Analisi della relazione tra popolarità musicale e sentiment sui social media: uno studio sugli artisti della Billboard Hot 100 Artists (2023)

De Pierro Giovanni
Università degli Studi di Salerno
Dipartimento di Informatica
Corso di Reti Geografiche: Struttura, Analisi e Prestazioni

***Abstract*—Il presente lavoro esplora la relazione tra popolarità musicale e sentiment espresso sui social media, analizzando gli artisti della classifica Billboard Hot 100 del 2023. Utilizzando scraper custom per Facebook e YouTube e applicando la sentiment analysis con VADER sui commenti degli utenti, lo studio valuta il gradimento del pubblico nei confronti degli artisti più ascoltati. I risultati mostrano che la maggior parte degli artisti ottiene sentiment neutro o positivo, mentre non emergono valori negativi, suggerendo che la popolarità non sempre corrisponde a un apprezzamento elevato. L'analisi evidenzia l'importanza di considerare il sentiment degli utenti come indicatore complementare alla popolarità, offrendo spunti per futuri studi sul rapporto tra visibilità mediatica e percezione sociale nel contesto musicale.**

I. Introduzione

La musica è da sempre uno degli intrattenimenti più accattivanti per le persone. Con la vasta disponibilità di piattaforme digitali e social network, gli artisti si trovano ad affrontare una nuova serie di sfide ed opportunità date dalla possibilità di una massiccia interazione con il pubblico e tra il pubblico stesso. Questo lavoro ha come obiettivo l'analisi del rapporto tra popolarità musicale e percezione sociale. L'intento è di valutare in che misura gli artisti più ascoltati siano anche associati a un sentiment positivo, quindi individuare se si tratti di artisti che uniscono ascolti elevati e forte gradimento oppure se la loro esposizione sia contrassegnata da critiche e opinioni negative. In questo modo si vuole contribuire a una comprensione più articolata della popolarità musicale contemporanea, mettendo in evidenza se il numero di ascolti non sempre coincida con il livello di apprezzamento.

II. Stato dell'Arte

La letteratura scientifica sul tema della musica digitale ha affrontato numerosi aspetti legati alla diffusione degli artisti attraverso le piattaforme di streaming e all'impatto dei social media nella formazione delle opinioni del pubblico [1,2]. Alcuni studi si sono concentrati sull'analisi delle classifiche di ascolto per descrivere l'evoluzione dei gusti musicali e le dinamiche di popolarità legate a generi o artisti emergenti. Diversi studi hanno analizzato i dati provenienti dai social network per esplorare i sentimenti degli utenti con lo scopo specifico di investigare la relazione tra le caratteristiche audio e la popolarità di una canzone [3]. Parallelamente, esiste una vasta produzione dedicata alla sentiment analysis applicata ai contenuti testuali dei social network, spesso utilizzata per monitorare il gradimento di prodotti, brand, eventi o figure pubbliche.

Nonostante questi contributi, dalla revisione della letteratura, non risultano studi che abbiano analizzato in modo specifico la relazione tra popolarità musicale degli artisti e sentiment espresso sui social media.

Il presente lavoro si colloca quindi in questa direzione al fine di indagare in che misura la visibilità degli artisti si accompagni a un apprezzamento positivo da parte del pubblico.

III. Background e Approccio

La sentiment analysis è una tecnica fondamentale nel natural language processing (NPL) che consiste nell'identificare ed estrarre informazioni soggettive da dati testuali. Questo processo è fondamentale per comprendere il tono emotivo delle parole, e ciò può fornire preziose informazioni in vari campi, come il marketing, l'analisi del feedback dei clienti e i mercati finanziari. Con questo lavoro si è voluto analizzare il sentiment del pubblico nei confronti degli artisti più popolari. Il primo passo è stato, quindi, individuare gli artisti da considerare.

È stata scelta come riferimento la classifica di fine anno `Billboard Hot 100 Artists` del 2023 che elenca i 100 artisti più popolari dell'anno, per poi recuperare i commenti degli utenti presenti su post e video, rispettivamente delle pagine Facebook e Youtube. I commenti sono stati ottenuti tramite scraping utilizzando scraper sviluppati appositamente con l'ausilio di Selenium WebDriver.

Selenium WebDriver è uno strumento open source per l'automazione dei test funzionali di applicazioni web. Fa parte della suite Selenium, insieme a Selenium IDE e Selenium Grid, e rappresenta l'evoluzione dei metodi di automazione basati su browser rispetto a strumenti più semplici come l'IDE. A differenza di Selenium IDE, WebDriver interagisce direttamente con il browser a livello nativo, simulando le azioni dell'utente come clic, inserimento di testo, selezione di elementi e navigazione tra pagine. Questa architettura consente un controllo più preciso della pagina web ed è stata particolarmente utile in questo contesto per navigare le piattaforme considerate.

Infine, per rilevare il sentiment dei commenti, è stato utilizzato VADER (Valence Aware Dictionary and Sentiment Reasoner). VADER è uno strumento molto conosciuto in questo ambito, progettato per funzionare bene sui testi dei social media ma efficace anche su altri formati di testo. Combina un lessico robusto con regole euristiche per catturare le sfumature di contesto, il che lo rende facile da usare e altamente accurato. Dato che VADER supporta soltanto la lingua inglese, è stata utilizzata la libreria fastText per rilevare la lingua di ogni commento e solo i commenti in inglese sono stati analizzati: fastText è una libreria open-source sviluppata da Facebook AI Research per l'apprendimento efficiente di rappresentazioni distribuite di parole e per la classificazione di testi. Uno degli utilizzi più diffusi della libreria è la language identification. In particolare, il modello lid.176.bin, rilasciato insieme a fastText, è un classificatore pre-addestrato in grado di riconoscere automaticamente la lingua di un testo. L'accuratezza è elevata anche su testi molto brevi, come potrebbero essere dei commenti, per questo è stato scelto come modello per questo lavoro.

I risultati ottenuti dall'analisi del sentiment verranno quindi mostrati e discussi in un capitolo dedicato.

Il capitolo seguente affronta nel dettaglio come è stato eseguito il lavoro di recupero e preprocessing dei dati, partendo dalla costruzione della lista degli artisti fino ad arrivare a commenti in formato testuale pronti per essere analizzati, per poi procedere con la sentiment analysis.

IV. Metodologia

Lista degli artisti. Il primo passo è stato ottenere la lista degli artisti da considerare. È stata recuperata la classifica Billboard consultabile all'indirizzo https://www.billboard.com/charts/year-end/2023/hot-100-artists/ ed elaborata tramite un semplice programma scritto in Java (`getNomi.java`) per estrarre i nomi dei 100 artisti ed inserirli in un file XML con la seguente struttura:

```
<names>
  <name>Morgan Wallen</name>
  <name>SZA</name>
  ...
<\names>
```

Sono stati, successivamente, recuperati le pagine Facebook e i canali Youtube degli artisti ed aggiunti come attributi al file XML. Questa operazione è stata eseguita manualmente al fine di assicurarsi che si trattasse di pagine ufficiali. Per gli artisti per i quali non è stato possibile recuperare le pagine, i valori degli attributi sono stati lasciati vuoti. Il file XML così costruito, disponibile come `nomi.xml`, presenta una struttura di questo tipo:

```
<names>
  <name youtube="morganwallen" facebook="
      morgancwallen">Morgan Wallen</name>
  <name youtube="sza" facebook="sza">SZA</name
      >
  ...
<\names>
```

Post e video. Una volta costruita la lista degli artisti con relative informazioni, il passo successivo è stato scrivere in Python due scraper, uno per Facebook e uno per Youtube. I due script, come altri programmi che verranno descritti in seguito, hanno una struttura molto simile dato che eseguono le stesse operazioni ma lavorando su due siti diversi. In particolare lo script `facebook_getPostsHtml.py` utilizza il file `nomi.xml` per eseguire un insieme di operazioni che verrà ripetuto per ogni artista.

```
1  driver.get("https://www.facebook.com/" +
       pagina + "/")
2  cookie_div = driver.find_element(By.XPATH, '
       //div/div/div/div/span/span[text()="
       Rifiuta cookie facoltativi"]')
3  cookie_div.click()
4  account_div = driver.find_element(By.XPATH,
       '//div[@aria-label="Chiudi"]')
5  account_div.click()
6
7  for i in range(2000):
8    scroll_script = "window.scrollTo(" + str(i
         *100) + ", +" + str((i+1)*100) + ");"
9    driver.execute_script(scroll_script)
10   time.sleep(10)
11   html_content = driver.page_source
12   with open(file_html, "w", encoding="utf-8"
         ) as file:
13       file.write(html_content)
```

Come si può osservare nel codice mostrato sopra, viene recuperata la pagina Facebook utilizzando le informazioni presenti nel file XML, e vengono eseguite alcune operazioni che servono a ripulire la pagina da messaggi del sito che informano, per esempio, sull'utilizzo dei cookie (righe 1-5). Le righe successive si occupano di scorrere la pagina per un pò di tempo in modo da visualizzare un numero sufficiente di post, poi la pagina viene salvata come file HTML. Questa operazione viene eseguita per ogni artista, in modo da ottenere alla fine per ogni artista un file HTML che abbia in sè contenuti i link degli ultimi post Facebook pubblicati. Allo stesso modo lo script `youtube_getVideosHtml.py` esegue la stessa operazione per i video Youtube, quindi cicla sulla lista degli artisti e per ognuno di loro recupera gli ultimi video pubblicati.

A questo punto, tramite altri due programmi scritti in Java (`facebook_getPostLinks.java` e `youtube_getVideosLinks.java`), vengono navigati i file HTML per recuperare i link ai post e ai video pubblicati e salvarli in file di testo.

Per quanto riguarda i video Youtube è stato considerato opportuno selezionare gli ultimi 25 video degli artisti che avessero almeno 20 commenti. Sono stati quindi esclusi gli artisti che non raggiungessero il numero minimo di video assieme al numero minimo di commenti. I due valori sono stati scelti in modo da cercare di ottenere un buon compromesso tra l'avere molti artisti che rientrassero nei parametri e allo stesso tempo avere una buona quantità di commenti da analizzare, magari distribuita anche su un intervallo di tempo sufficientemente ampio perchè si avesse una visione dell'opinione degli

utenti che non fosse limitata ad un periodo specifico. In questo modo si sono potuti considerare commenti anche di qualche anno precedenti al 2023. Un lavoro simile è stato effettuato riguardo i post Facebook per i quali sono stati considerati un numero minimo di post di 15 e un numero minimo di commenti per post di 15.

Commenti. Ottenuti i link dei post e dei video, altri due scraper Python (`facebook_getCommentsHtml.py` e `youtube_getCommentsHtml.py`) navigano le pagine per poter visualizzare i commenti degli utenti. Anche in questo caso vengono salvati i file HTML delle pagine, e altri due programmi Java (`facebook_estraiCommenti.java` e `youtube_estraiCommenti.java`) estraggono i commenti e li organizzano in file XML con la seguente struttura:

```
<Name_cardib _01="1235" _02="464" ... >
  <File_1.html commenti="1235">
    <Commento_1 testo="Still the song was not
        a hit"/>
    <Commento_2 testo="Raman is grazy you dont
         have to like you're comment only
        cardi b video"/>
    ...
  </File_1.html>
  <File_2.html commenti="464">
    ...
  </File_2.html>
  ...
</Name_cardib>
```

Per ogni artista, in questo modo, possono esistere fino a due file XML, uno contenente i commenti Facebook e uno quelli Youtube, ed entrambi hanno la struttura mostrata sopra. L'elemento radice rappresenta l'artista ed ha come attributi i file considerati per l'estrazione dei commenti (ovvero post o video) numerati dal più recente al meno recente e per ognuno il relativo valore indica il numero di commenti che sono stati estratti per quel file. Gli elementi figli della radice rappresentano i file, ed ognuno di essi ha tanti figli quanti sono i commenti che contiene. Ogni elemento che rappresenta un commento ha un'attributo che contiene il contenuto del commento.

Preprocessing. La fase di raccolta dati si è così conclusa con l'organizzazione dei commenti in file XML. Parte del preprocessing si è svolto durante questa fase stessa dato che durante l'estrazione dei commenti le emoji e alcune immagini o alcuni video che avevano una rappresentazione diversa da un URL, non potevano essere considerati testo e quindi non venivano inclusi nel testo dei commenti. Un'ulteriore controllo è servito soltanto per rimuovere eventuali URL ed eliminare commenti che magari, successivamente alla rimozione degli URL, non contenevano più testo se non delimitatori lessicali. Sono stati, quindi, sviluppati gli script `calculate_sentiment_facebook.py` e `calculate_sentiment_youtube.py` che, prima di procedere con l'analisi, si occupano di eseguire questa ultima operazione di preprocessing.

Sentiment analysis. Terminate le fasi di raccolta e preparazione dei dati, i due script presentati nel paragrafo precedente, eseguono l'analisi del sentiment e salvano i risultati.

```
1  def detect_lang(txt, top_k=3, min_ratio
       =0.95):
2    txt = txt.lower().strip()
3    labels, probs = lang_model.predict(txt.
         replace("\n", " ")[:500], k=top_k)
4    max_prob = probs[0]
5    for label, prob in zip(labels, probs):
6      language_code = label.replace("__label__
           ", "")
7      if language_code == "en" and prob >=
           min_ratio * max_prob:
8        return "en"
9
10   return labels[0].replace("__label__", "")
```

Per ogni commento si utilizza il modello lid.176.bin con la libreria fastText per rilevare la lingua. Prima di tutto si converte tutto il testo in minuscolo e si calcolano le tre lingue più probabili considerando i primi 500 caratteri del commento (righe 2-3). Si salva la probabilità della lingua più probabile, quindi quella massima, e si esegue un controllo: invece di procedere con l'analisi soltanto se la lingua più probabile è l'inglese, si procede anche se la probabilità che sia inglese è non più piccola del 95% rispetto alla lingua più probabile (riga 7). Questo permette di analizzare anche commenti dove la probabilità che siano scritti in inglese non è la massima ma si avvicina molto. Empiricamente è stato osservato che si riuscivano, in questo modo, ad includere nell'analisi anche alcuni commenti che erroneamente il modello non considerava scritti in inglese, limitando comunque i falsi positivi.

Dopo questo controllo gli script procedono con l'analisi e i risultati vengono salvati in file XML con la seguente struttura:

```
<Artisti>
  <Artista nome="_1997JungKook" sentiment="
      0.619">
    <Post nome="File_1" sentiment="0.798"/>
    <Post nome="File_2" sentiment="0.707"/>
    ...
  </Artista>
  <Artista nome="_21Savage" sentiment="0.111">
    ...
  </Artista>
  ...
</Artisti>
```

Si hanno così due file XML (`sentiment_totali_facebook.xml` e `sentiment_totali_youtube.xml`) ognuno dei quali contiene per ogni artista il sentiment ottenuto aggregando i sentiment sui post o video, e ognuno di questi sentiment è stato a sua volta ottenuto aggregando i sentiment di tutti i commenti analizzati relativi a quel file. Tutti i sentiment sono stati considerati secondo il punteggio complessivo di sentiment, normalizzato tra -1 e 1. Gli autori di VADER interpretano questo punteggio come neutro se compreso tra -0.05 e 0.05, e positivo o negativo rispettivamente se maggiore di 0.05 o minore di -0.05. I risultati ottenuti verranno presentati nel capitolo seguente.

V. Risultati

I risultati ottenuti sono stati rappresentati sotto forma di grafico tramite gli ultimi due script `print_sentiment_facebook.py` e `print_sentiment_youtube.py`. L'asse delle ascisse indica gli artisti seguendo l'ordine della classifica, quindi con una numerazione da 1 a 100, mentre quella delle ordinate indica il punteggio complessivo di sentiment, con valori tra -1 e 1. Per gli artisti per i quali non è stato possibile recuperare la pagina Facebook o il canale Youtube ufficiali, o che non hanno superato i requisiti minimi di numero di post o video e commenti, è presente la dicitura NaN che sta ad indicare un'assenza di valore di sentiment, dato che in quei casi non è stata eseguita l'analisi.

Il sentiment Facebook (Fig. 1) varia da valori compresi tra 0 e 0.05, che possono essere considerati neutri, a valori leggermente superiori lo 0.6, quindi non sono presenti sentiment negativi. È possibile osservare che per 32 artisti il NaN indica l'assenza di un sentiment.

Il sentiment Youtube (Fig. 2) in modo simile varia da valori tra 0 e 0.05, quindi neutri, a valori leggermente superiori lo 0.5. Anche in questo caso non sono presenti sentiment negativi. Per 13 artisti non è stato possibile calcolare il sentiment.

VI. Conclusioni e Sviluppi Futuri

Il presente studio ha analizzato la relazione tra popolarità musicale e sentiment espresso sui social media, concentrandosi sugli artisti presenti nella classifica Billboard Hot 100 Artists del 2023.

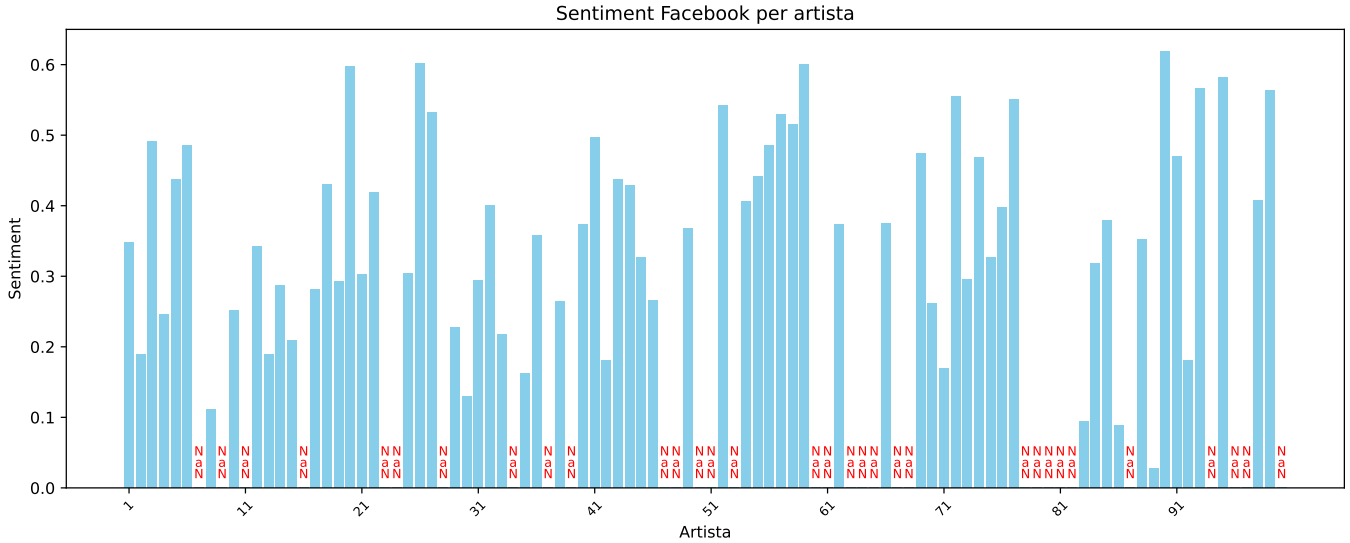

Fig. 1: Sentiment Facebook calcolato per gli artisti

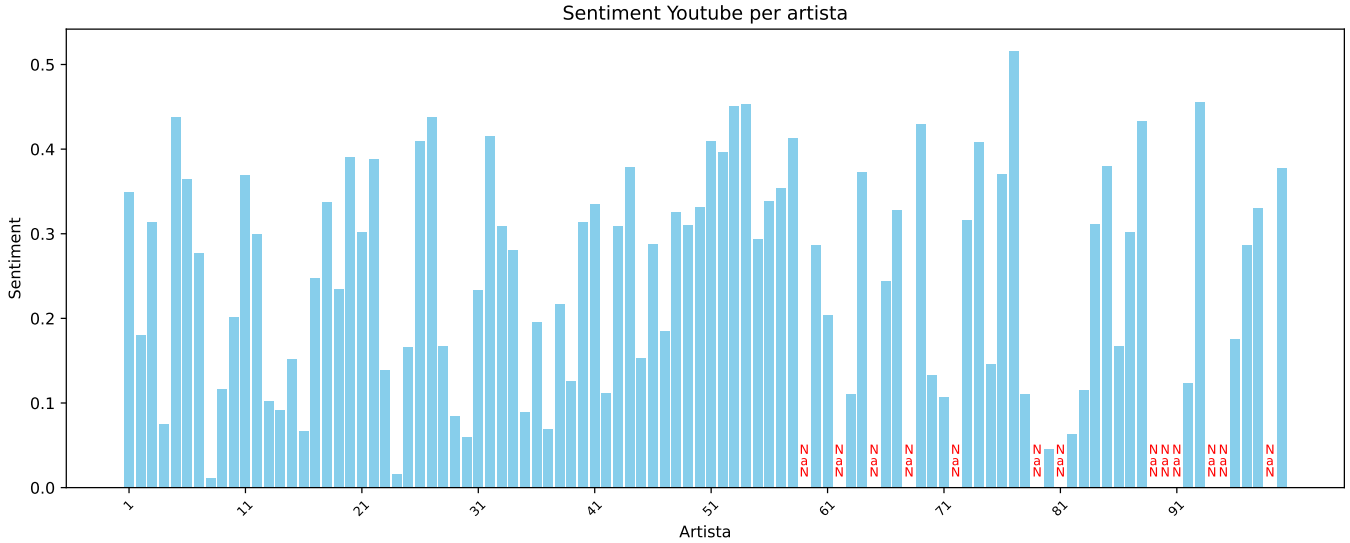

Fig. 2: Sentiment Youtube calcolato per gli artisti

Attraverso l'uso di scraper custom per Facebook e Youtube e l'applicazione della sentiment analysis con VADER, è stato possibile valutare l'apprezzamento del pubblico nei confronti degli artisti più ascoltati.

I risultati evidenziano che la maggior parte degli artisti ottiene sentiment positivo e solo una piccola parte un sentiment neutro, mentre non emergono valori negativi. Inoltre, l'analisi mostra come non tutti gli artisti più popolari abbiano un sentiment elevato, indicando che la popolarità non coincide necessariamente con il gradimento del pubblico. Questo suggerisce che, nel contesto musicale contemporaneo, l'ascolto e la visibilità mediatica possono essere influenzati da fattori diversi dal consenso emotivo espresso dai fan.

Studi futuri potrebbero ampliare l'analisi includendo altre piattaforme social, analizzare il sentiment anche in lingue diverse dall'inglese o confrontare l'evoluzione del sentiment nel tempo per comprendere meglio le dinamiche di popolarità e gradimento degli artisti.

BIBLIOGRAFIA

[1] Rompolas, G.: Exploiting time-series analysis to predict customers' behavioural dynamics in social networks. In: 13th International Conference on Information, Intelligence, Systems & Applications, IISA 2022, Corfu, Greece, 18-20 July 2022, pp. 1-7. IEEE (2022)

[2] Rompolas, G., Karavoulia, K.: The use of the twitter graph for analyzing user emotion for businesses. In: Proceedings of the CIKM 2021 Workshops co-located with 30th ACM International Conference on Information and Knowledge Man- agement (CIKM 2021), Gold Coast, Queensland, Australia, 1-5 November 2021. CEUR Workshop Proceedings, vol. 3052. CEUR-WS.org (2021)

[3] Gulmatico, J.S., Susa, J.A.B., Malbog, M.A.F., Acoba, A., Nipas, M.D., Mindoro, J.N.: SpotiPred: a machine learning approach prediction of spotify music popular- ity by audio features. In: 2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T), pp. 1-5. IEEE (2022)