della durata dei ticket in “Validazione dei dati”

La matrice di confusione tra second level ed il secondo argomento rivela che le richieste di informazioni siano predominanti per tutti i secondi argomenti, in particolare per "Assenza di dati o incongruenze" (431 richieste) e "Errori di validazione" (587 richieste). Questo suggerisce che questi problemi spesso richiedano chiarimenti o ulteriori dati per essere risolti.

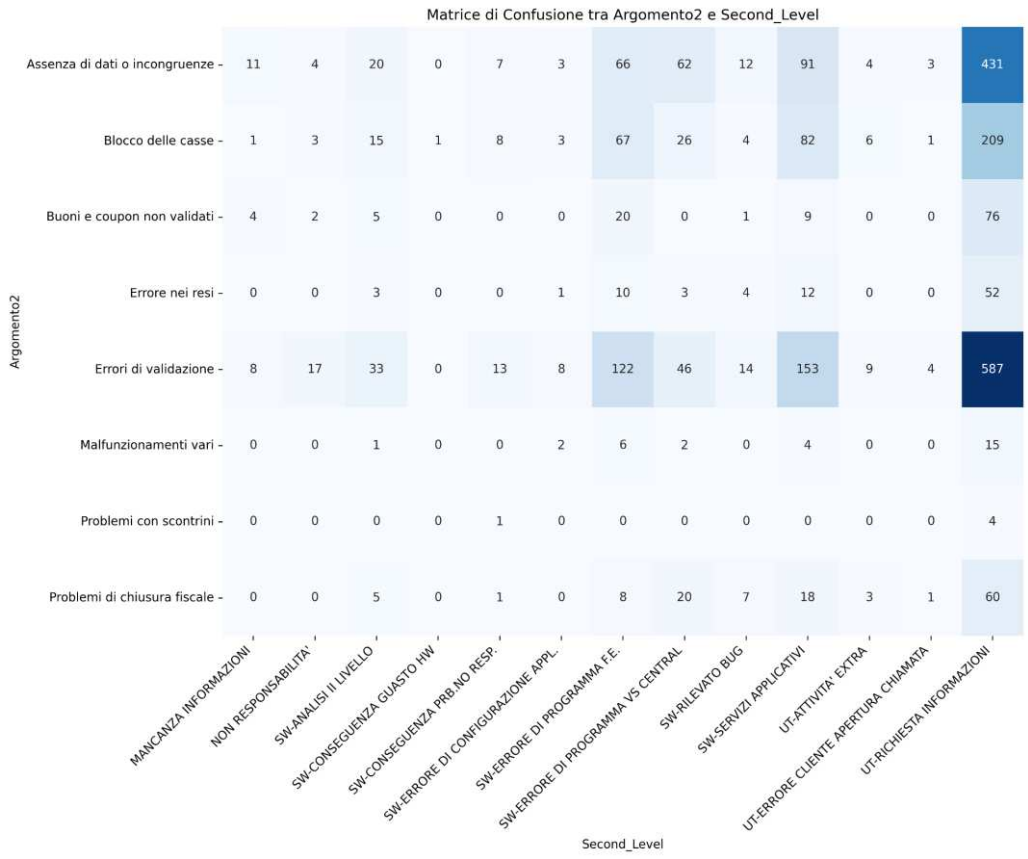

Figura 3.19: Matrice di confusione tra Argomento2 e Second Level

La matrice di confusione con il third level conferma che la maggior parte delle interazioni riguarda il supporto all'utente, con numeri particolarmente elevati per "Assenza di dati o incongruenze" (518) e "Errori di validazione" (700). Questo sottolinea l'importanza del supporto continuo e della comunicazione con l'utente per risolvere efficacemente questi problemi. Altri secondi argomenti come "Blocco delle casse" e "Buoni e coupon non validati" mostrano anch'essi una forte dipendenza dal supporto all'utente, riflettendo la necessità di interventi diretti per risolvere i problemi operativi.

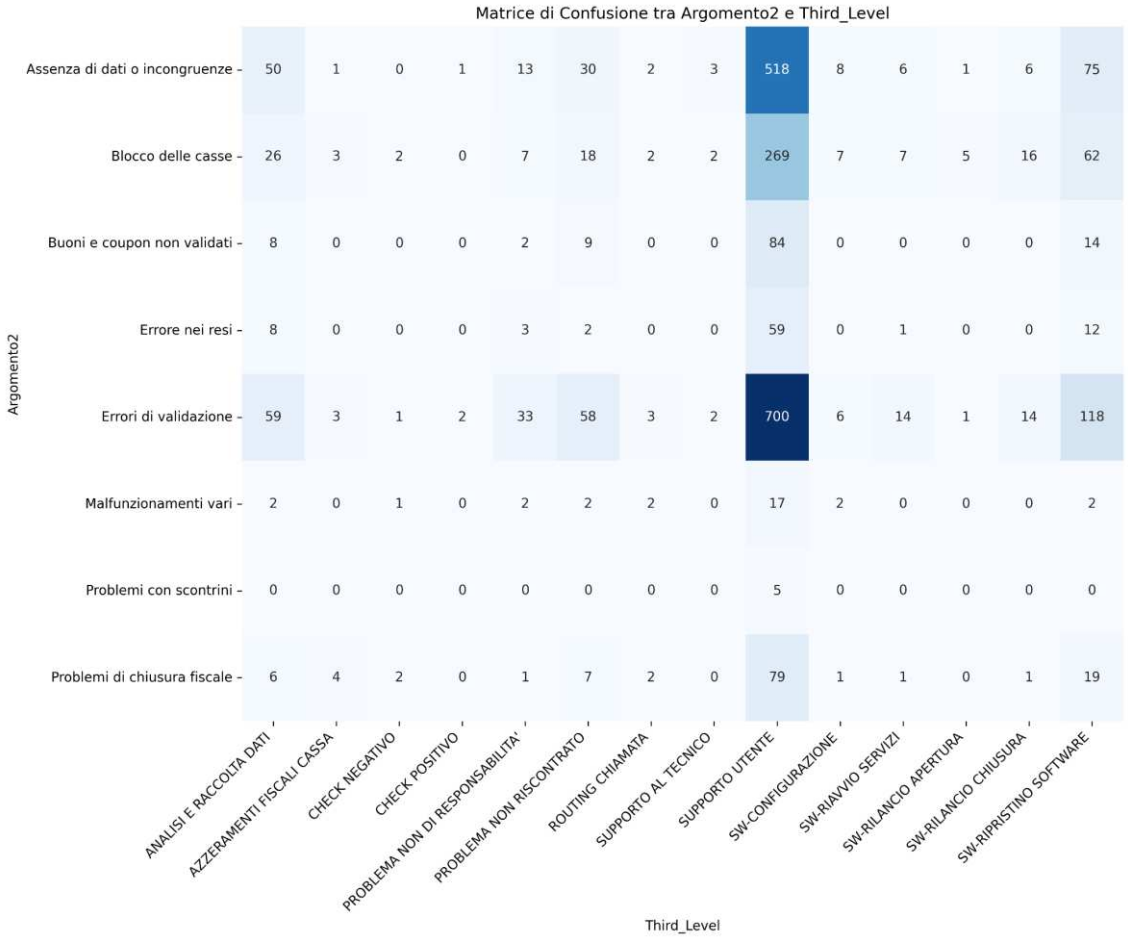

Figura 3.20: Matrice di confusione tra Argomento2 e Third Level

3.4.4 SW-VISUALSTORE: Problemi di chiusura fiscale

Riguardo ai ticket classificati come “Problemi di chiusura fiscale”, i tre topic più comuni sono “Chiusure pendenti”, “Differenze tra totale finanziario e totale fiscale” e “Cassa bloccata durante la chiusura” al cui interno rispettivamente troviamo 513 (36,41%), 397 (28,18%) e 283 (20,09%) ticket.

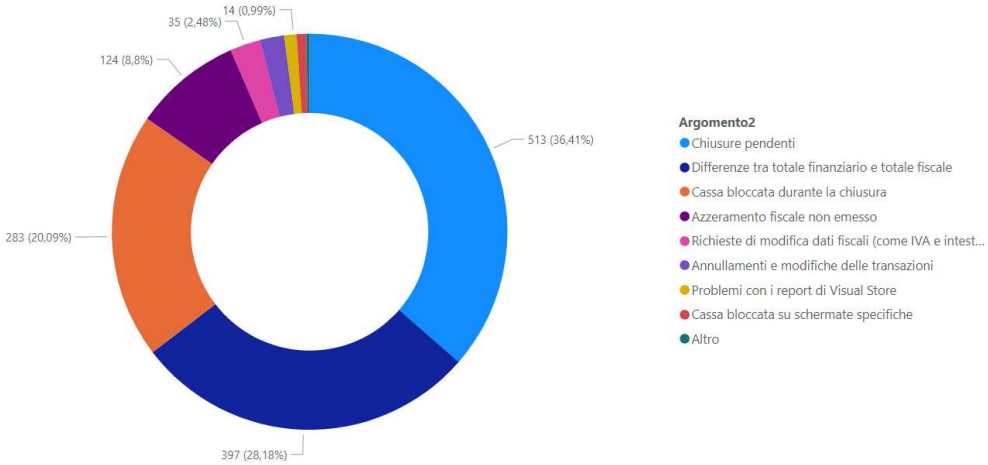

Figura 3.21: Argomento dei ticket in Problemi di chiusura fiscale

Per "Cassa bloccata durante la chiusura" la durata media è di 1.64 giorni, con una deviazione standard di 11.88. "Chiusure pendenti" ha una durata media di 1.99 giorni, con una deviazione standard di 13.18 giorni. Le "Differenze tra totale finanziario e totale fiscale" hanno una media di 6.36 giorni, con una deviazione standard di 37.32 giorni. In generale è possibile notare come le deviazioni standard degli argomenti inerenti a “Problemi di chiusura fiscale” siano tendenzialmente più elevate rispetto a quelle viste negli scorsi casi, indice di come queste problematiche siano spesso più complesse e lente anche per una questione di tempi di natura burocratica.

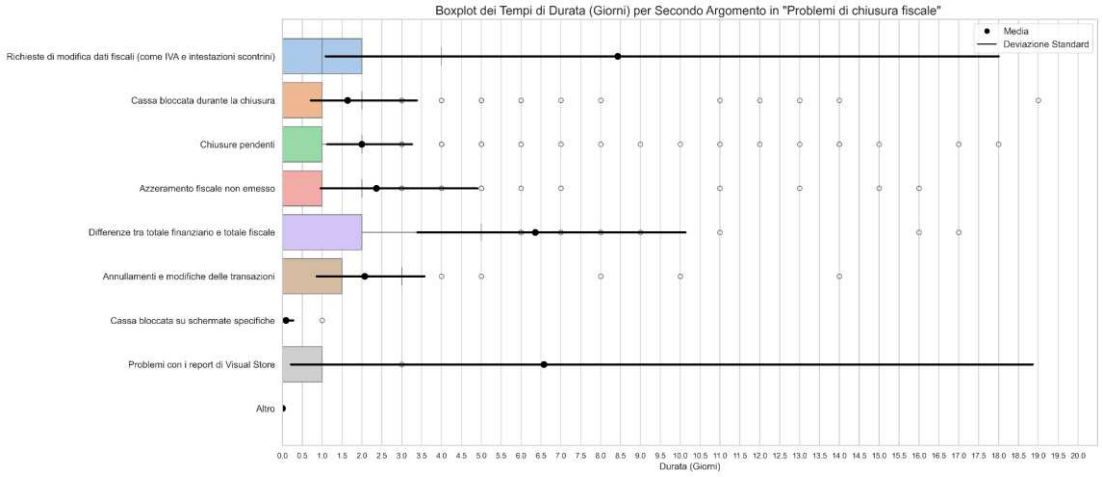

Figura 3.22: Boxplot della durata dei ticket in Problemi di chiusura fiscale

Osservando la matrice di confusione, notiamo una netta prevalenza di domande volte ad ottenere delucidazioni, specialmente in relazione ai temi "Chiusure pendenti" (con 235 richieste) e "Differenza tra totale finanziario e totale fiscale" (con 233 richieste).

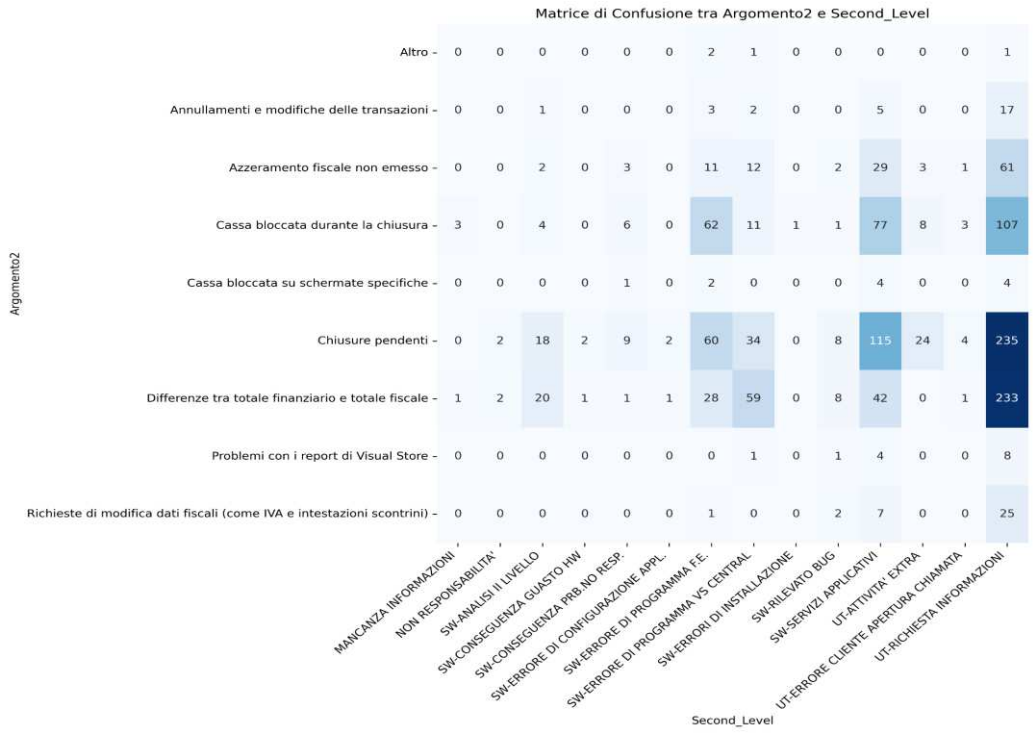

Figura 3.23: Matrice di confusione tra Argomento2 e Second Level

Infine, la matrice di confusione di terzo livello evidenzia come "Cassa bloccata durante la chiusura" e "Chiusure pendenti" richiedano frequentemente "Supporto Utente" con 156 e 318 casi rispettivamente. "Differenze tra totale finanziario e totale fiscale" e "Azzeramento fiscale non emesso" necessitano spesso di "Supporto Utente" e "Analisi e Raccolta Dati".

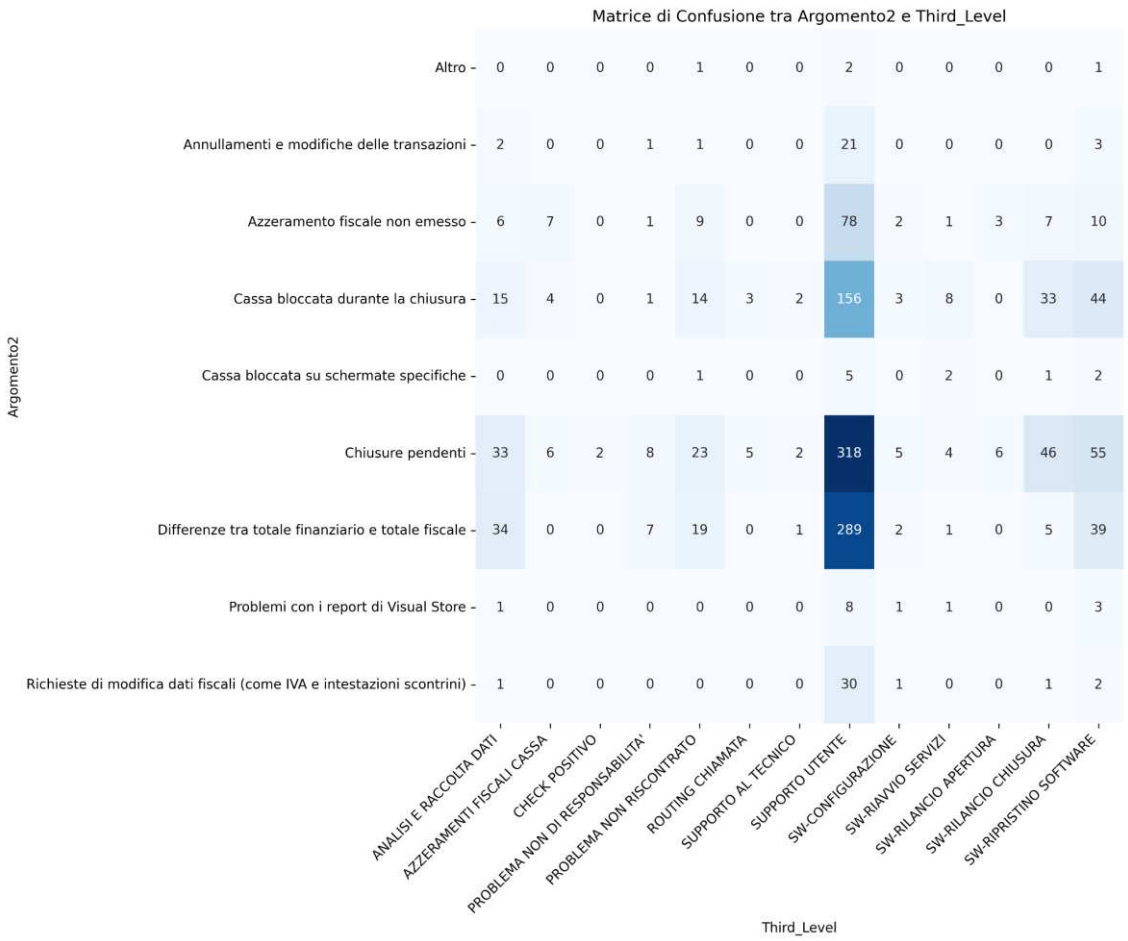

Figura 3.24: Matrice di confusione tra Argomento2 e Third Level

Conclusioni

In conclusione, questa tesi ha dimostrato come un'analisi dettagliata e strutturata dei dati raccolti dal call center esterno possa fornire a Magazzini Gabrielli strumenti essenziali per migliorare la qualità del servizio clienti. Utilizzando tecniche avanzate di Business Intelligence, Text Mining e Natural Language Processing, è stato possibile trasformare i dati grezzi in informazioni strategiche di grande valore.

In primo luogo, attraverso l'uso di strumenti di Business Intelligence come Power BI, si è potuto effettuare un'analisi descrittiva dei dati dei ticket. Questo ha permesso di monitorare vari aspetti del servizio clienti, come i tempi di risposta e i tipi di problemi riscontrati. Queste informazioni sono fondamentali per comprendere le prestazioni attuali e identificare aree di miglioramento.

Successivamente, sono state impiegate tecniche di text mining per analizzare i contenuti testuali dei ticket, identificando temi ricorrenti e tendenze nei problemi segnalati dagli utenti. Questa analisi ha consentito di andare oltre i semplici indicatori quantitativi, offrendo una comprensione più profonda delle problematiche affrontate.

Infine, l'adozione di modelli di elaborazione del linguaggio naturale (NLP), in particolare dei Large Language Models, ha permesso di sintetizzare e interpretare i contenuti dei ticket in modo più sofisticato. Questo ha facilitato

la comprensione delle problematiche ed il loro andamento nel tempo, fornendo una visione più completa delle esigenze dell'azienda.

L'applicazione di queste tecnologie all'analisi dei dati del call center rappresenta un esempio significativo di come l'innovazione digitale possa trasformare le operazioni aziendali, portando benefici tangibili in termini di efficienza e soddisfazione del cliente. Questo approccio integrato non solo migliora la capacità di risposta dell'azienda, ma contribuisce anche a creare un vantaggio competitivo nel settore della Grande Distribuzione Organizzata.

Bibliografia

Cer, D., Yang, Y., Kong, S., Hua, N., Limtiaco, N., John, R. st., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y.-H., Strope, B., & Kurzweil, R. (2018). Universal Sentence Encoder. http://arxiv.org/abs/1803.11175

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. de las, Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. le, Lavril, T., Wang, T., Lacroix, T., & Sayed, W. el. (2023). Mistral 7B. http://arxiv.org/abs/2310.06825

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need. http://arxiv.org/abs/1706.03762