AMAZON REVIEWS ANALYSIS

Università degli Studi di Milano-Bicocca
Corso di Data Analytics

Basso Matteo 807628
Ferri Marco 807130

Le recensioni utente possono contribuire al miglioramento del business di un'azienda come Amazon?

Sono in grado di fornire informazioni per migliorare l’esperienza utente?

1.

DOMINIO

OBIETTIVI

- ➔ Esplorazione del dataset
- ➔ Sentiment Analysis
- ➔ Recommender Systems

COSA È POSSIBILE OTTENERE?

DATASET

278.677 RECENSIONI

5-core

39.387 Utenti

23.033 Prodotti

Categoria Clothing, Shoes and Jewelry

Lingua Inglese **Periodo** 1996 - 2014

Fonte http://jmcauley.ucsd.edu/data/amazon

2.

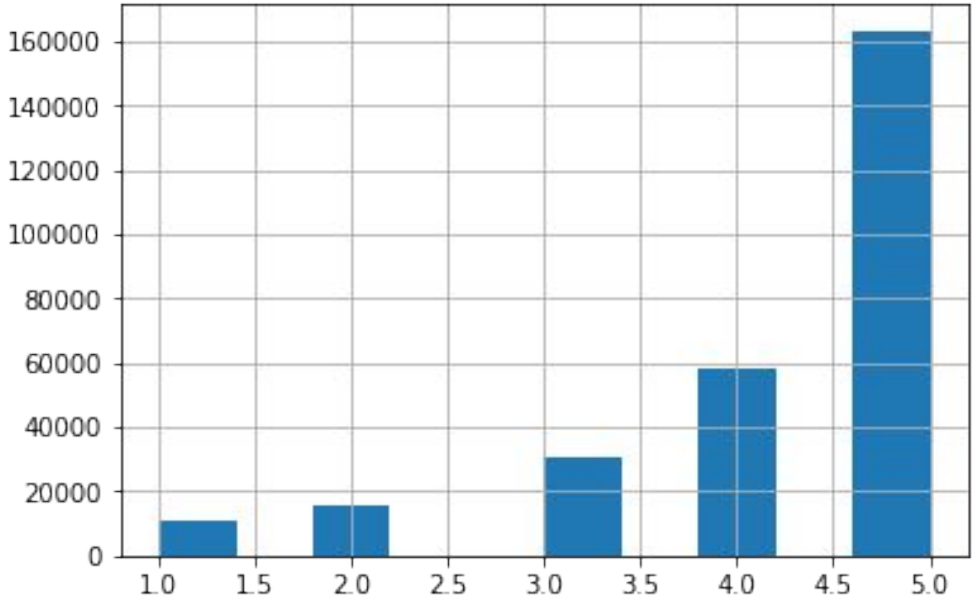
BASIC ANALYSIS

Analisi di carattere generale sul dataset

SCHEMA

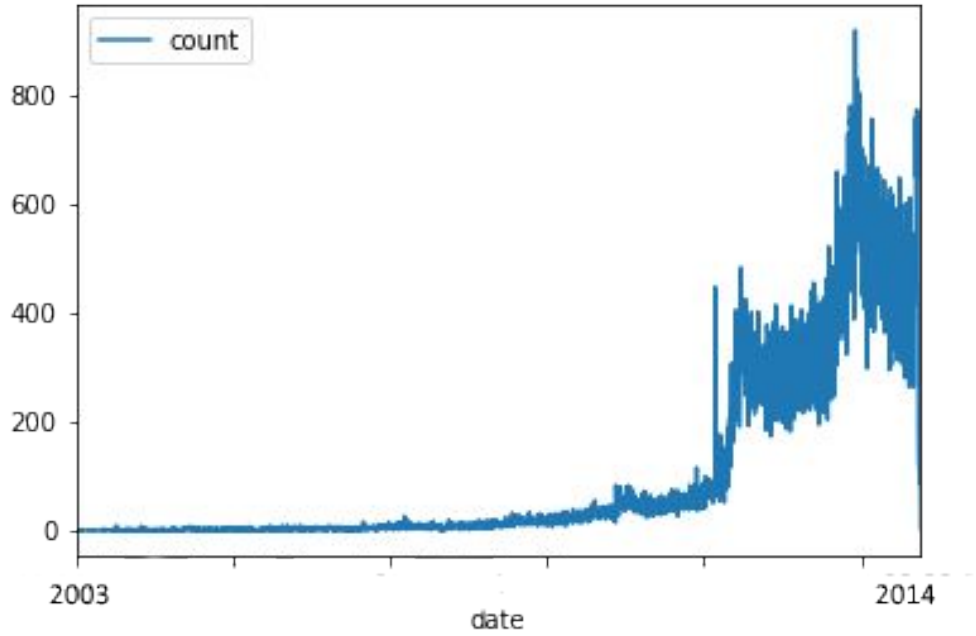
Campo	Descrizione
userID	ID utente
productID	ID prodotto
text	Testo della recensione
summary	Titolo della recensione
rating	Punteggio
date	Timestamp in formato datetime

DISTRIBUZIONE RECENSIONI PER **RATING**



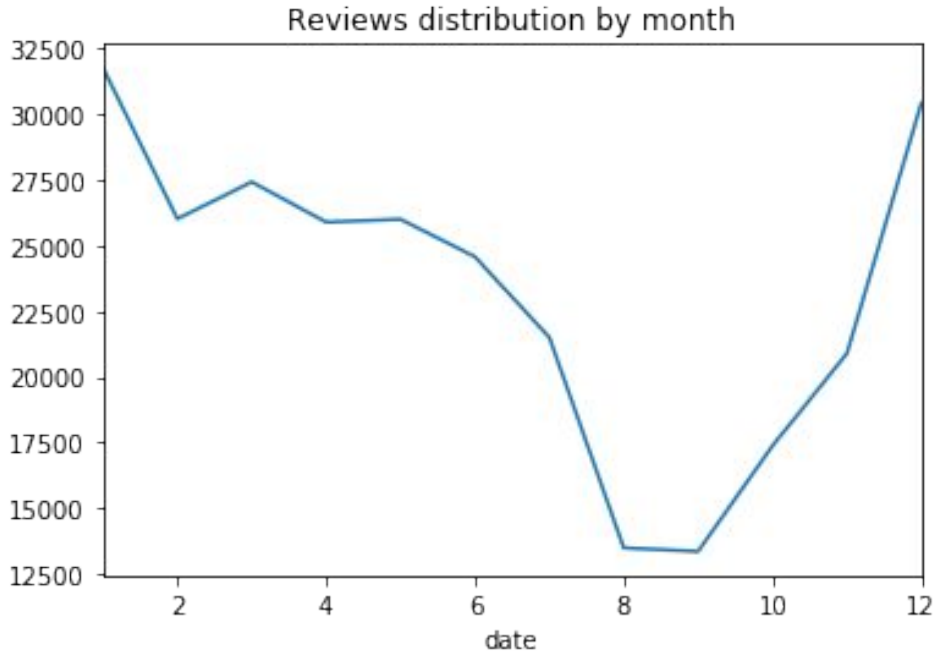
Classi fortemente sbilanciate

DISTRIBUZIONE RECENSIONI PER **DATA**



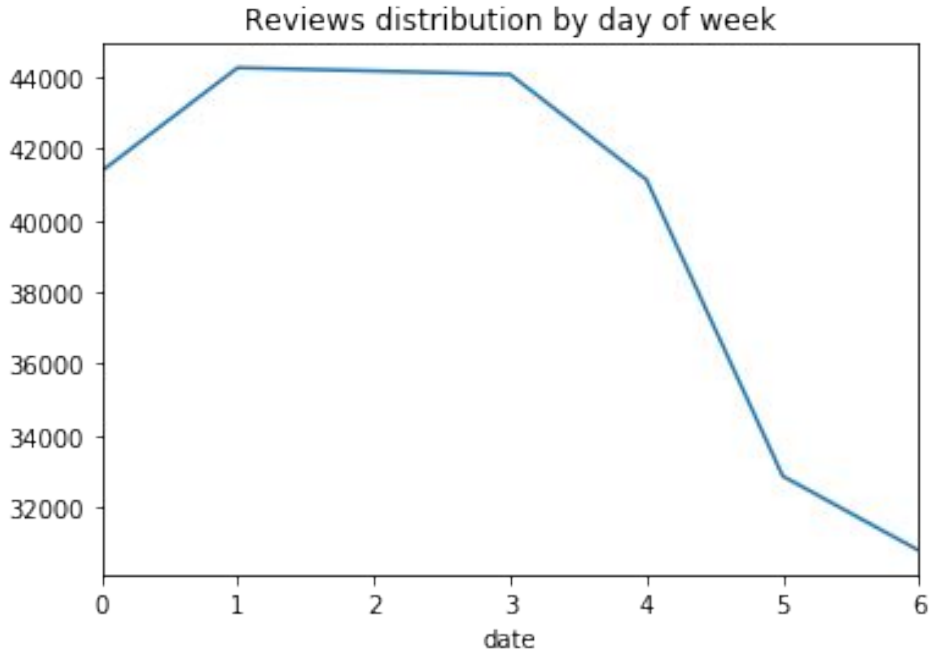
Medesima distribuzione del `rating`

DISTRIBUZIONE RECENSIONI PER **MESE**



Medesima distribuzione del `rating`

DISTRIBUZIONE RECENSIONI PER **GIORNO DELLA SETTIMANA**



Medesima distribuzione del rating

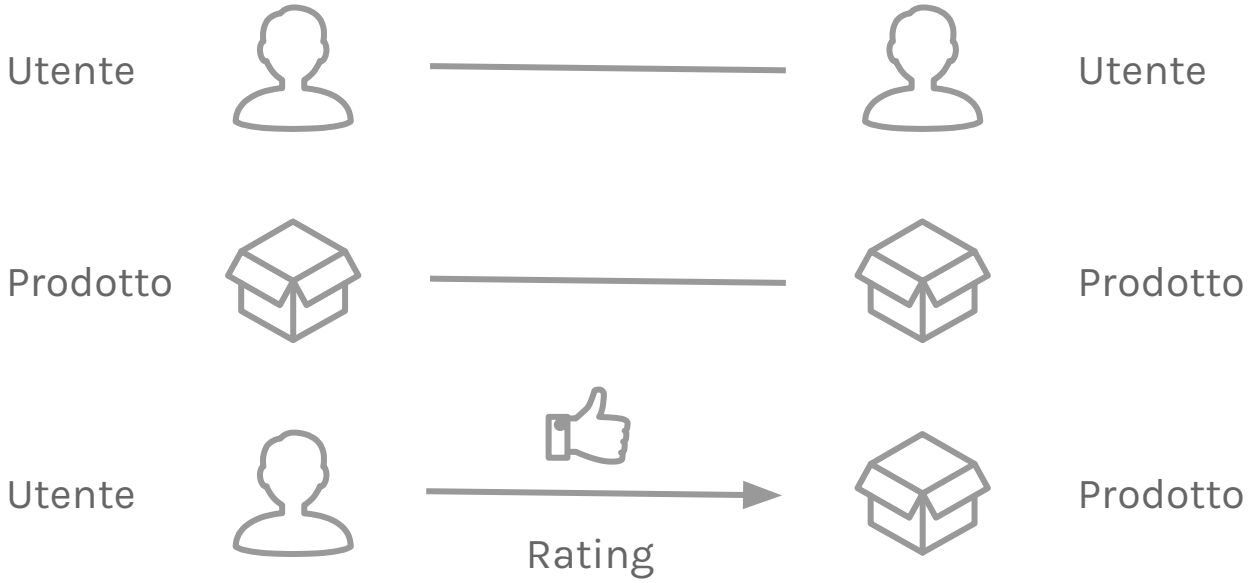
3.

NETWORK ANALYSIS

Costruzione e analisi della rete di utenti e prodotti

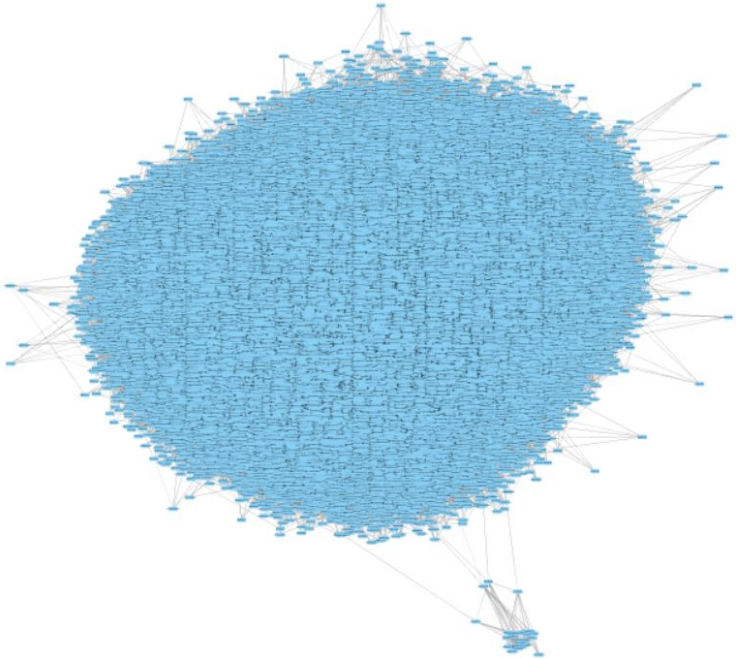
TIPOLOGIE DI RETE

Con il dataset di recensioni è possibile costruire tre diverse tipologie di rete:



RETE GENERATA

Poiché non sono disponibili i metadati dei prodotti è stato deciso di costruire la terza rete

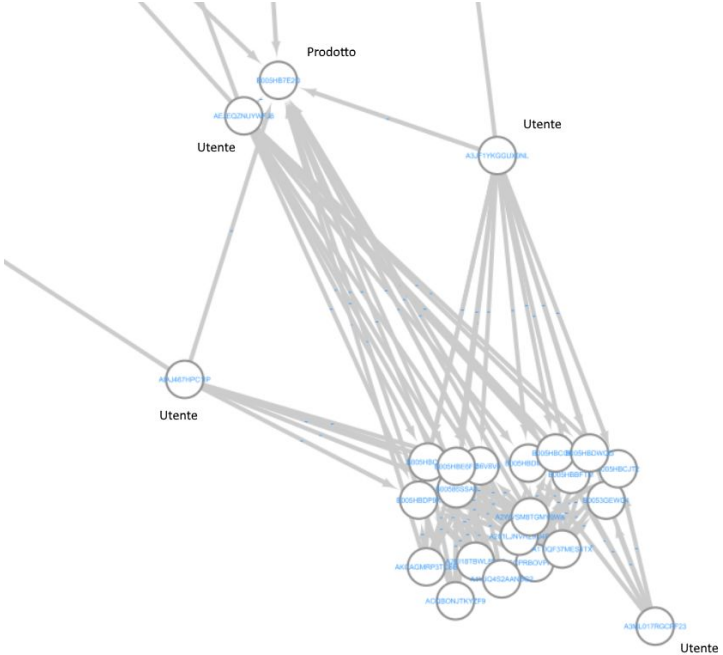


278.677 archi (recensioni)

62.420 nodi (utenti + prodotti)

Gli utenti sono sorgenti
I prodotti sono pozzi

CLUSTER INFERIORE

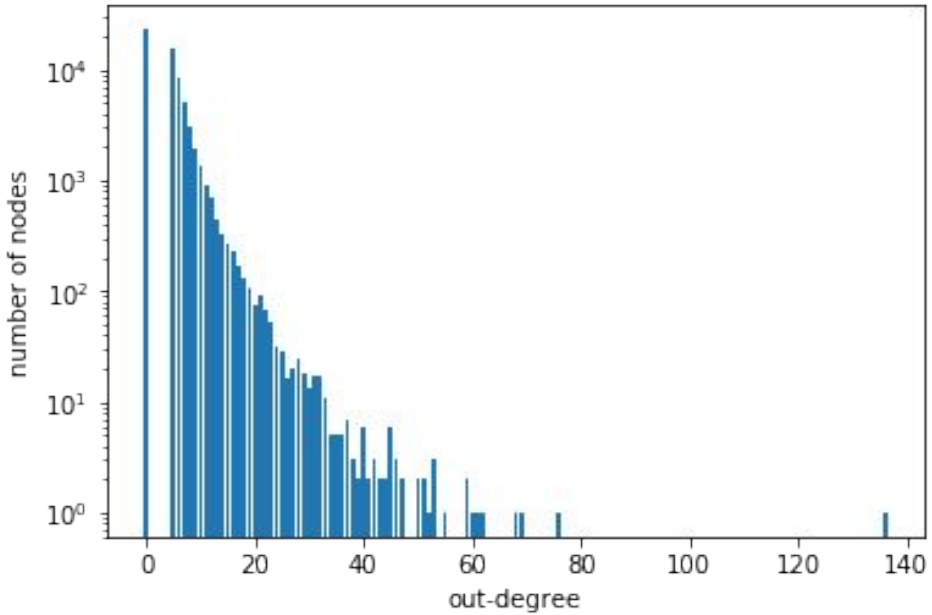


Prodotti di nicchia

Provenienti da mercati stranieri?

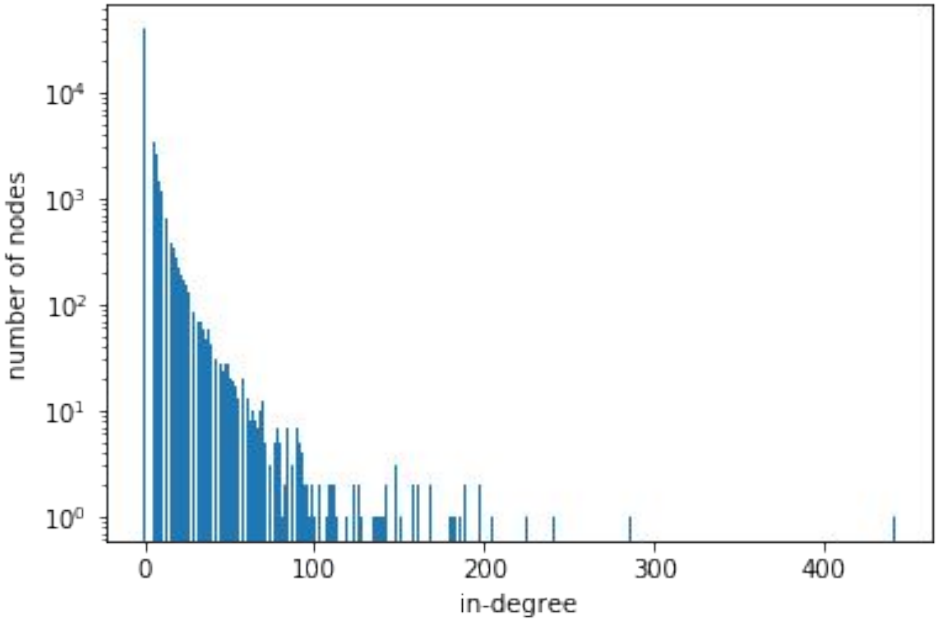
Oppure destinati ad un tipo di vendita tra gruppi ristretti di persone?

DISTRIBUZIONE DELL'OUT-DEGREE DEGLI UTENTI



Distribuzione plausibile visto il dominio di interesse

DISTRIBUZIONE DELL'IN-DEGREE DEI PRODOTTI



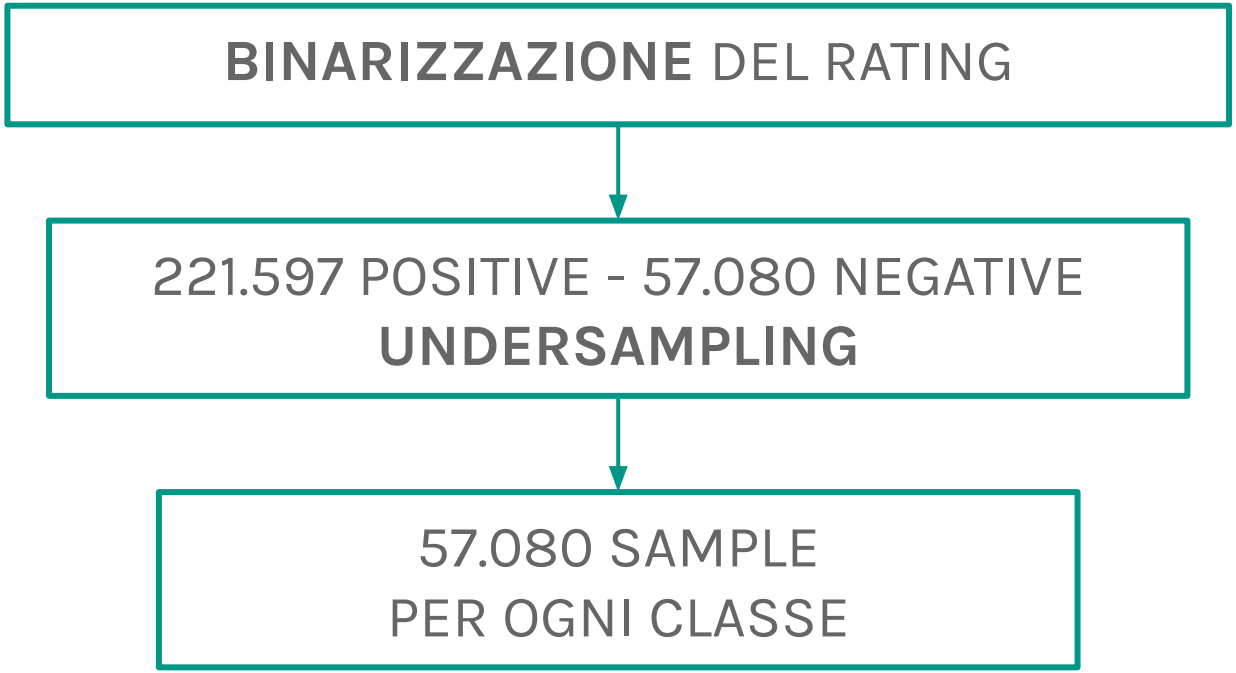
Non emergono prodotti particolarmente popolari (nessun *hub*)

4.

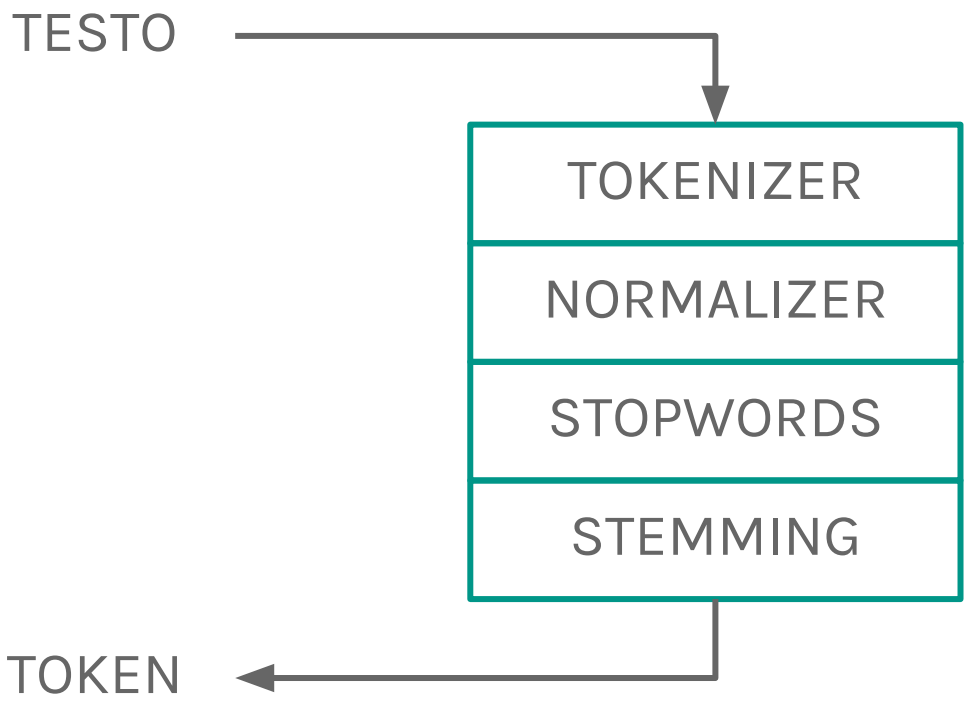
SENTIMENT ANALYSIS

Sentiment analysis sulle recensioni

ELABORAZIONE DATASET



ELABORAZIONE DEL TESTO



ESEMPIO

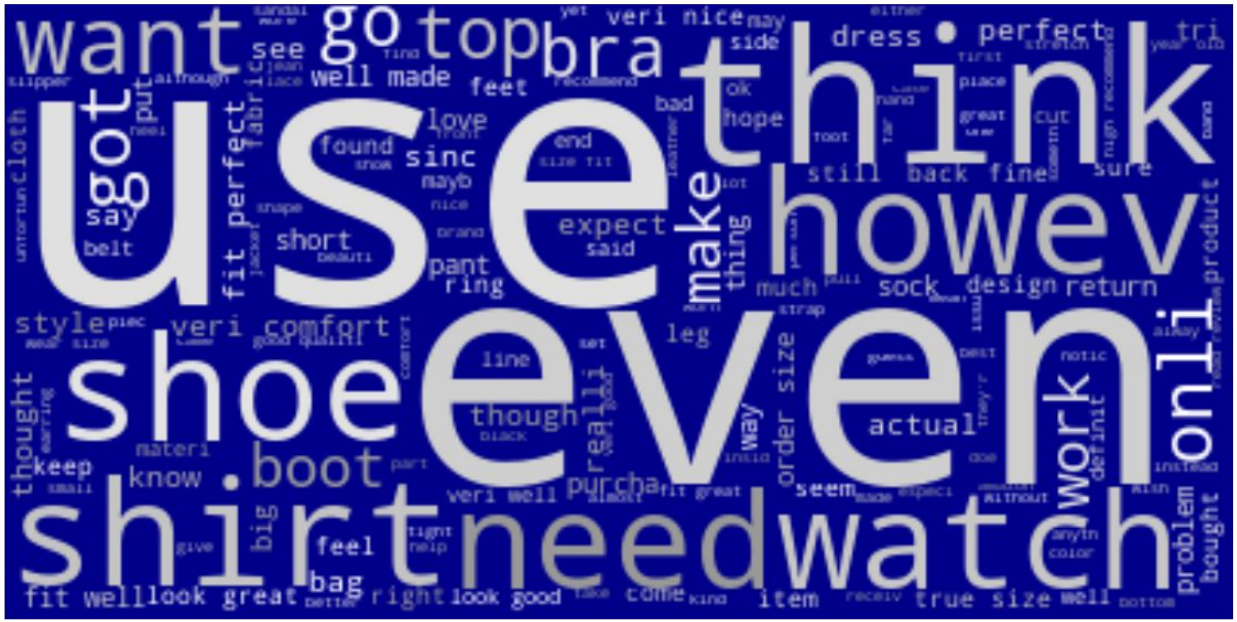
Testo:
"Hi! This... isn't a beautiful sentence with some interesting
$70 and $5,50 features like people's names and Mr. Fox thoughts
for number such as 23, 4 and 7 or peer2peer and wi-fi with
snake_case but not kebab-case."

Tokenization e lowercasing:
['hi', 'this', "isn't", 'a', 'beautiful', 'sentence', 'with',
'some', 'interesting', 'and', 'features', 'like', "people's",
'names', 'and', 'mr', 'fox', 'thoughts', 'for', 'number',
'such', 'as', 'and', 'or', 'peer2peer', 'and', 'wi', 'fi',
'with', 'snake_case', 'but', 'not', 'kebab', 'case']

Rimozione stopwords:
['hi', "isn't", 'beautiful', 'sentence', 'interesting', 'features',
'like', "people's", 'names', 'mr', 'fox', 'thoughts', 'number',
'peer2peer', 'wi', 'fi', 'snake_case', 'but', 'not', 'kebab', 'case']

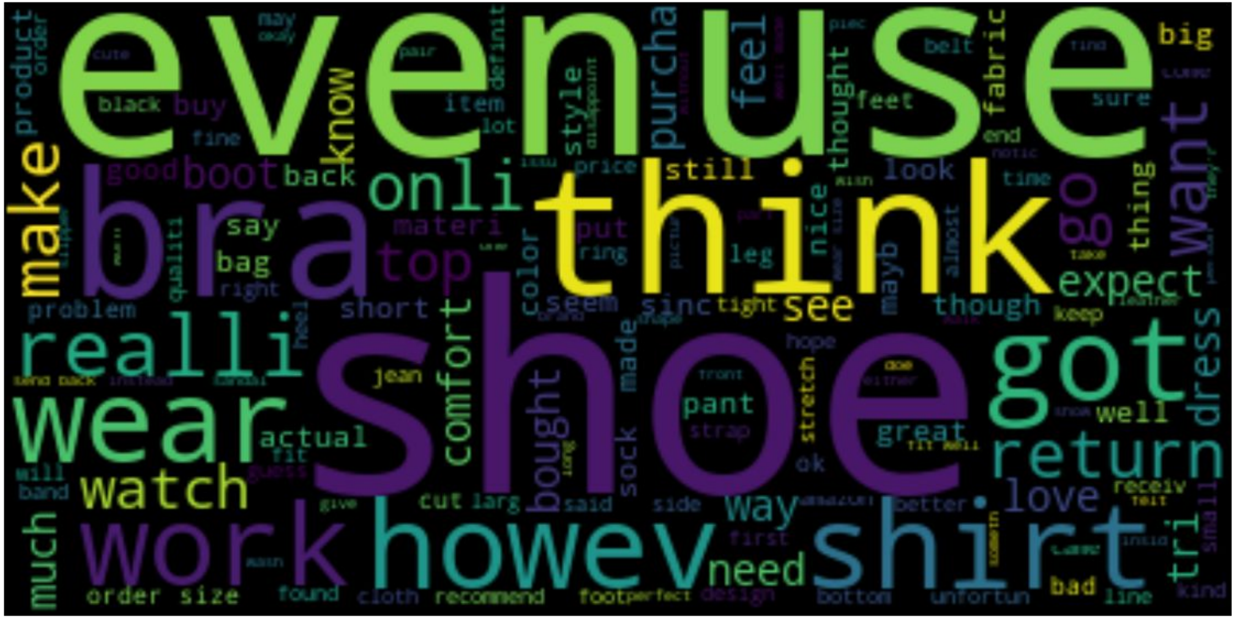
Stemming:
['hi', "isn't", 'beauti', 'sentenc', 'interest', 'featur', 'like',
'peopl', 'name', 'mr', 'fox', 'thought', 'number', 'peer2peer',
'wi', 'fi', 'snake_cas', 'but', 'not', 'kebab', 'case']

PAROLE PIÙ COMUNI



PAROLE PIÙ COMUNI NELLE RECENSIONI **POSITIVE**

PAROLE PIÙ COMUNI NELLE RECENSIONI NEGATIVE



5.

SENTIMENT PREDICTION

Predizione del sentiment associato ad una recensione

OBIETTIVO

Predizione del `rating` basata sul dizionario ricavato dalle recensioni a disposizione.

- **Rappresentazione** della conoscenza
- **Apprendimento**
- **Predizione**
- Valutazione delle **performance**

Implementazione tramite `Scikit-learn`

BAG OF WORDS

Dizionario di termini → recensioni in cui occorrono

Recensione → Termine ↓	**R001**	**R011**	**R023**	**R055**	**R786**
dress	0	3	0	0	1
machine	5	0	3	0	2
teacher	0	0	0	2	0
learning	5	0	0	1	0

MATRICE SPARSA

PESATURA DEI TERMINI

Term frequency - Inverse document frequency (**TF-IDF**)

Da occorrenze a **frequenze**

Normalizzate sulla lunghezza delle recensioni e la popolarità del termine stesso

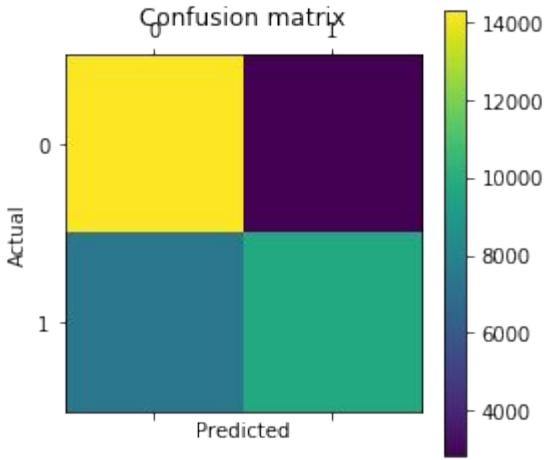
$$w_{ij} = \frac{tf_i}{\max tf_j} \times log\frac{N}{d_{fi}}$$

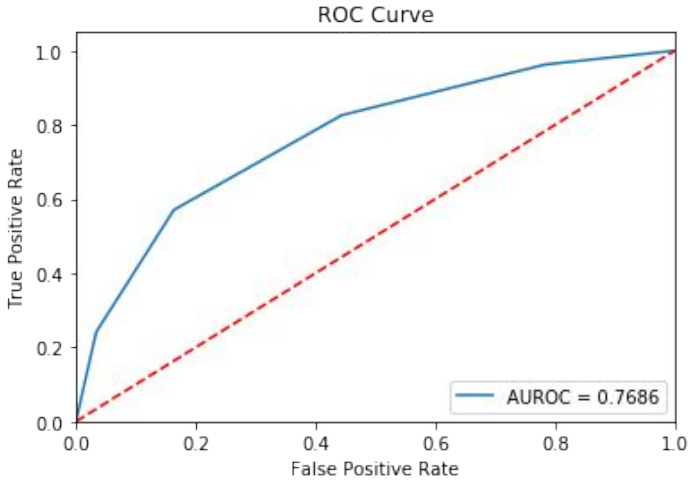
ID	Termine	Peso
4121	but	0.040886
19248	not	0.037178
25748	size	0.033456
31509	veri	0.032994
10467	fit	0.032224
16499	look	0.030117
16151	like	0.029839
32213	wear	0.029403
25314	shoe	0.027903
16656	love	0.025427

MODELLO DI PREDIZIONE

Modelli di machine learning
di classificazione supervisionata
per apprendere la correlazione
fra **termini** nelle recensioni
e **sentiment** associato.

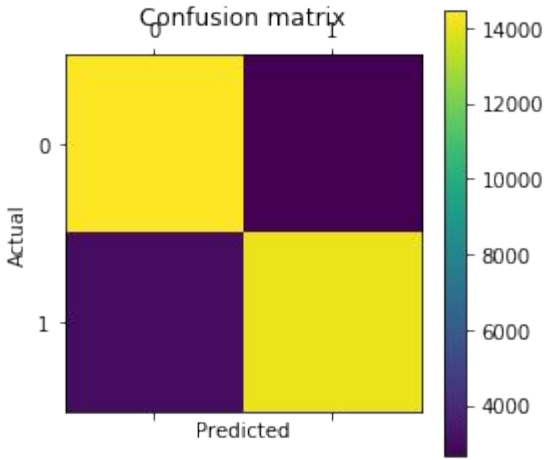
PERFORMANCE RANDOM FOREST

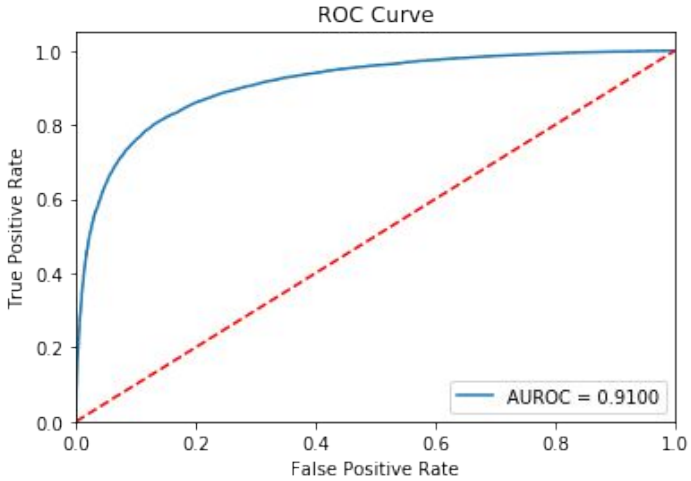




ACCURACY	10-FOLD CROSS VALIDATION:	0.7327	(std dev 0.005981)
PRECISION	10-FOLD CROSS VALIDATION:	0.7348	(std dev 0.009141)
RECALL	10-FOLD CROSS VALIDATION:	0.7348	(std dev 0.008359)
F1	10-FOLD CROSS VALIDATION:	0.7279	(std dev 0.003274)

PERFORMANCE SVM





ACCURACY	10-FOLD CROSS VALIDATION:	0.8290 (std dev 0.005217)
PRECISION	10-FOLD CROSS VALIDATION:	0.8371 (std dev 0.009706)
RECALL	10-FOLD CROSS VALIDATION:	0.8176 (std dev 0.008412)
F1	10-FOLD CROSS VALIDATION:	0.8273 (std dev 0.004608)

SCELTA DEL MODELLO

Random Forest è il modello che apprende più lentamente e ottiene risultati peggiori.

Naive Bayes e **Support Vector Machines** ottengono risultati simili in tempi del tutto comparabili, ma quest'ultimo è leggermente più preciso e quindi il candidato migliore.

PIPELINE

Sequenza di operazioni **salvabile** e **riutilizzabile** per la predizione su nuovi input.

```
text_clf = Pipeline([
    ('vect', TfidfVectorizer(
                use_idf = True,
                strip_accents = 'ascii',
                stop_words = stopset,
                lowercase = True)),
    ('tfidf', TfidfTransformer()),
    ('clf', SGDClassifier(loss='log')),
])
```

6.

ASPECT BASED SENTIMENT ANALYSIS

STRUMENTI

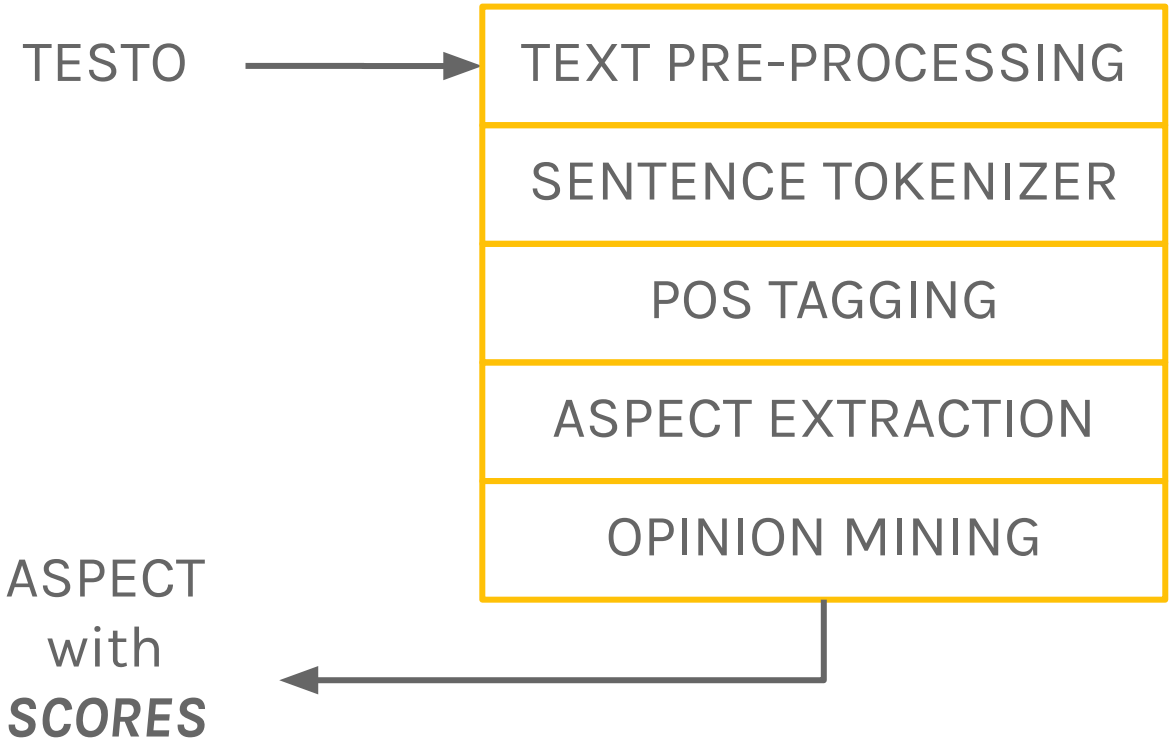
ASUM Java

- API poco user friendly e scarsa documentazione
- Effettuati dei test poco proficui

Si è scelto di utilizzare un progetto GitHub adeguatamente modificato per adattarsi alle esigenze del caso in esame e migliorare le prestazioni.

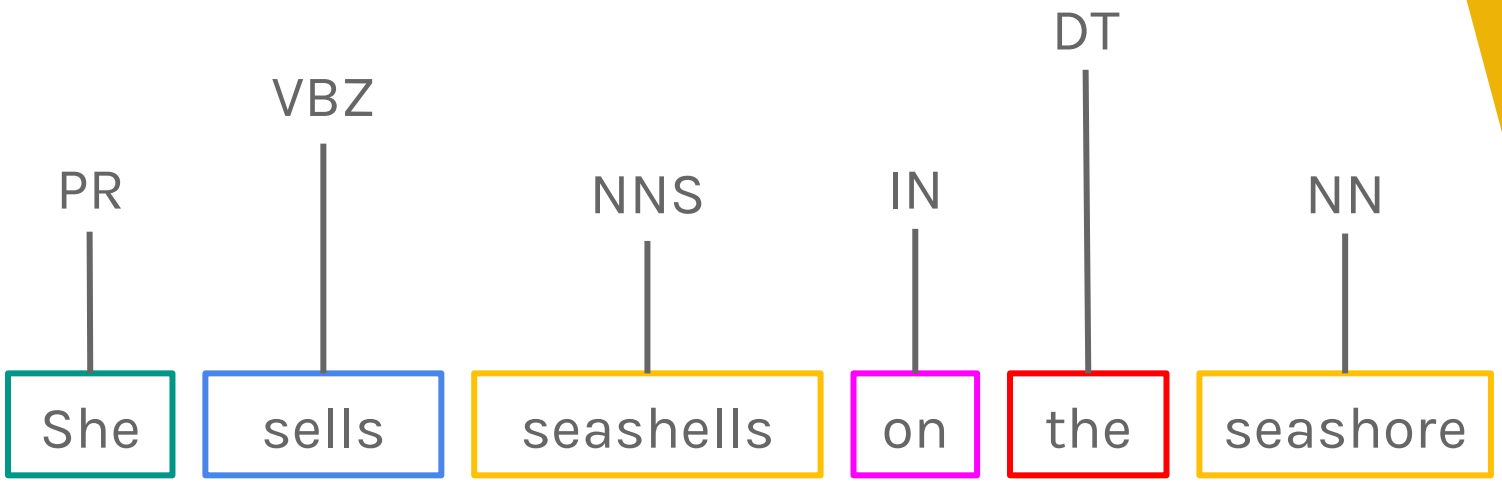
https://github.com/jonm01/absa

FASI DELL'ELABORAZIONE



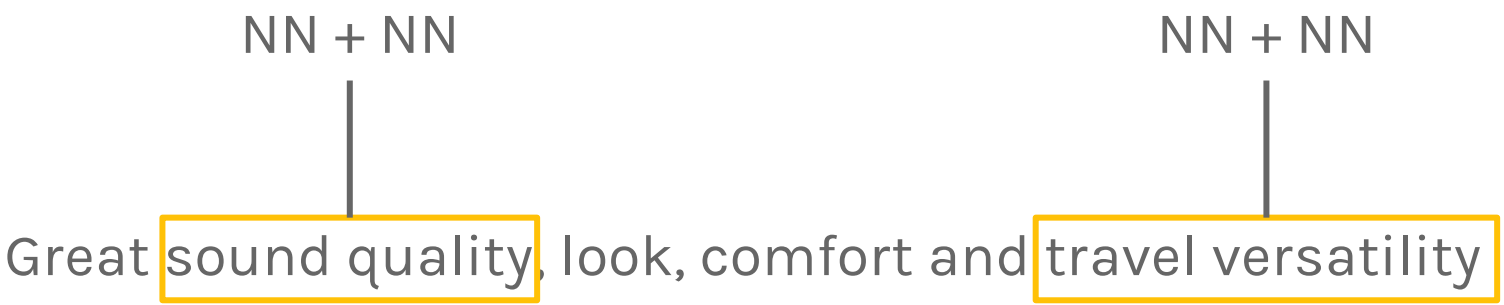
POS TAGGING

Assegnazione di un tag per ciascuna parola del testo al fine di identificare la categoria grammaticale a cui appartiene.

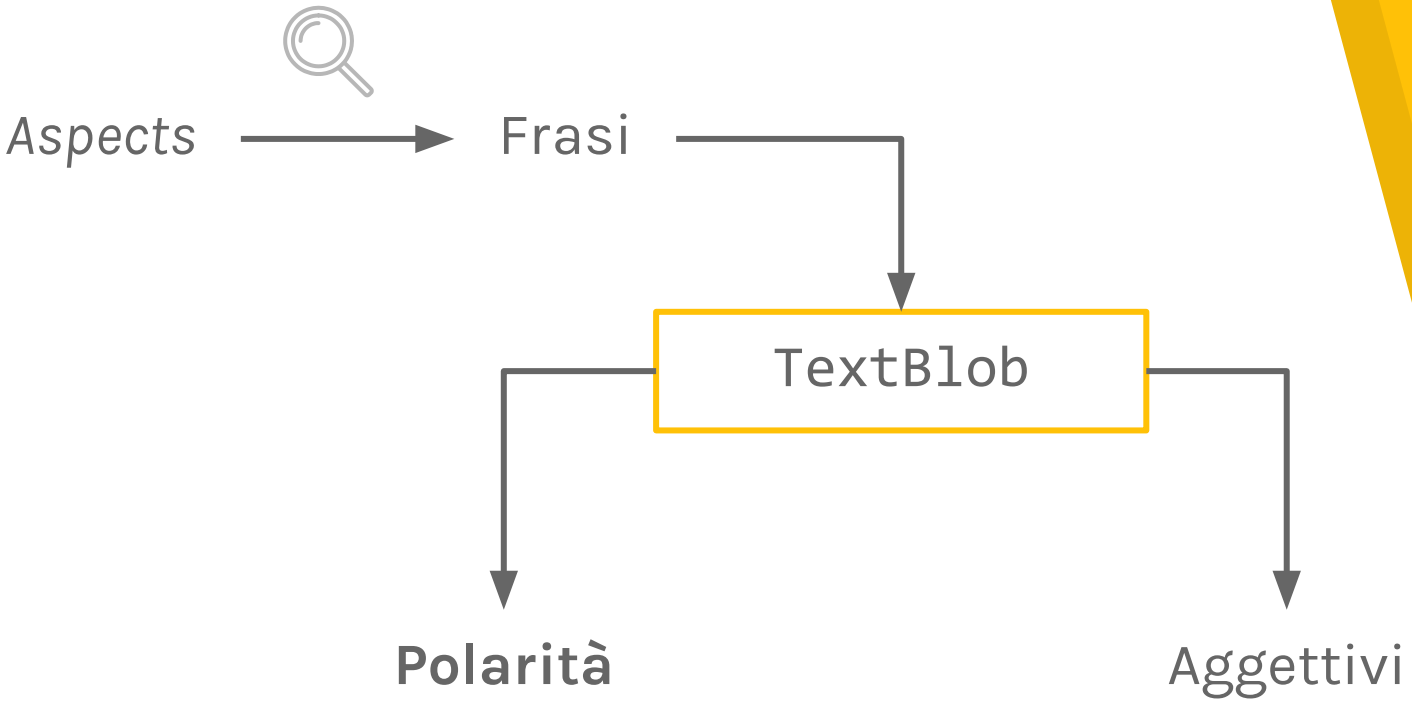


ESTRAZIONE DEGLI ASPECT

Identificazione degli aspect come congiunzione di **uno o più sostantivi contigui**



IDENTIFICAZIONE DEL SENTIMENT



RISULTATI

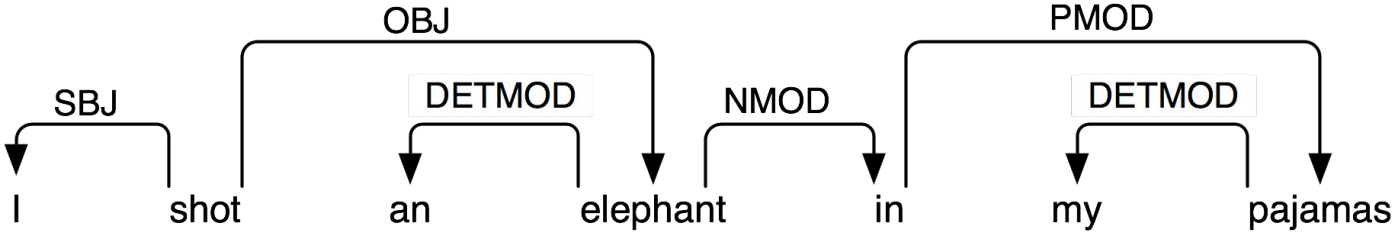
NECKLACE	**Positive** - score 0.4 - sentences 76%
PRICE	**Positive** - score 0.5 - sentences 76%
CHAIN	**Positive** - score 0.3 - sentences 58%
OWL	**Positive** - score 0.4 - sentences 72%
CUTE	**Positive** - score 0.4 - sentences 98%
QUALITY	**Positive** - score 0.4 - sentences 82%
GIFT	**Positive** - score 0.4 - sentences 76%
TIME	**Positive** - score 0.3 - sentences 55%

Estrazione di aspect plausibili

Qualche errore di natura comprensibile

ULTERIORI SVILUPPI

Introduzione di un sistema di Dependency Parsing per ottenere i tipi di **dipendenza** tra le varie unità linguistiche, potendo così migliorare l'accuratezza dell'analisi.



7.

RECOMMENDER SYSTEMS

SISTEMI DI RACCOMANDAZIONE

- **Content-based**

Basati su **contenuto** e **metadati**
Genere, caratteristiche, prezzo, …

- NLP applicato a modelli di machine learning

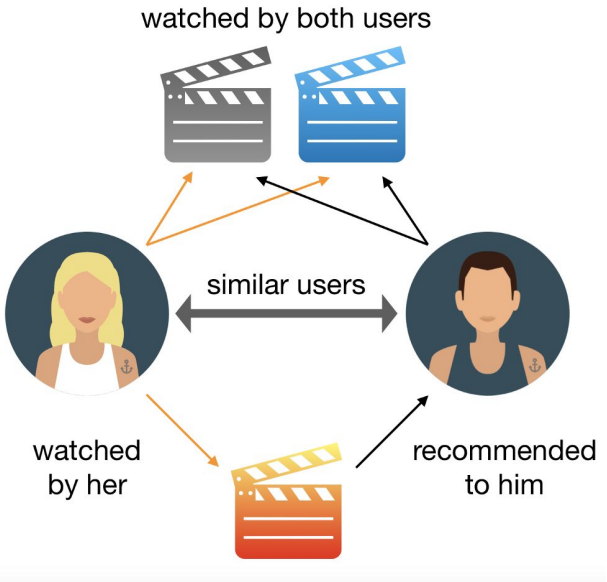
- **Collaborative filtering**

Basati sullo **storico** degli acquisti e delle **preferenze** espresse dagli utenti

- Analisi della rete sul concetto di similarità

COLLABORATIVE FILTERING

Utile per effettuare predizioni in merito alle valutazioni mancanti sulla base degli utenti con **preferenze simili.**



COLLABORATIVE FILTERING

EMBEDDING

Necessità di **rappresentare** e **comparare** gli elementi facenti parte del dominio.

Una tecnica molto utilizzata consiste nel proiettare ciascun prodotto e utente in uno **spazio vettoriale** attraverso la **fattorizzazione** della matrice.

Prodotti e utenti rappresentati come **vettori**.

BIAS

Parametro ausiliare per la
normalizzazione di ciascun utente e prodotto.

Ad esempio per trovare un compromesso di rappresentazione per la descrizione di utenti particolarmente critici o prodotti molto popolari.

SIMILARITA’

Tecniche di comparazione

- prodotto scalare
- distanza euclidea
- cosine similarity

Mettono in relazione i vettori nello spazio per trovare elementi simili su cui apprendere embedding e bias tramite **regressione**.

PREDIZIONE

Predizione del `rating` per tutte le coppie <*u*, *p*> di utenti e prodotti. Tramite prodotto scalare:

$$rating_{up} = dot\_product(user, product) + bias_u + bias_p$$

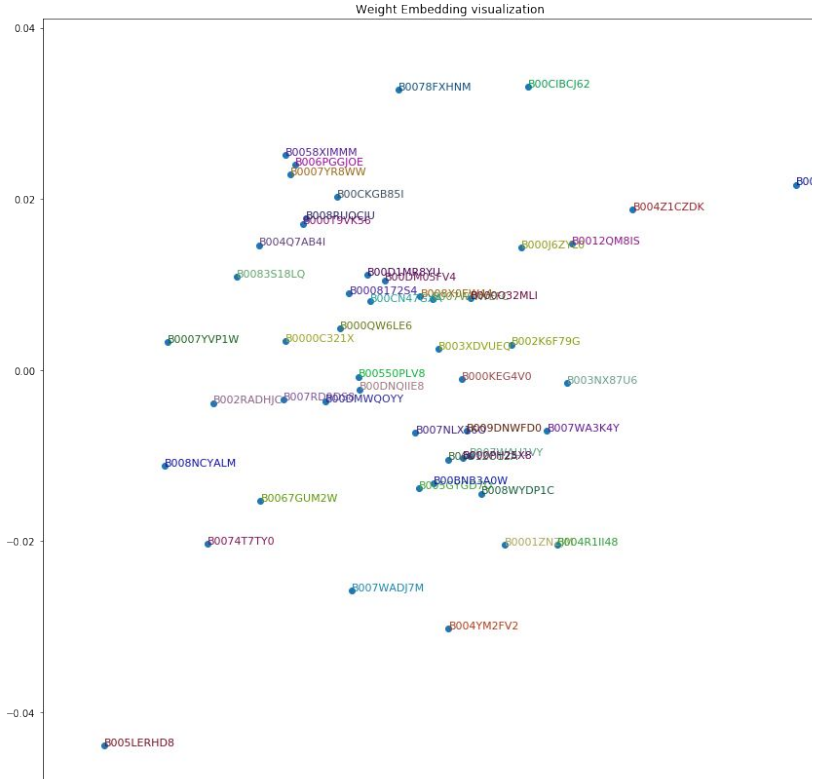
Embedding e bias vengono appresi tramite modelli di regressione per **minimizzare l'errore** (*loss*) rispetto ai `rating` reali ottenuto dalla formula di predizione per ciascuna coppia utente - prodotto.

IMPLEMENTAZIONE

FastAI

DotProduct

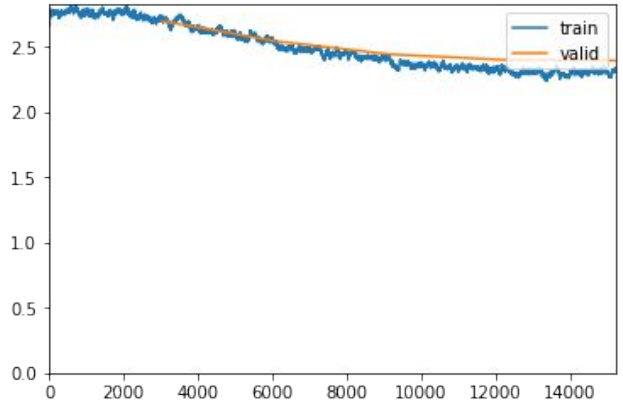
ESEMPIO DI EMBEDDING



Punti spazialmente vicini dovrebbero essere classificati dal modello di collaborative filtering come fossero prodotti simili.

TRAINING

epoch	train_loss	valid_loss	time
0	2.717241	2.697971	00:36
1	2.539666	2.543803	00:37
2	2.429567	2.443485	00:36
3	2.344157	2.401317	00:36
4	2.330287	2.394638	00:36



1-cycle policy

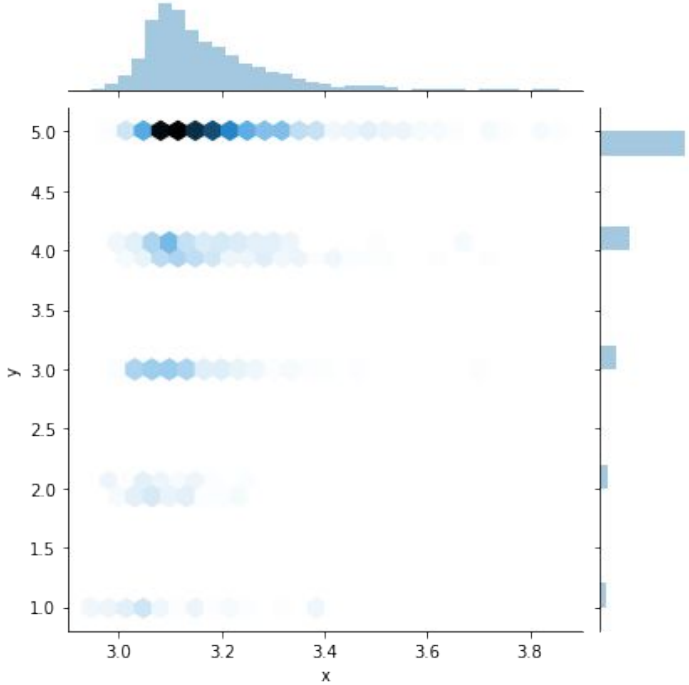
La MSE converge a 2.07 dopo 15 cicli

Coerenza fra *loss* del **train** e del **test set** con un miglioramento molto modesto.

RISULTATO

Il `rating` per qualsiasi coppia utente - prodotto viene classificato in un intorno del valore medio 3.

ANOMALIA!



RISULTATO

Il `rating` per qualsiasi coppia utente - prodotto viene classificato in un intorno del valore medio 3.

ANOMALIA!

Medesimo risultato con **undersampling** o **cambiando il modello** di apprendimento.

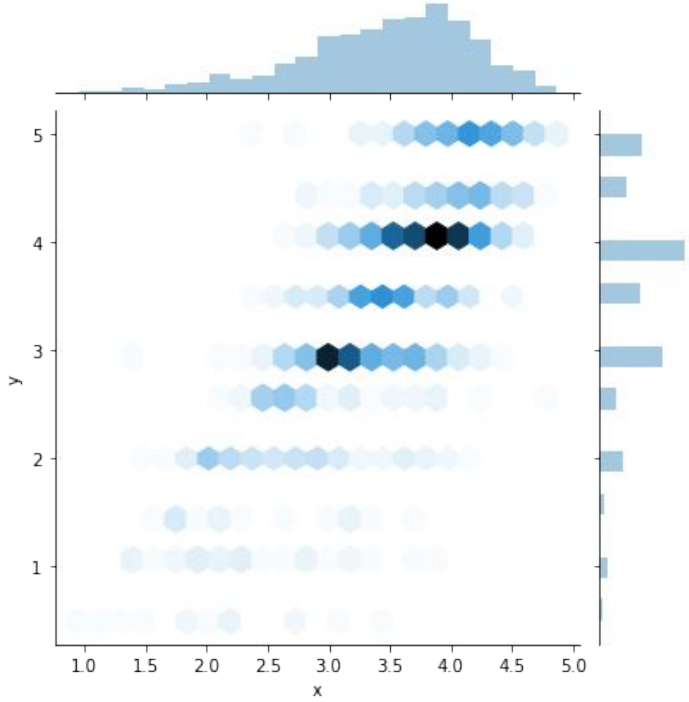
Problema nel dataset?

MOVIELENS

Testing del modello su un **dataset differente** per verificarne la validità.

Movielens offre un dataset contenente 100.000 recensioni di film, spesso usato come esempio per problemi di questo tipo.

MOVIELENS



Predizioni **ben distribuite**!
Dimostrano la correttezza del modello utilizzato.

Problema nel dataset?

Il modello classifica tutto con il **valore medio** per minimizzare l'errore commesso nella regressione.

Potrebbe significare che non vi sono utenti e prodotti con abbastanza recensioni in comune, concetto stesso su cui è basato collaborative filtering?

ANALISI DELLA RETE

COEFFICIENTE TOPOLOGICO

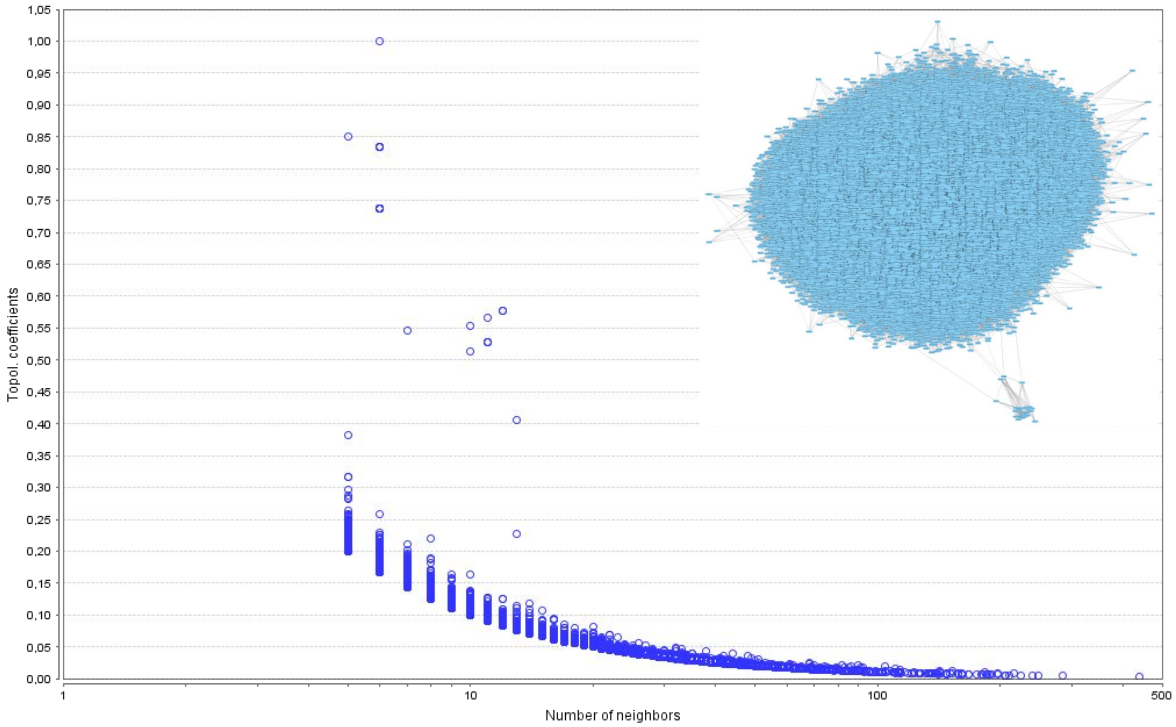
Calcolato per ogni nodo *n* esprime una misura relativa alla quantità di **vicini** che il nodo stesso **condivide** con gli altri nodi del grafo.

$$T_n = \frac{avg(J(n,m))}{k_n}$$

Un valore alto può indicare:

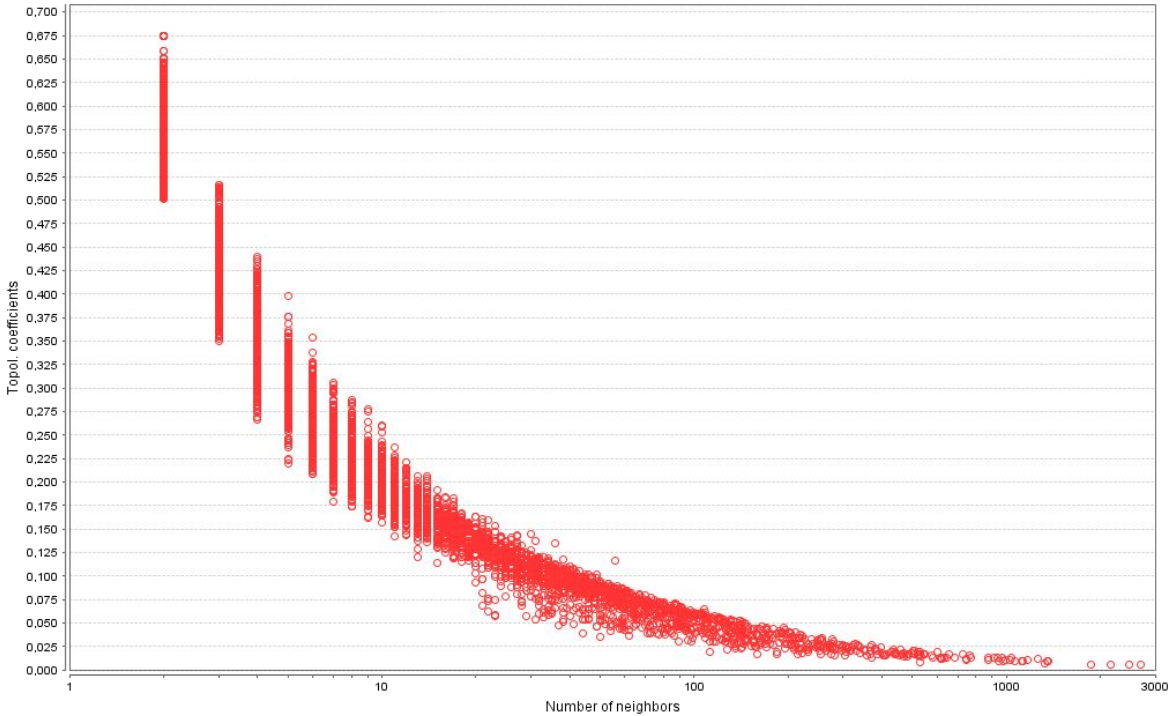
- Un utente che ha recensito prodotti in comune con molti altri utenti
- Un prodotto che condivide molti recensori con altri prodotti

COEFF. TOPOLOGICO - AMAZON



Tutti < 0.25
Molti < 0.05

COEFF. TOPOLOGICO - MOVIELENS



Molti da 0.65 a 0.15

Problema nel dataset

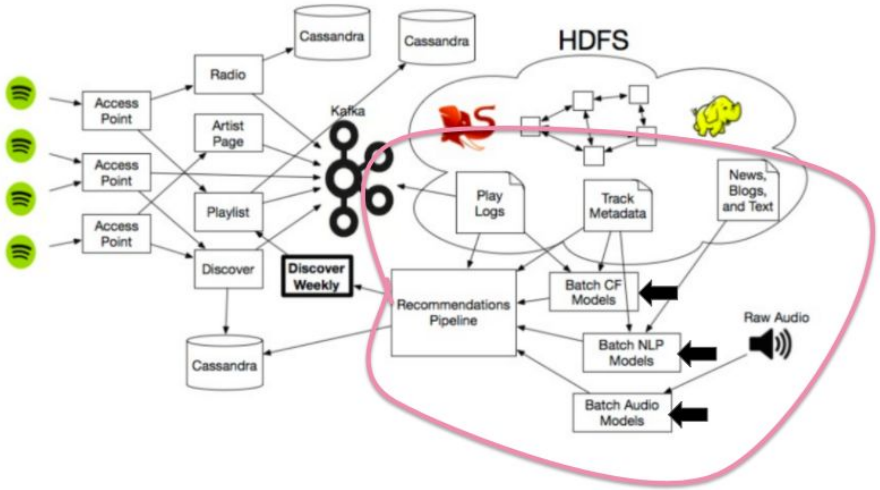
È stato dimostrato

Il dataset delle recensioni di Amazon non è adatto per essere utilizzato con un modello di collaborative filtering.

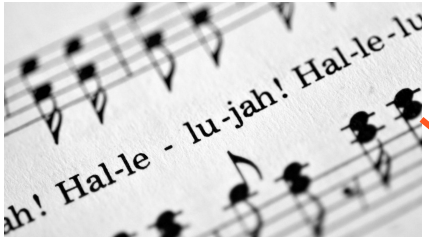
La **soluzione** è utilizzare altri approcci o **combinazione di diverse strategie**.

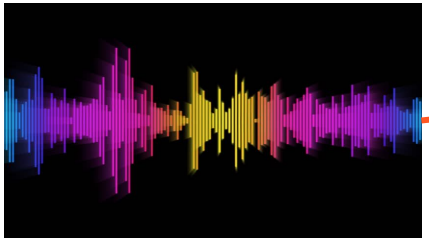
SPOTIFY - DISCOVER WEEKLY

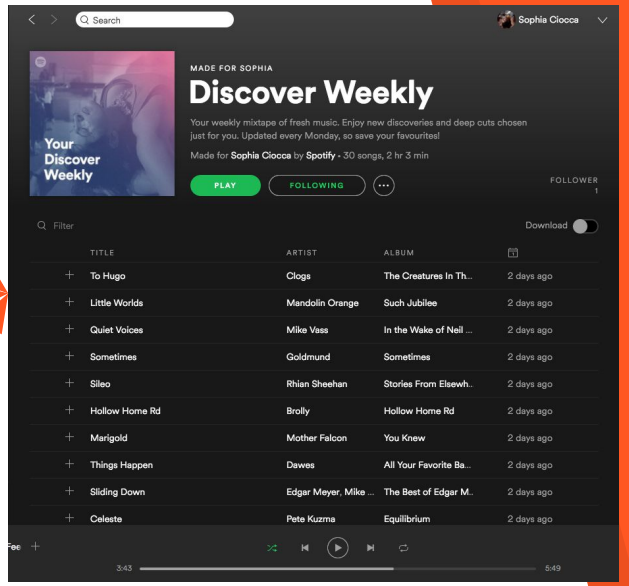
Collaborative Filtering
Natural Language Processing
Raw Audio Deep Learning



SPOTIFY - WEEKLY DISCOVER





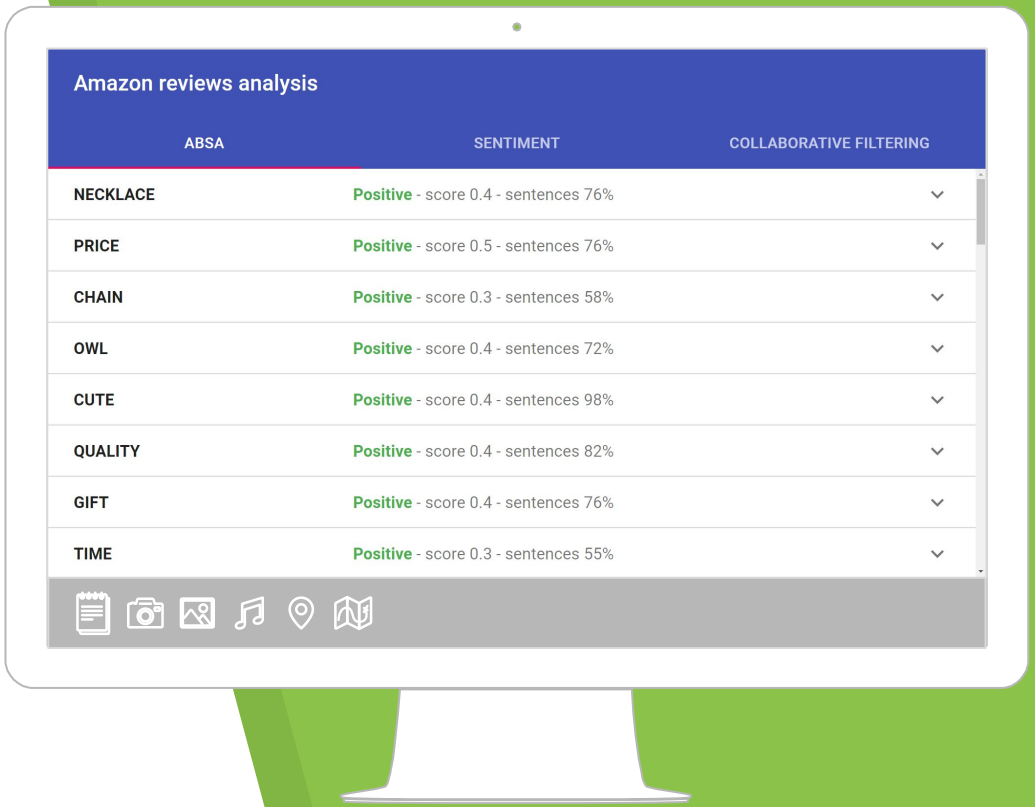


8.
DEMO

Web demo interattiva per l’esplorazione e utilizzo dei modelli impiegati

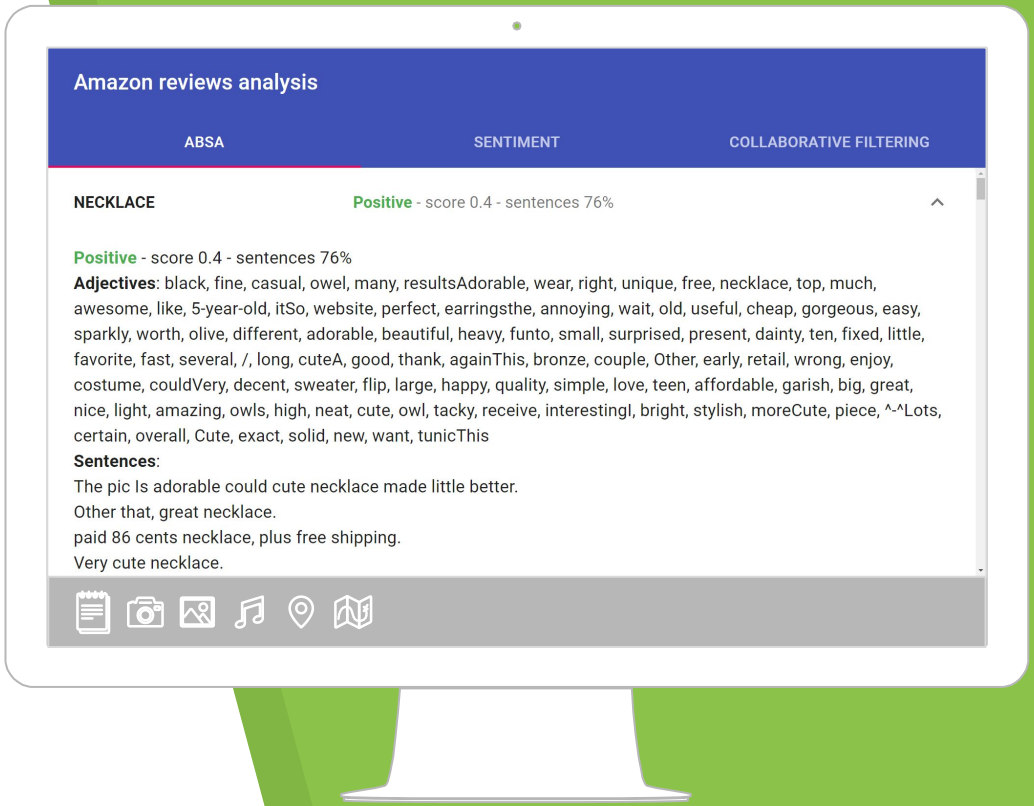
ASPECT BASED SENTIMENT

Elenco **aspetti** e **sentiment** nel dataset di Amazon considerato



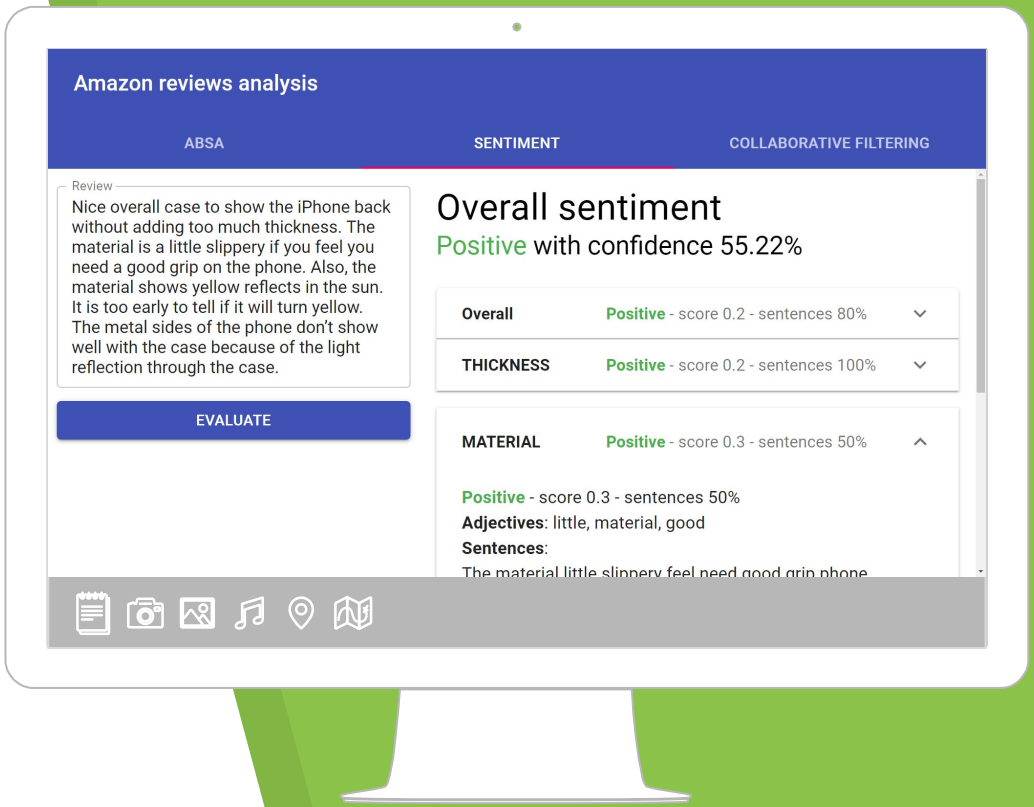
ASPECT BASED SENTIMENT

Polarità, aggettivi e frasi che caratterizzano ciascun aspect identificato



SENTIMENT PREDICTION

Inserimento di una recensione e **predizione** del **sentiment** associato



9.

CONCLUSIONI

CONCLUSIONI

- ➔ Sentiment analysis utile sia ai produttori che ad Amazon stesso per **migliorare i propri servizi** o sapere come ampliare l'offerta commerciale
- ➔ Gli utenti possono beneficiare di servizi di raccomandazione in favore di **acquisti più pertinenti**

Per ottimizzare l'utilizzo dei dati a disposizione è auspicabile l'utilizzo di **molteplici tecniche** di analisi e modelli in grado di influenzarsi a vicenda, nell'ottica di ottenere risultati più completi e precisi.

Basso Matteo 807628
Ferri Marco 807130

GRAZIE PER L'ATTENZIONE