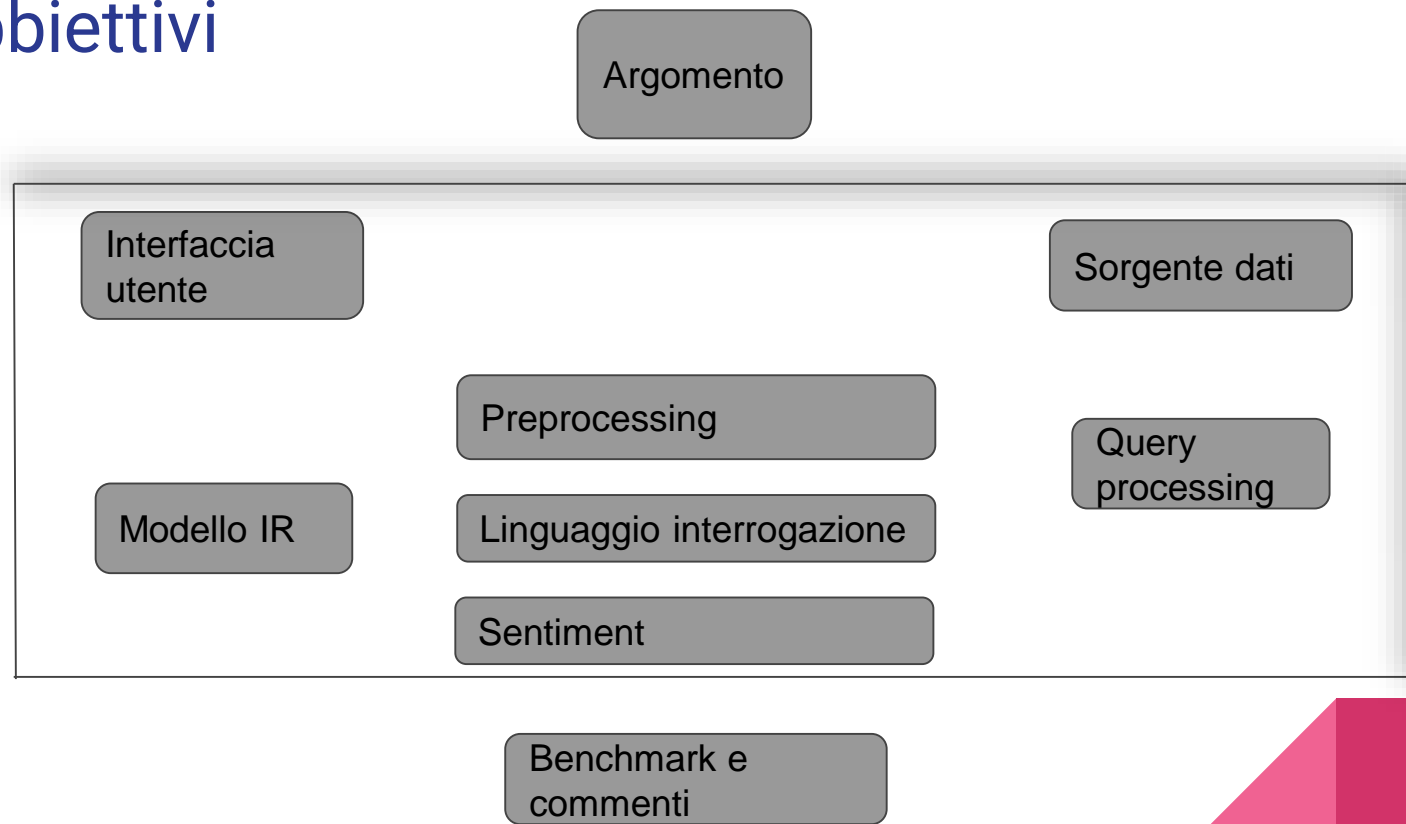


# Index Retrieval Project

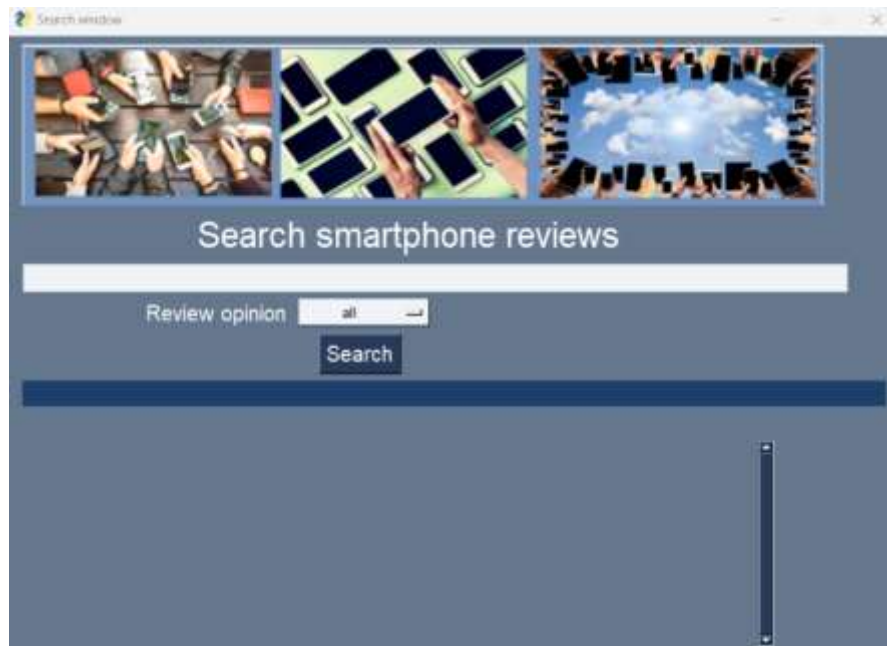
Guida agli acquisti con la sentiment analysis

# Obbiettivi



# Argomento

Si vuole realizzare un Software che permetta di effettuare la ricerca tra recensioni di amazon e filtrarle in base al sentimento espresso nella recensione



# IR

L'utente inserisce il suo UIN attraverso l'interfaccia grafica, il software lo trasforma in una query, attraverso l'indice vengono recuperati i documenti più rilevanti, anch'essi restituiti all'utente attraverso l'interfaccia grafica



# Sorgente dati

Per la creazione del dataset sono state pre-processate circa 30000 recensioni di amazon, ottenute dall'[AMAZON MOBILE PHONE REVIEWS](#).

Le singole recensioni sono state rappresentate con prodotto, testo, link, sentimento, stelle e documento

E successivamente si è proseguito con la costruzione dell'indice



# Indice

Per la creazione dell'indice abbiamo formulato lo schema per memorizzare le informazioni principali.

stored = True → per salvare il campo nel record.

```
Schema(nome_prodotto=TEXT(stored=True), # nome del prodotto  
        sentiment=NUMERIC(stored=True), # sentimento estratto dalla recensione  
        document=ID(stored=True), # nome del documento contenente la recensione  
        testo_processato=TEXT(stored=True)) # testo della recensione pre-processato
```

# Linguaggio d'interrogazione

Il linguaggio d'interrogazione è fornito da Whoosh, in cui troviamo

Ricerca semplice = effettuando una ricerca, le singole parole sono concatenate in OR

Ricerca Phrasal Retrieval = Effettuando una ricerca tra virgolette "" viene compiuto Phrasal Retrieval sul titolo della recensione

Ricerca miste booleane = Permette la ricerca di parole la cui presenza è regolata dagli operatori booleani, quali AND, OR, NOT

In aggiunta

Ricerca in AND= Iniziando una recensione con il simbolo &, viene effettuata una ricerca applicando l'operatore and alle parole presenti nella query

I risultati di qualunque modalità di ricerca sono ordinati sulla sentiment.



# Tecniche di Sentiment

Al fine di ricavare il sentimento delle recensioni è stata implementata la funzione `getScore`, tramite il pacchetto “Transformer” usando modelli di intelligenza artificiale si attribuisce un valore float che va da -1 a +1, che rappresenta il sentimento della recensione:

Da -1.0 a -0.6  $\Rightarrow$  “Molto male”

Da -0.6 a -0.2  $\Rightarrow$  “Male”

Da -0.2 a 0.2  $\Rightarrow$  “Neutro”

Da 0.2 a 0.6  $\Rightarrow$  “Positivo”

Da 0.6 a 1.0  $\Rightarrow$  “Molto positivo”





# Modelli di sentiment

Sono state implementate due classi, rispettivamente

- ReviewsHuggingFaceAnalyzer(SentimentAnalyzer)
- AmazonHuggingFaceAnalyzer(SentimentAnalyzer)

Il primo utilizza il modello juliensimon/reviews-sentiment-analysis = restituisce il sentimento in label\_0 e label\_1 con valori compresi tra 0 e 1

il secondo utilizza LiYuan/amazon-review-sentiment-analysis= analizza il testo e associa una probabilità che la recensione abbia determinate stelle

Dopo averli testati e aver visionato i documenti, il più realistico ci è sembrato il modello

juliensimon



# Modello di IR

Per effettuare interrogazioni si è optato per il modello default del pacchetto Whoosh, BM25F, in quanto migliora con il passare delle interrogazioni.

E' un'evoluzione del modello probabilistico che associa ad ogni documento la probabilità di essere rilevante per una query  $q$

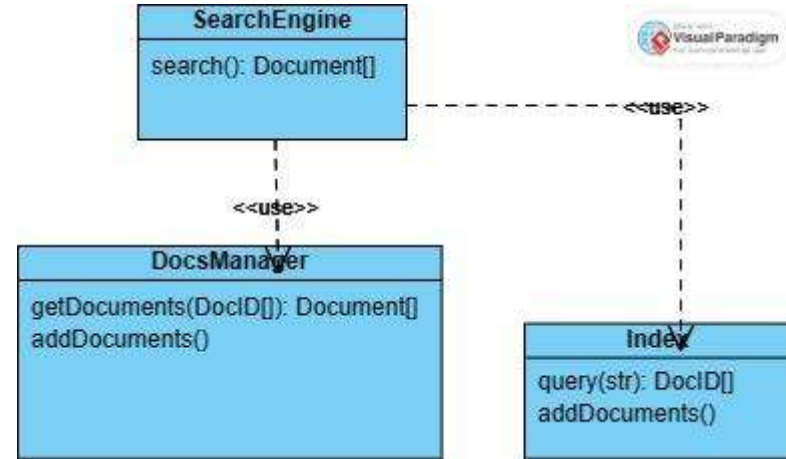
Tuttavia è possibile implementare altri modelli tramite la libreria Whoosh



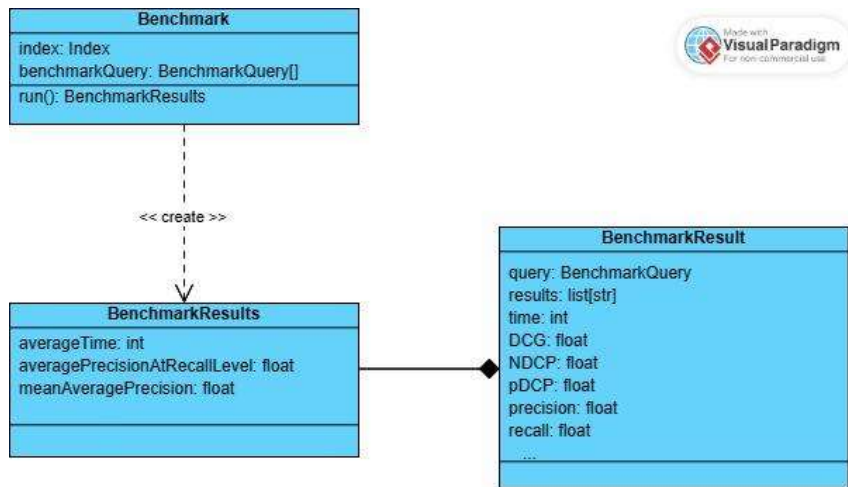
# Modalità di interrogazione del modello per query processing

Al fine di ottenere le informazioni tramite una query:

- Vengono presi i token dalla query
- Vengono cercati i token nell'indice
- Dall'indice si recuperano gli ID dei documenti contenenti i termini
- Gli ID vengono filtrati o ordinati tramite la sentiment
- Vengono ritirati i documenti dalla collezione



# Benchmark

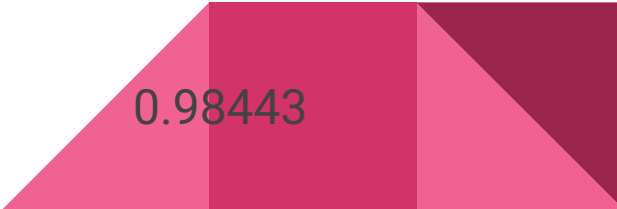


Abbiamo sezionato 10 query eterogenee e manualmente abbiamo selezionato tutti i documenti dandogli un punteggio di inerenza.

Sono state create classi apposite per la valutazione di questi benchmark

# Benchmark e commenti

TF-IDF		Frequency	BM25
Average time	0.55109	0.59812	0.60949
Mean average precision	1.54995	1.54997	1.54994
Average DCG	25024	25021	24930
Average normalized DCG	0.98363	0.98136	0.98443



# Grazie per l'attenzione

Federico Matarante - 152767

Andrea Bonfatti - 137417

Giovanni Cocchi - 142044

