



UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica

Corso di Laurea Triennale in Informatica

TESI DI LAUREA

# **Approccio Innovativo al Data Mining per l'Estrazione delle Interfacce Grafiche nelle Applicazioni Mobile**

RELATORE

Prof. Fabio Palomba

Dott. Vincenzo De Martino

CANDIDATO

**Luca Pastore**

Matricola: 0512113201

Anno Accademico 2022-2023

*Questa tesi è stata realizzata nel*

sesa<sup>lab</sup>  
SOFTWARE ENGINEERING  
SALERNO

*A mia madre e a mio fratello per la vita che mi hanno donato e per essermi sempre stati vicini.*

*A mio padre, spero tu possa essere orgoglioso di me*

## **Abstract**

Questa tesi si concentra sull'esecuzione di un approfondito processo di mining delle interfacce grafiche delle applicazioni Android, con l'obiettivo di estrarre informazioni significative. Questa metodologia è accompagnata da un'analisi avanzata dei dati estratti, finalizzata a rilevare pattern e relazioni di rilievo tra le varie applicazioni considerate. I risultati ottenuti consentono una comprensione profonda della popolarità di ciascuna applicazione e offrono una panoramica completa sull'ambito e sull'evoluzione di ogni singola interfaccia grafica. Questo studio non solo contribuisce al campo del design delle interfacce, ma fornisce anche una prospettiva innovativa per valutare in modo approfondito l'esperienza utente e l'evoluzione delle applicazioni nell'ecosistema mobile. La combinazione di estrazione dati e analisi avanzata apre nuove strade per comprendere in modo più completo la relazione tra design visuale e successo delle applicazioni, fornendo insight preziosi per sviluppatori e designer.

---

## Indice

---

<b>Elenco delle Figure</b>	<b>iii</b>
<b>1 Introduzione</b>	<b>1</b>
1.1 Contesto e Motivazioni . . . . .	2
1.2 Risultati . . . . .	3
1.3 Struttura della tesi . . . . .	3
<b>2 Background e Stato dell'arte</b>	<b>5</b>
2.1 Introduzione al Data Mining . . . . .	5
2.1.1 Fasi fondamentali . . . . .	8
2.2 Introduzione alle interfacce grafiche . . . . .	11
2.2.1 Ruolo delle interfacce grafiche nelle applicazioni mobile . . .	13
2.2.2 Esempi di estrazione di un interfaccia nella letteratura odierna	15
<b>3 Metodologia di estrazione e analisi delle interfacce</b>	<b>19</b>
3.1 Panoramica generale . . . . .	19
3.2 Prima fase: raccolta delle applicazioni da testare . . . . .	20
3.3 Seconda fase: estrazione delle interfacce . . . . .	20
3.3.1 Android Debug Bridge e UiAutomator . . . . .	21
3.4 Algoritmo BOCP-BOCC . . . . .	23

<b>4</b>	<b>Risultati dello studio</b>	<b>26</b>
4.1	OP1: Quali sono le caratteristiche delle app analizzate . . . . .	26
4.2	OP2: Quali sono i colori più usati nelle applicazioni considerate . . .	30
<b>5</b>	<b>Conclusione e sviluppi futuri</b>	<b>32</b>
	<b>Bibliografia</b>	<b>34</b>

---

## Elenco delle figure

---

2.1	Processo di estrazione della conoscenza . . . . .	9
4.1	Frequenza delle categorie di ciascuna app . . . . .	27
4.2	Frequenza del numero di download di ciascuna app . . . . .	28
4.3	Frequenza del numero di recensioni di ciascuna app . . . . .	28
4.4	Frequenza del numero di stelle di ciascuna app . . . . .	29
4.5	Analisi dei colori delle interfacce . . . . .	30

# CAPITOLO 1

---

## Introduzione

---

Negli ultimi anni, l'evoluzione della tecnologia mobile ha rivoluzionato radicalmente il modo in cui interagiamo con i dispositivi digitali, trasformando gli smartphone in strumenti indispensabili per la vita quotidiana. Questa crescente dipendenza ha portato a una richiesta sempre maggiore di interfacce grafiche intuitive ed efficienti, in grado di fornire un'esperienza utente ottimale.

In questo contesto, il Data Mining emerge come un potente strumento per analizzare e estrarre informazioni significative da grandi set di dati, con l'obiettivo di identificare modelli e tendenze che possono essere impiegati per migliorare le prestazioni delle interfacce grafiche mobile.

Nel contesto in continua evoluzione delle applicazioni per dispositivi Android, la comprensione delle interfacce grafiche riveste un ruolo fondamentale per valutare l'impatto e la popolarità di tali applicazioni. Questa ricerca si inserisce in un quadro più ampio di analisi, concentrando la sua attenzione sul mining delle interfacce grafiche e sull'elaborazione avanzata dei dati estratti al fine di individuare pattern e relazioni significative tra le diverse applicazioni prese in esame.

L'approccio si basa su una metodologia di estrazione mirata, che ci permette di ottenere un campione rappresentativo delle interfacce grafiche di applicazioni Android. Questo processo di mining è supportato da un'analisi approfondita dei dati estratti,



consentendo una valutazione dettagliata della popolarità e dell'evoluzione di ogni singola applicazione nel panorama Android. Un aspetto chiave di questa indagine è l'analisi dei colori predominanti nelle interfacce, un elemento spesso trascurato ma cruciale nell'influenzare l'esperienza dell'utente.

I risultati di questa ricerca offrono una visione chiara e completa della popolarità delle applicazioni Android e dell'andamento del loro design. La possibilità di comprenderne le dinamiche interne e di identificare pattern comuni ha fornito un quadro informativo essenziale per gli sviluppatori, gli studiosi e gli utenti stessi.

## **1.1 Contesto e Motivazioni**

La crescente complessità delle applicazioni mobili e la varietà di dispositivi su cui vengono eseguite pongono sfide significative nella progettazione di interfacce grafiche efficaci e adattabili. La necessità di fare uno studio sull'estrazione di interfacce grafiche diventa rilevante per gli sviluppatori e i progettisti di applicazioni mobili, in quanto si vuole offrire un'esperienza utente superiore. Tuttavia, l'estrazione manuale delle interfacce è un compito arduo e richiede un notevole sforzo e risorse umane. È qui che entra in gioco l'automazione attraverso l'impiego di algoritmi che permettono di estrarre le interfacce grafiche in modo da avere una panoramica generale della struttura di una generica applicazione. Questa tesi è motivata dall'esigenza di sfruttare il potenziale del Data Mining per automatizzare e migliorare il processo di estrazione delle interfacce grafiche mobili, consentendo una comprensione profonda della popolarità di ciascuna applicazione e offrendo una panoramica completa sull'ambito e sull'evoluzione di ogni singola interfaccia grafica.

## 1.2 Risultati

La ricerca condotta sull'estrazione delle interfacce grafiche da applicazioni differenti ha evidenziato risultati che meritano una riflessione più approfondita. L'impiego di algoritmi di Data Mining e tecniche di estrazione ha dimostrato di essere un approccio promettente per migliorare l'usabilità e l'esperienza visiva degli utenti che usano dispositivi Android.

Uno degli aspetti significativi emersi dalla ricerca è l'efficacia dell'algoritmo implementato nell'estrazione dettagliata delle componenti pixel delle interfacce grafiche delle applicazioni mobili. Questo approccio ha permesso di analizzare in modo approfondito la struttura e la disposizione degli elementi visivi, consentendo di identificare pattern e relazioni significative che possono influenzare l'esperienza degli utenti. Questo approccio su misura ha contribuito a migliorare la percezione e la distinguibilità dei colori, creando un ambiente visivo più accessibile e accogliente per diverse categorie di utenti.

## 1.3 Struttura della tesi

Questa tesi è strutturata nel seguente modo:

- **Background e Stato dell'arte:** In questo capitolo verrà fatta una breve introduzione al concetto di Data Mining, alle interfacce grafiche, ponendo un focus maggiore sul ruolo che assumono le interfacce grafiche all'interno dei dispositivi mobile, ma allo stesso tempo di quali sono, ad oggi, gli studi presenti in letteratura, in modo da evidenziare il contributo di questa tesi rispetto ai lavori precedenti.
- **Metodologia di estrazione e analisi delle interfacce:** In questo capitolo verrà definita la metodologia alla base di questo studio. Tratterà della fase di raccolta delle applicazioni da testare, input fondamentale per iniziare il nostro studio. Seguirà la fase di estrazione delle interfacce tramite un approccio automatico ed infine seguirà lo sviluppo dell'algoritmo BOCP-BOCC fondamentale per estrar-

re le componenti pixel dalle interfacce acquisite. Inoltre, verranno evidenziati gli obiettivi alla base di questo studio.

- **Risultati dello studio:** In questo capitolo verranno visualizzati i risultati dello studio, rispondendo agli obiettivi di ricerca prefissati. Verranno visualizzati i grafici che mostrano l'analisi della distribuzione dei dati all'interno del dataset di applicazioni e un'analisi dei colori sulla base delle interfacce acquisite, in modo da avere una panoramica dettagliata sulle applicazioni scelte come test all'inizio dello studio.
- **Conclusioni e Prospettive Future:** In questo capitolo si parlerà delle conclusioni e dei possibili sviluppi futuri alla base di questo studio. Verrà inoltre affrontato il tema della privacy e verranno definite tecniche, che potranno essere implementate in futuro, che permetteranno di analizzare le diverse applicazioni senza incorrere nella violazione della privacy

## CAPITOLO 2

---

### Background e Stato dell'arte

---

#### 2.1 Introduzione al Data Mining

La crescente disponibilità di dati nella società attuale ha evidenziato la necessità di disporre di determinati meccanismi che consentano la loro analisi. La grossa quantità di dati di cui disponiamo oggi costituisce un notevole potenziale che può essere valorizzato, permettendo di interpretare i dati. Secondo l'articolo "Big Data: a Review"[1], i big data e la loro analisi sono al centro della scienza e dell'economia moderna. Essi vengono archiviati in database che crescono in modo massivo e diventano difficili da catturare, formare, memorizzare, gestire, condividere, analizzare e visualizzare tramite gli strumenti software di database tipici. Fino al 2003, erano stati creati 5 exabyte<sup>1</sup> di dati da parte dell'uomo. Oggi questa quantità di informazioni viene creata in soli due giorni. Nel 2012, il mondo digitale dei dati si è espanso fino a raggiungere 2,72 zettabyte<sup>2</sup>. La quantità di questi dati è aumentata notevolmente, arrivando a circa 8 zettabyte di dati nel 2015. IBM indica che ogni giorno vengono

---

<sup>1</sup>Un exabyte (EB) è un'unità di misura per l'archiviazione di informazioni digitali e la capacità dei dati. Un exabyte equivale a 1,024 petabyte

<sup>2</sup>"Zetta" è un cosiddetto prefisso decimale, il significato del quale è definito nel sistema internazionale di unità di misura, e che corrisponde a  $10^{21}$

creati 2,5 exabyte di dati, corrispondenti anche al 90% dei dati prodotti negli ultimi due anni. Un personal computer, da solo, non riuscirebbe a contenere una grandissima quantità di dati, quindi sarebbero necessari miliardi e miliardi di computer per archiviare tutti i dati del mondo. Ad oggi, solo Google possiede più di un milione di server in tutto il mondo. Ogni giorno vengono inviati 10 miliardi di messaggi di testo. Nel 2020, 50 miliardi di dispositivi sono stati connessi alle reti e a Internet. Nel 2012, "The Human Face of Big Data" è stato realizzato come progetto globale, incentrato sulla raccolta, visualizzazione e analisi in tempo reale di grandi quantità di dati. Secondo questo progetto mediatico, sono emersi numerosi dati statistici. Facebook conta 955 milioni di account attivi mensilmente, utilizzando 70 lingue, con 140 miliardi di foto caricate, 125 miliardi di connessioni amichevoli, 30 miliardi di contenuti e 2,7 miliardi di like e commenti postati ogni giorno. Ogni minuto, vengono caricati 48 ore di video e ogni giorno si registrano 4 miliardi di visualizzazioni su YouTube. Google supporta numerosi servizi, monitorizzando 7,2 miliardi di pagine al giorno e processando 20 petabyte<sup>3</sup> di dati ogni giorno, traducendo anche in 66 lingue. Ogni 72 ore vengono pubblicati 1 miliardo di tweet da parte di oltre 140 milioni di utenti attivi su Twitter. Vengono creati 571 nuovi siti web ogni minuto della giornata. Entro il prossimo decennio, il numero di informazioni aumenterà di 50 volte, tuttavia il numero di specialisti delle tecnologie dell'informazione in grado di gestire tutti questi dati aumenterà solo di 1,5 volte. In tal contesto, il data mining si pone come obiettivo di scoprire regolarità o relazioni non note a priori allo scopo di ottenere un risultato chiaro e utile per affrontare problemi di decisione attraverso l'uso di processi di selezione, esplorazione e modellazione di grandi quantità di dati.

"Data mining is a set of processes by which knowledge is extracted from huge amounts of data. Data mining is used to extract useful patterns and hidden information from this data. Machine learning techniques help in the comprehension of the hidden knowledge in the data. Data mining is considered an important field of research and is used in many different fields such as fraud detection, financial banking, education, healthcare, agriculture, industry, etc. In this paper, we will highlight some

---

<sup>3</sup>Un petabyte rappresenta 1,024 terabyte o un milione di gigabyte

fundamentals of data mining and its applications. Also, we will conduct a comparative study among different reviews, combining literary studies that employed data mining techniques in various fields and reviewing the latest developments in this field."

Nell'articolo "A Review: Data Mining Techniques and Its Applications" [2], il data mining è definito come un insieme di processi attraverso i quali conoscenze vengono estratte da grandi quantità di dati.

Il data mining è utilizzato per estrarre modelli utili e informazioni nascoste dai dati, con l'aiuto delle tecniche di machine learning per comprendere le conoscenze nascoste nei dati. Le aziende, attraverso software di data mining, possono cercare tra i dati grezzi raccolti, le informazioni utili per il proprio business come: sviluppare strategie di marketing più efficaci, sapere di più sui propri clienti, aumentare le vendite e ridurre i costi.

"Data mining is a collection of methods and techniques that allow the analysis of large amounts of data to extract knowledge through the use of algorithms and techniques derived from the fields of machine learning, statistics, and database management systems"

L'impiego dei sistemi di data mining diventa imprescindibile per svariati motivi. Inizialmente, si assiste a una costante crescita della mole di dati archiviati su supporti informatici, tra cui informazioni web, dati di sistemi di e-commerce, dettagli sugli acquisti e ricevute fiscali, oltre a transazioni bancarie relative alle carte di credito. Parallelamente, è rilevante considerare l'incremento delle capacità dell'hardware e la sua riduzione di costo. Dal punto di vista aziendale, la crescente competizione sottolinea il valore cruciale dell'informazione nel superare i rivali. Nel settore scientifico, la produzione e l'accumulo di dati avvengono a ritmo accelerato (nell'ordine dei gigabyte all'ora), provenienti da sensori su satelliti, telescopi, microarray generanti espressioni genetiche e simulazioni scientifiche che generano terabyte di dati. Le metodologie tradizionali si dimostrano inadeguate per gestire queste enormi quantità di dati grezzi; il data mining, al contrario, fornisce ai ricercatori la possibilità di classificare, segmentare e formulare ipotesi su tali dati, contribuisce a identificare tendenze, modelli e relazioni nei dati che possono essere utilizzati per ottimizzare processi,

migliorare la produttività e fornire servizi più efficienti e personalizzati. Ulteriori ragioni per adottare il data mining derivano dal fatto che molte informazioni nei dati non emergono immediatamente; le analisi condotte manualmente possono richiedere settimane per scoprire informazioni utili, e una significativa parte dei dati rimane spesso non analizzata.

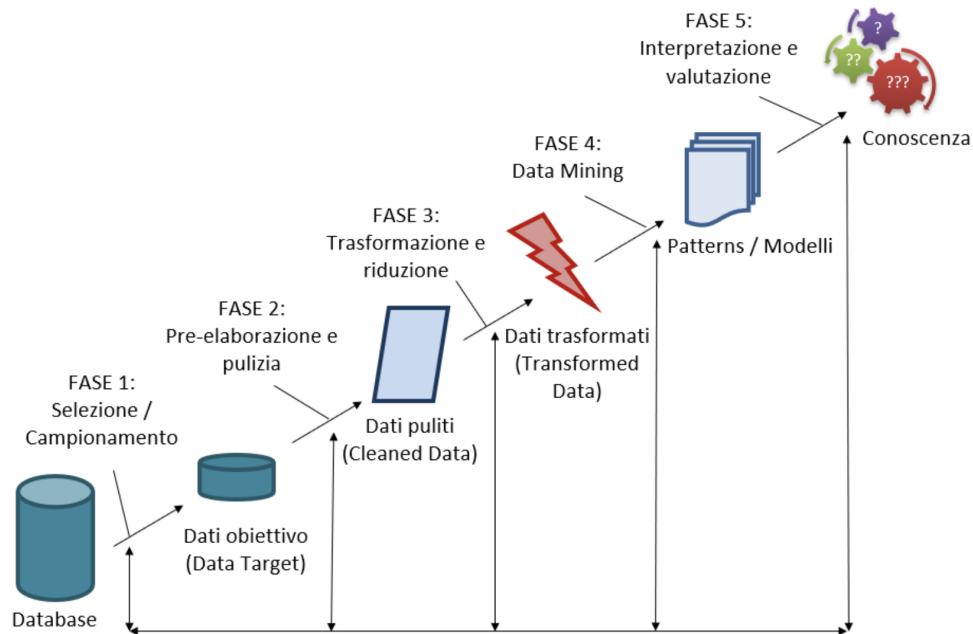
### 2.1.1 Fasi fondamentali

Secondo il libro "*Una breve introduzione alle tecniche di Data Mining*", il Data Mining comprende diverse fasi fondamentali per l'identificazione efficace di pattern nei dati. È essenziale che il database sia organizzato in modo da integrare dati che riflettano accuratamente la realtà oggetto dell'analisi, permettendo così un'analisi efficace per recuperare le informazioni rilevanti. È importante eliminare ridondanze e informazioni superflue che potrebbero interferire con l'analisi e la valutazione dei dati. Un obiettivo chiave del Data Mining è scoprire informazioni nascoste nei dati per migliorare la conoscenza.

Il processo di Knowledge Discovery in Database (KDD) mira a scoprire informazioni nascoste nei dati per migliorare la conoscenza. Le fasi del processo di KDD includono:

- **Selezione:** Definizione di criteri specifici e estrazione dei dati in base all'obiettivo dell'analisi, spesso chiamata campionamento in statistica.
- **Pre-elaborazione:** Analisi e pulizia dei dati per rimuovere elementi che potrebbero ostacolare la definizione di pattern. I dati possono essere trasformati per essere utilizzati con algoritmi matematici, ad esempio, trasformando variabili come il sesso in valori numerici (0 per maschio, 1 per femmina).
- **Trasformazione:** Arricchimento dei dati con ulteriori informazioni, ad esempio unendo dati da diversi database per massimizzare la disponibilità di informazioni.
- **Data Mining:** Analisi dei dati con l'obiettivo di estrarre regole e modelli significativi.

- **Interpretazione e Valutazione:** Valutazione e interpretazione dei modelli generati per fornire conoscenze utili che possono supportare le decisioni, come la previsione di trend o la classificazione di nuovi elementi.



**Figura 2.1:** Processo di estrazione della conoscenza

Inoltre, nel libro, vengono definiti due approcci principali del Data Mining per l'analisi dei dati: il metodo di verifica e il metodo di scoperta.

### 1. Metodo di verifica:

Questo metodo si basa sull'ipotesi dell'utente e utilizza i dati per confermare o smentire tali ipotesi senza generare nuove informazioni. In pratica, si parte da ipotesi predefinite e si verifica se i dati raccolti le confermano o meno. Tuttavia, questo approccio non porta alla creazione di nuove conoscenze, ma serve piuttosto a validare le ipotesi esistenti. Per superare questa limitazione, è stata introdotta la tecnica chiamata *Exploratory Data Analysis* in statistica, che consente di esplorare i dati in modo più approfondito senza preconcetti.



## 2. Metodo di scoperta:

Questo metodo consente al sistema di individuare informazioni nascoste in modo completamente automatico, senza l'intervento diretto dell'utente. L'obiettivo è quello di identificare similitudini e generalizzazioni nei dati senza presupporre ipotesi specifiche. Questo approccio è particolarmente utile per rivelare pattern e relazioni non evidenti a una prima analisi dei dati.

Le applicazioni operative del Data Mining includono diverse tecniche:

- **Scoring system (predictive modelling):** utilizzato per assegnare punteggi in base alla probabilità di un evento futuro, come nel credit scoring per valutare la solvibilità finanziaria di un individuo o di un'azienda.
- **Segmentazione della clientela (customer profiling):** impiegato per suddividere i clienti in gruppi omogenei in base a comportamenti o caratteristiche socio-demografiche, al fine di personalizzare le strategie di marketing.
- **Market basket analysis (affinity analysis):** analizza i prodotti acquistati insieme per ottimizzare la disposizione sugli scaffali e migliorare le vendite.
- **Rilevazione di frodi (fraud detection):** identifica profili sospetti che potrebbero essere coinvolti in attività fraudolente o comportamenti non conformi.
- **Analisi degli abbandoni (churn analysis):** individua i clienti che potrebbero abbandonare un prodotto o servizio, consentendo di adottare misure preventive.
- **Text mining:** applicato all'analisi di documenti testuali per estrarre informazioni significative e raggruppare i documenti in base agli argomenti trattati.
- **Web Mining:** comprende tecniche come la click-stream analysis per comprendere il comportamento degli utenti online e il dynamic content targeting per personalizzare la visualizzazione delle pagine web in base al profilo del visitatore.

Queste tecniche consentono di estrarre valore dai dati, identificare pattern significativi e prendere decisioni informate basate sull'analisi approfondita dei dati disponibili.

## 2.2 Introduzione alle interfacce grafiche

Le interfacce grafiche rappresentano un elemento fondamentale nel mondo dell'informatica, trasformando radicalmente il modo in cui gli utenti interagiscono con i dispositivi digitali. Queste interfacce sono essenziali per la nostra esperienza digitale in un mondo sempre più tecnologico, rendendo accessibili complessi sistemi informatici a chiunque, indipendentemente dal livello di esperienza tecnica. Immaginiamo di poter usare un dispositivo senza dover memorizzare comandi complicati o imparare un certo linguaggio; la forza delle interfacce grafiche si manifesta in questo momento. Queste interfacce offrono un accesso visuale e interattivo tramite icone intuitive a finestre che aprono mondi digitali. Un semplice clic del mouse o un tocco sullo schermo nasconde una complessa architettura di design in cui ogni componente è stato ponderato per migliorare la fruibilità e l'esperienza dell'utente.

L'introduzione delle interfacce grafiche ha rappresentato una svolta epocale nell'evoluzione dei sistemi informatici, come analizzato nel documento "In the Beginning was the Command Line" di Neal Stephenson [3]. Prima di questo cambiamento rivoluzionario, l'interazione con i computer avveniva esclusivamente attraverso l'uso di linee di comando testuali, un ambiente dominato dalla scrittura e dalla logica dei linguaggi di programmazione. Le interfacce a riga di comando richiedevano una conoscenza approfondita dei comandi e dei linguaggi informatici, limitando l'accessibilità e l'usabilità dei sistemi informatici a una cerchia ristretta di utenti tecnici. Tuttavia, con l'avvento delle interfacce grafiche, questo paradigma è stato rivoluzionato: gli utenti hanno potuto abbandonare la complessità delle righe di comando per abbracciare un'interfaccia visuale e intuitiva, basata su icone, finestre e menu interattivi. L'introduzione delle interfacce grafiche ha reso l'interazione con i computer più accessibile e user-friendly, consentendo a un pubblico più ampio di utilizzare la tecnologia informatica in modo più semplice e immediato.

Questo passaggio ha aperto nuove possibilità di interazione e ha contribuito a democratizzare l'accesso alla tecnologia, trasformando i computer da strumenti accessibili esclusivamente a tecnici di settore, a dispositivi utilizzabili da un vasto numero di utenti.

Le interfacce grafiche hanno reso la tecnologia informatica visivamente più attraente e hanno semplificato le operazioni quotidiane, consentendo agli utenti di eseguire compiti complessi in modo più efficiente e intuitivo.

Nell'articolo "*User Interface: A Personal View*" di Alan Kay [4], si ha una visione completa dell'evoluzione della progettazione dell'interfaccia utente e del suo impatto sull'interazione uomo-macchina. Il lavoro di Kay, basato sulle sue esperienze presso l'Università dello Utah e presso lo Xerox PARC, evidenzia i progressi rivoluzionari culminati nello sviluppo dell'interfaccia grafica utente (GUI<sup>4</sup>). Il suo approccio enfatizzava l'importanza di comprendere la percezione e la cognizione umana, attingendo alle teorie di psicologi come Jean Piaget e Jerome Bruner.

La consapevolezza di Kay che i computer potevano servire come mezzo per l'espressione creativa e l'apprendimento portò alla concettualizzazione del **Dynabook**, un prototipo di personal computer che fornisce una rappresentazione dinamica di informazioni e concetti. Inoltre, ha sottolineato l'importanza di integrare molteplici mentalità nella progettazione dell'interfaccia utente, sottolineando la necessità di una perfetta integrazione di questi processi cognitivi.

L'esplorazione di Kay della relazione tra cognizione umana e progettazione dell'interfaccia utente sottolinea il ruolo vitale dell'apprendimento e dell'arricchimento ambientale nel modellare la progettazione di sistemi interattivi. Le sue osservazioni sulle distinte mentalità della cognizione umana e la loro rilevanza per la progettazione dell'interfaccia utente offrono preziose informazioni sui principi fondamentali alla base dell'efficace interazione uomo-macchina. Inoltre, la sua enfasi sull'unificazione di elementi concreti e astratti nella progettazione dell'interfaccia utente riflette un approccio fondamentale volto a migliorare l'esperienza dell'utente e a semplificare le interazioni complesse.

---

<sup>4</sup>In informatica è un tipo di interfaccia utente che consente l'interazione uomo-macchina in modo visuale utilizzando rappresentazioni grafiche (es. widget) piuttosto che utilizzando i comandi tipici di un'interfaccia a riga di comando.

Nel complesso, "*User Interface: A Personal View*" di Alan Kay fornisce una profonda esplorazione delle dimensioni storiche, teoriche e pratiche della progettazione dell'interfaccia utente, evidenziando il suo forte impatto sulla cognizione umana e sull'interazione con i sistemi informatici. Le sue intuizioni fungono da quadro fondamentale per comprendere l'intricata relazione tra la progettazione dell'interfaccia utente, i processi cognitivi e il panorama in evoluzione della collaborazione e comunicazione digitale.

### 2.2.1 Ruolo delle interfacce grafiche nelle applicazioni mobile

Le interfacce grafiche svolgono un ruolo cruciale nelle applicazioni mobili, fungendo da ponte visivo tra l'utente e le complesse funzionalità sottostanti. La progettazione di un'interfaccia efficace è essenziale per garantire un'esperienza utente positiva e per favorire l'interazione intuitiva con le applicazioni. Elementi come la disposizione dei componenti, la dimensione dei pulsanti e l'organizzazione generale dell'interfaccia giocano un ruolo chiave nel rendere l'applicazione accessibile a una vasta gamma di utenti.

Un panorama cruciale per comprendere l'importanza di un design ottimizzato e accessibile per le interfacce grafiche in ambito mobile viene discusso da M. Pieri. Secondo quest'ultimo [5], esperto nel campo dell'accessibilità del **Mobile Learning**<sup>5</sup>, emerge chiaramente che la progettazione di interfacce grafiche accessibili riveste un ruolo fondamentale nell'assicurare un'esperienza inclusiva per tutti gli utenti, compresi coloro con disabilità. Questo concetto è particolarmente rilevante nell'ambito dell'educazione mobile, dove l'accessibilità diventa un pilastro per garantire l'inclusione di tutti gli studenti, indipendentemente dalle loro capacità.

L'articolo di Pieri sottolinea che l'ottimizzazione delle interfacce grafiche non riguarda solo l'aspetto tecnico, ma implica anche una riflessione sui cambiamenti necessari negli approcci pedagogici e nelle strategie didattiche.

In questo contesto, le interfacce grafiche accessibili diventano uno strumento chiave

---

<sup>5</sup>M-learning o mobile learning, è l'insieme dei modelli e delle soluzioni di e-learning basati sull'utilizzo di uno o più dispositivi portatili come risposta alle istanze di soggetti che necessitano di accedere a risorse educative in modo continuativo e indipendentemente da una localizzazione predefinita.

per favorire l'inclusione e l'accesso equo all'istruzione, contribuendo a superare le barriere che potrebbero limitare la partecipazione degli studenti con disabilità.

Inoltre, è importante approfondire come le interfacce grafiche mobile possano essere ottimizzate non solo dal punto di vista tecnologico, ma anche considerando le esigenze specifiche degli utenti, inclusi coloro con disabilità visive, motorie o uditive. Integrare principi di design inclusivo e accessibile nelle interfacce grafiche delle applicazioni mobile può non solo migliorare l'esperienza degli utenti, ma anche promuovere un ambiente educativo più equo e inclusivo.

L'articolo "*From Skeuomorphism to Material Design and back. The language of colours in the 2nd generation of mobile interface design*" [6] discute l'evoluzione del design delle interfacce mobili dallo skeuomorfismo <sup>6</sup> al flat design <sup>7</sup>, concentrandosi in particolare sul ruolo dei colori nell'esperienza utente.

Inizialmente, con l'introduzione dell'iPhone nel 2007, le interfacce mobili seguivano un'approccio di design skeuomorfo, imitando oggetti del mondo reale e utilizzando i colori in modo naturale per creare un'esperienza utente familiare.

Tuttavia, nel 2013, il passaggio al flat design, esemplificato da iOS7 e dal material design in Android, ha segnato un cambiamento significativo nell'uso dei colori nelle interfacce grafiche e tattili su varie piattaforme.

Il flat design ha introdotto un uso audace e astratto dei colori, rendendoli asset potenti nelle interazioni degli utenti. I colori hanno assunto ruoli connotativi e astratti nella guida degli utenti attraverso le interfacce. Si sottolinea, inoltre, l'importanza strategica del design visivo negli studi sull'esperienza utente, evidenziando il valore estetico nelle interfacce utente grafiche e multimodali <sup>8</sup>.

---

<sup>6</sup>Lo skeuomorfismo è un approccio di design che cerca di riprodurre gli elementi visivi di oggetti del mondo reale, portandoli all'interno di un ambiente digitale

<sup>7</sup>Il flat design è un approccio minimalista caratterizzato dalla semplificazione degli elementi grafici, evitando ombre, rilievi e dettagli tridimensionali

<sup>8</sup>Si parla di multimodalità quando un qualsiasi tipo di interazione coinvolge più di un canale percettivo (o input di comunicazione). L'esempio più eclatante è senz'altro la comunicazione umana, durante una conversazione vengono stimolati più sensi contemporaneamente: la vista, l'udito, il tatto ed anche l'olfatto

Con l'avvento del flat design e del material design, i colori assumono un ruolo fondamentale nelle interfacce grafiche, diventando mezzi più potenti ed espressivi. In particolare, vengono utilizzati colori audaci, saturi, vividi e acidi in modo molto astratto e connotativo. I contrasti sono forti ed efficaci, diventando uno dei principali elementi espressivi nel design delle interfacce mobili.

Nel contesto del flat design e del material design, i colori vengono impiegati in modo deliberato e intenzionale per creare gerarchie, significato e focalizzazione. Scelte di colore deliberate, immagini a pieno schermo, tipografia di grande scala e spazi bianchi intenzionali vengono utilizzati per creare immersione e chiarezza nell'interfaccia. Questo approccio mira a comunicare in modo chiaro e coinvolgente con gli utenti, utilizzando i colori come strumento per guidare l'esperienza e la navigazione attraverso le varie sezioni e azioni dell'interfaccia.

Inoltre, il documento menziona che le linee guida di design, come il Material Design di Google, forniscono principi e esempi pratici per aiutare i professionisti a progettare app coerenti con l'approccio di sviluppo Android. Queste linee guida includono spiegazioni sulla scelta, sull'uso e sull'implementazione dei colori per garantire coerenza e coesione nel design delle interfacce.

### 2.2.2 Esempi di estrazione di un'interfaccia nella letteratura odierna

L'estrazione di interfacce grafiche riveste un ruolo di crescente importanza nella nostra vita quotidiana, soprattutto considerando l'ampia diffusione della tecnologia e delle interfacce utente in vari contesti. Le interfacce grafiche sono elementi visivi e interattivi che consentono agli utenti di comunicare e interagire con i dispositivi e i software. Comprendere i meccanismi e le metodologie utili ad estrarre interfacce grafiche è cruciale nella nostra quotidianità:

- **Facilità d'uso:** Le interfacce grafiche forniscono un modo intuitivo per gli utenti di interagire con i dispositivi e i programmi senza la necessità di conoscenze tecniche avanzate. Icone, pulsanti e elementi visivi rendono più facile comprendere e utilizzare le funzionalità senza dover imparare comandi complessi o linguaggi specifici.

- **Accessibilità:** Un'interfaccia grafica ben progettata contribuisce all'accessibilità, consentendo a un pubblico più ampio di usufruire dei servizi digitali. Elementi come il contrasto, la dimensione del testo e le opzioni di navigazione intuitiva sono cruciali per garantire che le interfacce siano accessibili a persone con diverse abilità e capacità.
- **Esperienza utente migliorata:** Un'estrazione accurata delle interfacce grafiche contribuisce a un'esperienza utente positiva. La progettazione attenta di layout, colori e elementi interattivi può influenzare significativamente il modo in cui gli utenti percepiscono e interagiscono con un'applicazione o un sito web.
- **Interazione multi-dispositivo:** Con la crescente diversità di dispositivi, dalla telefonia mobile ai tablet e ai computer desktop, le interfacce grafiche svolgono un ruolo cruciale nell'adattamento e nell'ottimizzazione dell'esperienza utente su diversi schermi e dimensioni.
- **Comunicazione efficace delle informazioni:** Le interfacce grafiche sono fondamentali per la presentazione visuale delle informazioni. Icone, grafici e layout ben progettati aiutano a comunicare in modo chiaro e rapido le informazioni essenziali, migliorando la comprensione e la fruibilità dei contenuti.
- **Sviluppo della tecnologia:** L'avanzamento della tecnologia è strettamente legato all'evoluzione delle interfacce grafiche. L'estrazione di interfacce è fondamentale per consentire nuove funzionalità e interazioni innovative che tengono conto dei cambiamenti tecnologici e delle esigenze degli utenti.

A tal proposito, un ottimo punto di partenza per estrarre le interfacce grafiche mobile in modo efficiente, viene fornito da Vincenzo De Martino. La sua tesi "*Consumo Energetico ed Accessibilità di Dispositivi Android: Un Approccio per l'Ottimizzazione di Interfacce Grafiche*" [7] si concentra sull'ottimizzazione delle interfacce grafiche per dispositivi Android al fine di ridurre il consumo energetico durante la visualizzazione delle interfacce. Il suo lavoro include la misurazione dei consumi energetici, la stima dei consumi energetici dei display, e l'implementazione di un algoritmo genetico per ottimizzare le interfacce grafiche. Inoltre, la tesi affronta obiettivi di ricerca come la definizione dell'adeguato contrasto delle interfacce, la distinguibilità delle componenti per persone con problemi visivi, e l'identificazione dell'approccio di ottimizzazione multi-obiettivo più adatto. La ricerca si basa su una re-implementazione di una tecnica search-based per l'ottimizzazione delle interfacce grafiche, con una fase di ottenimento dei dati, l'architettura di un algoritmo genetico, e la valutazione dei consumi energetici delle nuove interfacce prodotte. Viene incluso anche uno studio empirico sul design, gli obiettivi di ricerca e i risultati ottenuti, fornendo una panoramica approfondita sull'approccio e i risultati raggiunti. L'approccio utilizzato da Vincenzo De Marino per l'estrazione delle interfacce grafiche è un approccio manuale, basato sull'utilizzo di uno strumento chiamato UI Automator Viewer. Nella sua tesi, viene menzionato che UI Automator View è stato utilizzato per acquisire le interfacce visualizzate dalle applicazioni selezionate, restituendo screenshot e snapshot delle schermate.

Partendo dall'approccio di Vincenzo De Martino, in questo studio, si vuole realizzare un algoritmo che estragga le interfacce grafiche da una generica applicazione in modo automatico. Inoltre, si vuole assicurare:

- **Efficienza:** L'estrazione automatica consente di risparmiare tempo ed energia rispetto all'estrazione manuale delle interfacce. Gli algoritmi possono analizzare rapidamente grandi quantità di dati e generare risultati in modo efficiente.
- **Approccio innovativo:** L'utilizzo di un algoritmo automatico creato appositamente per l'estrazione delle interfacce grafiche rappresenta un approccio innovativo e avanzato nel campo del data mining e del design delle interfacce .



- **Profondità dell'analisi:** La metodologia di estrazione e analisi delle interfacce permette di ottenere una comprensione approfondita della popolarità di ciascuna applicazione, offrendo una panoramica completa sull'ambito e sull'evoluzione di ogni singola interfaccia grafica .
- **Miglioramento dell'esperienza utente:** L'analisi dettagliata delle componenti pixel e delle strutture delle interfacce grafiche contribuisce a migliorare la percezione e la distinguibilità dei colori, creando un ambiente visivo migliore per gli utenti Android.
- **Utilità pratica:** I risultati dello studio forniscono una visione chiara e completa della popolarità delle applicazioni Android e dell'andamento del loro design, offrendo un quadro informativo essenziale per sviluppatori, studiosi e utenti .

---

### Metodologia di estrazione e analisi delle interfacce

---

#### 3.1 Panoramica generale

In questo capitolo verrà descritto il lavoro svolto per la realizzazione del software che permetterà di estrarre interfacce grafiche da una generica applicazione. Il lavoro viene svolto in tre fasi. La prima fase si baserà sulla raccolta di applicazioni che intendiamo testare, che proverà la fattibilità e la veridicità di questo studio. La seconda fase riguarderà l'estrazione di  $k$  interfacce da una generica applicazione presa dal dataset di applicazioni precedentemente creato; da questo seguirà la terza e ultima fase che riguarderà l'estrazione delle componenti presenti su una generica interfaccia e i colori dei pixel delle immagini.

Sulla base del contesto di studio svolto sono stati proposti due obiettivi di ricerca. Il primo obiettivo di ricerca si focalizza sul comprendere quali sono le caratteristiche delle app analizzate. Questo ci permetterà di capire la soddisfazione degli utenti e la popolarità delle app che verranno analizzate.

Il secondo obiettivo di ricerca si focalizza sul comprendere quali sono i colori più usati nelle applicazioni che saranno testate. Questo consentirà una comprensione approfondita della distribuzione cromatica e delle relazioni tra i colori e gli elementi dell'interfaccia.

## 3.2 Prima fase: raccolta delle applicazioni da testare

La prima cosa da fare per lo studio dell'estrazione di interfacce grafiche è quella di individuare le applicazioni da analizzare. Il numero di applicazioni presenti nel dataset non influenzerà in alcun modo la qualità dello studio. Il dataset presenta circa 50 applicazioni, catalogate secondo: **versione, categoria, numero di download, numero di recensioni e numero di stelle**. Questi sono tutti parametri rilevanti dal punto di vista dell'utente. Le stelle, le recensioni e i download riflettono la soddisfazione degli utenti e la popolarità dell'app, mentre la categoria e la versione forniscono contesto sull'ambito e sull'evoluzione dell'app.

- **Indicatori di qualità e successo:** Il numero di stelle e il numero di recensioni sono indicatori chiave della qualità e del gradimento dell'app. Un alto numero di download può indicare il successo dell'app sul mercato. Questi parametri possono aiutare a misurare l'accoglienza e la soddisfazione degli utenti.
- **Evoluzione delle versioni:** Seguire le versioni delle app è importante per comprendere come gli sviluppatori migliorano o modificano l'app nel tempo. Questo è utile per testare lo stesso studio su versioni diverse e vedere come gli algoritmi si comportano.
- **Variazioni nelle categorie:** L'analisi delle categorie può rivelare tendenze e cambiamenti nei gusti degli utenti. Ad esempio, se un'app cambia categoria nel tempo, potrebbe indicare adattamenti strategici degli sviluppatori.
- **Utilità nella raccolta di dati:** Il set di dati acquisito fornisce una visione completa delle applicazioni, permettendo un'analisi più approfondita e significativa.

## 3.3 Seconda fase: estrazione delle interfacce

Una volta scelte le applicazioni da testare, si passa alla fase di estrazione delle interfacce. Nel contesto di questa ricerca è stato sviluppato un algoritmo di esplorazione delle interfacce utente per applicazioni Android.

### 3.3.1 Android Debug Bridge e UiAutomator

All'interno dello studio, è stato implementato un algoritmo di esplorazione delle interfacce utente di applicazioni Android, avvalendosi delle potenzialità di Android Debug Bridge (ADB) e UiAutomator. L'obiettivo primario è acquisire in modo sistematico dati dettagliati sulle interfacce utente, consentendo un'analisi approfondita delle caratteristiche e del comportamento delle applicazioni selezionate. Utilizzando Android Debug Bridge (ADB) è stato possibile effettuare le seguenti operazioni:

- **Collegamento al dispositivo:** Inizialmente, il programma stabilisce una connessione con il dispositivo Android tramite ADB. Questo permette una comunicazione bidirezionale tra il computer e il dispositivo, essenziale per l'esplorazione delle applicazioni.
- **Ricerca e avvio dell'applicazione:** ADB viene impiegato per individuare e avviare l'applicazione di interesse sul dispositivo. Ciò è fondamentale per iniziare l'esplorazione mirata delle interfacce utente.
- **Acquisizione di screenshot e dati XML<sup>1</sup>:** Attraverso comandi ADB, vengono catturati screenshot delle interfacce e ottenuti i dati XML corrispondenti. Questi dati saranno successivamente utilizzati per l'analisi dettagliata.

L'interazione con UiAutomator è stata cruciale in quanto:

- **Uso della libreria uiautomator:** L'algoritmo fa ampio uso della libreria UiAutomator in Python. Questa libreria semplifica notevolmente l'interazione con gli elementi dell'interfaccia utente e fornisce strumenti per l'esplorazione automatica.
- **Identificazione e clic degli elementi:** UiAutomator viene utilizzato per identificare e cliccare gli elementi dell'interfaccia utente. La funzionalità click() è fondamentale per simulare l'interazione utente e per raccogliere ulteriori dati.

---

<sup>1</sup>metalinguaggio per la definizione di linguaggi di markup, ovvero un linguaggio basato su un meccanismo sintattico che consente di definire e controllare il significato degli elementi contenuti in un documento o in un testo.

- **Gestione degli errori:** Attraverso UiAutomator, il programma è in grado di gestire eventuali errori durante l'esplorazione, come ad esempio clic su elementi non cliccabili o errori di caricamento.
- **Ordinamento degli elementi:** UiAutomator supporta la raccolta e l'ordinamento degli elementi in base alla loro posizione sull'interfaccia. Ciò è cruciale per una navigazione sistematica attraverso le schermate dell'applicazione.

L'obiettivo fondamentale di questa fase è quello di esplorare tutte le interfacce che compongono un'applicazione restituendo in output gli screenshot e gli snapshot<sup>2</sup> di tutte le interfacce. Gli screenshot e gli snapshot delle interfacce possono essere catturate manualmente o automatizzando il processo. L'algoritmo creato mira ad automatizzare il processo di estrazione utilizzando la libreria UiAutomator per interagire con l'UI dell'applicazione e raccogliere dati dettagliati per l'analisi. Inizialmente, il programma identifica e avvia l'applicazione da noi scelta sul dispositivo Android attraverso una connessione ADB. Successivamente, utilizzando una coda di esplorazione, l'algoritmo percorre in modo sistematico le diverse interfacce dell'applicazione.

Ad ogni passo dell'esplorazione, viene catturato uno screenshot dell'interfaccia corrente, e i dati relativi alla struttura dell'UI vengono salvati in file XML. Gli elementi cliccabili sull'interfaccia vengono ordinati in base alla loro posizione, e l'algoritmo procede a cliccarli uno per uno. Durante questo processo, vengono gestiti eventuali errori, come ad esempio clic su elementi non cliccabili. Per evitare clic ripetuti sugli stessi elementi, l'algoritmo tiene traccia degli elementi già visitati attraverso insiemi dedicati. Viene, inoltre, definito un limite  $k$  di interfacce che saranno acquisite. Questo limite può essere modificato in base alle proprie esigenze. Per rendere lo studio coerente il valore di  $k$  è lo stesso per tutte le applicazioni che sono state testate. L'algoritmo di esplorazione delle interfacce utente implementato può contribuire positivamente all'accessibilità per le persone affette da daltonismo attraverso alcune implementazioni

---

<sup>2</sup>Uno snapshot, letteralmente una "istantanea", è generalmente la cattura di stato di un oggetto in un determinato momento nel tempo.

mirate, per esempio introducendo strategie di design e interazione specifiche rendendo l'applicazione più inclusiva e migliorare l'esperienza degli utenti con disabilità visive. UiAutomator ha un grandissimo potenziale in quanto Patil [8] ha realizzato uno strumento simile ad UiAutomator, ma più potente in quanto migliora notevolmente l'accessibilità per tutti gli utenti: l'Enhanced UI Automator Viewer.

Questo strumento introduce nuove funzionalità cruciali, come la possibilità di selezionare il dispositivo di destinazione, catturare e riprodurre eventi touch e tastiera, e controllare il contrasto dei colori per ottimizzare l'esperienza degli utenti con disabilità visive e daltonismo. L'obiettivo principale di questa innovazione è individuare e correggere le problematiche di accessibilità delle interfacce utente delle app Android durante la fase di sviluppo, garantendo un'esperienza inclusiva e accessibile per tutti gli utenti.

### 3.4 Algoritmo BOCP-BOCC

Dopo aver acquisito un insieme di  $k$  interfacce, insieme ai rispettivi snapshot e screenshot, procederemo con l'implementazione dell'algoritmo BOCP\_BOCC. Questo algoritmo mira a estrarre due strutture dati fondamentali da un numero  $k$  di interfacce, ovvero due HashMap: BOCP (Bag of Color Pixels) e BOCC (Bag of Color Components). Il BOCP è concepito come una struttura hash-map che tiene traccia di tutti i pixel presenti nell'interfaccia grafica. Ogni chiave di questa mappa rappresenta un colore quantizzato (colore), e il valore corrispondente è un elenco di pixel assegnati a quel particolare colore, insieme all'indicazione dell'interfaccia associata. Il BOCC, anch'esso un'hash-map, ha come chiavi i colori quantizzati (colore), mentre i valori associati contengono un elenco completo degli elementi delle componenti che sono correlati ai pixel presenti in BOCP[colore]. Questa struttura consente di stabilire una connessione diretta tra i colori dei pixel, le componenti e le interfacce. L'algoritmo BOCP\_BOCC è progettato per organizzare in maniera efficiente le informazioni sulle interfacce grafiche, offrendo una rappresentazione chiara e strutturata dei colori, dei pixel e delle componenti correlate nelle interfacce utente considerate. Esso è strutturato nel seguente modo: la funzione principale,

bocp\_bocc(screenshotPath, snapshotPath), inizia inizializzando tre strutture dati essenziali: bocp, bocc, e pixel\_interface. Questi sono rispettivamente un dizionario che mappa i colori quantizzati ai pixel e alle relative interfacce, un altro dizionario che mappa i colori quantizzati agli elementi delle componenti e alle interfacce, e un dizionario per tracciare i pixel elaborati nell'immagine.

Successivamente, l'immagine dello screenshot viene caricata nella variabile screenshot, e i valori RGB dei pixel nell'immagine vengono quantizzati utilizzando la funzione quantizeImage da un modulo chiamato ImageUtilities. Un altro modulo, ImageXML, rappresenta una delle librerie utilizzate per caricare l'XML dello snapshot e ottenere la struttura dell'interfaccia.

La funzione scorre i componenti in ordine discendente (dal basso verso l'alto). Per ciascun componente, i pixel di interesse vengono estratti utilizzando la funzione getPixels, la quale si avvale di funzioni ausiliarie come checkPixel e getColor. I risultati vengono quindi aggiunti ai dizionari bocp e bocc per tenere traccia delle informazioni organizzate per colore, pixel e componenti.

Infine, la funzione restituisce i dati raccolti dalle strutture dati bocp e bocc. Le altre funzioni nel set di dati includono:

- **getComponents(root):** Estrae tutti gli elementi discendenti dall'albero XML.
- **getPixels(screenshot, c\_item, pixel\_interface, type):** Restituisce i pixel rilevanti da un componente.
- **checkPixel(pixel\_x, pixel\_y, pixel\_interface):** Controlla se un pixel è già stato considerato.
- **getColor(S, x, y):** Restituisce il colore di un pixel.
- **ImageUtilities.quantizeImage(img):** Quantizza un'immagine per semplificare i colori.

Terminato l'algoritmo le due strutture HashMap verranno salvate in file .pickle per poterle utilizzare durante lo sviluppo di eventuali algoritmi di intelligenza artificiale.

L'algoritmo Bag of Color Pixels e Bag of Color Components (BOCP\_BOCC) è stato ampiamente utilizzato in diversi lavori di ricerca per obiettivi diversi.

L'articolo "*Multi-Objective Optimization of Energy Consumption of GUIs in Android Apps*" [9] si focalizza sull'ottimizzazione del consumo energetico delle interfacce utente grafiche (GUIs) nelle app Android attraverso l'utilizzo di una tecnica, chiamata GEMMA, di ottimizzazione multi-obiettivo per la scelta dei colori.

In GEMMA è stato implementato l'algoritmo BOCP-BOCC, che consente di identificare e rappresentare in modo dettagliato le palette di colori delle GUI delle app Android, fornendo una base solida per l'ottimizzazione multi-obiettivo del consumo energetico e del design visivo delle interfacce utente.



## CAPITOLO 4

---

### Risultati dello studio

---

In questo capitolo verranno visualizzati i risultati dello studio, rispondendo agli obiettivi di ricerca prefissati. Verranno visualizzati i grafici che mostrano l'analisi della distribuzione dei dati all'interno del dataset di applicazioni e l'analisi dei colori delle interfacce, in modo da avere una panoramica dettagliata sulle applicazioni scelte come test all'inizio dello studio.

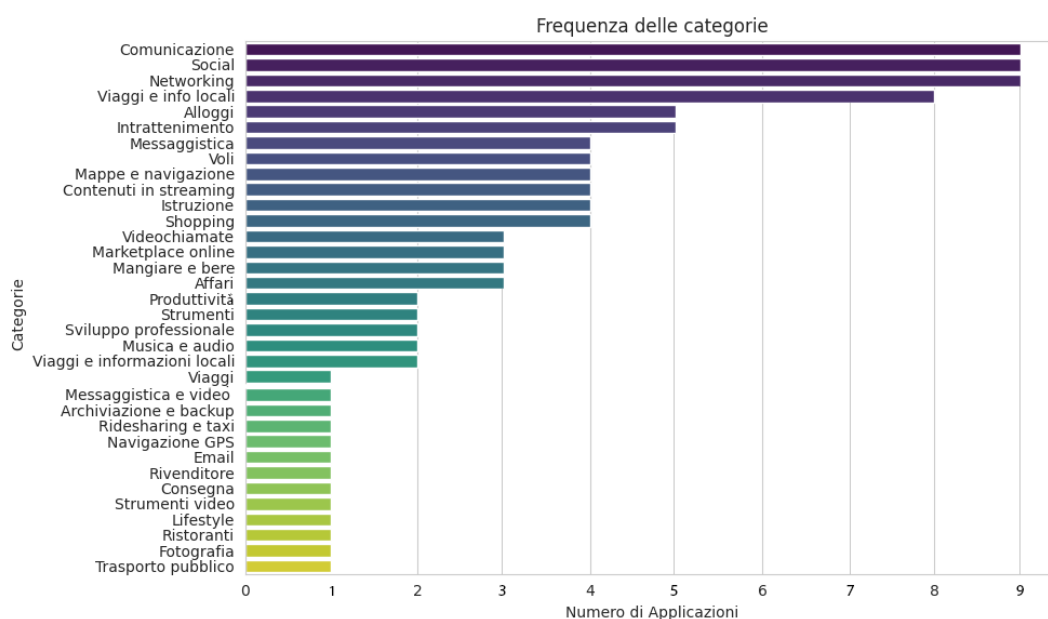
#### 4.1 OP1: Quali sono le caratteristiche delle app analizzate

Per rispondere alla prima domanda di ricerca, le applicazioni utilizzate per testare questo studio sono state circa 50, catalogate secondo: versione, categoria, numero di download, numero di recensioni e numero di stelle. si possono ricavare parametri rilevanti come la soddisfazione degli utenti, la popolarità di un'app e avere una panoramica generale sull'ambito e sull'evoluzione di un'app.

L'analisi di questo studio ha portato ha portato diversi risultati interessanti. Sulla base delle applicazioni utilizzate per eseguire il test di questo studio, sono stati prodotti dei grafici in base:

- categoria
- numero di download
- numero di recensioni
- numero di stelle

Iniziamo ad analizzarli, uno per volta:

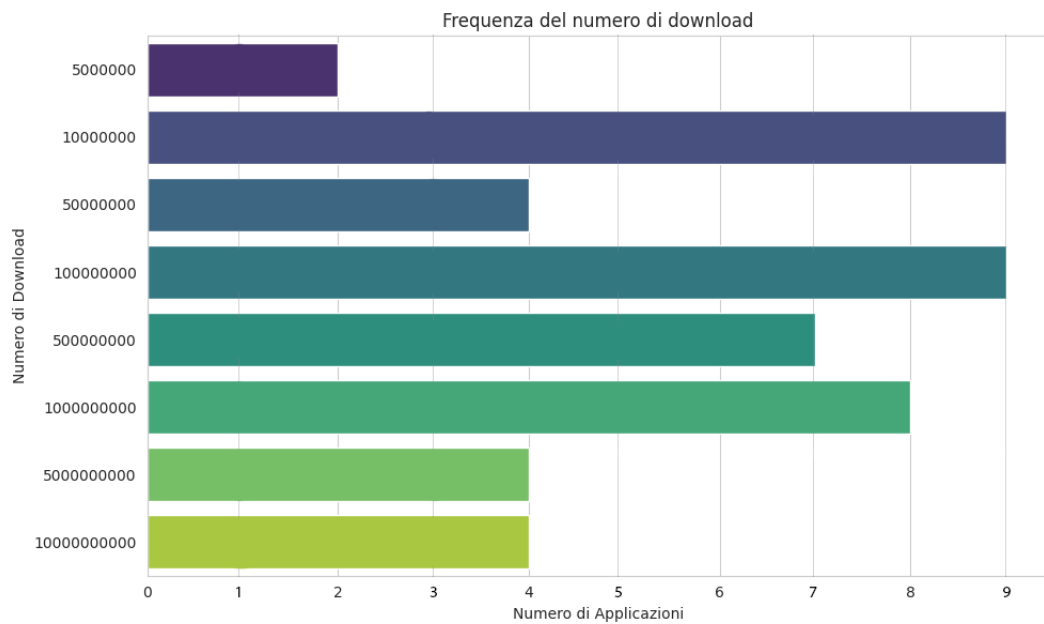


**Figura 4.1:** Frequenza delle categorie di ciascuna app

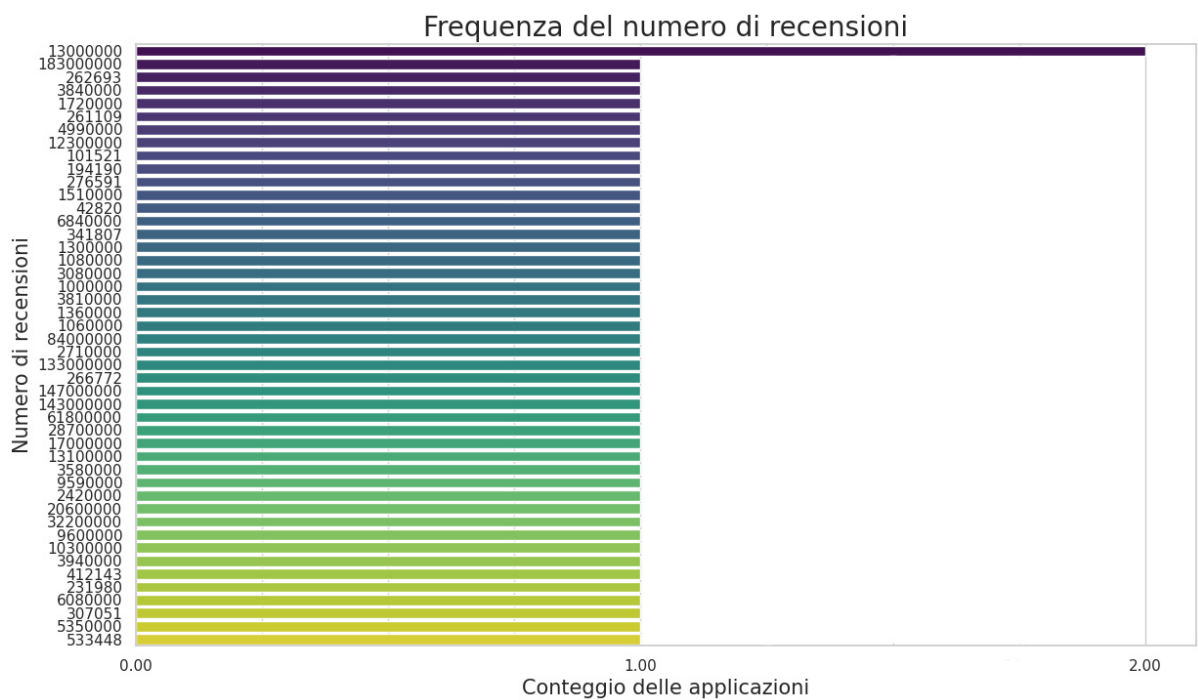
Il grafico in figura 5.1 mostra la distribuzione delle app in base alla loro categoria di appartenenza. L'asse delle x rappresenta il numero di applicazioni, mentre l'asse delle y rappresenta il nome di ogni categoria. Da questo grafico si evince la frequenza delle categorie per ciascuna delle app presente nel dataset.

La figura 5.2 mostra la distribuzione della colonna numero di download nel mio dataset. L'asse delle x rappresenta il numero di applicazioni, mentre l'asse delle y rappresenta il numero di download. Da questo grafico si evince la frequenza del numero di download per ciascuna delle app presenti nel dataset.

#### 4.1 – OP1: Quali sono le caratteristiche delle app analizzate

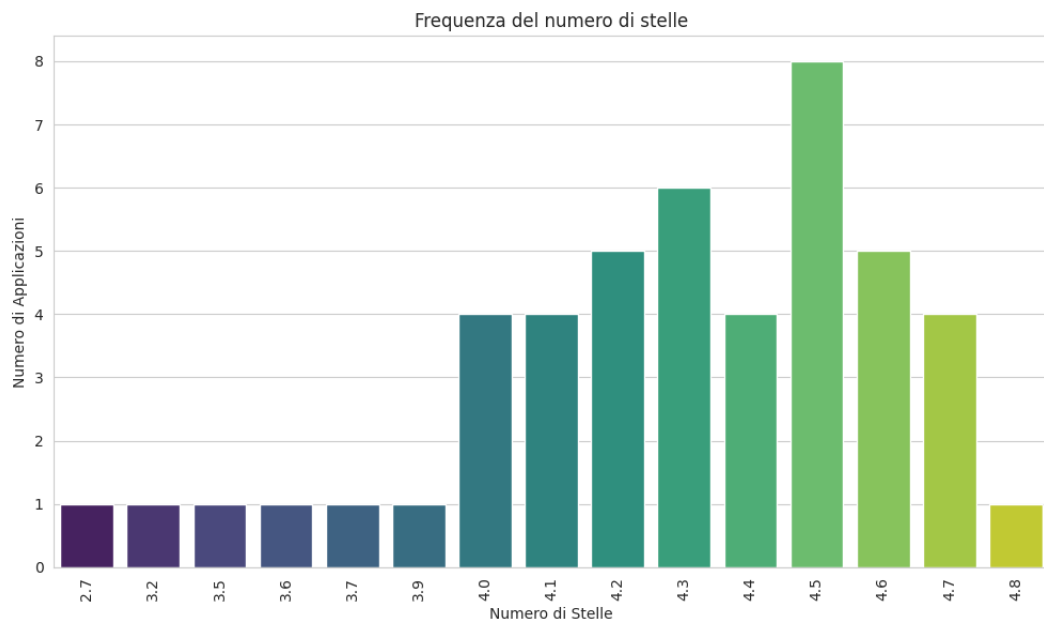


**Figura 4.2:** Frequenza del numero di download di ciascuna app



**Figura 4.3:** Frequenza del numero di recensioni di ciascuna app

La figura 5.3 mostra la distribuzione del numero di recensioni per le app all'interno del dataset. L'asse delle x rappresenta il numero di applicazioni, mentre l'asse delle y rappresenta il numero di recensioni. Dal seguente grafico possiamo evincere che solo due applicazioni hanno lo stesso numero di recensioni.



**Figura 4.4:** Frequenza del numero di stelle di ciascuna app

Infine, la figura 5.4 mostra la distribuzione delle valutazioni a stelle per le app nel set di dati. In particolare, l’asse delle x rappresenta il numero di stelle, mentre l’asse delle y rappresenta il numero di applicazioni. Dal grafico si evince che il numero di stelle pari a 4.5 è il più rilevante all’interno del dataset in quanto più applicazioni presenti nel dataset hanno quel numero di stelle.

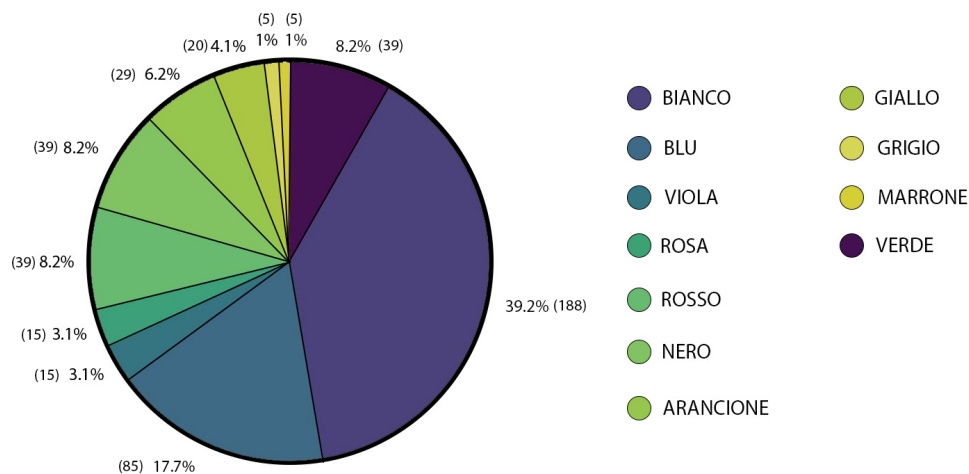
I risultati dello studio evidenziano l’efficacia dell’algoritmo utilizzato per estrarre e analizzare le interfacce utente delle applicazioni Android. L’analisi della distribuzione dei dati ha rivelato informazioni preziose su varie caratteristiche delle applicazioni, come la categoria di appartenenza, il numero di download, il numero di recensioni e le valutazioni a stelle. Queste informazioni possono essere utilizzate per comprendere meglio il panorama attuale delle applicazioni Android e per guidare lo sviluppo futuro di applicazioni più efficaci e coinvolgenti. Inoltre, i risultati dello studio possono servire come base per ulteriori ricerche in questo campo.

## 4.2 OP2: Quali sono i colori più usati nelle applicazioni considerate

Per rispondere alla seconda domanda di ricerca, nel contesto dell'algoritmo BOCP\_BOCC, i risultati ottenuti rivestono un'importanza significativa per l'estrazione e l'analisi dei colori impiegati nelle interfacce catturate. L'algoritmo si propone di estrarre due strutture dati fondamentali: il BOCP (Bag of Color Pixels) e il BOCC (Bag of Color Components).

Il BOCP si configura come una struttura hash-map che registra tutti i pixel presenti nell'interfaccia grafica, associando ogni colore quantizzato a un elenco di pixel assegnati a quel particolare colore. Ciò consente un'identificazione efficiente dei colori utilizzati e della loro distribuzione all'interno dell'interfaccia.

D'altra parte, il BOCC, anch'esso un'hash-map, collega i colori quantizzati a un elenco completo degli elementi delle componenti correlati ai pixel di quel colore. Tale struttura facilita l'instaurazione di una connessione diretta tra i colori dei pixel, le componenti dell'interfaccia e le interfacce stesse, fornendo una rappresentazione chiara e organizzata dei colori e dei loro utilizzi nelle interfacce grafiche.



**Figura 4.5:** Analisi dei colori delle interfacce

I risultati derivanti dall'attuazione dell'algoritmo BOCP\_BOCC hanno dimostrato di essere fondamentali per condurre un'analisi dettagliata dei colori presenti nelle interfacce acquisite. Dalla figura 5.5 possiamo notare che è stata fatta un'analisi

dei colori delle interfacce che ha rilevato una percentuale del colore bianco pari al 39,2%. Questo significa che il colore bianco, rispetto agli altri colori, assume un ruolo preponderante all'interno delle applicazioni testate.

---

### Conclusione e sviluppi futuri

---

Le conclusioni di questa tesi suggeriscono una prospettiva innovativa e promettente nell'estrazione delle interfacce grafiche delle applicazioni Android. L'utilizzo di un algoritmo dedicato come BOCP/BOCC per analizzare dettagliatamente i pixel e le strutture delle interfacce grafiche ha portato a diversi risultati. La comprensione della popolarità di un'app attraverso l'analisi delle sue interfacce, la panoramica sull'ambito e sull'evoluzione di ciascuna app, insieme all'analisi dei colori più rilevanti, offre un approccio completo per valutare e comprendere il mondo delle applicazioni Android.

La prospettiva di applicare gli algoritmi alle interfacce estratte per compiti medici, di classificazione e generative AI è un passo avanti notevole. Questo suggerisce l'applicabilità e la versatilità delle informazioni estratte per svariate finalità, ampliando così il campo di utilizzo dei risultati ottenuti.

Guardando al futuro, l'obiettivo di estendere la ricerca al maggior numero possibile di applicazioni è una mossa ambiziosa e significativa. Ciò potrebbe portare a una comprensione ancora più approfondita del panorama delle interfacce delle applicazioni Android, consentendo ulteriori sviluppi e applicazioni pratiche.

Questa tesi non solo ha contribuito alla comprensione delle interfacce delle applicazioni Android, ma ha anche aperto la strada a possibili sviluppi futuri e applicazioni in settori diversi, dimostrando il potenziale di trasferimento e utilizzo delle conoscenze acquisite in contesti più ampi.



---

## Bibliografia

---

- [1] S. Sagioglu and D. Sinanc, "Big data: A review," in *2013 International Conference on Collaboration Technologies and Systems (CTS)*, 2013, pp. 42–47. (Citato a pagina 5)
- [2] A. A. Maryoosh and E. M. Hussein, "A review: Data mining techniques and its applications," *International Journal of Computer Science and Mobile Applications*, vol. 10, pp. 1–14, 2022. (Citato a pagina 7)
- [3] N. Stephenson, *In the Beginning...was the Command Line*. William Morrow Paperbacks, 1999. (Citato a pagina 11)
- [4] A. Kay, "User interface: A personal view," *The art of human-computer interface design*, pp. 191–207, 1990. (Citato a pagina 12)
- [5] M. Pieri, "L'accessibilità del mobile learning," *TD-Tecnologie Didattiche*, vol. 52, pp. 49–56, 2011. (Citato a pagina 13)
- [6] L. Bollini *et al.*, "From skeuomorphism to material design and back. the language of colours in the 2nd generation of mobile interface design," *Colour and colorimetry. Multidisciplinary contributions*, vol. 12, pp. 309–320, 2016. (Citato a pagina 14)
- [7] D. M. Vincenzo, "Consumo energetico ed accessibilità di dispositivi android: Un approccio per l'ottimizzazione di interfacce grafiche," 2021-2022. (Citato a pagina 17)

- [8] N. Patil, D. Bhole, and P. Shete, “Enhanced ui automator viewer with improved android accessibility evaluation features,” in *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, 2016, pp. 977–983. (Citato a pagina 23)
- [9] M. Linares-Vásquez, G. Bavota, C. Bernal-Cárdenas, M. D. Penta, R. Oliveto, and D. Poshyvanyk, “Multi-objective optimization of energy consumption of guis in android apps,” *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 27, no. 3, pp. 1–47, 2018. (Citato a pagina 25)