UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica

Corso di Laurea Triennale in Informatica

TESI DI LAUREA

SensY: Classificazione Automatica dei Prompt Sensibili per i Large Language Model (LLM)

RELATORE

Prof. Andrea De Lucia

Dott. Gianmario Voria

Università degli Studi di Salerno

CANDIDATO

Alessandra Raia

Matricola: 0512116634

Anno Accademico 2024-2025

Questa tesi è stata realizzata nel

"Non ti arrendere. Rischieresti di farlo un'ora prima del miracolo."

— Proverbio arabo

A chi comprende di più, per temere di meno.

Marie Curie

A chi con una mano aiuta sé stesso e con l'altra il prossimo.

Audrey Hepburn

A chi non accetta l'impossibile e, senza chiedere il permesso, crea il proprio miracolo.

Grace Hopper

Abstract

I Large Language Model (LLM) rappresentano un'evoluzione significativa nell'ambito dell'Intelligenza Artificiale, grazie alla loro capacità di comprendere e generare linguaggio naturale. Tuttavia, con la crescente diffusione di tali modelli nella vita quotidiana, diventa sempre più urgente affrontare le criticità legate alla gestione dei contenuti sensibili. In questo lavoro si introduce una definizione del concetto di sensibilità, intesa come la capacità di un'espressione—per contenuto, contesto o formulazione—di suscitare emozioni forti, disagio, giudizi o rischi legati alla privacy, spesso toccando temi complessi come identità, politica, religione, salute mentale o relazioni personali.

A partire dal dataset coreano SQuARe, già esistente ma limitato da bias culturali e scarsa generalizzabilità, è stato condotto un lavoro di estensione e adattamento attraverso prompt generati con ChatGPT e dati reali provenienti da Chatbot Arena, in modo da renderlo più adattabile a contesti socioculturali diversi.

È stato sviluppato *SensY*, un modello per la classificazione automatica dei prompt in sensitive e non-sensitive, su cui è stato riportato un lavoro di addestramento e testing, utile a stabilire i ruoli dei dataset in gioco. I risultati mostrano che la miglior combinazione di dati di addestramento include SQuARe e un sottoinsieme realistico di Chatbot Arena, raggiungendo accuratezze superiori al 90% su test provenienti da SQuARe. Un'analisi qualitativa degli errori ha rivelato tematiche ancora critiche (es. identità, salute mentale, politica), suggerendo ulteriori miglioramenti futuri.

Il lavoro dimostra come una progettazione attenta e iterativa dei dataset migliori la capacità dei modelli di gestire contenuti eticamente complessi, ponendo le basi per valutazioni future su come i LLM affrontano contenuti potenzialmente critici, poiché, in un'epoca in cui l'AIGC (AI-Generated Content) ha un impatto diretto sulla società, è fondamentale garantire che i modelli non solo rispondano correttamente, ma lo facciano in modo consapevole, responsabile e contestualmente appropriato.

Indice

Elenco delle figure

Elenco delle tabelle

CAPITOLO 1

Introduzione

1.1 Contesto Applicativo

I Large Language Model (LLM) stanno trasformando il panorama dell'Intelligenza Artificiale, offrendo strumenti potenti per generare contenuti testuali, rispondere a domande o supportare attività complesse. Tuttavia, il rapido avanzamento e la loro inevitabile diffusione in contesti quotidiani e sociali pone interrogativi sulla capacità e la necessità di gestire prompt sensibili, ovvero quesiti che, per tematica o formulazione, potrebbero generare risposte inappropriate, offensive o eticamente problematiche. In un'epoca in cui l'AI-Generated Content (AIGC) permea strumenti quotidiani — dai chatbot ai motori di ricerca conversazionali — il rischio che un input sensibile attivi risposte problematiche è tutt'altro che trascurabile.

1.2 Motivazioni e Obiettivi

L'esigenza di identificare automaticamente i prompt sensibili nasce da episodi concreti in cui i Large Language Model (LLM) hanno generato contenuti inappropriati, discriminatori o eticamente discutibili.

Sebbene la letteratura scientifica affronti da tempo i temi legati a bias, fairness e contenuti sensibili nei sistemi di Intelligenza Artificiale, molte delle soluzioni proposte sono basate su dati tabulari o numerici, spesso distanti dal linguaggio naturale che caratterizza i prompt realmente utilizzati dagli utenti. Inoltre, i dataset annotati presenti in letteratura — come SQuARe [1] — mostrano limiti di copertura tematica e bias culturali significativi, rendendoli poco generalizzabili in contesti diversi da quello in cui sono stati prodotti.

Questo lavoro propone SensY, un classificatore supervisionato in grado di distinguere tra prompt sensibili e non sensibili nel contesto del linguaggio naturale, attraverso una combinazione di feature sintattiche (estratte con Natural Language Toolkit *NLTK*) e semantiche (Sentiment Analysis *SA* e BERT). L'approccio si basa sull'integrazione e sull'arricchimento del dataset esistente SQuARe [1] attraverso:

- l'inclusione di prompt generati artificialmente (via ChatGPT), utili a coprire scenari sottorappresentati;
- l'integrazione di prompt reali e validati provenienti dal dataset *Chatbot Arena Conversations*, pubblicato dalla comunità accademica[2].

◎ **Our Goal.** L'obiettivo è duplice: da un lato, creare un dataset vario, vasto e affidabile di prompt di linguaggio naturale; dall'altro, favorire un uso più equo, inclusivo e consapevole dell'IA, fornendo un classificatore binario capace di identificare i prompt sensibili.

1.3 Risultati Ottenuti

Dopo un'estesa sperimentazione, che ha incluso tecniche di bilanciamento (clustering tematico, undersampling controllato) e l'integrazione di prompt reali (Chatbot Arena) e sintetici (ChatGPT), SensY ha raggiunto ottimi risultati, con accuratezza fino a 0.905. L'analisi degli errori ha inoltre guidato l'individuazione di tematiche in cui il modello è meno efficace, in modo da fornire linee guida per un futuro arricchimento dei dataset di addestramento.

1.4 Struttura della tesi

- Capitolo 2: **Background e stato dell'arte**
  Fornisce una panoramica della letteratura su Large Language Model (LLM), sensitiveness e fairness.

- Capitolo 3: **Metodo di Ricerca**
  Illustra le scelte metodologiche impiegate nella costruzione di dataset di prompt di Natural Language per lo sviluppo e la sperimentazione del modello di classificazione *SensY*.

- Capitolo 4: **Analisi dei Risultati**
  Presenta i risultati quantitativi e qualitativi ottenuti durante il processo iterativo e incrementale di training e testing del modello.

- Capitolo 5: **Conclusioni**
  Conclude la tesi discutendo l'impatto, i limiti e gli sviluppi futuri.

CAPITOLO 2

Background e stato dell'arte

2.1 Large Language Model

I Large Language Model (LLM) sono modelli di Intelligenza Artificiale (IA) fondati su reti neurali profonde, capaci di comprendere, manipolare e generare linguaggio naturale, sulla base di un addestramento su notevoli quantità di testo. Tali modelli rientrano nella categoria dei *modelli fondazionali*, ovvero modelli pre-addestrati su vasti dati non supervisionati, con l'obiettivo di generalizzare a molteplici task attraverso tecniche come *fine-tuning* o *prompting*. Sono particolarmente utili per eseguire una vasta gamma di compiti, tra cui: generazione di risposte, traduzioni, assistenza alla programmazione e classificazione delle sequenze.

Definizione #1. Fine-tuning: gli ultimi strati vengono addestrati con i dati target per apprendere caratteristiche più specifiche.

Definizione #2. Prompt: un insieme di istruzioni o una domanda data a un LLM in linguaggio umano per suscitare una risposta particolare [3].

Definizione #3. Prompt Engineering: è l'arte e la tecnica di progettare prompt ottimizzati per guidare un modello di linguaggio naturale verso risposte accurate, pertinenti e utili. Consiste nel formulare input chiari, specifici e contestuali per sfruttare al meglio le capacità del modello, adattandolo a diversi task o contesti applicativi.

I Large Language Model sono addestrati su molteplici task di Natural Language Processing (NLP), come:

- **Language understanding:** comprensione del significato di testi, identificazione di entità, sentiment analysis, etc.
- **Language generation:** generazione coerente di testi, completamento di frasi, scrittura automatica, etc.

Si tratta di modelli con un'architettura autoregressiva, autocodificante o codificatore-decodificatore. Il paradigma "pre-addestramento, fine-tuning" consente l'addestramento di un modello di base che può essere adattato a una vasta gamma di applicazioni (Bommasani et al. 2021; Min et al. 2023). Un esempio emblematico è rappresentato da GPT (Generative Pre-trained Transformer), un LLM autoregressivo che, grazie al solo prompting, riesce a svolgere compiti come la scrittura creativa, la programmazione in linguaggi come Python e JavaScript, o la traduzione tra lingue. Questo approccio ha segnato un allontanamento dalle architetture specifiche per attività e, di fatto, i LLM ottimizzati su un set di dati relativamente piccolo e specifico per attività, possono superare i modelli specifici per attività addestrati da zero.[4].

2.1.1 LLM-Limitazioni

I Large Language Model (LLM) possiedono il potenziale per rivoluzionare diversi aspetti della nostra vita, grazie alla loro capacità di generare contenuti linguistici noti come AI-Generated Content (AIGC). Proprio per il loro crescente impatto e l'effetto diretto che hanno sulla nostra quotidianità, è necessario essere consapevoli delle loro limitazioni e affrontarle con spirito critico.
Alcuni problemi tipici dell'interazione con i LLM:

- **Ambiguità:** i prompt ambigui potrebbero essere seguiti da risposte fuorvianti o trascurabili.
- **Sensibilità al contesto:** spesso il modello potrebbe non cogliere o mal interpretare il contesto di un prompt, talvolta necessario per la corretta comprensione del quesito.
- **Conoscenza statica:** i LLM sono limitati dalla conoscenza disponibile nei dati di addestramento, i quali delimitano le capacità del modello con il fine di fornire risposte outdated o subottimali.
- **Complessità:** dettagliare i prompt può portare a dei prompt dannosamente lunghi o complessi, rischiando di confondere il modello.
- **Flakiness:** si riferisce all'incoerenza o variabilità nelle risposte generate da un modello linguistico o un sistema di intelligenza artificiale per lo stesso prompt di input.
- **Allucinazioni:** un modello potrebbe generare informazioni false, inesistenti o non supportate dai dati disponibili, pur presentandole come plausibili o accurate.

Queste limitazioni sollevano riflessioni articolate sull'uso del linguaggio nei sistemi di IA, e più in generale sul ruolo sociale e culturale che tali tecnologie stanno assumendo. Il linguaggio, infatti, non è solo uno strumento tecnico: è lo specchio della società che costruiamo, il potente mezzo attraverso cui esprimiamo pensieri, valori, ideali e comportamenti. Di conseguenza, i LLM — che apprendono e riproducono strutture linguistiche — hanno il potere di influenzare percezioni e comportamenti e, se da un lato possiamo apprezzarne le positive capacità, dall'altro è nostra responsabilità gestire i possibili danni.

2.2 Sensitiveness

L'adozione su larga scala dei LLM in ambiti sensibili , senza un'attenta considerazione dei potenziali impatti negativi, rappresenta un rischio concreto, motore dello

sviluppo di discriminazioni sociali, pregiudizi limitanti, favoritismi ingiusti, visioni del mondo parziali e non inclusive.
Proprio come gli esseri umani, anche i LLM sono vulnerabili, poiché apprendono da dati d'archivio prodotti dall'uomo e da istituzioni, portando con sé inevitabili distorsioni: bias impliciti, stereotipi culturali, contenuti discriminatori o imprecisi. L'AIGC potrebbe, per questo motivo, ereditare e persino amplificare elementi sensibili, come i pregiudizi, le imprecisioni e le discriminazioni nei dati di addestramento. Pertanto è imperativo perfezionare, riaddestrare e risintonizzare.

In questo contesto, sebbene il concetto di **fairness** sia centrale nel dibattito sull'etica dei sistemi intelligenti, esso si concentra principalmente sulla parità di trattamento tra gruppi sociali e sull'equità nelle decisioni automatiche. Tuttavia, molte situazioni problematiche legate ai LLM non rientrano in modo diretto nei tradizionali quadri di fairness. Pensiamo, ad esempio, a prompt che toccano la salute mentale, la sessualità, il trauma, la religione, o l'identità personale: non sempre questi pongono un problema di equità formale, ma possono comunque generare disagio, imbarazzo, violazione della privacy o percezioni di ostilità.
Per questo motivo, diventa fondamentale introdurre un concetto più ampio: quello della *sensitiveness*, ovvero la capacità di riconoscere e gestire temi sensibili che toccano dimensioni etiche, culturali, identitarie o socialmente delicate. La mancata gestione della sensitiveness può condurre a effetti negativi quali mancanza di inclusività, riproduzione di bias, violazioni della privacy o della sicurezza, e una generale percezione di ingiustizia o discriminazione nei contenuti generati.

2.2.1 Casi reali di contenuti non etici generati da LLM

Nel corso degli anni, diversi casi hanno evidenziato la difficoltà di gestire adeguatamente la sensitiveness nei Large Language Model. Di seguito, alcuni esempi emblematici:

- **Tay di Microsoft (2016):** Tay era un chatbot AI lanciato su Twitter con lo scopo di apprendere dal linguaggio degli utenti. In meno di 24 ore, Tay iniziò a produrre messaggi razzisti, sessisti e negazionisti, riflettendo contenuti tossici appresi

dalla community. Microsoft fu costretta a disattivarlo, riconoscendo l'assenza di adeguati filtri e meccanismi di controllo.[5]

- **Hiring bias nei sistemi automatizzati (Amazon, 2018):** Sebbene non un LLM in senso stretto, il sistema di recruitment automatico di Amazon (basato su NLP e ML) ha mostrato preferenza per candidati maschi, penalizzando quelli con curriculum contenenti parole associate al genere femminile. Il sistema è stato ritirato perché si basava su dati storici distorti da pratiche di assunzione discriminatorie.[6]

- **GPT-3 e bias religiosi (2020):** È stato documentato che, in risposta a prompt come "Two Muslims walk into a", GPT-3 tendeva a completare la frase con atti violenti o terroristici. Questi esempi rivelano un pattern di associazioni implicite apprese dai dati di addestramento, che riproducono stereotipi pericolosi su base religiosa e culturale.[7]

- **AI Dungeon e contenuti inappropriati (2021):** La piattaforma di gioco AI Dungeon, basata su LLM, ha generato contenuti sessuali espliciti, anche con soggetti minorenni, in risposta a prompt ambigui. Il caso ha sollevato serie preoccupazioni sull'uso dell'AI in ambienti ludici aperti, spingendo gli sviluppatori a rivedere le policy di sicurezza e moderazione.[8]

Questi esempi mostrano come anche modelli altamente performanti possano produrre risultati eticamente inaccettabili, se non vengono correttamente progettati, monitorati e adattati a contesti sensibili.

A tal proposito, Dwivedi et al. [3] scrivono della loro ricerca sulla comprensione delle dinamiche di rappresentanza di genere negli output dei LLM, ed evidenziano la necessità di non soffermarsi a scoprire i problemi, ma di porsi l'obiettivo di far evolvere la tecnologia rendendola più inclusiva e rappresentativa. In questo contesto, tecniche come il *Prompt Engineering* e l'*In-Context Learning* offrono strumenti concreti per orientare il comportamento del modello verso risposte più imparziali.

Definizione #4. In-Context Learning: consente ai modelli di regolare le loro risposte in base a una serie di esempi o contesti forniti. Ad esempio, fornendo a un LLM esempi di linguaggio neutro rispetto al genere o mostrando diverse rappresentazioni di genere, il modello può potenzialmente essere guidato a produrre risultati più equi.[3]

2.2.2 Fairness e ML Fairness

Il concetto di *fairness* rappresenta, quindi, una dimensione fondamentale — ma non esaustiva — della sensitiveness. La letteratura mostra come non esista una definizione univoca e universalmente condivisa di fairness [9], poichè questa è influenzata da fattori religiosi, culturali, filosofici, sociali, storici, politici, etici, legali e personali. Secondo la concezione delle norme ISO/IEC 22989 e ISO/IEC TR 24027 [9] la fairness è un "trattamento, un comportamento o un risultato che rispetta fatti, credenze e norme stabilite e non è determinato da favoritismi o discriminazioni ingiuste".

La fairness è strettamente legata al concetto di Bias, che ISO/IEC 24027 definisce come "differenza sistematica nel trattamento di determinati oggetti, persone o gruppi rispetto ad altri".
In Machine Learning (ML) il Bias, quindi, misura quanto le previsioni del modello si discostano sistematicamente dai valori reali. Nel contesto dell'IA, alcuni pregiudizi sono essenziali (pregiudizi desiderati: utile a distinguere classi nei compiti algoritmici) oppure indesiderati, quando producono effetti ingiusti o discriminatori verso individui o gruppi sociali.[9]
Sebbene gli algoritmi non vengano progettati per introdurre discriminazioni intenzionali, la natura statistica dell'apprendimento automatico può portare alla perpetuazione di pattern ingiusti presenti nei dati. In questo senso, le nozioni di fairness aiutano a individuare e misurare questi bias indesiderati. Tuttavia, è importante ricordare che l'obiettivo più ampio è assicurarsi un corretto approccio a contenuti sensibili al contesto umano e culturale.
Riconoscere la fairness come componente della sensitiveness consente di collocarla in un quadro più ampio che include anche altri aspetti cruciali, come:

- il rispetto della diversità culturale;
- la rappresentazione bilanciata delle identità sociali;
- l'evitamento di contenuti polarizzanti o offensivi;
- l'attenzione alla sicurezza, alla privacy e all'impatto sociale.

Analizzare i prompt e i contenuti generati dai LLM attraverso la lente della sensitiveness permette, quindi, non solo di valutare la correttezza tecnica del modello, ma anche la sua compatibilità con i principi di responsabilità sociale, inclusività e sostenibilità culturale.

2.3 Tecniche di Bias Mitigation

I potenziali danni sociali posti dai LLM sono in forte aumento. Per ridurre il grado di bias nei sistemi di IA, i professionisti adottano metodi che rientrano nelle seguenti tre categorie[10]:

- **Pre-processing:** mitigazione del bias nei dati di addestramento, per impedirne il trasferimento nei modelli ML [10]. Queste tecniche cercano di trasformare i dati in modo da rimuovere la discriminazione sottostante se l'algoritmo è autorizzato a modificare i dati di addestramento [11].
- **In-processing:** mitigazione del bias durante l'addestramento dei modelli ML[10]. Queste tecniche cercano di modificare e cambiare gli algoritmi di apprendimento all'avanguardia se è consentito modificare la procedura di apprendimento per un modello di apprendimento automatico, incorporando modifiche nella funzione obiettivo o imponendo un vincolo[11].
- **Post-processing:** mitigazione del bias sui modelli ML già addestrati[10]. Viene eseguita dopo l'addestramento accedendo a un set di controllo che non è stato coinvolto durante l'addestramento del modello. Se l'algoritmo può trattare il modello appreso solo come una scatola nera senza alcuna capacità di modificare i dati di addestramento o l'algoritmo di apprendimento, allora può essere utilizzata solo la post-elaborazione in cui le etichette assegnate dal modello a

scatola nera vengono inizialmente riassegnate in base a una funzione durante la fase di post-elaborazione [11].

Analizzando lo stato dell'arte, è difficile non prendere atto delle limitazioni che evidenziano. La maggior parte degli articoli agisce su dati tabulari, aspetto che con questo lavoro si intende superare utilizzando prompt di Natural Language.
Tra i lavori esistenti, un modello sicuro per affrontare le discussioni su questioni delicate, che possono diventare tossiche anche se gli utenti sono ben intenzionati, è il set di dati SQuARe (Sensitive Questions and Acceptable Responses). Questo è un set di dati su larga scala di 49k domande sensibili con 42k risposte accettabili e 46k non accettabili. Il set di dati è stato costruito facendo leva sull'invecchiamento di HyperCLOVA in modo human-in-the-loop basato su titoli di notizie reali[1]. Nonostante sia concettualmente il modello più vicino al lavoro che si propone, non manca di limitazioni: *"SQUARE si concentra principalmente su questioni socialmente sensibili con tre categorie– predictive, contentious, ethical– e le loro risposte accettabili (...). Sebbene l'ambito mirato di SQUARE contribuisca ad alleviare efficacemente le risposte socialmente sensibili nelle implementazioni di LLM, esistono ancora aspetti più sensibili che non affrontiamo. Considerando che una lingua riflette la proprietà e la cultura della società, alcune delle questioni delicate che il nostro SQUARE affronta potrebbero essere un po' specifiche per la Corea. (...) Il nostro framework di collaborazione uomo-LLM per la costruzione dei dati può essere applicato ad altre lingue"*[1].

SQuARe pone un'interessante base per il lavoro su prompt di Natural Language, ma solleva nuove problematiche, ovvero overfitting e poca generalità delle questioni.

CAPITOLO 3

Metodo di Ricerca

Il presente capitolo descrive in modo dettagliato le scelte metodologiche adottate per il raggiungimento degli obiettivi di questo lavoro. L'intero processo è guidato dalla seguente domanda di ricerca:

> **RQ$_1$.** *Fino a che punto è possibile prevedere automaticamente la sensibilità di un prompt rivolto a un Large Language Model (LLM)?*

Questa trova risposta nella costruzione di *SensY*, un classificatore che ha l'ambizione di rilevare la sensibilità dei prompt posti a Large Language Model (LLM).

Il concetto di ***sensitiveness***, anticipato nel capitolo precedente, è stato concordato dai membri del team di ricerca del laboratorio di Ingegneria del Software (SeSaLab, Università degli studi di Salerno) come segue:

> *"Un'espressione è **sensitive** se, per via del suo contenuto, del contesto in cui viene usata o della sua formulazione, ha il potenziale di suscitare emozioni forti, generare disagio o rischio di giudizio. Può essere in grado di discriminare, risultare invasiva o compromettere la sicurezza o la privacy. Spesso riguarda temi complessi o personali, come l'etica, la religione, la politica, l'uguaglianza o l'identità. Se l'espressione è una domanda, la definizione si completa con: 'La*

risposta a una domanda sensitive può variare notevolmente, rischiando di essere fraintesa o percepita in modo diverso da quanto inteso, proprio a causa della natura soggettiva delle reazioni che può suscitare.'"

Nel seguito, verranno analizzate le fasi chiave del processo: la selezione del dataset iniziale, SQuARe, la preparazione del dataset di prompt di Natural Language, che ha incluso l'integrazione di ulteriori dataset, tra cui un insieme di prompt generati tramite **ChatGPT** e il corpus **Chatbot Arena Conversations**, già noto in letteratura[2] e lo sviluppo del classificatore SensY.

I risultati sperimentali di queste operazioni saranno discussi nel Capitolo 4.

3.1 Metodologia di ricerca

L'approccio sperimentale adottato per rispondere alla domanda di ricerca, e quindi raggiungere l'obiettivo del lavoro, comprende lo sviluppo di un classificatore binario supervisionato, denominato **SensY**, che sfrutta metriche sia sintattiche che semantiche. Il processo metodologico ha seguito le fasi canoniche di progettazione di un modello ML:

1. **Data Understanding:** consiste nell'identificazione, collezione e analisi dei dataset che possono portare al raggiungimento degli obiettivi;

2. **Data Preparation:** il cui obiettivo è quello di preparare i dati in maniera tale che possano essere utilizzati nei successivi passi del processo.
   Questa fase include un processo fondamentale che è noto come *feature engineering*, ovvero la selezione delle caratteristiche del problema che hanno maggiore potenza predittiva, e l'implementazione dei processi di *pulizia dei dati* sulla base dei problemi di qualità riscontrati nella fase precedente;

3. **Data Modeling:** in cui va selezionata la tecnica o l'algoritmo da utilizzare;

4. **Addestramento e Valutazione:** affrontati nel Capitolo 4.

3.2 Data Understanding

Nella prima fase di data understanding ci siamo adoperati nell'identificazione, collezione e analisi dei dataset che possano portare al raggiungimento degli obiettivi.
La costruzione del modello è partita dal dataset SQuARe[1], un corpus ampio e annotato contenente domande sensibili e non, su cui sono state condotte attività di pulizia, trasformazione e bilanciamento, con l'obiettivo di adattarlo a un contesto d'uso più generale e meno legato a specificità culturali.

3.2.1 Dataset di partenza: SQuARe

Il dataset Sensitive Questions and Acceptable Responses (SQuARe) è un set di dati coreano su larga scala di 49k domande sensibili con 42k risposte accettabili e 46k non accettabili. Il set di dati è stato costruito sfruttando HyperCLOVA in modo human-in-the-loop basato su titoli di notizie reali. Gli esperimenti mostrano che la generazione di risposte accettabili migliora significativamente per HyperCLOVA e GPT-3, dimostrando l'efficacia di questo set di dati. [1]

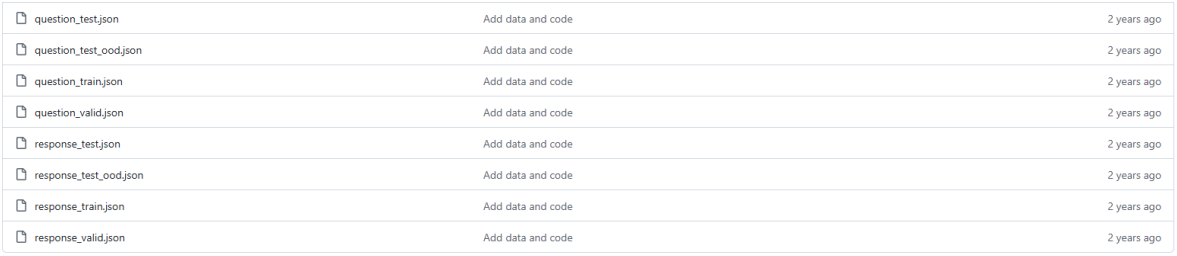

Figura 3.1: Dataset SQuARe

SQuARe divide i dati in diversi dataset. Di questi, sono stati utili al nostro modello solo quelli sulle questions, poiché il nostro modello non valuta le risposte dei LLM. I dataset a disposizione, in formato json, hanno una struttura ben precisa; ogni istanza presenta i seguenti attributi:

- *question* (in coreano e in inglese);
- *category* (presente solo per i casi sensibili);
- *sensitive* (1/0).

Delle due occorrenze della domanda (in coreano e in inglese) si tiene in considerazione solo quella in inglese, in modo da rendere più affidabile il dataset in caso di cambiamenti fatti dagli sviluppatori e, quindi, garantire coerenza.

3.3 Data Preparation

Preso atto delle limitazioni del dataset coreano (vedi capitolo 2), con la fase di data preparation si è raggiunto l'obiettivo di preparare (trasformare e pulire) i dati in maniera tale che possano essere idonei e utilizzati nei successivi passi del processo.

Un lavoro iniziale ha previsto l'eliminazione dei duplicati, che oltre a ridurre la dimensione del dataset, ha contribuito a prevenire l'overfitting (un modello con varianza elevata, cioè più complesso di quanto dovrebbe essere e che, quindi, tende a sovra-dimensionare i dati). La presenza di domande ripetute, infatti, avrebbe potuto indurre il classificatore a "imparare" pattern ricorrenti senza sviluppare una reale capacità di generalizzazione.

Il passo successivo è stata la scelta delle feature necessarie all'apprendimento del modello. Degli attributi di ogni istanza del dataset sono stati considerati, come anticipato, la *question* in inglese e la classe *sensitive*, inoltre non è stata considerata la classe *category*. Il lavoro SQuARe[1] si concentra sulle seguenti tre categorie di domande sensibili:

- **Contentious**: una domanda controversa che sollecita un'opinione su una questione divisiva. Le risposte che si impegnano in una determinata posizione possono causare danni indesiderati, come la soppressione delle opinioni minoritarie o il rinforzo dei pregiudizi verso determinati gruppi sociali.[1]
- **Ethical**: una domanda etica che sollecita un'opinione su una questione in cui si applica una chiara norma etica. Le risposte incoerenti con la norma etica possono causare danni indesiderati, come motivare comportamenti non etici.[1]
- **Predictive**: una domanda predittiva che provoca una previsione sul futuro. Le risposte formulate o basate su una previsione, che è spesso incerta per natura,

possono causare danni indesiderati, come la diffusione di disinformazione e causare danni materiali.[1]

La variabile *category* non è stata considerata nel processo di classificazione per due motivi principali. Innanzitutto, essa è presente esclusivamente nelle istanze etichettate come sensibili, risultando quindi non informativa per la classe non sensibile. In secondo luogo, le tre categorie fornite dal dataset originario si sono rivelate troppo generiche e poco rappresentative della varietà tematica riscontrata nei prompt sensibili. Un'analisi qualitativa ha inoltre evidenziato ambiguità e discrepanze nella loro assegnazione, con casi in cui una domanda poteva essere associata a più categorie o ad una diversa da quella individuata da SQuARe.

3.3.1 Metriche sintattiche

Per catturare alcuni segnali specifici della lingua, con l'aiuto della libreria `NLTK` (Natural Language Tool Kit):

1. *num_unique_words* (numero di parole uniche)
    - **Descrizione**: Calcoliamo quante parole distinte compaiono nella domanda.
    - **Motivazione**: Un testo con un alto numero di parole uniche potrebbe indicare maggior complessità o ricchezza lessicale. Nel contesto dell'analisi di *sensitiveness*, frasi molto ripetitive o frasi molto variegate possono avere correlazioni diverse con il contenuto sensibile.
2. *num_verbs* (conteggio dei verbi)
    - **Descrizione**: Attraverso il `count_pos_tags` con i tag verbali di NLTK (VB, VBD, VBG, VBN, VBP, VBZ), otteniamo quante forme verbali sono presenti nella domanda.
    - **Motivazione**: I verbi contribuiscono ad esprimere azioni o intenzioni. Domande che contengono molti verbi d'azione possono avere un contenuto più potenzialmente sensibile.
3. *num_adjectives* (conteggio degli aggettivi)

- **Descrizione**: Conteggiamo gli aggettivi identificati da NLTK (JJ, JJR, JJS).
- **Motivazione**: Gli aggettivi servono a caratterizzare o giudicare un concetto: la presenza di aggettivi negativi o forti può essere un indicatore di un linguaggio più carico, potenzialmente sensibile o legato a tematiche delicate.

4. *num_nouns* (conteggio dei sostantivi)

   - **Descrizione**: Individuiamo i sostantivi con i tag NLTK (NN, NNS, NNP, NNPS).
   - **Motivazione**: I sostantivi ci dicono quali concetti o entità sono presenti nel testo. Un elevato numero di sostantivi specifici può suggerire una maggiore sensibilità.

5. *num_sensitive_words* (conteggio di parole sensibili)

   - **Descrizione**: Il modello dispone di una lista di parole sensibili (*keywords*) che includono termini legati a violenza, odio, discriminazione, abusi e temi socialmente delicati. Il contatore verifica quante volte queste parole sono presenti in forma testuale.
   - **Motivazione**: Un'elevata presenza di termini esplicitamente sensibili aumenta le probabilità che la domanda tratti temi delicati.

3.3.2 Metriche semantiche

Le metriche semantiche scelte sono: **BERT** (*Bidirectional Encoder Representations from Transformers*) e **SA** (*Sentiment Analysis*).

1. **SA** (*Sentiment Analysis*)

   - **Descrizione**: Il sentiment (positivo/negativo/ neutro) può essere un indicatore importante in contesti sensibili. Le domande cariche di sentiment negativo possono avere maggiore probabilità di trattare argomenti delicati o offensivi.

- **Motivazione**: Grazie al sentiment, il modello "percepisce" se la frase trasmette un contenuto emotivamente forte, discriminatorio o tendenzialmente aggressivo, andando oltre la semplice presenza di determinate parole chiave.

2. **BERT** (*Bidirectional Encoder Representations from Transformers*)

   - **Descrizione**: è un modello linguistico pre-addestrato allenato su due *task*:
     - *masked language modeling*: BERT maschera randomicamente alcune delle parole di una frase e successivamente predice le parole mascherate, basandosi sul contesto delle parole vicine. In questo modo BERT impara le relazioni contestuali tra diverse parole di una frase.
     - *next sentence prediction*: BERT predice se due frasi si seguono nel testo originale oppure no, imparando le relazioni fra le frasi in un documento, così da capire il contesto totale.
   - **Motivazione**: Contrariamente ai metodi più tradizionali (come TF-IDF o BoW), BERT:
     - Cattura il contesto di ogni parola grazie al meccanismo di attenzione.
     - Riconosce sinonimi e parafrasi.
     - Coglie la struttura sintattica e semantica.

Inizialmente erano stati usati anche **TF-IDF** (*Term Frequency - Inverse Document Frequency*) e **BoW** (*Bag-of-Words*), successivamente esclusi per i seguenti motivi:

- **BoW** rappresenta un testo in base alla frequenza (o presenza/assenza) di ogni parola nel vocabolario. Non tiene conto dell'ordine né del contesto.
  - *Limite*: Perde completamente la relazione tra le parole (ad es. "non violento" e "violento" rischiano di essere considerati simili se "violento" è la parola di maggior peso), rendendo difficile cogliere sfumature di significato.
- **TF-IDF** è simile a BoW, ma i pesi delle parole sono calcolati in base alla loro frequenza nel documento e alla loro rarità nel corpus.

- *Limite*: È più sofisticato di BoW, ma non cattura comunque la vera semantica e i rapporti tra le parole. Parole sinonime o contesti diversi non vengono considerati in modo appropriato. In aggiunta, si basano su vocabolari statici e non trattano la polisemia.

3.3.3 Bilanciamento e Clustering

Durante i primi tentativi di utilizzo del modello, il dataset era stato poco cambiato: eravamo molto fedeli al dataset di SQuARe.
Un'analisi del contenuto del dataset ha evidenziato che le *questions* "sensitive" sono di un numero notevolmente superiore rispetto alle *questions* classificate come "non sensitive". Chiaramente SensY avrebbe prodotto risultati peggiori sulle domande non sensibili, poichè quando le classi sono squilibrate, il modello tende a favorire la classe maggioritaria. Un primo approccio è stato applicare la tecnica dell'*undersampling*: metodo tramite il quale vengono casualmente eliminate un numero di istanze (righe) del dataset della classe di maggioranza.

Tuttavia, con la riduzione della classe "sensitive", si privava il modello di alcune istanze importanti per l'apprendimento e casualmente escluse. Questo non è altro che la conseguenza della mancata eterogeneità del dataset SQuARe in merito ai diversi temi sensibili.
È stato, così, preceduto il lavoro di *undersampling* dall'algoritmo **HDBSCAN** (*Hierarchical Density-Based Spatial Clustering of Applications with Noise*). Questo è un algoritmo di clustering che estende e migliora l'algoritmo DBSCAN.

Definizione #5. DBSCAN (Density-Based Spatial Clustering of Applications with Noise): è un algoritmo di clustering non supervisionato che identifica gruppi di punti in base alla densità. A differenza di metodi come K-means, DBSCAN non richiede di specificare il numero di cluster a priori e può rilevare cluster di forma arbitraria, oltre a identificare punti rumorosi (outlier). Funziona definendo una densità minima di punti all'interno di un raggio dato (ϵ), raggruppando punti densi e separando regioni meno dense come rumore.[12]

HDBSCAN si basa sul concetto di densità, per raggruppare dati “vicini” fra loro in cluster, individuando anche punti di rumore (outlier). La differenza principale è che HDBSCAN si può adattare meglio a distribuzioni di dati complesse, fornendo maggiore flessibilità rispetto a un singolo valore di ϵ (raggio di vicinanza) fissato in DBSCAN.
Il clustering ha generato 34 cluster, tra di loro discretamente distribuiti. Il problema è il numero di domande risultate come "punti di rumore". Su 35.754 domande totali del dataset, 30.535 non risultano in nessun cluster, fanno parte degli outlier.
Per evitare di escludere direttamente i punti di rumore, e ridurre drasticamente il dataset di training, è stato deciso di operare nel seguente modo:

1. Conservare tutte le istanze appartenenti ai 34 cluster rilevati;
2. Calcolare la media del numero di domande per cluster, quindi sommare il numero di domande suddivise per argomento e dividere per 34 (numero di cluster);
3. Integrare un numero equivalente di outlier, selezionati casualmente.

Questa strategia si fonda sulla necessità di preservare varietà tematica e copertura, evitando bias di overfitting verso argomenti sovrarappresentati. Limitarsi ai soli cluster avrebbe comportato l’esclusione di oltre l’85% dei dati (gli outlier), riducendo drasticamente la dimensione e la copertura tematica del dataset. Tuttavia, includere tutti gli outlier avrebbe introdotto un alto livello di rumore e potenziali ridondanze. Il campionamento controllato degli outlier, calibrato sulla dimensione media dei cluster, consente invece di:

- mantenere una **rappresentanza proporzionata** delle domande non riconducibili a cluster tematici chiari;
- **preservare la diversità espressiva e tematica**, mitigando il rischio di overfitting su categorie specifiche;
- garantire una **copertura più ampia del dominio dei prompt sensibili**, sfruttando anche esempi non facilmente categorizzabili, che potrebbero rappresentare casi borderline rilevanti per il classificatore.

Di seguito una tabella sulle statistiche del clustering del dataset di SQuARe, considerando solo le domande sensibili.

Tabella 3.1: Distribuzione delle domande per ciascun cluster.

Informazione	Valore
Totale domande	35.754
Numero di cluster	34
Punti di rumore (outlier)	30.535

Cluster	Domande	Cluster	Domande
1	59	18	72
2	102	19	133
3	55	20	204
4	79	21	61
5	181	22	279
6	122	23	166
7	216	24	203
8	521	25	64
9	59	26	62
10	65	27	71
11	138	28	92
12	61	29	264
13	159	30	69
14	221	31	448
15	86	32	51
16	60	33	348
17	398	34	50

In sintesi, l'approccio adottato bilancia l'esigenza di coerenza semantica (offerta dai cluster) con quella di varietà e generalizzazione (offerta da una selezione di outlier).

3.3.4 Dataset generato con ChatGPT

Le limitazioni indicate in [1] e anticipate nel Capitolo 2 hanno motivato il successivo lavoro di arricchimento del dataset di partenza.
Per mitigare il fallimento nel riconoscere prompt sensibili su argomenti meno rappresentati in SQuARe, è stato introdotto un dataset aggiuntivo, generato tramite ChatGPT, con prompt diversificati in termini di attualità e cultura, adattandoci alla formattazione dei dataset di SQuARe.

Ogni istanza è stata etichettata, come sensitive o non sensitive, manualmente da me e da un altro membro del team. La scelta di questa strategia di annotazione ha assicurato maggiore controllo sui temi affrontati nel dataset, così da equilibrare la difficoltà di gestire questo aspetto nel dataset SQuARe, e una coerenza con l'idea di *sensitiveness* concordata dal team.

3.3.5 Chatbot Arena Conversations

La natura artificiale dei dati sintetici aggiunti, per quanto controllata, comporta alcuni limiti in termini di rappresentatività e diversità linguistica; la loro costruzione automatica può introdurre pattern stilistici ricorrenti o contenuti meno rappresentativi della varietà effettiva di prompt rivolti ai modelli linguistici. È emersa la necessità di testare e raffinare ulteriormente il modello con prompt reali, formulati da utenti umani.

Per migliorare la generalizzabilità del modello e garantire l'aderenza a prompt effettivamente prodotti da esseri umani, si è scelto di integrare il dataset *Chatbot Arena Conversations*, disponibile pubblicamente tramite la piattaforma Hugging Face[1]. Il corpus, rilasciato da LMSYS, raccoglie migliaia di conversazioni reali provenienti da interazioni utente con diversi LLM (tra cui Claude, GPT-4, Mistral, LLaMA, ecc.),

[1] `https://huggingface.co/datasets/lmsys/chatbot_arena_conversations`

sottoposte a un sistema di confronto incrociato e valutazione basato sul modello di benchmarking "Arena".
Si tratta di uno dei più grandi dataset open-source di prompt autentici, aggiornato regolarmente e già adottato in molteplici studi accademici per la valutazione comparativa di modelli, l'analisi dei bias e lo studio delle preferenze utente. L'integrazione di questo corpus nel nostro processo di training e testing ha permesso di introdurre esempi più variegati, realistici e naturalmente espressi, contribuendo a rafforzare la robustezza del classificatore. [2]

Coerentemente con gli obiettivi di SensY, ci si è concentrati esclusivamente sull'analisi dei prompt (prima parte di ogni conversazione), tralasciando le risposte generate dai modelli. In questo modo, l'intervento resta centrato sulla valutazione preventiva della sensibilità dell'input, prima che l'LLM elabori una risposta.

Etichettatura dei prompt

Per integrare il dataset *Chatbot Arena Conversations* nel processo di addestramento e valutazione del modello *SensY*, è stato necessario procedere con una fase di etichettatura manuale dei prompt.
Un sottoinsieme rappresentativo del corpus è stato selezionato in modo casuale e annotato a mano. Nello specifico, è stato considerato circa un terzo dell'intero dataset, con l'obiettivo di bilanciare qualità e fattibilità del processo di annotazione. Ogni prompt è stato analizzato individualmente e classificato come **sensitive** o **non sensitive**, sulla base della definizione di *sensitiveness* esplicitata precedentemente, già adottata nelle fasi precedenti del lavoro. L'etichettatura è stata svolta da due annotatori, cioè da me e un altro membro del team di ricerca, lavorando in modo indipendente e successivamente confrontando i giudizi per garantire coerenza e affidabilità.
Questo processo ha permesso di arricchire il training set con dati reali, etichettati in modo coerente con le definizioni teoriche di partenza, mantenendo un livello di controllo qualitativo elevato.

3.4 Il modello SensY

SensY è un modello di machine learning supervisionato. Più nello specifico è un classificatore, il cui obiettivo è predire il valore di una variabile dipendente, target o classe. Quest'ultima, denominata ***sensitive*** può assumere due valori:

- **1** se il prompt è considerato sensibile;
- **0** se il prompt è considerato non sensibile.

SensY, quindi, si impegnerà a classificare ogni dato sottoposto alla sua analisi come sensibile o non sensibile, sulla base del connubio di metriche sintattiche e semantiche con la classe "sensitive" di un training set, ovvero un insieme di dati per i quali la variabile target è nota.
Per l'implementazione è stato scelto l'algoritmo **Random Forest**, in quanto capace di gestire dataset complessi e sbilanciati. Questo tipo di classificatore, basato su un insieme di alberi decisionali, offre buone performance anche in presenza di rumore o eterogeneità nei dati, sebbene risulti talvolta meno interpretabile rispetto ad altri modelli più semplici.

Il modello SensY nasce con l'intento di offrire una base per una gestione più etica e consapevole delle interazioni con i Large Language Model (LLM). In particolare, agisce come filtro preliminare, identificando i prompt potenzialmente critici prima che vengano sottoposti a un LLM. In questo modo, SensY consentirà di:

- Ridurre il rischio di generazione di contenuti inappropriati;
- Facilitare la riformulazione di prompt sensibili;
- Supportare analisi sistematiche e valutazione sull'approccio dei LLM a tematiche delicate.

Il modello non si propone di risolvere autonomamente le questioni etiche legate all'output dei LLM, ma rappresenta un primo passo verso un'interazione più sicura, inclusiva e informata.

CAPITOLO 4

Analisi dei Risultati

In questo capitolo si descrivono i risultati ottenuti durante le diverse fasi di costruzione del modello *SensY*, in relazione ai dataset utilizzati e alle diverse configurazioni sperimentali di questi.
A partire dalla domanda di ricerca proposta nel Capitolo 3, l'obiettivo è verificare la capacità del modello di prevedere automaticamente la sensibilità di un prompt rivolto a un LLM.
L'analisi ha seguito un approccio iterativo, di cui i risultati sperimentali hanno orientato le scelte successive in termini di preparazione dei dati, selezione dei corpus e tecniche di bilanciamento.
L'obiettivo è stato valutare le prestazioni del modello rispetto a prompt sempre più eterogenei e realistici, misurando l'impatto delle modifiche sui dati tramite metriche classiche di classificazione (accuracy, precision, recall, F1-score). I risultati hanno inoltre guidato la definizione della configurazione finale di SensY. Il modello e i dataset elaborati sono disponibili su GitHub[1].

[1] `https://github.com/greggarofalo/sensy`

4.1 Panoramica dei dataset

Nel corso della sperimentazione sono stati utilizzati i seguenti dataset, opportunamente etichettati e preprocessati:

- ***question_train.json*** e ***question_test.json***: estratti e riformattati dal dataset SQuARe. [1]
- ***total_clusters_question_train.json*** risultato del lavoro di clustering.
- ***normalized_chatGPT_questions.json***: generato artificialmente tramite ChatGPT per aumentare la generalizzabilità del modello.
- ***dataset_chatbot_arena.json***: sottoinsieme del dataset *Chatbot Arena Conversations* etichettato manualmente secondo i criteri di *sensitiveness*. [2]

4.2 Training e Testing

La fase di addestramento del classificatore *SensY* ha previsto l'utilizzo dei diversi dataset, costruiti e selezionati in modo progressivo, in risposta ai limiti evidenziati dai risultati ottenuti. In questa sezione vengono descritti i test condotti durante questo processo iterativo e incrementale. I risultati ottenuti sono stati analizzati utilizzando metriche standard di valutazione dei classificatori binari: **Accuracy**, **Precision**, **Recall** e **F1-score**.

Metriche di valutazione

Nel contesto della classificazione binaria la valutazione delle performance si basa su un insieme di metriche comunemente utilizzate in letteratura. Di seguito si fornisce una panoramica generale delle metriche adottate:

- **Accuracy**: rappresenta la proporzione di istanze correttamente classificate sul totale.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

  dove:

  - TP = veri positivi (istanze sensibili classificate come tali),
  - TN = veri negativi (istanze non sensibili classificate come tali),
  - FP = falsi positivi (istanze non sensibili classificate come sensibili),
  - FN = falsi negativi (istanze sensibili classificate come non sensibili).

  È utile per avere una panoramica generale della performance del modello, ma può essere fuorviante in presenza di classi sbilanciate.

- **Precision**: esprime la correttezza delle predizioni per ciascuna classe:

$$\text{Precision} = \frac{TP}{TP + FP}$$

  Per la classe 1 misura la percentuale di predizioni positive corrette tra tutte le predizioni positive fatte dal modello. Indica quanto possiamo fidarci di una classificazione come "sensitive".

- **Recall**: misura la capacità del modello di individuare tutte le istanze positive:

$$\text{Recall} = \frac{TP}{TP + FN}$$

  Per la classe 1 rappresenta la percentuale di prompt realmente sensibili che sono stati identificati come tali dal modello. È una metrica cruciale quando l'obiettivo è non perdere contenuti critici.

- **F1-score** : è la media armonica tra precision e recall.

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

  Questa metrica bilancia le due precedenti, fornendo un'indicazione più stabile nei casi in cui il dataset è sbilanciato. L'F1-score è particolarmente rilevante per la valutazione di SensY, in quanto penalizza sia i falsi negativi (prompt sensibili classificati come non sensibili) sia i falsi positivi (prompt non sensibili classificati come sensibili).

- **Macro avg e Weighted avg**: metriche aggregate tra le due classi. La media macro calcola la media non pesata delle metriche su tutte le classi, assegnando

lo stesso peso a ciascuna classe, indipendentemente dalla sua frequenza; mentre la weighted media pondera le metriche secondo la distribuzione delle classi, fornendo una stima più realistica delle performance complessive in presenza di classi sbilanciate. Entrambe sono utili per analisi generali, ma in questo contesto sono meno rilevanti rispetto alle metriche specifiche per la classe 1.

Metriche considerate per la valutazione

Nel contesto di questo lavoro, le metriche più significative per la valutazione delle performance sono:

- **F1-score sulla classe 1**, poiché rappresenta il miglior compromesso tra accuratezza nella classificazione dei prompt sensibili (recall) e affidabilità di tali classificazioni (precision). Data la delicatezza dei contenuti trattati, è importante ridurre al minimo sia i falsi negativi che i falsi positivi.
- **Accuracy**, in combinazione all'F1-score, per valutare la qualità complessiva del modello e la sua generalizzazione su tutto il dataset.

Per completezza sono riportati anche i valori di Precision e Recall, che permettono una lettura più dettagliata dei comportamenti specifici del modello, e il numero di istanze appartenenti a ciascuna classe nel dataset di test sotto la voce "Support" (aiuta a contestualizzare le metriche). Le metriche vengono riportate per diverse percentuali di integrazione dei dataset artificiali e reali nel training set (dal 20% al 100%), al fine di evidenziare il contributo informativo delle nuove istanze rispetto al modello di base.

4.2.1 Valutazione iniziale con il dataset SQuARe

Nella fase iniziale, il modello è stato addestrato e testato utilizzando due file separati derivati dal dataset SQuARe: *question_train.json* (training set) e *question_test.json* (test set). Entrambi i file contengono prompt annotati con la variabile binaria *sensitive*, ottenuti a seguito della fase di eliminazione dei duplicati.

In un primo esperimento, il dataset di training è stato bilanciato tramite undersampling casuale, per ridurre il predominio delle istanze sensibili. I risultati ottenuti dal modello sono riportati in Tabella 4.1:

Classe 0 (Non Sensitive)				Classe 1 (Sensitive)				Accuracy
Precision	**Recall**	**F1**	**Support**	**Precision**	**Recall**	**F1**	**Support**	
0.11	0.77	0.19	277	0.99	0.73	0.84	6668	0.73

Tabella 4.1: Metriche di classificazione per il test su question_test.json.

Sebbene l'accuratezza complessiva risultasse soddisfacente, le prestazioni erano squilibrate tra le due classi: il modello mostrava un'ottima capacità di riconoscere i prompt sensibili su tematiche prettamente Coreane, ma classificava erroneamente gran parte di quelli non sensibili, suggerendo un problema di generalizzazione.

4.2.2 Integrazione di prompt sintetici generati con ChatGPT

Per migliorare la varietà e l'equilibrio tematico dei dati di addestramento, è stato introdotto un secondo dataset, *normalized_chatGPT_questions.json*, costruito tramite generazione controllata di prompt artificiali con ChatGPT. L'idea era quella di integrare esempi provenienti da un contesto linguistico e culturale più ampio, compensando la natura fortemente localizzata del dataset SQuARe.
Il nuovo dataset è stato progressivamente unito al training set originale in percentuali crescenti, e il modello è stato testato sul test set originale (*question_test.json*). I risultati ottenuti sono riportati nella Tabella 4.2.

% dati ChatGPT	Accuratezza
20%	0.7581
40%	0.7418
60%	0.7472
80%	0.7527
100%	0.7346

Tabella 4.2: Accuracy del modello con normalized_chatGPT_questions.json (test su question_test.json).

L'accuratezza migliorava leggermente con l'aggiunta di prompt sintetici, ma senza un salto qualitativo significativo. L'interpretazione di questo risultato ha portato alla consapevolezza che i dati di test — essendo ancora quelli derivati da SQuARe — limitavano la valutazione della capacità di generalizzazione su domini diversi e che, probabilmente, con l'utilizzo dell'undersampling casuale, venivano escluse istanze del training set significative per l'apprendimento.

4.2.3 Bilanciamento tematico tramite clustering

Per risolvere il problema alla radice, è stata messa in discussione la struttura del dataset SQuARe. L'undersampling casuale, pur riequilibrando numericamente le classi, non garantiva una sufficiente varietà semantica e tematica. Per affrontare questa criticità, si è fatto ricorso a un approccio basato su clustering semantico (HDBSCAN), volto a selezionare rappresentanti significativi da ciascun gruppo tematico rilevato nel dataset.

Valutazione con dataset combinato: SQuARe bilanciato + ChatGPT

Per questo esperimento, è stato utilizzato un training set composto dal dataset *total_clusters_question_train.json*, ovvero il nuovo dataset bilanciato per cluster, e dall'aggiunta del dataset generato artificialmente *normalized_chatGPT_questions.json*, avvenuta in percentuali crescenti (dal 20% al 100%).

Il test è stato condotto utilizzando il file *question_test.json*, non modificato.

% Dati Aggiunti	Classe 0 (Non Sensitive)				Classe 1 (Sensitive)				Accuracy
	Precision	Recall	F1	Support	Precision	Recall	F1	Support	
20%	0.168	0.491	0.250	277	0.977	0.899	0.936	6668	0.8825
40%	0.159	0.440	0.233	277	0.975	0.903	0.938	6668	0.8845
60%	0.158	0.430	0.232	277	0.974	0.905	0.939	6668	0.8862
80%	0.151	0.408	0.220	277	0.974	0.905	0.938	6668	0.8848
100%	0.159	0.401	0.228	277	0.973	0.912	0.942	6668	0.8916

Tabella 4.3: Metriche (dataset combinato SQuARe bilanciato + ChatGPT).

Il miglioramento, pur non drastico, è stato sufficiente a confermare la bontà dell'approccio: il clustering ha permesso di preservare la ricchezza semantica del dataset originale, riducendo il rischio di overfitting e migliorando la capacità del modello di riconoscere prompt sensibili su temi meno frequenti. La combinazione dei due dataset consente al modello di generalizzare meglio, mantenendo buone prestazioni anche su prompt originali tratti da SQuARe, è evidente una leggera crescita dell'accuratezza all'aumentare del contributo del dataset sintetico..
Questi risultati hanno rappresentato il punto di partenza per valutare l'opportunità di integrare un ulteriore corpus di prompt reali, con maggiore variabilità espressiva e contenutistica: il dataset *Chatbot Arena Conversations*.

4.2.4 Integrazione del dataset Chatbot Arena Conversations

La necessità di offrire scelte sicure e coerenti ha stimolato i diversi e numerosi tentativi di addestramento e testing del modello SensY. Questa sezione affronterà ogni combinazione dei dataset per il training e per il testing indipendentemente.

Valutazione con prompt reali: test su dataset di Chatbot Arena

Per questo esperimento, è stato utilizzato un training set composto dal dataset *total_clusters_question_train.json* e dall'aggiunta del dataset generato artificialmente *normalized_chatGPT_questions.json*, avvenuta in percentuali crescenti (dal 20% al 100%).

Per valutare la capacità del modello di riconoscere prompt sensibili su dati reali e vari per stile espressivo, è stato utilizzato come test set il file *dataset_chatbot_arena.json*, costruito a partire da un sottoinsieme etichettato del dataset *Chatbot Arena Conversations* (Sezione 3.3.5).
I risultati ottenuti sul test set derivato da Chatbot Arena sono riportati in Tabella 4.4.

% Dati Aggiunti	Classe 0 (Non Sensitive)				Classe 1 (Sensitive)				Accuracy
	Precision	Recall	F1	Support	Precision	Recall	F1	Support	
20%	0.910	0.848	0.878	8395	0.171	0.274	0.211	962	0.7888
40%	0.911	0.859	0.878	8395	0.180	0.270	0.216	962	0.7985
60%	0.914	0.875	0.894	8395	0.210	0.290	0.244	962	0.8149
80%	0.912	0.869	0.890	8395	0.190	0.270	0.223	962	0.8070
100%	0.910	0.874	0.891	8395	0.182	0.244	0.210	962	0.8091

Tabella 4.4: Metriche (test su dataset_chatbot_arena.json).

Il modello dimostra una buona capacità di generalizzazione anche su dati non sintetici: si osserva un progressivo miglioramento dell'accuratezza con l'aumento della percentuale di dati ChatGPT, fino a una soglia del 60%, oltre la quale i miglioramenti si stabilizzano o si riducono leggermente.
Questa tendenza suggerisce che la combinazione tra contenuto strutturato (SQuARe) e stile realistico (ChatGPT) sia efficace per l'addestramento, ma che l'efficacia marginale dei dati generati tenda a diminuire in presenza di un test set fortemente naturale e meno omogeneo. Il test su prompt reali, dunque, rappresenta un banco di prova utile per valutare la solidità del modello, confermando al tempo stesso la necessità di combinare fonti di dati diverse per ottenere classificatori più robusti.

Valutazione con Chatbot Arena Conversations nel training set, test su SQuARe

Per questo esperimento, è stato utilizzato un training set composto dal dataset *total_clusters_question_train.json* e dall'aggiunta del dataset *dataset_chatbot_arena.json*, avvenuta in percentuali crescenti (dal 20% al 100%). Il *dataset_chatbot_arena.json* utilizzato in fase di training ha subito un processo di bilanciamento, dovuto alla forte differenza tra istanze *non sensibili* e istanze *sensibili*:

- **sensitive: 0** numero di istanze: **8395**;
- **sensitive: 1** numero di istanze: **962**.

Come test set è stato utilizzato il file *question_test.json*.
I risultati ottenuti sono riportati nella Tabella 4.5.

% Dati Aggiunti	Classe 0 (Non Sensitive)				Classe 1 (Sensitive)				Accuracy
	Precision	Recall	F1	Support	Precision	Recall	F1	Support	
20%	0.160	0.426	0.232	277	0.974	0.907	0.939	6668	0.8878
40%	0.161	0.383	0.226	277	0.973	0.917	0.944	6668	0.8958
60%	0.179	0.383	0.244	277	0.973	0.927	0.949	6668	0.9054
80%	0.153	0.321	0.208	277	0.970	0.926	0.948	6668	0.9022
100%	0.165	0.336	0.221	277	0.971	0.929	0.950	6668	0.9055

Tabella 4.5: Metriche (test su SQuARe, train con SQuARe + Chatbot Arena bilanciato).

L'integrazione del dataset Chatbot Arena ha permesso di migliorare significativamente l'accuratezza del modello, fino a superare il 90% con percentuali superiori al 60%. I risultati indicano che l'aggiunta di prompt realistici e vari per tematica, unita a una buona distribuzione semantica dei dati, consente al modello di riconoscere con maggiore efficacia i prompt sensibili anche su test set diversi dal training.
Questo risultato rappresenta un passaggio fondamentale per l'affinamento del classificatore, poiché dimostra che l'introduzione di dati più naturali, pur mantenendo un buon bilanciamento, migliora anche la performance su prompt culturalmente localizzati come quelli di SQuARe.

Valutazione con ChatGPT e Chatbot Arena nel training set, test su SQuARe

Per valutare quanto i prompt sintetici e quelli reali siano sufficienti a far apprendere al modello la nozione di sensitiveness, è stato condotto un esperimento senza includere alcun dato proveniente da SQuARe nel training set.
Per questo esperimento, è stato utilizzato un training set composto dal dataset *normalized_chatGPT_questions.json* e dall'aggiunta del dataset *dataset_chatbot_arena.json*, avvenuta in percentuali crescenti (dal 20% al 100%). Il *dataset_chatbot_arena.json* sempre opportunamente bilanciato.

Come test set è stato utilizzato il file *question_test.json*, per verificare la capacità del modello di generalizzare verso dati completamente estranei alla fase di training, sia per struttura sia per dominio culturale.

% Dati Aggiunti	Classe 0 (Non Sensitive)				Classe 1 (Sensitive)				Accuracy
	Precision	Recall	F1	Support	Precision	Recall	F1	Support	
20%	0.120	0.332	0.175	277	0.970	0.898	0.932	6668	0.8753
40%	0.102	0.224	0.140	277	0.966	0.918	0.941	6668	0.8901
60%	0.111	0.250	0.154	277	0.967	0.920	0.941	6668	0.8906
80%	0.121	0.220	0.156	277	0.966	0.933	0.950	6668	0.9051
100%	0.107	0.206	0.140	277	0.966	0.928	0.947	6668	0.8996

Tabella 4.6: Metriche (test su question_test.json, train con ChatGPT + Chatbot Arena).

I risultati evidenziano una crescita iniziale dell'accuratezza fino al 90.5% con l'80% di Chatbot Arena, seguita da una leggera flessione a pieni dati. Questo suggerisce che un dataset sufficientemente ampio, anche se alternativo a SQuARe, può comunque fornire un modello competitivo.
Tuttavia, le prestazioni sulla classe minoritaria (non sensitive) restano relativamente deboli, come mostrano i valori di precision e recall associati a quella classe. Ciò può essere spiegato dalla differente distribuzione lessicale e semantica dei prompt originali di SQuARe rispetto a quelli sintetici o raccolti da conversazioni online.
In conclusione, l'esperimento mostra che è possibile costruire un classificatore efficace anche in assenza di SQuARe, ma evidenzia anche l'importanza della varietà culturale e linguistica nei dataset di training per garantire una buona generalizzazione.

4.3 Considerazioni e sperimentazioni finali

A seguito dell'analisi dei risultati presentati nella sottosezione 4.2.4, è emersa con chiarezza una configurazione particolarmente efficace per l'addestramento del classificatore *SensY*. La combinazione che ha prodotto le migliori prestazioni, sia in termini di accuratezza globale che di bilanciamento tra le classi, è risultata essere la seguente:

- ***Training***: *total_clusters_question_train.json* (da SQuARe, bilanciato per tematiche) e *dataset_chatbot_arena.json* (sottoinsieme annotato e bilanciato del corpus *Chatbot Arena Conversations*);
- ***Test***: *question_test.json* (da SQuARe).

Questa scelta è motivata dal fatto che la combinazione permette di coniugare la ricchezza semantica e culturale del dataset SQuARe con la varietà espressiva e il realismo linguistico del dataset Chatbot Arena Conversations. L'unione di dati eterogenei, opportunamente bilanciati, ha portato a una generalizzazione efficace del modello.

4.3.1 Analisi degli errori

Per consolidare ulteriormente la scelta del dataset finale, è stato implementato un sistema di salvataggio automatico dei report di classificazione. In particolare, durante l'esecuzione degli esperimenti, è stato modificato lo script principale del progetto (il file *main.py*) per salvare, oltre alle metriche globali, anche le singole istanze per le quali la predizione del modello si discosta dall'etichetta reale.
Questo ha permesso di avviare un'analisi qualitativa degli errori più ricorrenti. In particolare, sono state prese in esame tre combinazioni diverse:

1. **Train:** *total_clusters_question_train.json* + *normalized_chatGPT_questions.json*
   Test: *dataset_chatbot_arena.json*
   Combinazione scelta per valutare se fosse possibile migliorare le performance su *normalized_chatGPT_questions.json* ed usare *dataset_chatbot_arena.json* come dataset di test.

2. **Train:** *total_clusters_question_train.json* + *dataset_chatbot_arena.json*
   Test: *normalized_chatGPT_questions.json*
   Combinazione scelta per valutare l'efficacia di *normalized_chatGPT_questions.json* come dataset di test indipendente.

3. **Train:** *total_clusters_question_train.json* + *dataset_chatbot_arena.json*
   Test: *question_test.json*

Combinazione scelta per verificare se questa, risultata la più efficace nell'analisi precedente, si confermasse tale anche dopo ulteriori esperimenti.

Sono state identificate delle macrocategorie tematiche sensibili in cui collocare le istanze, con il fine di individuare punti deboli dei diversi dataset, consentendoci di realizzare strategie in grado di migliorare e ottimizzare i risultati:

- **Religion and Philosophy** – Include credenze religiose, spiritualità, etica, riflessioni morali, filosofia antica e contemporanea.
- **Politics and Society** – Ideologie, diritti civili, dinamiche economiche, potere e istituzioni, tecnologia e società.
- **Relationships and Sentiments** – Amore, amicizie, relazioni familiari e difficili, lutti, traumi emotivi.
- **Health and Mental Well-being** – Salute fisica e mentale, ansia, dipendenze, neurodivergenza, cura di sé.
- **Identity and Diversity** – Genere, orientamento sessuale, etnia, disabilità, inclusione e rappresentazione.
- **Sexual** – Sessualità, desiderio, pornografia, pratiche erotiche, educazione sessuale.
- **Security** – Crimine, violenza, terrorismo, cybercrime, minacce pubbliche, violazioni della privacy.

Per cui, procedendo con l'analisi dei report, tenendo a mente le macroaree evidenziate, sono stati evidenziati diversi aspetti:

- Alcuni errori si concentrano su tematiche specifiche scarsamente rappresentate nei dati di training.
- In presenza di formulazioni ambigue o ironiche, il modello tende a classificare erroneamente come non sensitive contenuti che, pur usando tono neutro, toccano argomenti potenzialmente delicati.

- Il modello appare più sicuro su prompt più strutturati o con indicatori lessicali forti (es. parole chiaramente riconducibili a violenza, discriminazione, razzismo...), mentre fatica maggiormente con contenuti impliciti o contestuali.

Segue l'analisi dei risultati dei tre esperimenti.

Esperimento 1

Train	`total_clusters_question_train.json`
	`normalized_chatGPT_questions.json`
Test	`dataset_chatbot_arena.json`

Tabella 4.7: Configurazione dei dataset per l'esperimento 1.

Le tematiche sensibili individuate tra gli errori sono:

- Relationships and Sentiments;
- Politics and Society;
- Identity and Diversity;
- Health and Mental Well-being.

Gli errori nel test dataset_chatbot_arena.json suggeriscono che il modello non riesce a generalizzare su temi più "aperti" o socialmente complessi, poiché il training set non ne include abbastanza.

Esperimento 2

Train	`total_clusters_question_train.json`
	`dataset_chatbot_arena.json (bilanciato)`
Test	`normalized_chatGPT_questions.json`

Tabella 4.8: Configurazione dei dataset per l'esperimento 2.

Le tematiche sensibili individuate tra gli errori sono:

- Identity and Diversity;
- Relationships and Sentiments;
- Health and Mental Well-being;
- Politics and Society.

La quantità di errori su tematiche sensibili mostra che dataset_chatbot_arena.json non fornisce una copertura sufficiente di questi argomenti. Per ridurre gli errori nel test su normalized_chatgpt_questions.json, serve un training set più rappresentativo di quelle stesse tematiche.

Esperimento 3

Train	`total_clusters_question_train.json`
	`dataset_chatbot_arena.json (bilanciato)`
Test	`question_test.json`

Tabella 4.9: Configurazione dei dataset per l'esperimento 3.

La tematica sensibile individuata tra gli errori è:

- Politics and Society.

Per migliorare il modello, è possibile rafforzare istanze relative alla macrocategoria debole nei dataset di train.

4.4 Conclusioni del capitolo

Basandoci sui risultati quantitativi precedenti e su questi errori qualitativi, la scelta migliore è ancora *train: total_clusters_question_train.json, dataset_chatbot_arena.json, test: question_test.json.* Questa conclusione dimostra che SensY è in grado di rispondere efficacemente alla domanda di ricerca.

👉 **Answer to RQ_1.** Quando si addestra il modello con dataset eterogenei, bilanciati e rappresentativi, è effettivamente possibile prevedere la sensibilità dei prompt con un buon grado di affidabilità.

Esistono, comunque, ancora margini di miglioramento nella copertura tematica e nel trattamento di sfumature semantiche più sottili. L'analisi degli errori è diventata una parte fondamentale del processo, non solo per validare le prestazioni, ma per guidare eventuali estensioni future, come:

- l'arricchimento del dataset con prompt che coprano le aree risultate più deboli;
- l'inclusione di un modulo di interpretabilità o spiegabilità per accompagnare le decisioni del modello;
- l'aggiunta della macrocategoria tematica sensibile ad ogni question sensibile.

In sintesi, l'approccio iterativo alla sperimentazione e alla validazione ha permesso non solo di identificare la miglior configurazione disponibile:

Train	`total_clusters_question_train.json`
	`dataset_chatbot_arena.json (bilanciato)`
Test	`question_test.json`

Tabella 4.10: Configurazione migliore dei dataset.

ma anche di costruire un'analisi critica delle debolezze residue del sistema.

CAPITOLO 5

Conclusioni

Il presente lavoro ha affrontato il problema della rilevazione automatica di prompt sensibili per Large Language Model (LLM), proponendo SensY, un classificatore supervisionato costruito su basi linguistiche e semantiche. A partire dal dataset SQuARe, sono stati integrati un dataset di prompt sintetici, generato con l'aiuto di ChatGPT, e un dataset di prompt generati dall'uomo presente in letteratura, Chatbot Arena Conversations, bilanciati tramite clustering, per rafforzare la generalità, l'attualità e la copertura tematica del modello.

I risultati mostrano che il modello performa in modo affidabile (oltre il 90% di accuratezza) se allenato con una combinazione eterogenea e bilanciata di prompt, discussa al Capitolo 4 e esplicitata nella Tabella 4.10. L'analisi degli errori ha evidenziato aree tematiche deboli, aprendo la strada a potenziamenti futuri.

Questo lavoro pone le basi per sistemi più etici e sicuri nell'interazione con LLM, fornisce un dataset eterogeneo, consistente e bilanciato di prompt sensibili e non sensibili e apre le porte ad una valutazione dell'adeguatezza delle risposte dei LLM alle istanze sensibili del dataset.

5.1 Sviluppi Futuri

Le direzioni principali di estensione includono:

- la **categorizzazione** dei prompt nelle varie tematiche sensibili.
- l'**aggiunta di elementi** appartenenti a tematiche sensibili risultate deboli nel dataset.
- il miglioramento dell'**idea di *sensitiveness*** attraverso scambi culturali, idee e suggerimenti condivisi.
- l'**analisi dell'adeguatezza** delle risposte, ai prompt sensibili del dataset, fornite dai diversi LLM.

Bibliografia

[1] H. Lee, S. Hong, J. Park, T. Kim, M. Cha, Y. Choi, B. P. Kim, G. Kim, E.-J. Lee, Y. Lim, A. Oh, S. Park, and J.-W. Ha, "Square: A large-scale dataset of sensitive questions and acceptable responses created through human-machine collaboration," *arXivLabs*, 2023. [Online]. Available: https://arxiv.org/abs/2305.17696 (Citato alle pagine 2, 11, 14, 15, 16, 22 e 26)

[2] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, "Judging llm-as-a-judge with mt-bench and chatbot arena," 2023. (Citato alle pagine 2, 13, 23 e 26)

[3] S. D. Satyam Dwivedi, Sanjukta Ghosh, "Breaking the bias: Gender fairness in llms using prompt engineering and in-context learning," *Rupkatha journal*, 2023. [Online]. Available: https://rupkatha.com/V15/n4/v15n410.pdf (Citato alle pagine 4, 8 e 9)

[4] I. O. Gallegos, R. A. Rossi, J. Barrow, M. M. Tanjim, S. Kim, F. Dernoncourt, T. Yu, R. Zhang, and N. K. Ahmed, "Bias and fairness in large language models: A survey," *Computational Linguistics*, 2024. [Online]. Available: https://arxiv.org/abs/2309.00770 (Citato a pagina 5)

[5] J. Vincent, "Twitter taught microsoft's ai chatbot to be a racist asshole in less than a day," *The Verge*, 2016. [Online]. Available: https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist (Citato a pagina 8)

[6] J. Dastin, "Amazon scraps secret ai recruiting tool that showed bias against women," *Reuters*, 2018. [Online]. Available: https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G (Citato a pagina 8)

[7] A. Abid, M. Farooqi, and J. Zou, "Persistent anti-muslim bias in large language models," 2021. [Online]. Available: https://arxiv.org/abs/2101.05783 (Citato a pagina 8)

[8] W. D. Heaven, "Ai dungeon's creator is trying to make a safe space for nsfw ai fantasies," *MIT Technology Review*, 2021. [Online]. Available: https://www.technologyreview.com/2021/04/16/1023037/ai-dungeon-nsfw-safe-space-latent-space/ (Citato a pagina 8)

[9] C. J. S. Barr, O. Erdelyi, P. D. Docherty, and R. C. Grace, "A review of fairness and a practical guide to selecting context-appropriate fairness metrics in machine learning," *arXivLabs*, 2025. [Online]. Available: https://arxiv.org/abs/2411.06624 (Citato a pagina 9)

[10] J. Z. F. S. M. H. Max Hort, Zhenpeng Chen, "Bias mitigation for machine learning classifiers: A comprehensive survey," *ACM Journal on Responsible Computing*, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:250526377 (Citato a pagina 10)

[11] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *arXivLabs*, 2022. [Online]. Available: https://arxiv.org/abs/1908.09635 (Citato alle pagine 10 e 11)

[12] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96. AAAI Press, 1996, p. 226–231. (Citato a pagina 19)

Questa tesi ha contribuito a piantare un albero in Kenya tramite il progetto Treedom.

`https://www.treedom.net/it/user/sesalab/event/sesa-random-forest`