



UNIVERSITÀ DEGLI STUDI DI BARI ALDO MORO

DIPARTIMENTO DI INFORMATICA

CORSO DI LAUREA TRIENNALE IN INFORMATICA

TESI DI LAUREA IN

METODI PER IL RITROVAMENTO DELL'INFORMAZIONE

Progettazione e Sviluppo di un Sistema Basato su Vision-Language-Model per la Creazione Personalizzata di Contenuti Pubblicitari nel Settore Moda

Relatori:

Prof. Pasquale Lops

Prof. Cataldo Musto

Dott. Alessandro F. M. Martina

Laureando:

Alessandro Piergiovanni

ANNO ACCADEMICO 2023-2024

INDICE

1. INTRODUZIONE	8
1.1 Obiettivi e Motivazioni	10
1.2 Struttura della Tesi	12
2. ANALISI DELLO STATO DELL'ARTE	13
2.1 Large Language Models (LLM)	13
2.1.1 Architettura Transformer	13
2.1.2 Prompt Engineering	16
2.1.3 Addestramento e Fine-Tuning	18
2.2 Vision Language Models (VLM)	19
2.2.1 Struttura Generale dei VLM	19
2.2.2 Addestramento dei VLM	20
2.3 Latent Diffusion Modes (LDM)	20
2.3.1 Struttura dei LDM	21
2.3.2 Addestramento dei LDM	22
2.3.3 Differenze tra LDM e Altri Modelli Generativi	22
2.4 Esempi di Campagne di Successo Basate sull'IA	23
2.4.1 Coca-Cola e l'IA Creativa	24
2.4.2 L'Oréal e i Modelli Virtuali	26
2.4.3 Nike, Realtà Aumentata e IA Generativa	28
3. METODOLOGIA	36
3.1 Architettura del Sistema di Generazione	37
3.2 Modellazione Utenti	39
3.3 Scelta delle Immagini di Baseline	40
3.3.1 Esclusione delle Fonti Commerciali	40
3.3.2 Criteri per la Selezione delle Immagini Baseline	40
3.4 Selezione dei Modelli Generativi	42
3.5 Strategie di Prompting per i Modelli Impiegati	43
3.5.1 Prompting del Vision Language Model	43
3.5.2 Prompting del Latent Diffusion Model	44
3.6 Tecniche di Gestione dei Dati (CSV e Database)	45
3.6.1 Progettazione del Database (Modello E-R)	46
3.7 Architettura della Web App	51
3.7.1 Web App per Utente Standard	51

3.7.2 Web App per Super User	52
3.8 Metriche di Valutazione	55
4. IMPLEMENTAZIONE	60
4.1 Diagramma delle Componenti	60
4.2 Tecnologie Utilizzate	62
4.3 Selezione delle Immagini di Baseline	65
4.4 Scelta dei Modelli	70
4.4.1 Scelta del Modello image-text-to-text	71
4.4.2 Scelta del Modello text-to-image	76
4.5 Generazione delle Immagini e Prompt Engineering	81
4.5.1 Fase Iniziale: Prompt lunghi e strutturati per Hugging Face	81
4.5.2 Transizione a Stability API: degrado prestazionale a parità di prompt	82
4.5.3 Ottimizzazione e test iterativi: verso una pipeline coerente	83
4.5.4 Definizione del Prompt Finale: bilanciamento tra regole e creatività	88
4.6 Progettazione Logica del Database e Scelte Progettuali	97
4.6.1 Discussione delle Scelte Progettuali, delle Chiavi e degli Attributi	98
4.7 Gestione e Organizzazione dei Dati	102
4.7.1 Struttura e Caricamento dei File CSV	102
4.7.2 Importazione nel Database SQLite	105
4.7.3 Dataset Creati Ad Hoc	107
4.7.4 Tracciamento e Aggiornamento dei Dati	108
4.8 Realizzazione della Web App con Gradio	110
4.8.1 Interfaccia Utente: Registrazione e Raccolta Profilo	111
4.8.2 Interfaccia Utente: Generazione delle Immagini e Valutazione	115
4.8.3 Interfaccia Super-User: Analisi e Valutazione dei Dati	120
4.9 Approfondimenti Tecnici sulle Logiche del Sistema	126
4.9.1 Gestione della Randomizzazione delle Immagini	126
4.9.2 Caricamento Dinamico delle Domande	129
4.9.3 Controlli Sequenziali sulle Tab (Blocco Avanzamento)	132
4.9.4 Query Complesse nella Super-User Interface	134

4.9.5 Caricamento Dinamico delle Domande nella Super-User Interface Mediante SetID	135
4.9.6 Assegnazione delle Immagini Baseline in base al Genere	136
4.10 Implementazione delle Metriche di Valutazione	137
4.10.1 Correlazioni Lineari Semplici	137
4.10.2 Regressione Lineare (OLS)	139
4.10.3 Modello Bradley-Terry (Standard)	140
4.10.4 Modello Bradley-Terry Pesato	142
4.11 Considerazioni Finali sullo Sviluppo	143
5. Sperimentazione e Risultati	144
5.1 Protocollo Sperimentale	144
5.2 Risultati	145
5.2.1 Analisi Preliminari del Campione Utente	146
5.2.2 Analisi del Questionario	149
5.2.3 Analisi delle performance delle immagini di baseline	156
5.2.4 Correlazioni Semplici	158
5.2.5 Regressione Lineare	161
5.2.6 Bradley Terry Standard e Bradley Terry Pesato	164
6. Conclusioni	167
6.1. Principio di Utilizzo ed Etica	168
6.2 Limitazioni	170
6.3 Sviluppi Futuri	172
BIBLIOGRAFIA	174
SITOGRAFIA	179
APPENDICI	185
Appendice A – Input Prompt e Output dei Modelli image-text-to-text	185
A.1 Prompt di Input usato per i modelli	185
A.2 Risposta Lunga 1 di LLaMa	185
A.3 Risposta Lunga 2 di LLaMa	186
A.4 Risposte di Gemini 2.0 Flash Thinking	187
A.5 Riposta di Gemini Flash Experimental	188
Appendice B – Prompt di Input per i modelli text-to-image	189
B.1 Prompt 1 Generato da Gemini Flash Experimental	189
B.2 Prompt 2 Generato da Gemini Flash Experimental	190

B.3 Prompt 3 Generato da Gemini Flash Experimental	190
B.4 Prompt 4 Generato da Gemini Flash Experimental	192
Appendice C – Codici Utilizzati per la Sezione Super-User	193
C.1 Query Usata nella Sezione 2 della 3° Schermata - “Analisi Performances Immagini Baseline”	193
C.2 Query Usata nella Sezione 4 “Efficacia Immagini Baseline”	194
Appendice D – Prompt Finale	195
D.1 Prompt Finale Utilizzato per Gemini	195
D.2 Prompt 1 Generato da Gemini con Prompt Definitivo	196
D.3 Prompt 2 Generato da Gemini con Prompt Definitivo	196
D.4 Prompt 3 Generato da Gemini con Prompt Definitivo	197

ABSTRACT

La presente tesi descrive la progettazione e lo sviluppo di un sistema interattivo per la generazione personalizzata di immagini pubblicitarie in ambito moda, basato su tecnologie di intelligenza artificiale multimodale. Il sistema combina modelli *image-to-text* e *text-to-image* per trasformare un capo reale, in una nuova immagine generata coerente con il profilo utente.

Il profilo viene raccolto attraverso un'interfaccia web sviluppata con Gradio, mediante cui l'utente inserisce dati personali e preferenze stilistiche. Un vision-language-model (VLM) analizza le informazioni e l'immagine di partenza del capo, generando un prompt descrittivo che viene utilizzato come input per il modello di diffusione (LDM) incaricato di produrre l'immagine. L'immagine generata viene quindi confrontata con quella di partenza tramite un questionario interattivo sottoposto all'utente, volto a misurare la percezione dell'utente in termini di efficacia comunicativa, realismo e potenziale impatto commerciale.

Alla base di questo lavoro si colloca una domanda di ricerca centrale: *in che misura la personalizzazione visiva di un contenuto pubblicitario, basata sul profilo dell'utente, può costituire un fattore determinante nell'influarne le preferenze e incentivare la propensione all'acquisto?*

L'interfaccia include anche una sezione avanzata, destinata a un utente *super-user*, per l'analisi statistica dei dati raccolti. In tale ambiente sono implementate quattro metriche di valutazione: correlazioni lineari semplici, regressione lineare multipla, modello di Bradley-Terry e una sua variante pesata, che tiene conto della competenza e dell'interesse dichiarato degli utenti.

L'intero sistema è supportato da un database SQLite, che viene inizializzato automaticamente all'avvio e popolato in modo incrementale ogni volta che vengono aggiunti nuovi dati. Le immagini baseline e le domande del questionario, create ad hoc nella fase pre-sperimentale, assicurano coerenza e controllo sperimentale nella valutazione.

Il risultato è un prototipo *completo, modulare e interattivo*, in grado di integrare *generazione creativa, personalizzazione visiva, interazione utente e valutazione* quantitativa delle preferenze soggettive. Le analisi statistiche implementate permettono di correlare le scelte degli utenti con i loro profili individuali, fornendo insight potenzialmente rilevanti per la pubblicità generativa e la moda personalizzata basata su intelligenza artificiale.

1. INTRODUZIONE

Negli ultimi decenni, l'intelligenza artificiale (IA) ha avuto un ruolo dirompente nel plasmare le nostre vite, ridefinendo il nostro modo di lavorare, comunicare e creare. Quella che una volta era considerata una tecnologia futuristica confinata al regno della fantascienza è ora parte integrante delle nostre attività quotidiane, influenzando settori diversificati come la sanità, i trasporti, l'intrattenimento e la pubblicità [*Bibliografia 12*]. Assistenti virtuali come Siri e Alexa, algoritmi di raccomandazione utilizzati da piattaforme di streaming come Netflix e Spotify, e persino le applicazioni nei veicoli autonomi sono solo alcune delle manifestazioni dell'IA nella nostra quotidianità. Questo progresso tecnologico non ha solo migliorato l'efficienza operativa delle aziende e la qualità della vita delle persone, ma ha anche aperto nuovi orizzonti per la creatività e l'innovazione.

L'evoluzione dell'IA è stata accelerata dallo sviluppo di tecnologie di apprendimento automatico, in particolare il deep learning, che hanno permesso alle macchine di elaborare grandi quantità di dati e di apprendere schemi complessi. Questo ha portato alla nascita di sistemi generativi, capaci di creare contenuti originali sotto forma di testi, immagini, video e audio. In particolare, l'IA generativa rappresenta un'area emergente e rivoluzionaria, caratterizzata da modelli che non si limitano a rispondere a input predefiniti, ma che sono in grado di produrre output creativi e unici. Strumenti come GPT per il testo, DALL-E per le immagini e altre piattaforme basate su reti generative avversarie (GAN) o modelli Transformer hanno ridefinito ciò che è possibile nel campo della produzione creativa.

Un settore che ha abbracciato con entusiasmo queste innovazioni è quello del marketing e della pubblicità. Sin dagli anni '90, l'introduzione di algoritmi predittivi e tecniche di data mining aveva già trasformato il modo in cui le aziende raggiungevano i consumatori, offrendo campagne mirate basate sull'analisi dei dati comportamentali [*Sitografia 28, 29*]. Tuttavia, con l'avvento dell'IA generativa, l'interazione tra brand e pubblico ha raggiunto un livello di personalizzazione e creatività senza precedenti. Le aziende ora possono creare contenuti pubblicitari che non solo rispondono alle esigenze

specifiche dei consumatori, ma che catturano anche la loro attenzione attraverso immagini ed esperienze uniche e coinvolgenti.

L'IA nella pubblicità: un contesto storico

La pubblicità, in quanto disciplina, è da sempre strettamente legata alla capacità di catturare l'attenzione del pubblico attraverso messaggi visivi e narrativi accattivanti. Già negli anni '60, l'avvento dei computer aveva iniziato a influenzare il settore, ma è solo con l'introduzione delle tecnologie digitali negli anni '90 che si è verificato un vero cambiamento paradigmatico. Con l'avvento di Internet e dei primi sistemi di pubblicità online, come Google Ads, il focus si è spostato verso la raccolta e l'analisi dei dati, ponendo le basi per l'intelligenza artificiale moderna.

Negli anni 2010, la comparsa delle reti neurali e, in particolare, delle GAN, ha segnato una svolta significativa. Le GAN, introdotte per la prima volta da Ian Goodfellow nel 2014 [*Bibliografia 13*], hanno aperto la strada alla creazione di immagini sintetiche che sembrano straordinariamente realistiche. Ciò ha consentito ai professionisti della pubblicità di sperimentare con contenuti visivi generati automaticamente, come modelli virtuali per campagne di moda o scenari personalizzati per target di mercato specifici. Più recentemente, modelli come DALL-E, MidJourney e Stable Diffusion hanno ulteriormente raffinato questa capacità, combinando input testuali e visivi per creare immagini pubblicitarie personalizzate e di alta qualità in pochi secondi.

L'IA generativa e la personalizzazione nella moda

Il settore della moda, caratterizzato da un'estetica visiva forte e da una comunicazione emozionale, è particolarmente adatto all'utilizzo di queste tecnologie. La moda non si limita a vendere prodotti; racconta storie, trasmette valori e crea connessioni profonde con il pubblico. In un mercato sempre più competitivo, la capacità di distinguersi e di offrire esperienze uniche ai consumatori è diventata cruciale. L'IA, e in particolare l'IA generativa, offre uno strumento potente per raggiungere questi obiettivi.

Un esempio rilevante è la pipeline generativa adottata in questo progetto, che combina due modelli multimodali: uno **image-text-to-text** e uno **text-to-image**. Il primo modello analizza un'immagine di riferimento e un profilo utente, generando un prompt testuale descrittivo e personalizzato. Il prompt ottenuto viene poi utilizzato dal secondo modello, il generatore visivo, per creare un contenuto pubblicitario sotto forma di immagine coerente con lo stile dell'utente e il capo iniziale. Tale approccio rappresenta una nuova frontiera nella creazione di contenuti pubblicitari personalizzati e visivamente coinvolgenti.

Questo approccio non solo riduce il tempo e i costi di produzione delle campagne, ma apre nuove possibilità creative, permettendo di realizzare immagini che riflettono con precisione l'identità del brand e le preferenze del pubblico di riferimento.

Il lavoro presentato in questa tesi si propone di esplorare queste potenzialità, applicandole specificamente al settore della moda. Attraverso l'utilizzo di modelli avanzati di IA, verrà analizzato come sia possibile generare contenuti pubblicitari personalizzati, la coerenza del messaggio, l'interazione con il consumatore e il confronto tra diversi modelli. Inoltre, verranno approfondite le implicazioni etiche e culturali di questa tecnologia, considerando sia i vantaggi che le sfide associate all'adozione di strumenti generativi.

1.1 Obiettivi e Motivazioni

La personalizzazione nell'era digitale rappresenta una leva strategica per catturare l'attenzione di consumatori sempre più esigenti e immersi in un flusso costante di stimoli visivi. Tuttavia, nonostante i progressi dell'IA generativa, rimane una lacuna significativa nella comprensione di come immagini pubblicitarie iper-personalizzate influenzino concretamente le decisioni d'acquisto, specialmente in settori ad alta intensità estetica come la moda. Questo lavoro nasce dall'esigenza di colmare tale divario, proponendo una metodologia innovativa che unisce creatività umana e lavoro dell'IA, con risvolti sia teorici che pratici.

La tesi esplora come la coerenza tra identità del brand, preferenze dell'utente e output generativo influenzano l'efficacia del messaggio della pubblicità incentivando il cliente all'acquisto. Inoltre, il lavoro si rivolge a professionisti del marketing e designer, proponendo:

1. **Riduzione dei Costi Operativi:** Automatizzare la generazione di varianti pubblicitarie per micro-target (es. stagionalità, eventi culturali, trend social), eliminando la necessità di sessioni fotografiche multiple o riadattamenti manuali.
2. **Supporto Accelerato al Processo Creativo:** Consentire ai team creativi di testare decine di varianti di un'immagine pubblicitaria (es. cambiando scenari, modelli virtuali, illuminazione) in minuti anziché settimane, velocizzando iterazioni prima della fase di produzione e fornendo input creativi per il team di designer come fatto da Nike per il progetto A.I.R. [Sitografia 15].
3. **Etica della Personalizzazione:** Analisi critica su rischi come l'omologazione estetica, l'eccessiva dipendenza da algoritmi e la potenziale marginalizzazione di figure creative, con linee guida per un uso responsabile (es. mantenere il controllo umano sulle fasi chiave, integrare watermark per distinguere contenuti generati).

Lo studio non si limita a una validazione tecnica, ma delinea un nuovo paradigma di co-creazione che prevede:

1. **Sinergia uomo-macchina:** Esempi dimostrano come designer possano utilizzare gli output generativi come bozze iniziali, risparmiando tempo su iterazioni minori e concentrandosi su aspetti concettuali.
2. **Scalabilità cross-settoriale:** L'architettura proposta è adattabile a settori come l'arredamento, il beauty e altri settori dove la personalizzazione visiva è cruciale.
3. **Integrazione con tecnologie emergenti:** In futuro si potrebbe prevedere l'uso di tecnologie AR per provare virtualmente i capi generati, creando un ecosistema di shopping immersivo.

L'obiettivo fondamentale di questo lavoro è verificare in che misura le immagini personalizzate generate con la pipeline proposta migliorino le intenzioni d'acquisto rispetto alle immagini dei capi di partenza. Diversi studi indicano che la contestualizzazione emozionale — ad esempio, l'inserimento di un capo in scenari affini agli interessi personali dell'utente — può incrementare significativamente la percezione di valore e l'efficacia comunicativa dell'annuncio, con ricerche che segnalano un aumento della memorabilità fino al 40% quando il contesto è percepito come rilevante [Bibliografia 14, Sitografia 30], un dato promettente per aziende alla ricerca di un vantaggio competitivo.

Inoltre, con questo lavoro si intende non solo contribuire alla comprensione di come l'IA possa essere impiegata per la generazione di contenuti creativi, ma anche delineare uno scenario futuro in cui la tecnologia diventi un alleato fondamentale per la creatività umana. Immaginiamo campagne pubblicitarie in cui designer impostano direttive artistiche (es. palette cromatiche, mood narrativo) mentre l'IA genera migliaia di varianti testabili in tempo reale, ottimizzando così il dialogo tra intuizione umana e precisione algoritmica.

1.2 Struttura della Tesi

Il documento di tesi si articola in diverse sezioni. Dopo questa introduzione, verrà fornita una panoramica sullo stato dell'arte esplorando alcuni esempi di campagne pubblicitarie di successo basate sull'IA, alcuni elementi teorici dell'IA generativa e dei modelli utilizzati. Successivamente, verrà descritto il processo realizzativo del progetto illustrando la metodologia applicata e le applicazioni specifiche nel settore della moda. Infine, saranno discussi i risultati ottenuti e le implicazioni pratiche, creative ed etiche di questa innovazione. Per la consultazione del codice, dei materiali integrativi e delle sperimentazioni aggiuntive non incluse nel presente documento, è disponibile la repository ufficiale del progetto al seguente indirizzo: <https://github.com/alessandropier/Personalized-Fashion-Ads>. Al suo interno è possibile trovare tutto il necessario per approfondire ulteriormente gli aspetti implementativi e analitici del sistema sviluppato.

2. ANALISI DELLO STATO DELL'ARTE

In questo capitolo si analizzerà lo stato dell'arte esplorando alcuni esempi di campagne pubblicitarie di successo basate sull'IA ed approfondendo alcuni elementi teorici dell'IA generativa e dei modelli utilizzati, con un focus sulle tecnologie text-to-image e sulla loro evoluzione.

2.1 Large Language Models (LLM)

I Large Language Models (LLM) rappresentano una delle più recenti e avanzate innovazioni nel campo dell'Intelligenza Artificiale (IA) e del Natural Language Processing (NLP). Essi sono modelli di apprendimento profondo, addestrati su enormi quantità di dati testuali, che hanno la capacità di comprendere, generare e manipolare il linguaggio naturale con un alto grado di coerenza e pertinenza contestuale.

Gli LLM sono alla base di molte applicazioni moderne, tra cui chatbot, assistenti virtuali, generazione automatica di testo, traduzione automatica, e modelli multimodali in grado di processare input testuali e visivi [Bibliografia 22]. Grazie alla loro architettura basata sulle reti neurali transformer, questi modelli riescono a catturare relazioni complesse tra le parole e i concetti, permettendo loro di rispondere a domande, completare testi e generare descrizioni dettagliate di immagini.

2.1.1 Architettura Transformer

Gli LLM si basano principalmente sull'architettura Transformer, introdotta dal paper “Attention Is All You Need” pubblicato nel 2017 [Bibliografia 32, 33, 34]. Questa architettura ha rivoluzionato il settore del NLP grazie alla sua capacità di gestire in modo efficiente l'apprendimento contestuale attraverso il meccanismo di self-attention. Questo meccanismo consente al modello di attribuire pesi diversi alle parole in una sequenza, permettendogli di mantenere una memoria a lungo raggio delle dipendenze semantiche e sintattiche.

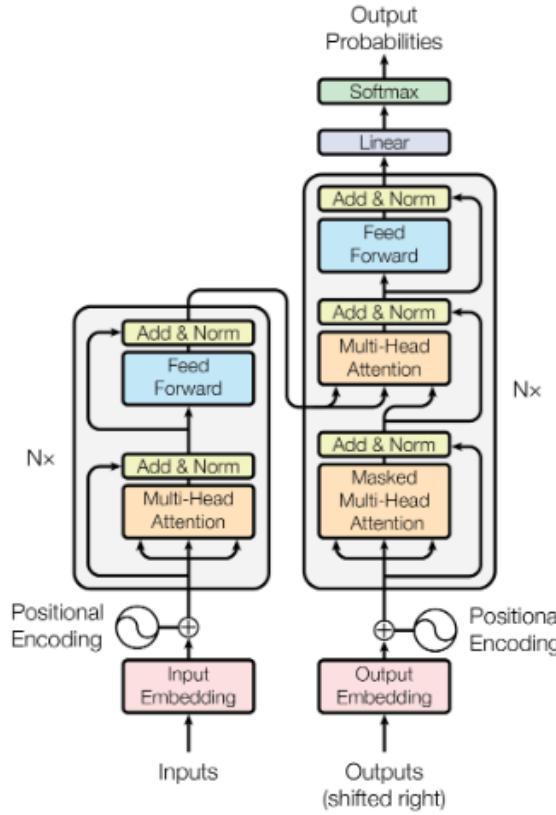


Figura 1: Architettura Transformer

Un tipico modello Transformer è composto da:

- **Embeddings:** rappresentano le parole o i token in uno spazio vettoriale ad alta dimensione.
- **Stack di livelli encoder-decoder:** il modello è costituito da più strati (layers) di encoder e decoder, ciascuno con i propri meccanismi di attenzione e feed-forward.
- **Self-Attention Mechanism:** ogni parola della sequenza viene messa in relazione con tutte le altre, calcolando un punteggio di rilevanza tramite una matrice di attenzione.
- **Feed-Forward Neural Network:** dopo il meccanismo di attenzione, i dati passano attraverso una rete neurale completamente connessa per l'elaborazione finale.
- **Add & Norm Layers:** ogni strato contiene meccanismi di normalizzazione per stabilizzare l'apprendimento e migliorare la convergenza del modello.

Il modello Transformer utilizza il meccanismo positional encoding per mantenere informazioni sulla sequenza delle parole, dato che, a differenza delle reti ricorrenti (RNN), non elabora i dati in ordine sequenziale. Questo permette un'elaborazione parallela più efficiente, migliorando drasticamente la velocità di addestramento e inferenza.

Encoder-Decoder

L'architettura originale dei Transformer, introdotta da Vaswani (2017) [*Bibliografia 8*], seguiva un paradigma encoder-decoder, il quale è particolarmente utile nei modelli generativi e nella traduzione automatica. L'encoder e il decoder lavorano insieme per trasformare un input testuale in un output elaborato. Tuttavia, molti dei modelli linguistici di grandi dimensioni (LLM) oggi più diffusi adottano una struttura **decoder-only**, ottimizzata per la generazione autoregressiva di testo.

Encoder

Ha il compito di processare l'input trasformandolo in una rappresentazione interna comprensibile per il modello. È composto da più strati identici, ognuno dei quali contiene:

1. Un meccanismo di self-attention, che permette di pesare le parole in base alla loro rilevanza rispetto al contesto.
2. Una rete neurale feed-forward, che elabora le informazioni ricevute dal layer di attenzione.
3. Strati di normalizzazione (layer normalization) e connessioni residue per stabilizzare l'apprendimento. L'encoder riceve in ingresso una sequenza di token (parole o sottoparole) e genera una rappresentazione astratta dell'input che verrà poi utilizzata dal decoder.

Decoder

Prende la rappresentazione generata dall'encoder e la trasforma in una sequenza di output comprensibile. Anch'esso è composto da più strati, con le seguenti componenti:

1. Un meccanismo di masked self-attention, che permette di generare il testo in maniera autoregressiva, impedendo al modello di vedere parole future durante la generazione.
2. Un meccanismo di cross-attention, che mette in relazione la rappresentazione dell'input prodotta dall'encoder con le informazioni già generate dal decoder.
3. Una rete feed-forward e strati di normalizzazione simili a quelli dell'encoder.

Grazie a questa architettura, il modello è in grado di apprendere relazioni complesse tra input e output, permettendo la generazione di testi coerenti e pertinenti. Il meccanismo **encoder-decoder** è alla base di modelli come T5 (Text-to-Text Transfer Transformer) [*Bibliografia 15*] e BART (Bidirectional and Auto-Regressive Transformer) [*Bibliografia 16*], utilizzati per compiti di generazione testuale avanzata. Tuttavia, molti Large Language Model (LLM) moderni sono basati su un'architettura **decoder-only**, progettata per la generazione autoregressiva di testo: tra questi si annoverano GPT-2 [*Bibliografia 17*], GPT-3 e GPT-4, che si distinguono per l'elevata capacità di completamento, traduzione e risposta a domande.

2.1.2 Prompt Engineering

Il Prompt Engineering è una tecnica utilizzata per guidare e controllare il comportamento degli LLM (Large Language Models) durante la generazione di testo per compiti specifici. Consiste nel fornire al modello un prompt o una serie di istruzioni che rappresentano le linee guida che deve seguire durante la generazione. Questa strategia è fondamentale per ottenere risultati coerenti e accurati nei task specifici, consentendo di personalizzare l'output del modello in base alle esigenze dell'utente.

Per migliorare la generazione delle risposte del modello, esistono diversi parametri che possono essere regolati:

- **Temperature**: regola la casualità delle risposte generate. Un valore basso (es. 0.2) rende il modello più deterministico e conservativo, mentre un valore alto (es. 0.8) favorisce risposte più creative e variegate.
- **Top-p** (nucleus sampling): permette di selezionare solo i token con una determinata probabilità cumulativa. Aiuta a bilanciare coerenza e creatività, evitando scelte meno rilevanti.
- **Max Length**: definisce il numero massimo di token che il modello può generare in risposta a un prompt.
- **Stop Sequences**: sono specifiche sequenze di caratteri o parole che inducono il modello ad arrestare la generazione se incontrate.
- **Repetition Penalty**: regola la tendenza del modello a ripetere parole o frasi già presenti nel testo generato.

Questi parametri possono essere configurati direttamente nelle API dei modelli generativi, consentendo agli sviluppatori di adattare la generazione del testo in base alle necessità del contesto.

Esistono diverse tecniche di prompting che variano in base alla loro complessità e alla quantità di informazioni fornite al modello per la generazione delle risposte:

- **Zero-shot prompting**: consiste nel fornire una richiesta al modello senza esempi o contesto aggiuntivo. Il modello elabora la risposta basandosi esclusivamente sulle conoscenze acquisite durante la fase di addestramento.
- **Few-shot prompting**: include alcuni esempi di input-output all'interno del prompt, guidando il modello nella generazione della risposta. È particolarmente utile per compiti che richiedono un formato specifico o che beneficiano di una maggiore contestualizzazione.

In generale, il prompt engineering è essenziale per sfruttare appieno il potenziale degli LLM. Stabilire un buon prompt permette di migliorare la qualità delle risposte e di adattare i modelli a diverse applicazioni.

2.1.3 Addestramento e Fine-Tuning

L'addestramento di un LLM avviene in due fasi principali:

1. **Pre-Training:** in questa fase, il modello viene esposto a una grande quantità di dati testuali per apprendere strutture linguistiche, relazioni semantiche e conoscenze generali. Questo avviene tramite tecniche come:
 - a. Masked Language Modeling (MLM), dove alcune parole vengono nascoste e il modello deve prevederle.
 - b. Auto-Regressive Modeling (AR), dove il modello apprende a predire il prossimo token in una sequenza.
 - c. Contrastive Learning, usato nei modelli multimodali per correlare testo e immagini.
2. **Fine-Tuning:** dopo il pre-addestramento, il modello può essere adattato a compiti specifici attraverso un fine-tuning su dataset più piccoli ma specializzati. Questo processo permette di:
 - a. Migliorare le prestazioni su task specifici come la classificazione del testo, la generazione di contenuti e la traduzione.
 - b. Personalizzare il comportamento del modello per adattarlo alle esigenze di specifici settori.
 - c. Ridurre i bias presenti nel modello grazie all'uso di dati curati e annotati manualmente.

Il fine-tuning può avvenire tramite metodi come:

- **LoRA** (Low-Rank Adaptation), che riduce i parametri aggiornati, rendendo il processo più efficiente.
- **Instruction Tuning**, dove il modello viene addestrato con esempi di istruzioni per migliorare la capacità di seguire comandi.
- **RLHF** (Reinforcement Learning from Human Feedback), che affina il modello tramite valutazioni umane per migliorarne l'affidabilità.

Queste fasi permettono agli LLM di essere applicabili a una vasta gamma di domini, migliorandone la precisione e la rilevanza nei diversi contesti.

2.2 Vision Language Models (VLM)

I Vision-Language Models (VLM) rappresentano una classe avanzata di modelli di intelligenza artificiale progettati per elaborare e comprendere sia input visivi che testuali. Questi modelli combinano la capacità di interpretare immagini e video con la comprensione del linguaggio naturale, permettendo applicazioni sofisticate come la generazione di descrizioni di immagini, la ricerca multimodale, la risposta a domande basate su contenuti visivi e la creazione di immagini a partire da descrizioni testuali.

L'avvento dei VLM ha rivoluzionato il campo della computer vision e del natural language processing (NLP), consentendo l'integrazione di due dei principali domini dell'intelligenza artificiale. I modelli più avanzati, come CLIP, ALIGN, Flamingo [Bibliografia 18, 19, 20] e Gemini, hanno dimostrato capacità notevoli nella comprensione semantica di immagini e testi, riducendo il divario tra la percezione umana e quella artificiale.

2.2.1 Struttura Generale dei VLM

I Vision-Language Models si basano su un'architettura multimodale, che integra due componenti fondamentali:

- **Encoder Visivo:** responsabile dell'analisi dell'input visivo. Può essere una rete neurale convoluzionale (CNN), un Vision Transformer (ViT) o una combinazione di entrambi. Il compito di questo modulo è estrarre feature significative dall'immagine, creando una rappresentazione vettoriale utile per l'integrazione con il testo.
- **Encoder Testuale:** basato su modelli di linguaggio di ultima generazione, come BERT, GPT o T5, il cui scopo è trasformare l'input testuale in una rappresentazione densa e significativa. Questi modelli possono operare su token singoli, frasi o interi paragrafi, consentendo una comprensione contestuale avanzata.

L'interazione tra i due encoder avviene attraverso vari metodi, tra cui:

- **Allineamento nello spazio latente**: immagini e testo vengono mappati in uno spazio vettoriale condiviso, consentendo una facile associazione tra concetti visivi e linguistici. Questo approccio è usato in CLIP e ALIGN.
- **Cross-attention**: le rappresentazioni visive e testuali vengono fuse in una pipeline unificata che permette l'interazione tra le due modalità. Modelli come Flamingo utilizzano questa tecnica per migliorare la comprensione contestuale tra le informazioni.
- **Feature fusion**: le feature estratte dai due encoder vengono concatenate o combinate tramite reti neurali dense, per generare una rappresentazione multimodale più ricca.

2.2.2 Addestramento dei VLM

L'addestramento dei Vision-Language Models si basa su dataset multimodali di grande scala, contenenti milioni di coppie immagine-testo. Le principali tecniche di training includono:

- **Contrastive Learning**: massimizza la similarità tra coppie correlate e la minimizza tra coppie non correlate.
- **Masked Language Modeling (MLM)**: applicato all'elaborazione testuale, aiuta il modello a comprendere relazioni semantiche nascoste.
- **Image-Conditioned Text Generation**: addestra il modello a generare descrizioni testuali accurate basandosi su input visivi.

2.3 Latent Diffusion Modes (LDM)

I Latent Diffusion Models (LDM) rappresentano una classe avanzata di modelli generativi progettati per creare immagini a partire da input testuali o altri segnali condizionanti. Essi fanno parte della famiglia più ampia dei modelli di diffusione, che hanno rivoluzionato la generazione di contenuti visivi sintetici grazie alla loro capacità di produrre immagini ad alta qualità e risoluzione con un costo computazionale relativamente inferiore rispetto ai modelli di diffusione tradizionali.

L'introduzione degli LDM ha avuto un impatto significativo nel campo dell'intelligenza artificiale creativa, dando vita a modelli come Stable Diffusion, MidJourney e FLUX, ampiamente utilizzati per applicazioni artistiche, pubblicitarie e di design. Questi modelli si basano su una combinazione di tecniche di apprendimento profondo e rappresentazioni latenti che consentono di ottimizzare il processo generativo.

2.3.1 Struttura dei LDM

Gli LDM si basano su un'architettura di modelli di diffusione, ma operano in uno spazio latente piuttosto che nello spazio dei pixel, riducendo significativamente i costi computazionali e migliorando la qualità dell'output generato [Bibliografia 21]. Le componenti principali di un LDM includono:

- **Encoder Latente:** riduce l'input originale (es. immagini) in una rappresentazione compatta nello spazio latente.
- **Modello di Diffusione:** applica un processo di rumore additivo alla rappresentazione latente e successivamente apprende a invertire questo processo per generare nuove immagini.
- **Decoder:** riconverte la rappresentazione latente generata in un'immagine visibile ad alta risoluzione.

Questa struttura consente agli LDM di ottenere un'efficienza computazionale superiore rispetto ai modelli di diffusione classici, che operano direttamente su immagini ad alta risoluzione.

La fase di applicazione del rumore additivo utilizza una tecnica chiamata **denoising progressivo**, che consiste nei seguenti passaggi:

1. **Addestramento:** Il modello apprende un processo di diffusione inversa eliminando il rumore dalle immagini latenti corrotte.
2. **Generazione:** Viene aggiunto rumore casuale a una rappresentazione latente, che poi il modello decodifica progressivamente fino a ottenere un'immagine chiara e coerente con l'input testuale.
3. **Condizionamento:** Il modello può essere guidato da un prompt testuale o altri segnali condizionanti, migliorando la qualità e la coerenza del risultato.

Questo approccio ha permesso ai LDM di diventare lo standard per la generazione di immagini ad alta qualità, con applicazioni che spaziano dall'arte digitale alla pubblicità.

2.3.2 Addestramento dei LDM

L'addestramento degli LDM segue un processo che combina tecniche avanzate di apprendimento auto-supervisionato e modelli generativi. Le principali fasi di addestramento includono:

- **Pre-training con dataset visivi:** i modelli vengono addestrati su vasti insiemi di dati contenenti milioni di immagini e descrizioni testuali associate (es. LAION-5B per Stable Diffusion).
- **Fase di rumore e de-rumore:** il modello impara a prevedere e rimuovere progressivamente il rumore introdotto nelle rappresentazioni latenti, affinando la capacità di generare immagini realistiche.
- **Fine-tuning con tecniche di text-to-image:** consente di migliorare la qualità della generazione di immagini condizionate dal testo, permettendo maggiore fedeltà alle descrizioni testuali fornite dagli utenti.

L'addestramento degli LDM richiede un grande investimento computazionale, ma il loro design efficiente consente un'inferenza più rapida rispetto ai tradizionali modelli di diffusione operanti nello spazio dei pixel.

2.3.3 Differenze tra LDM e Altri Modelli Generativi

Gli LDM si distinguono da altre tecnologie di generazione di immagini, come **GAN (Generative Adversarial Networks)** e **VAE (Variational Autoencoders)**, per il loro **compromesso tra qualità e efficienza computazionale**. Rispetto ai modelli GAN, gli LDM tendono a produrre immagini più dettagliate e coerenti, riducendo i problemi di mode collapse e artefatti visivi. Rispetto ai VAE, gli LDM generano immagini con maggiore fedeltà e minore perdita di dettaglio [Bibliografia 21].

Inoltre, a differenza dei **Vision-Language Models (VLM) come Gemini**, che possono processare e comprendere sia testo che immagini, gli **LDM si**

limitano alla generazione di immagini a partire da testo, senza capacità di comprensione o interpretazione semantica del contenuto visivo.

2.4 Esempi di Campagne di Successo Basate sull'IA

L'intelligenza artificiale si è affermata come una risorsa strategica non solo per migliorare l'efficienza delle campagne pubblicitarie, ma anche per rivoluzionare il processo creativo. Oggi, i marchi leader a livello globale utilizzano l'IA per creare esperienze uniche, altamente personalizzate e spesso interattive, che coinvolgono i consumatori in modi mai visti prima [Sitografia 7, 8, 31]. Grazie a strumenti avanzati come modelli generativi, realtà aumentata e machine learning, è possibile ideare campagne che uniscono estetica e innovazione, offrendo messaggi mirati e rilevanti per diversi target di pubblico.

Esploriamo alcuni esempi concreti di come aziende iconiche abbiano sfruttato l'IA per realizzare campagne di successo. Questi casi mostrano come tecnologie all'avanguardia siano state applicate per migliorare il coinvolgimento, semplificare i processi creativi e consolidare il rapporto tra brand e consumatori.

Mostriamo ora alcune campagne emblematiche che hanno fatto scuola nell'utilizzo dell'intelligenza artificiale nel marketing e nella pubblicità.

2.4.1 Coca-Cola e l'IA Creativa

Coca-Cola ha utilizzato strumenti di intelligenza artificiale per migliorare l'efficacia delle sue campagne pubblicitarie. Alcuni esempi includono:

1. The Holiday Magic is Coming

Coca-Cola ha lanciato annunci televisivi generati da IA, inclusa una versione aggiornata della famosa campagna "The Holidays Are Coming". Questa campagna ha utilizzato AI per ricreare scenari natalizi con camion decorati che attraversano foreste innevate e città, illuminando l'ambiente con luci natalizie.



Figura 2: Frame del video pubblicitario “The Holiday Magic is coming” realizzato da Coca-Cola per la campagna pubblicitaria del Q4, 2024. [Sitografia 6]

2. Create Real Magic

Coca-Cola ha collaborato con DALL-E e altri strumenti generativi per permettere agli utenti di creare opere d'arte basate sul brand.



Figura 3: Immagine generata da un utente utilizzando l'AI generativa di Coca-Cola durante il Q3, 2024. [Sitografia 7]

Alcune delle immagini create dagli utenti sono state esposte in luoghi iconici a livello globale, rafforzando così il legame tra il brand e i clienti, facendoli sentire parte integrante della sua identità e creatività.



Figura 4: Alcune immagini generate dagli utenti mediante l'IA mostrate a New York City. [Sitografia 7]

2.4.2 L'Oréal e i Modelli Virtuali

Per affrontare i cambiamenti nella creazione di contenuti, L'Oréal collabora con oltre 50.000 influencer e creatori esperti di intelligenza artificiale, realtà aumentata, realtà virtuale e modellazione 3D. Queste partnership, presenti in tutto il mondo con un forte focus in mercati come la Cina, si estendono su piattaforme popolari come Facebook e Instagram di Meta, TikTok e Snapchat, raggiungendo una vasta gamma di consumatori.

L'Oréal si concentra su tre aree chiave: migliorare la preparazione tecnologica, garantire che l'IA sia adattabile e modulare, e gestire efficacemente i dati del marchio. L'obiettivo è combinare la creatività umana con l'intelligenza artificiale per personalizzare l'esperienza di bellezza attraverso la tecnologia.

Di seguito vengono presentate alcune delle strategie di marketing innovative adottate da L'Oréal, che evidenziano come il brand abbia integrato l'intelligenza artificiale e altre tecnologie avanzate nelle sue campagne.

1. Beauty Genius con ModiFace

Uno dei servizi innovativi di L'Oréal è Beauty Genius, un consulente virtuale che semplifica la scelta dei prodotti di bellezza per i consumatori. Sviluppato utilizzando tecnologie di intelligenza artificiale e realtà aumentata di ModiFace (acquisita da L'Oréal nel 2018), Beauty Genius offre analisi personalizzate della pelle, consente agli utenti di "provare" virtualmente il trucco e fornisce raccomandazioni su misura, selezionando tra oltre 750 opzioni di skincare, makeup e haircare. Aggrega contenuti dai siti web e dai social media dei marchi L'Oréal, fungendo da risorsa educativa con consigli e suggerimenti di bellezza. Inoltre, permette di diagnosticare le condizioni della pelle e suggerire routine di bellezza personalizzate.

2. Generazione di Contenuti Creativi

L'Oréal utilizza AI per creare contenuti pubblicitari che non dipendono da immagini falsamente generate di volti o corpi, ma enfatizzano la personalizzazione e il coinvolgimento.

Il GenAI Content Lab di L'Oréal è uno spazio dedicato alla sperimentazione creativa per la produzione di contenuti legati alla bellezza. Questo laboratorio favorisce la collaborazione con una rete di partner e sviluppa modelli di servizio che sfruttano l'IA per potenziare la creatività. Inoltre, stabilisce linee guida e standard etici per l'uso dell'IA, garantendo che le immagini generate (volti, pelle, corpi o capelli) non vengano utilizzate in modo fuorviante. L'IA viene impiegata internamente per attività come brainstorming, progettazione e pianificazione, assicurando un uso responsabile e trasparente della tecnologia.

Sperimentando con l'IA generativa, L'Oréal realizza contenuti di marketing pertinenti e coinvolgenti. Questa tecnologia si è dimostrata efficace nell'ottimizzazione delle immagini dei prodotti e nella creazione di sfondi, riducendo significativamente le risorse necessarie per i tradizionali servizi fotografici. Oltre alle immagini statiche, l'azienda sta esplorando la generazione di video per migliorare il marketing visivo.

L'Oréal utilizza la piattaforma Nvidia Omniverse per creare modelli 3D dettagliati dei propri prodotti, che possono essere inseriti in diversi contesti e ambientazioni. Questo approccio offre un duplice vantaggio: riduce i tempi e le risorse richiesti dai servizi fotografici tradizionali e garantisce flessibilità per la presentazione dei prodotti su diverse piattaforme, come i social media e l'e-commerce.

Grazie a Omniverse, L'Oréal può adattare facilmente le immagini al contesto, rendendole più adatte ai vari canali digitali. Secondo Asmita Dubey, questa innovazione ha rivoluzionato il marketing nel settore della bellezza, consentendo all'azienda di soddisfare l'infinita diversità delle esigenze dei consumatori e di offrire servizi personalizzati che facilitano la navigazione tra le numerose opzioni di bellezza disponibili.

2.4.3 Nike, Realtà Aumentata e IA Generativa

Nike ha integrato l'IA e la realtà aumentata per migliorare le esperienze pubblicitarie e di vendita:

1. Nike Fit

Un'applicazione basata su Intelligenza Artificiale, Realtà Aumentata e Computer Vision per scansionare i piedi dei clienti e suggerire la misura di scarpa perfetta. È stata usata come parte di campagne pubblicitarie per sottolineare l'innovazione del brand.

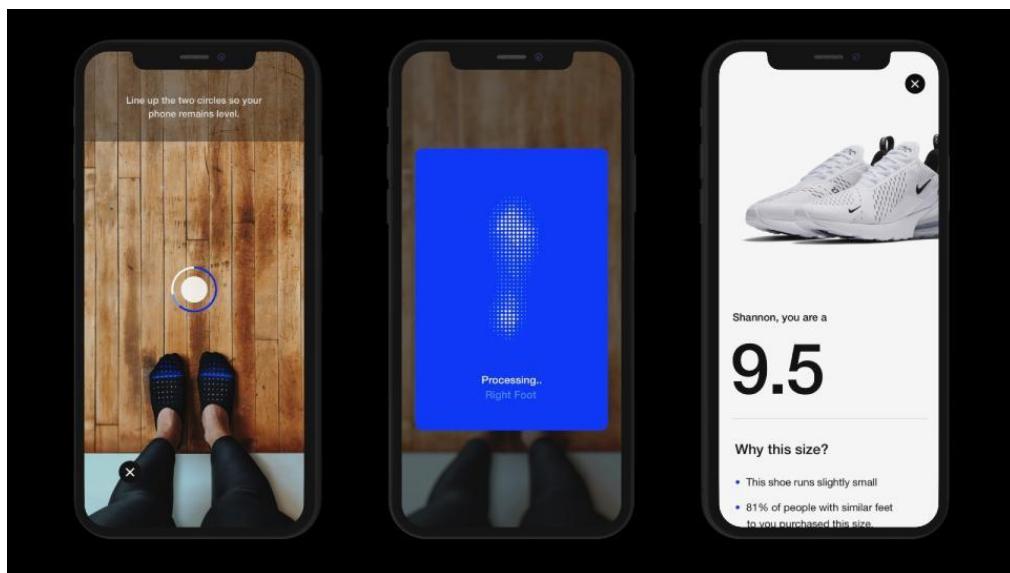


Figura 5: Funzionamento dell'applicazione Nike Fit. [Sitografia 13]

Nike ha integrato la tecnologia di realtà aumentata (AR) nella sua app, rendendo la funzionalità Nike Fit un elemento centrale e non solo un esperimento o una strategia di marketing temporanea. Attraverso Nike Fit, gli utenti possono misurare i propri piedi direttamente da casa, con i dati salvati nel profilo personale sull'app Nike. Questo elimina la necessità di ripetere la misurazione ogni volta che si desidera acquistare un nuovo paio di scarpe.

Nei negozi fisici, i dipendenti possono utilizzare le informazioni del profilo Nike del cliente, semplicemente scansionando un codice QR dall'app per suggerire le taglie migliori per diversi modelli sostituendo il tradizionale dispositivo Brannock. Ad esempio, chi indossa taglie diverse tra modelli

come Air Max e Epic Reacts può ricevere suggerimenti specifici sulla misura ideale, considerando le differenze di larghezza e vestibilità tra i modelli.

Nike evidenzia come questa tecnologia semplifichi anche l'esperienza per i genitori, consentendo loro di misurare i piedi dei propri figli a casa, evitando spostamenti e permettendo di effettuare ordini online con facilità. Nike Fit può essere usato periodicamente per aggiornare le misure man mano che i piedi dei bambini crescono.

2. IA Generativa per Product Design e Progetto Athlete Imagined Revolution (A.I.R.)

Nike sta sviluppando un modello di intelligenza artificiale generativa basato su un large-language-model (LLM) progettato su misura per il design di prodotti, utilizzando dati esclusivi forniti dagli atleti quindi con un'alta personalizzazione. John Hoke, Chief Innovation Officer di Nike, ha rivelato questa iniziativa durante un recente evento a Parigi, spiegando che il modello combina dati sulle performance degli atleti con dati pubblici, creando un "giardino privato" di informazioni per l'addestramento dell'IA.

Questa iniziativa rappresenta un passo importante nella strategia di Nike per rivoluzionare il design e la produzione dei prodotti. Integrando l'IA con tecnologie avanzate come la realtà virtuale (VR) e la stampa 3D, Nike sta trasformando il processo di prototipazione. Hoke ha descritto questa combinazione come una "nuova alchimia" che consente di passare dall'idea al prototipo in poche ore, anziché settimane o mesi. Questo approccio accelera il coinvolgimento con gli atleti, rendendo possibile tradurre rapidamente le loro visioni in prodotti tangibili.

Nonostante le limitazioni dell'IA generativa, come gli errori o le "allucinazioni" comuni a questa tecnologia, Hoke ha evidenziato il potenziale creativo dell'IA, definendola "carburante per la creatività". Ha sottolineato che l'IA non sostituirà la creatività umana, ma la potenzierà, affermando: "È uno strumento straordinario guidato dall'immaginazione umana". Hoke vede anche gli errori come opportunità per ampliare le prospettive creative e spingere ulteriormente i confini dell'innovazione.

Un esempio concreto di questa tecnologia è stato mostrato con il progetto Athlete Imagined Revolution (AIR), in cui l'IA generativa è stata utilizzata per creare prototipi di scarpe per atleti di alto livello, come Sha'Carri Richardson e Kylian Mbappé. I modelli sono stati progettati in base alle preferenze degli atleti, utilizzando input forniti all'IA per generare centinaia di design, successivamente perfezionati attraverso tecniche di fabbricazione digitale.

Sebbene Hoke abbia riconosciuto che le normative attuali potrebbero limitare l'utilizzo di queste scarpe iper-personalizzate nelle competizioni, ha suggerito che esse potrebbero offrire benefici significativi, sia fisici che psicologici. “È un'estensione di chi sono”, ha dichiarato, indicando come i prodotti personalizzati potrebbero fornire agli atleti un vantaggio unico.



Figura 6: Alcuni design di scarpe realizzati per gli atleti Nike. [Sitografia 15]

Nike ha sviluppato un approccio innovativo per progettare calzature personalizzate per 13 atleti provenienti da quattro discipline sportive: atletica leggera, calcio, basket e tennis. Il processo ha avuto inizio con sessioni di ascolto approfondito, in cui i designer hanno raccolto input direttamente dagli atleti, esplorando ogni dettaglio delle loro preferenze e ispirazioni. Le domande spaziavano dal tipo di design desiderato (tradizionale o audace, monolitico o frammentato) fino ai legami personali

degli atleti con luoghi, persone e momenti significativi. L'obiettivo era tradurre questi elementi nella progettazione delle scarpe, catturando ciò che Nike definisce "la verità dell'atleta".

Secondo Roger Chen, Vicepresidente di Nike NXT, Digital Product Creation, la "verità dell'atleta" rappresenta una comprensione quasi cellulare di ciò che consente a un atleta di pensare, sentire e performare al meglio. Questa filosofia ha guidato l'intero processo di design, basandosi su rapporti autentici tra designer e atleti, fondamentali per comprendere chi si sta servendo.

I dati raccolti durante queste sessioni di ascolto sono stati inseriti in comandi dettagliati per l'intelligenza artificiale, che ha generato centinaia di proposte visive istantaneamente. Questi output hanno offerto una quantità enorme di ispirazioni per i designer, accelerando significativamente il processo creativo. Mentre in passato creare i punti di partenza per i progetti richiedeva mesi; grazie all'IA questo è ora possibile in pochi secondi.

Chen ha paragonato l'IA a una "matita più affilata e intelligente", sottolineando che i designer mantengono sempre il controllo. L'IA serve come strumento per amplificare la creatività, restituendo mondi visivi complessi basati sugli input ricevuti, ma tutto parte dalla fase iniziale di ascolto e comprensione approfondita degli atleti. Questo approccio ha permesso di sviluppare prototipi più specifici e di rafforzare ulteriormente il legame tra Nike e i suoi atleti.

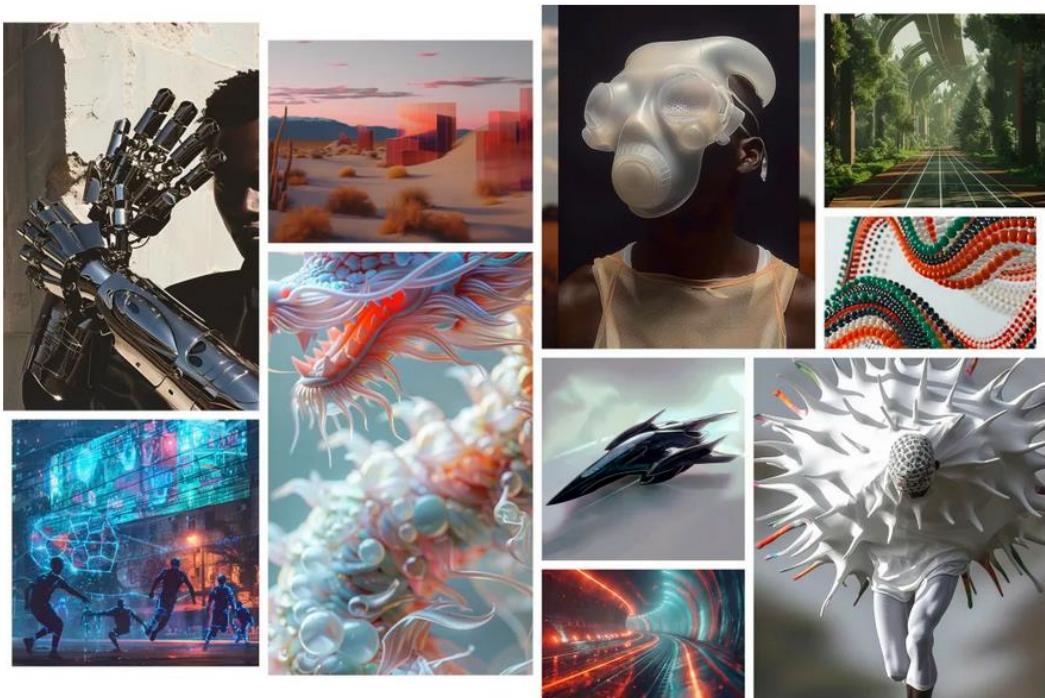


Figura 7: Un piccolo assaggio delle centinaia di visual create da una vasta gamma di strumenti generativi — tutti in un singolo pomeriggio. [Sitografia 15]

Dopo aver analizzato centinaia di visual creati dall'IA, i team di Nike hanno messo da parte l'"intelligente matita" dell'IA per fare ciò che sanno fare meglio: progettare scarpe in base alle specifiche esatte richieste dagli atleti. I 13 team hanno utilizzato le forme, le texture, le figure generate e persino interi mondi visivi ispirati dalle immagini create dall'IA per sviluppare tre concetti radicalmente innovativi di scarpe, pensati come una nuova espressione della linea *Air* di Nike. Quest'ultima non rappresenta solo una tecnologia di ammortizzazione basata su unità a gas inserite nella suola, ma incarna anche un'identità estetica ben riconoscibile, legata a concetti di leggerezza, sospensione e innovazione. Il processo creativo non è stato privo di ostacoli: in alcuni casi, i designer si sono trovati a dover superare i bias degli algoritmi generativi, che tendevano a interpretare il concetto di *Air* in maniera letterale o astratta, restituendo un'estetica fluida e organica poco coerente con l'identità visiva e tecnologica del brand.

Eliud Kipchoge, dal Concept al Feedback degli Atleti:

Una volta definiti i tre concetti principali, i team hanno coinvolto gli atleti per raccogliere il loro feedback. Ogni dettaglio delle scarpe è stato discusso, con particolare attenzione agli aspetti estetici, funzionali ed espressivi.

Un esempio significativo è emerso durante una sessione iniziale con il maratoneta Eliud Kipchoge. Analizzando un rendering digitale, Kipchoge ha individuato un problema pratico: il design aerodinamico del tallone scavato, simile a una molla, avrebbe potuto accumulare detriti su percorsi sterrati. Con un foglio di carta, Kipchoge ha schizzato una modifica che collegava l'area cava del tallone, adattando la scarpa al suo ambiente di allenamento. Questo esempio ha dimostrato come il feedback degli atleti possa sfidare le assunzioni progettuali, migliorando l'innovazione.

Il Caso di Sha'Carri Richardson:

Durante un confronto con la campionessa dei 100 metri Sha'Carri Richardson, un termine chiave emerso è stato "grazioso". Pur essendo associata a forza e determinazione, Sha'Carri ha richiesto un design che enfatizzasse l'armonia, rifiutando un'estetica aggressiva simile a un "sandalo da battaglia". Il team ha quindi sviluppato un prototipo che integrasse armoniosamente l'unità inferiore della scarpa con la tomaia e la struttura della manica lungo la gamba.

Collaborazione Creativa:

Questi esempi sottolineano come il processo innovativo di Nike si basi su una collaborazione continua tra designer, atleti e IA, sfidando le assunzioni e spingendo i confini del design per creare prototipi che incarnino le qualità uniche di ogni atleta puntando alla massima personalizzazione possibile.



Figura 8: Una delle ispirazioni che ha portato Eliud Kipchoge al suo prototipo finale. [Sitografia 15]



Figura 9: Una delle ispirazioni che ha portato Eliud Kipchoge al suo prototipo finale. [Sitografia 15]

La Visione di Hoke:

Per John Hoke, Chief Innovation Officer di Nike, il progetto A.I.R. incarna una visione ambiziosa. Gli atleti sono al centro del processo creativo, “Imagined” rappresenta l’ispirazione che Nike trae dall’IA come strumento collaborativo, e “Revolution” segna un cambiamento radicale nel modo di lavorare.

“Il nostro utilizzo degli strumenti generativi ci permette di ascoltare gli atleti con una precisione senza pari,” afferma Hoke. In mani inesperte, l’IA potrebbe produrre design generici, ma grazie all’ascolto attento degli atleti, Nike sfrutta il potere concettuale dell’IA per rispondere ai loro bisogni, trasformando il processo creativo in una collaborazione unica tra umani e macchine. Hoke sottolinea che A.I.R. è solo all’inizio: “Non si torna indietro,” afferma. *“Forma e funzione incontrano la fantasia.”*

Questo progetto non solo ridefinisce il design, ma consolida il ruolo dell’IA come co-creatore nella missione di Nike di creare prodotti che riflettano la visione e le esigenze degli atleti, spingendo sempre oltre i confini dell’innovazione.

3. METODOLOGIA

L'obiettivo di questo capitolo è illustrare la metodologia adottata per lo sviluppo di un sistema innovativo basato su Vision-Language Models (VLM), in grado di generare contenuti pubblicitari personalizzati nel settore moda. Il sistema proposto si distingue per unire, in un'unica pipeline, la capacità creativa dei modelli generativi con una personalizzazione fine basata su dati utente. Questo approccio si differenzia significativamente dalle soluzioni attualmente presenti nel panorama industriale, che tendono a trattare separatamente la generazione creativa e la targettizzazione.

Mentre strumenti come DALL-E o MidJourney producono immagini pubblicitarie generiche, e piattaforme come Meta Ads si limitano a selezionare contenuti preesistenti in base alla profilazione, il sistema proposto genera varianti uniche di uno stesso capo, adattate a micro-target specifici tramite un'analisi congiunta dell'immagine di partenza e del profilo utente. Il risultato è una forma di **Dynamic Prompt Engineering** che traduce automaticamente le preferenze personali in input per la generazione visiva, garantendo coerenza estetica, originalità e rilevanza comunicativa.

La pipeline è progettata per essere altamente modulare e scalabile, adattabile ad altri settori oltre la moda (es. automotive, design), e va oltre casi come **Nike A.I.R.** o le copertine personalizzate di **Netflix**, poiché non si limita alla scelta o modifica di contenuti esistenti, ma crea materiale pubblicitario ex-novo. In questo modo, l'IA non viene impiegata solo per automatizzare o selezionare, ma diventa un vero **strumento creativo e strategico**.

Di seguito si introdurrà la metodologia del sistema e come si distingue da soluzioni già esistenti. Verranno illustrati l'architettura del sistema, i dati utente raccolti in fase di registrazione, le tecniche di prompt engineering utilizzate e il metodo di gestione dei dati. Infine, verranno discusse le metriche di valutazione della metodologia adottata.

3.1 Architettura del Sistema di Generazione

Il cuore del sistema proposto è rappresentato dalla pipeline di generazione automatica dell'immagine personalizzata, mostrata nella **Figura 10**: Pipeline esecutiva di progetto**Figura 10**. Essa integra due componenti fondamentali dell'Intelligenza Artificiale multimodale: un **Vision-Language Model (VLM)** e un **modello generativo Text-to-Image**. L'interazione tra questi due moduli consente di trasformare l'immagine originale di un capo e le caratteristiche di un utente specifico in una nuova immagine pubblicitaria mirata e personalizzata.

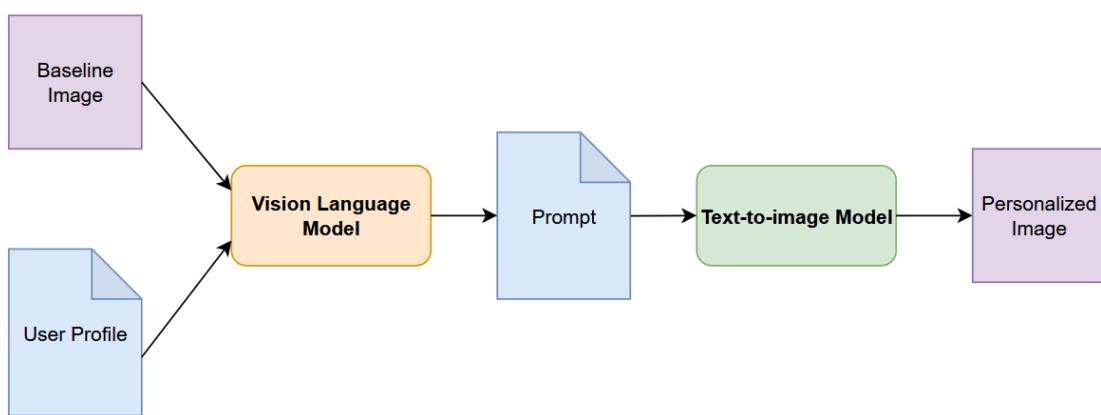


Figura 10: Pipeline esecutiva di progetto

Descrizione della Pipeline

La pipeline di generazione si articola in quattro fasi principali:

1. Input multimodale (Immagine + Profilo Utente)

Il processo ha inizio con due input fondamentali:

- **L'immagine di partenza**, chiamata “baseline”, rappresenta un capo d'abbigliamento reale (es. abito, giacca, gonna).
- **Il profilo utente**, raccolto in fase di registrazione, include dati demografici, preferenze stilistiche, colori preferiti, generi musicali, brand affini e altre informazioni rilevanti per il targeting creativo.

2. Vision-Language Model (VLM)

Il VLM riceve l'immagine baseline e il profilo utente in ingresso, ed elabora una **descrizione testuale personalizzata**. Questo prompt non

è generico: il modello è guidato da istruzioni specifiche a generare una descrizione immaginativa, coerente con l'estetica dell'immagine, ma modellata sulle preferenze dell'utente ottenendo un prompt strutturato pensato per essere compatibile con il modello text-to-image impiegato nella fase successiva.

3. Modello Text-to-Image (LDM)

il prompt testuale generato dal VLM viene utilizzato come input per un modello di generazione visiva che lo trasforma in una **nuova immagine pubblicitaria**. L'output è visivamente coerente con la descrizione e riflette il target personale indicato nei dati utente. Questa fase costituisce la generazione vera e propria dell'immagine passando da contenuto testuale a contenuto visuale.

4. Output: Immagine Personalizzata

L'immagine risultante è un contenuto pubblicitario completamente nuovo, concepito per essere visivamente accattivante e profondamente rilevante per il target. Essa viene utilizzata nella fase di valutazione comparativa da parte dell'utente, attraverso un questionario che confronta l'efficacia della generazione rispetto all'immagine originale.

Riflessioni sull'Architettura

La forza di questa architettura risiede nell'unione di **comprendione semantica** (tramite il VLM) e **capacità generativa visiva** (tramite il modello text-to-image). Questo approccio è particolarmente innovativo perché consente una personalizzazione raffinata e creativa che va oltre le tecniche di targhettizzazione classiche poiché non ci si limita a selezionare contenuti da un set predefinito, ma si generano *ex-novo* contenuti pubblicitari *su misura*, partendo dalla combinazione tra immagine e identità dell'utente.

Inoltre, la modularità della pipeline ne garantisce la scalabilità in quanto ciascun componente può essere sostituito, aggiornato o adattato a settori differenti (es. automotive, design, turismo) mantenendo intatta la logica di personalizzazione guidata da dati.

3.2 Modellazione Utenti

I dati utente rappresentano il core del sistema, in quanto determinano direttamente la rilevanza e l'efficacia delle generazioni. Per garantire output coerenti e ridurre il rischio di allucinazioni del modello (errori generativi come dettagli incongruenti o contesti irrealistici), è stata condotta un'analisi iterativa su oltre 30 parametri potenziali, mantenendo i soli campi determinanti per la personalizzazione. I campi chiave sono i seguenti:

- Nome
- Cognome
- Età
- Nazione
- Genere
- Corrente Artistica Preferita
- Professione
- Colori Preferiti
- Generi Musicali Preferiti
- Cosa Cerca Nei Capi
- Marchi Preferiti
- Interesse in Ambito Moda (in scala da 1 a 5)
- Competenze in Ambito Moda (in scala da 1 a 5)

Dall'analisi empirica emerge che tre categorie di dati utente sono determinanti nel guidare la generazione di immagini pubblicitarie rilevanti:

1. **Gusti Personalì**: corrente artistica preferita, colori preferiti, generi musicali preferiti, marchi preferiti
2. **Genere**
3. **Professione**

Specificando la professione e i gusti dell'utente, il modello generativo crea contenuti in contesti con strutture o palette esteticamente familiari all'utente in modo da migliorare la personalizzazione. Questo livello di granularità non solo migliora la personalizzazione, ma trasforma la pubblicità del capo in un dialogo visivo tra brand e consumatore.

3.3 Scelta delle Immagini di Baseline

3.3.1 Esclusione delle Fonti Commerciali

Prima di procedere con la selezione delle immagini baseline, è stata condotta una riflessione sulle possibili fonti da cui attingere i capi da utilizzare come riferimento. In particolare, si è deciso **di non utilizzare cataloghi digitali standardizzati** come quelli presenti su piattaforme commerciali quali Zalando, Asos o simili.

Questi ambienti, sebbene ricchi di immagini ad alta risoluzione e di ampia varietà, presentano un impianto comunicativo altamente **ripetitivo, impersonale e tabellare**: i capi vengono spesso presentati in griglie visive uniformi (es. 4xN), con modelle e modelli stilizzati, frontalmente in posa neutra, il cui ruolo si riduce a quello di meri "**manichini umani**". La marginalizzazione dell'espressività individuale impedisce una valutazione più completa del contenuto visivo, rendendo difficile la **personalizzazione e contestualizzazione** dei prompt generativi.

Per queste ragioni si è preferito adottare un approccio più qualitativo, attingendo da fonti editoriali in cui l'immagine del capo è **strettamente connessa a una narrazione visiva**, come avviene ad esempio nel portale *Vogue Runway*. Qui l'immagine non è solo documento, ma rappresentazione stilistica, permettendo al sistema di generazione di agire su contenuti visivamente più ricchi, profondi e significativi.

3.3.2 Criteri per la Selezione delle Immagini Baseline

All'interno del sistema proposto, le immagini di baseline rappresentano il punto di partenza per la generazione di contenuti personalizzati mediante modelli generativi. Per garantire l'efficacia dell'esperimento e la validità dei dati raccolti, è stato necessario adottare criteri chiari e strutturati per la selezione di tali immagini.

La selezione è stata effettuata manualmente a partire da materiale fotografico proveniente da "*Vogue Runway*", una delle piattaforme più autorevoli e aggiornate nel settore della moda contemporanea. Le immagini sono state scelte con l'obiettivo di garantire **rappresentatività**,

bilanciamento e varietà, pur mantenendo una coerenza stilistica funzionale all'analisi successiva.

In particolare, sono stati seguiti i seguenti criteri metodologici:

- **Bilanciamento di genere:** sono state selezionate otto immagini in totale, suddivise equamente tra capi femminili e maschili (quattro per ciascun genere). Questa suddivisione consente di esplorare eventuali differenze nelle preferenze degli utenti in base al genere del capo valutato.
- **Rappresentatività e rilevanza:** ciascun capo è stato scelto in modo da risultare significativo dal punto di vista estetico e creativo, privilegiando abiti appartenenti a collezioni recenti di brand di rilievo internazionale. Questo approccio garantisce che le immagini siano in linea con le tendenze attuali del settore moda.
- **Varietà stilistica:** per evitare una visione unilaterale o limitata, le immagini selezionate coprono una gamma ampia di stili (elegante, casual, concettuale, sperimentale), cromie, tessuti e silhouettes.
- **Coerenza visiva:** tutte le immagini presentano caratteristiche fotografiche coerenti, come un'inquadratura frontale e una buona illuminazione, in modo da evitare bias legati alla composizione dell'immagine stessa. Inoltre, è stata posta attenzione affinché lo sfondo non interferisse visivamente con il capo presentato.
- **Pertinenza rispetto al contesto d'uso:** i capi selezionati sono stati scelti considerando la loro capacità di stimolare un confronto estetico significativo tra immagine di partenza e immagine generata. Si è data priorità a contenuti visivamente interessanti, concettualmente distintivi o particolarmente rappresentativi di una proposta stilistica, in modo da garantire una valutazione consapevole e non banale da parte degli utenti coinvolti.

Le immagini di baseline così selezionate costituiscono quindi un campione di riferimento fondamentale per la successiva generazione e valutazione dei contenuti personalizzati. I dettagli tecnici saranno approfonditi nel Capitolo **4. IMPLEMENTAZIONE.**

3.4 Selezione dei Modelli Generativi

Durante la fase progettuale è stata condotta un'analisi comparativa tra diversi modelli generativi e linguistici multimodali, con l'obiettivo di individuare le soluzioni più adatte alle esigenze del sistema in termini di **coerenza semantica, qualità visiva e personalizzazione del contenuto.**

Per la componente **image-text-to-text** e più in generale per l'elaborazione multimodale, sono stati analizzati modelli VLM come *SMOL VLM*, *Qwen/QVQ-72B-Preview*, *LLaMA 3.2 (11B e 90B)* e *Gemini*. Parallelamente, per quanto riguarda i modelli **text-to-image**, sono stati testati diversi generatori, tra cui *FLUX.1-dev*, *FLUX-Super-Realism-LoRA*, *FLUX.1-schnell*, *Flux-Midjourney-Mix2-LoRA*, *Midjourney* e *Stable Diffusion*.

Tutti i modelli sono stati **adeguatamente testati** attraverso esempi reali e confronti qualitativi, valutandone la capacità di generare contenuti rilevanti, esteticamente validi e coerenti con il profilo dell'utente e il contesto richiesto.

A seguito di un'attenta valutazione, sono stati selezionati i due modelli più promettenti per le rispettive funzioni:

- **Gemini 2.0 Flash Experimental**, per la sua efficacia nella comprensione semantica e nella generazione linguistica multimodale;
- **Stable Diffusion 3.5 large**, per la sua versatilità nella generazione visuale e l'ottimo equilibrio tra controllo, qualità dell'immagine e integrazione tecnica.

Successivamente, sono stati condotti ulteriori test specifici al fine di verificare **quali configurazioni e stili generativi** risultassero più coerenti con le finalità del sistema, adattando progressivamente i parametri alle esigenze dell'utente e del contesto d'uso.

3.5 Strategie di Prompting per i Modelli Impiegati

Il sistema prevede l'utilizzo di due modelli generativi:

1. **Gemini Flash 2.0 Experimental**: generazione image-text-to-text
2. **Stable Diffusion 3.5 Large**: generazione text-to-image

I due modelli operano sinergicamente l'uno con l'altro, infatti, l'output del modello image-text-to-image (Gemini) è l'input del modello text-to-image (Stable Diffusion). Per questo motivo è importante impostare un adeguato prompt di input per gemini in maniera tale da avere un ottimo prompt di input per il modello text-to-image caratterizzato da un adeguato livello di descrittività.

3.5.1 Prompting del Vision Language Model

Fornire al modello istruzioni su cosa fare è un modo efficace ed efficiente per personalizzare il suo comportamento [Sitografia 23]. È importante assicurarsi che le istruzioni fornite siano chiare e specifiche. Le istruzioni possono essere semplici come un elenco di istruzioni passo passo o complesse come la mappatura dell'esperienza e del mindset di un utente. A tal fine, sono state sperimentate diverse tecniche di prompting e, dopo un'attenta valutazione, si è giunti alla versione definitiva.

La guida al prompting fornita da Google [Sitografia 23] si rivela un prezioso alleato nell'esecuzione di questo compito. Come evidenziato sia da Google che da altre risorse dedicate all'ingegneria del prompt per Gemini [Sitografia 22], questo modello si distingue per la sua straordinaria versatilità nell'interpretare un'ampia gamma di prompt, indipendentemente dalla loro formattazione. Tale capacità è resa possibile dalle sue avanzate abilità di analisi e sintesi testuale, nonché dalla sua attitudine a estrarre, rielaborare e strutturare le informazioni più rilevanti in maniera coerente ed efficace.

A tal proposito, è fondamentale calibrare con attenzione il prompt di input per Gemini, considerando in maniera approfondita il modello text-to-image che lo elaborerà successivamente. Affinché il processo generativo risulti efficace e coerente con le aspettative, è essenziale formulare il prompt di Gemini non solo in funzione delle sue capacità di interpretazione e rielaborazione, ma anche in perfetta sintonia con le specifiche linee guida

del modello text-to-image che lo segue. Questo approccio consente di massimizzare la qualità dell'output visivo, garantendo che le informazioni trasmesse siano ottimizzate per una resa finale accurata e fedele con quanto desiderato.

Sono stati condotti test di diversa natura, tra cui approcci Zero-Shot, One-Shot e Few-Shot, oltre a sperimentazioni che prevedevano vincoli di varia rigidità, l'ottimizzazione della formattazione delle informazioni fornite dall'utente e la gestione della posizione dell'immagine nel contesto generativo. Ogni test è stato attentamente valutato attraverso l'analisi delle immagini finali prodotte dal modello text-to-image, confrontando l'output con le aspettative iniziali. Sulla base di tali osservazioni, si è proceduto con un affinamento iterativo del prompt, intervenendo attivamente per migliorare l'allineamento tra le richieste testuali e i risultati visivi, garantendo così una maggiore precisione e coerenza nel processo generativo.

3.5.2 Prompting del Latent Diffusion Model

Fornire istruzioni precise e strutturate è essenziale per guidare efficacemente Stable Diffusion nella generazione di immagini allineate alle aspettative [Sitografia 17]. A differenza dei modelli text-to-text, qui la sfida risiede nel tradurre concetti astratti in elementi visivi coerenti, richiedendo coordinazione tra creatività e rigore tecnico.

La guida ufficiale di Stability AI [Sitografia 17] sottolinea l'importanza di una struttura gerarchica del prompt, suddivisa in:

1. **Soggetto Principale:** descrizione del soggetto principale dell'immagine
2. **Contesto e dettagli ambientali:** descrizione dello sfondo e dello spazio in cui è collocato il soggetto
3. **Stile artistico:** descrizione dello stile artistico dell'immagine
4. **Parametri tecnici:** aspetti tecnici dell'immagine da realizzare (es. "full body portrait", "standing shot")

Questa suddivisione più o meno ampliata facilita al modello l'interpretazione delle priorità compositive, riducendo ambiguità.

Sono stati condotti test di diversa natura, tra cui approcci One-Shot e l'uso di Stable tramite il parametro *Style*, oltre a sperimentazioni che prevedevano vincoli di varia rigidità e l'organizzazione del prompt. Sono state confrontate descrizioni minimaliste, descrizioni molto verbose e specifiche, descrizioni strutturate e descrizioni quasi colloquiali. Inoltre, sono stati condotti test che espandono l'utilizzo delle metriche seed, cfg scale, steps e sampler. Infine, sono anche state confrontate diverse versioni del modello stable diffusion tra cui 3-large, 3.5-large e 3.5-large di hugging face. Ogni test è stato attentamente valutato attraverso l'analisi delle immagini prodotte dal modello, confrontando l'output con le aspettative iniziali e le altre generazioni. Sulla base di tali osservazioni, si è proceduto intervenendo attivamente per migliorare l'allineamento tra le richieste testuali e i risultati visivi, garantendo così una maggiore precisione e coerenza nel processo generativo.

3.6 Tecniche di Gestione dei Dati (CSV e Database)

Il sistema prevede la gestione di una grande quantità di dati e richiede inoltre la possibilità di eseguire interrogazioni e analisi approfondite. Per gestire il volume crescente di dati e supportare query complesse, il sistema adotta un workflow a due livelli:

1. **Fase Iniziale – Raccolta dati in CSV:** I dati grezzi (registrazione utenti, log delle generazioni, feedback) vengono memorizzati in file CSV strutturati, ottimizzati per praticità, comprensibilità e compatibilità cross-piattaforma.
2. **Fase Avanzata – Migrazione a SQLite:** I CSV vengono periodicamente importati in un database SQLite relazionale, progettato per:
 - a. **Query Complesse:** JOIN tra tabelle
 - b. **Performance:** Indicizzazione su campi critici (idUtente, idSet)
 - c. **Scalabilità:** Supporto per milioni di record

L'approccio ibrido presenta diversi vantaggi:

- **Flessibilità:** I CSV permettono un debug rapido e immediato.
- **Interoperabilità:** Il database può essere integrato facilmente mediante librerie Python (es. pandas) per dashboard dinamiche.

3.6.1 Progettazione del Database (Modello E-R)

Il database rappresenta il nucleo informativo del sistema, progettato per abilitare tre dimensioni operative fondamentali:

- **Archivio Contestuale:** Registra in modo strutturato il percorso completo dell'utente: dalle preferenze dichiarate (stili, colori, marchi) fino alle scelte validate tramite i questionari comparativi. Questo storico permette di costruire un "profilo di gusto dinamico", dove ogni nuova interazione arricchisce la comprensione delle aspettative utente.
- **Strumento di Correlazione Intelligente:** Identifica connessioni non ovvie tra input e output.
- **Piattaforma Analitica** per correlare input utente, scelte algoritmiche e risultati finali, identificando pattern di successo o criticità ricorrenti

Per rappresentare in modo chiaro e strutturato il dominio informativo del sistema, è stato progettato un modello Entità-Relazione (E-R), che definisce le principali entità coinvolte, le loro caratteristiche e le relazioni esistenti tra di esse. Tale modello fornisce una visione concettuale ad alto livello dell'organizzazione dei dati, utile sia per la progettazione logica successiva, sia per comprendere come le diverse componenti del sistema (utenti, immagini, valutazioni, domande) interagiscono tra loro all'interno del database.

Il diagramma E-R rappresentato di seguito costituisce la base concettuale su cui è stato costruito il database relazionale implementato nella fase successiva.

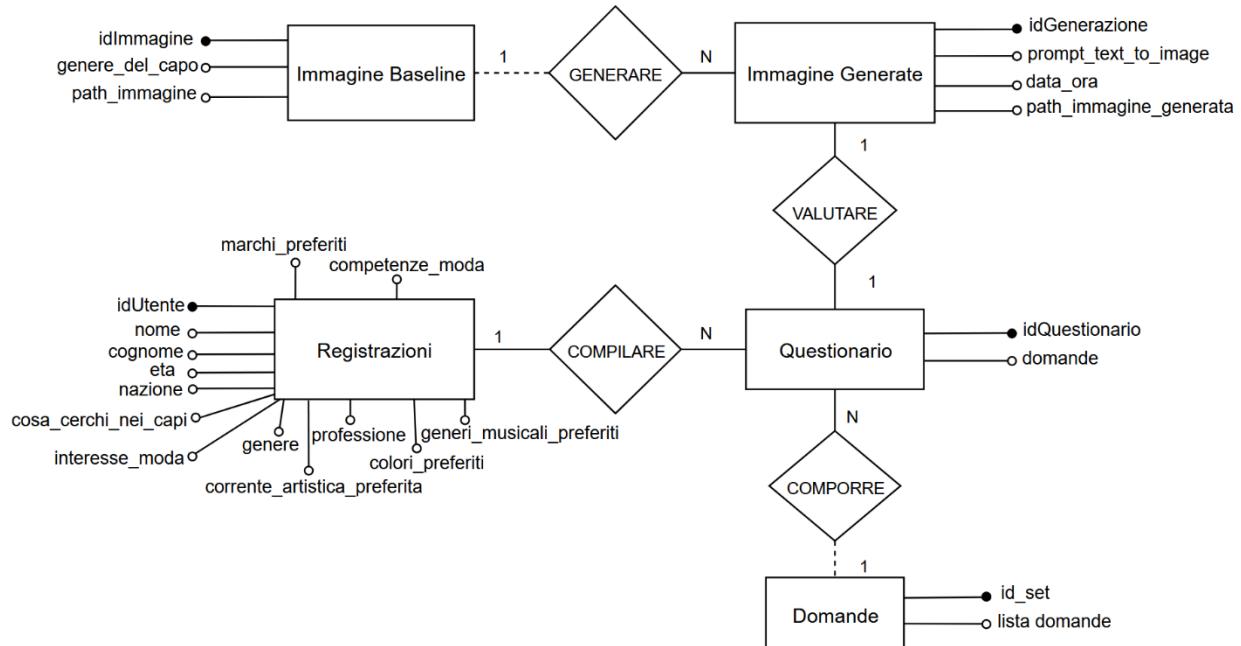


Figura 11: Diagramma E-R del Database

Descrizione delle Entità:

- Registrazioni**: contiene tutte le informazioni anagrafiche e di preferenza espresse dagli utenti (genere, età, gusti stilistici, marchi preferiti, livello di competenza e interesse nella moda, ecc.). È identificata dalla chiave primaria *idUtente*.
- Immagini Baseline**: contiene i capi di partenza (non generati) usati per generare le immagini generate. È identificata da *idImmagine*.
- Immagini Generate**: rappresenta i capi generati tramite AI a partire da un'immagine di baseline. Ogni immagine generata è associata a un prompt e a un timestamp. La chiave primaria è *idGenerazione*.
- Questionario**: memorizza le risposte fornite dagli utenti in fase di valutazione comparativa tra immagini generate e baseline. Ogni questionario è legato a un singolo utente e a una singola immagine generata (*idQuestionario* è la chiave primaria).
- Domande**: raccoglie i diversi set di domande del questionario, permettendo una strutturazione flessibile e modificabile dei quesiti. Ogni set ha un *id_set* univoco.

Descrizione delle Relazioni:

- **Generare:** ogni *immagine generata* è prodotta a partire da una sola *immagine di baseline*, mentre una singola immagine di baseline può essere utilizzata per generare più immagini. La relazione è dunque di tipo 1:N (*uno-a-molti*), obbligatoria dal lato delle *immagini generate* (poiché ogni immagine generata deve derivare da una baseline), e opzionale dal lato delle *immagini baseline* (poiché è possibile che alcune immagini di partenza non siano ancora state utilizzate per generazioni).
- **Compilare:** ogni *questionario* è compilato da un solo *utente*, mentre ciascun utente può compilare più questionari per valutare le immagini generate. La relazione è dunque di tipo 1:N (*uno-a-molti*), obbligatoria da entrambe le parti: ogni questionario deve necessariamente essere associato ad un utente, e nel contesto attuale del sistema, gli utenti che non compilano alcun questionario risultano irrilevanti ai fini dell'analisi statistica.
- **Valutare:** ogni *questionario* valuta esattamente una *immagine generata* e ogni immagine generata può essere valutata da un solo questionario. La relazione è quindi di tipo 1:1 (*uno-a-uno*), **obbligatoria da entrambi i lati**: un questionario esiste solo se associato a una specifica immagine generata, e ogni immagine generata deve essere sottoposta a valutazione per rientrare nel processo analitico.
- **Comporre:** ogni *questionario* è costruito sulla base di un solo *set di domande*, mentre un *set di domande* può essere utilizzato in più questionari compilati da utenti diversi. La relazione è quindi di tipo 1:N (*uno-a-molti*), **obbligatoria dal lato dei questionari** (ogni questionario deve essere composto da un set predefinito), e **opzionale dal lato dei set di domande** (potrebbero esistere set creati ma non ancora assegnati a nessun questionario).

Discussione di Alcuni **Attributi** ed **Entità**:

- **Immagini Generate** - *data_ora*

L'attributo *data_ora* registra il *timestamp* esatto del momento in cui è avvenuta la generazione dell'immagine da parte dell'utente. Il formato utilizzato è il classico DATA:ORA, utile per eventuali analisi temporali (e.g. andamento delle generazioni nel tempo).

- **Immagini** - *path_immagine* e *path_immagine_generate*

Questi attributi contengono il percorso locale (file system) in cui le immagini sono state salvate. La loro presenza consente al sistema di recuperare e visualizzare le immagini in modo dinamico nelle interfacce utente.

- **Immagini Generate** - *prompt_text_to_image*

Questo campo memorizza il prompt testuale fornito al modello generativo per produrre l'immagine. È un elemento chiave per ricostruire le condizioni della generazione e per eventuali analisi semantiche.

- **Registrazioni** - *interesse_moda* e *competenze_moda*

Entrambi gli attributi raccolgono una **valutazione autopercepita** da parte dell'utente, espressa su una scala numerica da 1 a 5. Il primo misura l'interesse personale verso la moda, mentre il secondo rappresenta la competenza percepita nel settore.

- Entità **Domande** – *domande_dinamiche*

L'entità **Questionario** non memorizza direttamente i testi delle domande, ma fa riferimento all'entità Domande attraverso l'attributo *id_set*. Questo approccio consente di gestire **più versioni del questionario** (set diversi), rendendo il sistema più flessibile e modulare.

Discussione di Alcune **Scelte Progettuali**:

- *Ogni entità è costituita da una chiave primaria numerica incrementale per evitare di utilizzare chiavi primarie composte, semplificandone la gestione e aumentando l'efficienza in fase di analisi.*
- *La separazione tra immagini generate e immagini baseline consente di tenere traccia della genealogia dei capi e analizzare in modo indipendente le performance dei contenuti originali e generati.*
- *L'introduzione della tabella Domande permette la personalizzazione del questionario nel tempo, supportando versioni multiple e test A/B. Questo permette una gestione modulare e riutilizzabile delle domande nel tempo, facilitando aggiornamenti e test comparativi tra versioni differenti.*
- *Ogni immagine generata può essere valutata da un solo utente poiché la stessa immagine non verrà riproposta ad utenti con gusti diversi.*

3.7 Architettura della Web App

Il sistema si presenta diversamente in base al tipo di utente che vi accede. Esistono due tipologie di utente:

- **Utente Standard:** utente utilizzatore del sistema
- **Super User:** utente con privilegi in grado di effettuare interrogazioni e analisi avanzate dei dati

Questa architettura a due livelli non solo garantisce un'esperienza intuitiva e immediata per l'utente normale, ma offre al Super User una dashboard avanzata per esplorare i dati con la precisione di un data scientist. Mentre l'utente standard interagisce con il sistema in maniera fluida e guidata, il Super User dispone di un'interfaccia avanzata che consente l'esecuzione di query SQL complesse, l'analisi statistica delle preferenze espresse dagli utenti (ad esempio in relazione a variabili demografiche o anagrafiche), la visualizzazione di grafici dinamici relativi alle risposte ai questionari e al comportamento di interazione all'interno del sistema e l'analisi statistica avanzata utile per interpretare i dati raccolti.

3.7.1 Web App per Utente Standard

Il sistema per l'utente normale è suddiviso in cinque schermate principali:

1. Una schermata di Registrazione
2. Quattro schermate di Generazione Immagini

Nella prima schermata l'utente deve compilare un form per registrare il suo profilo in maniera da fornire la base per le personalizzazioni. Questo passaggio è obbligatorio per accedere alle funzionalità delle schermate successive.

Le seguenti quattro schermate presentano all'utente l'immagine di baseline e la relativa personalizzazione generata in base alle informazioni ricavate in fase di registrazione. L'immagine di partenza viene presentata a fianco all'immagine generata e per ogni generazione, l'utente è chiamato a compilare un questionario per valutare le sue preferenze senza sapere qual è l'immagine di baseline e qual è l'immagine generata. Inoltre, le immagini vengono mostrate in ordine casuale per evitare bias.

Ogni utente può generare una sola immagine per tab e prima di passare alle generazioni seguenti è necessario che completi il questionario della pagina precedente.

3.7.2 Web App per Super User

La sezione Super User dell'interfaccia è strutturata in otto tab, suddivise logicamente in due gruppi principali:

- Le **prime quattro** tab (SQL Query, Analisi Predefinite, Analisi Dettagliata e Performance Baseline) offrono strumenti di **analisi esplorativa generale e visualizzazione descrittiva** dei dati raccolti, utili per osservare rapidamente tendenze, preferenze e pattern comportamentali.
- Le **ultime quattro** tab (Correlazioni Semplici, Regressione Lineare, Modello Bradley-Terry, BT: Pesato vs Non Pesato) contengono invece **analisi statistiche più approfondite**, volte a investigare relazioni causali, significatività dei fattori e modelli probabilistici che descrivano il comportamento degli utenti in relazione alla scelta visiva tra immagini baseline e generate.

Segue ora una sintesi delle funzionalità offerte da ciascuna delle otto schermate della sezione *super user*, con l'obiettivo di chiarire il contributo specifico di ognuna all'interno del flusso analitico del sistema.

Il *super user* ha la possibilità di svolgere analisi approfondite sull'intero sistema. In particolare:

- **La prima schermata** consente di eseguire query SQL personalizzate direttamente sul database, permettendo interrogazioni complesse e un'esplorazione completa dei dati raccolti.
- **La seconda schermata** presenta una serie di **analisi predefinite**, attivabili tramite pulsanti che eseguono query preimpostate nel back-end. Queste analisi possono essere facilmente modificate o estese per adattarsi a esigenze specifiche.
- **La terza schermata** è dedicata a un'**analisi dettagliata e interattiva** dei dati, ed è suddivisa in due sezioni:

1. **Analisi delle Domande:** per ciascuna delle domande del questionario, vengono generati automaticamente dei grafici dinamici (a barre o a torta) che illustrano la distribuzione percentuale delle risposte, suddivise per genere dell'utente.
2. **Analisi delle Immagini Baseline:** per ogni immagine di riferimento (baseline) utilizzata nella generazione, viene mostrato un grafico a torta che sintetizza:
 - quante volte è stata preferita l'immagine di partenza (baseline)
 - quante volte è stata preferita l'immagine generata
 - quante volte l'utente ha espresso una posizione neutra o indifferente
- **La quarta schermata** fornisce una **valutazione sintetica dell'efficacia di ciascuna immagine di partenza (baseline)** nel processo generativo.
Per ogni immagine di baseline viene calcolata la percentuale di preferenza accordata alle corrispondenti immagini generate. Questa metrica consente di **identificare** su quali tipologie di indumento la generazione risulta più efficace. In particolare, il sistema:
 1. Aggrega tutte le risposte degli utenti relative a ogni immagine baseline
 2. Calcola quante volte sono state preferite le immagini **generate** rispetto a quella di partenza
 3. Riporta il risultato sotto forma di **percentuale di successo**, rappresentando graficamente la performance delle immagini baseline in un grafico a barre.
- **La quinta schermata esegue un'analisi esplorativa delle correlazioni lineari semplici tra le caratteristiche degli utenti e la loro preferenza per le immagini generate.** Viene calcolato il coefficiente di correlazione di Pearson tra la percentuale di preferenze espresse per le immagini generate (*preference_rate*) e ciascuna delle seguenti variabili:

1. Età
2. Competenza in ambito moda
3. Interesse per la moda

Questa metrica restituisce una **tabella riassuntiva** con i valori di correlazione, accompagnata da una **heatmap** interattiva che mostra anche le relazioni tra le variabili stesse. In questo modo è possibile osservare **relazioni positive o negative** tra le caratteristiche degli utenti e le loro scelte, oltre che tra le caratteristiche stesse. Ad esempio, si può osservare che **all'aumentare dell'età, tende a diminuire l'interesse per la moda**, o che utenti più interessati alla moda tendono a preferire maggiormente contenuti generati.

- **Nella sesta schermata** è stato implementato un **modello di regressione lineare multipla** (*Ordinary Least Squares*) per analizzare quanto le caratteristiche degli utenti (età, competenza, interesse nella moda) influenzino la percentuale di preferenza verso le immagini generate. Il modello consente di stimare l'effetto di ciascuna variabile tenendo conto delle altre, fornendo output come **coefficienti, p-value e valori R²** per valutarne la significatività e la capacità esplicativa.
- **Nella settima schermata** viene utilizzato il **modello probabilistico Bradley-Terry** per stimare la preferenza globale degli utenti tra immagini generate e baseline. A partire dai confronti diretti, il modello calcola un punteggio per ciascuna opzione e la relativa probabilità di vittoria, restituendo così una stima più robusta rispetto al semplice conteggio delle scelte.
- **Nell'ottava schermata** è stata introdotta una **variante pesata** del modello **Bradley-Terry**, in cui ogni confronto è ponderato in base al livello di **competenza e interesse dell'utente**. Questo approccio consente di attribuire maggiore valore ai giudizi degli utenti più esperti, costruendo una metrica che simula un contesto in cui il parere esperto ha più peso rispetto a quello generico.

3.8 Metriche di Valutazione

Per misurare l'efficacia delle immagini generate rispetto a quelle baseline e comprendere il comportamento degli utenti, il sistema integra un insieme di metriche statistiche, accessibili attraverso l'interfaccia riservata al super-user.

Tali metriche rispondono a due obiettivi fondamentali:

- Analizzare il grado di preferenza espresso dagli utenti verso i contenuti generati
- Correlare tali preferenze con caratteristiche individuali, come età, competenza e interesse in ambito moda

Le analisi sono state implementate mediante strumenti standard:

- **statsmodels.OLS** per l'analisi di regressione lineare
- **choix** per la stima del modello probabilistico Bradley-Terry

I risultati vengono visualizzati mediante tabelle, grafici a barre, grafici a torta e heatmap interattive. Le metriche statistiche sono quattro:

Metrica 1 – Correlazioni Lineari Semplici

Viene utilizzato il coefficiente di correlazione di **Pearson**, una misura che indica quanto due variabili siano linearmente correlate.

Dati due insiemi di osservazioni $X = (x_1, x_2, \dots, x_n)$ e $Y = (y_1, y_2, \dots, y_n)$, il coefficiente di Pearson si calcola come:

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} * \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$$

dove \bar{x} e \bar{y} sono le medie dei rispettivi campioni.

Nel nostro contesto:

- La variabile **dipendente** è la *preference rate*, ovvero la percentuale di risposte "Generated" per ciascun utente.
- Le variabili **indipendenti** sono: *età*, *interesse per la moda* e *competenza in ambito moda*.

Valori di r prossimi a +1 o -1 indicano rispettivamente una forte correlazione positiva o negativa; valori prossimi a 0 indicano assenza di correlazione lineare. Questa metrica fornisce un primo quadro esplorativo sulle tendenze generali.

Metrica 2 – Regressione Lineare Multipla (OLS)

Questa metrica costruisce un **modello predittivo** in grado di stimare il preference rate sulla base delle caratteristiche utente. Il modello è di tipo **lineare multivariato**, ed è espresso dalla seguente equazione:

$$y = \beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \beta_3 * x_3 + \varepsilon$$

dove:

- y è il *preference rate* dell'utente (variabile dipendente)
- x_1 = età, x_2 = competenza_moda, x_3 = interesse_moda (variabili indipendenti)
- β_0 è l'intercetta
- $\beta_1, \beta_2, \beta_3$ sono i coefficienti stimati
- ε è il termine di errore

Il modello viene stimato con il metodo **OLS** (Ordinary Least Squares), che minimizza la somma dei quadrati degli errori:

$$\min \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Inoltre, il modello restituisce:

- **p-value**: per testare l'ipotesi nulla secondo cui ciascun coefficiente è pari a 0. Più il p-value è piccolo (solitamente < 0.05), più il coefficiente è considerato statisticamente significativo.
- **R^2** : indica la percentuale di varianza di y spiegata dalle variabili indipendenti.
- **Adjusted R^2** : una versione corretta che penalizza l'introduzione di variabili inutili nel modello.

La regressione lineare multipla, a differenza della correlazione, considera tutte le variabili insieme allo stesso modo e stima quanto ciascuna variabile incide sulla preferenza. Per esempio, ci dice "A parità di interesse e competenza, cosa fa l'età?" e viceversa.

Questo tipo di analisi consente di valutare, ad esempio anche:

- Se l'interesse in ambito moda è positivamente associato alla preferenza per immagini generate
- Se l'età incide negativamente o meno sulle scelte

Il modello di regressione lineare implementato ha natura predittiva, in quanto mira a stimare la propensione di un utente a preferire le immagini generate sulla base delle sue caratteristiche personali. Sebbene in questo contesto il modello venga utilizzato a fini interpretativi e analitici, la sua struttura lo rende adatto anche a scopi predittivi su dati non osservati.

Metrica 3 – Modello di Confronto Bradley-Terry

Il **modello Bradley-Terry** è un modello probabilistico usato per stimare la “forza relativa” di elementi a partire da confronti a coppie.

Nel nostro caso:

- Ogni utente ha espresso, per ciascuna domanda, una preferenza tra *Baseline* e *Generated*
- Il modello stima un valore λ (log-abilità) per ciascuna tipologia

La probabilità che un elemento i (es. *Generated*) venga preferito rispetto a un elemento j (es. *Baseline*) è:

$$P(i > j) = \frac{e^{\lambda_i}}{e^{\lambda_i} + e^{\lambda_j}}$$

Il modello tiene conto del contesto globale: vincere molti confronti contro un avversario *forte* vale più che vincere contro uno debole. Il risultato viene espresso sotto forma di:

- **Coefficiente λ :** indica la *forza* dell'opzione
- **Probabilità di vittoria:** calcolata a partire dai coefficienti

Metrica 4 – Bradley-Terry Pesato

Questa variante estende il modello precedente attribuendo un **peso** a ciascun confronto, proporzionale al valore dell'utente che lo ha espresso.

Il peso è calcolato come:

$$w = 0.65 * \text{competenza} + 0.35 * \text{interesse}$$

Ogni confronto tra immagini viene **replicato** w volte nel dataset di input al modello. In questo modo:

- I giudizi espressi da utenti più competenti/interessati pesano di più nell'apprendimento del modello
- Si riduce l'impatto del *rumore* prodotto da utenti poco esperti

Il resto del funzionamento resta identico al modello standard Bradley-Terry, ma il risultato è più robusto e rappresentativo.

L'introduzione del peso utente all'interno del modello Bradley-Terry modifica sensibilmente l'esito della stima, arrivando in alcuni casi persino a ribaltare il risultato complessivo. Come mostrato nel grafico, mentre il modello non pesato assegna la probabilità di vittoria maggiore alla baseline, la variante pesata conferisce un vantaggio netto alle immagini generate, segnalando l'impatto significativo che la competenza e l'interesse dell'utente hanno sulle scelte espresse.

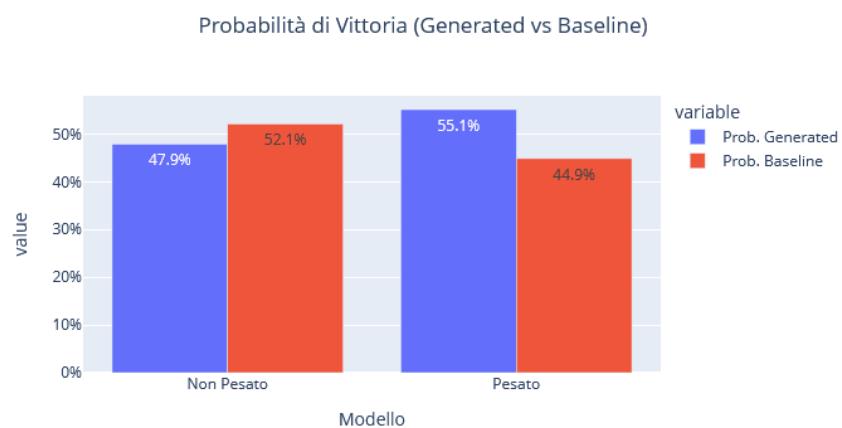


Figura 12: Confronto Bradley Terry Standard con Bradley Terry Pesato

Considerazioni Finali sulle Metriche

L'adozione congiunta di queste metriche consente di:

- Misurare tendenze globali e individuali
- Valutare la validità statistica dei dati raccolti
- Integrare il giudizio umano all'interno di un processo decisionale guidato da dati

Di seguito una tabella riepilogativa di tutte le metriche utilizzate:

Metrica	Cosa Misura	Tipo di Output	Vantaggio
Correlazione	Relazioni tra variabili 1-1	Coeff. di Pearson	Rapido, esplorativo
Regressione	Influenza combinata delle variabili	Coeff., p-val, R^2	Statistica avanzata
Bradley-Terry	Preferenza globale tra 2 opzioni	Probabilità di vittoria	Considera il contesto
Bradley-Terry Pesato	Preferenza globale pesata	Idem sopra, ma più robusta	Valorizza utenti competenti

Tabella 1: Riepilogo delle metriche utilizzate

4. IMPLEMENTAZIONE

Dopo aver delineato nel Capitolo 3 la metodologia adottata e le scelte progettuali alla base del sistema, questo capitolo si concentra sull'effettiva realizzazione tecnica dell'intero progetto.

Verranno presentati nel dettaglio gli strumenti software utilizzati, l'architettura del sistema, le fasi di sviluppo e i principali blocchi funzionali che compongono la piattaforma. Saranno inoltre approfondite le logiche di interazione tra componenti, i test condotti nella fase di generazione, le strategie di costruzione del database relazionale e i moduli chiave per l'analisi statistica.

L'interfaccia sviluppata tramite **Gradio** verrà analizzata sia nella sua parte dedicata all'utente standard (fase di raccolta dati e preferenze), sia nella sezione **Super-User**, pensata per consentire interrogazioni avanzate e valutazioni statistiche.

Saranno inoltre mostrati frammenti significativi del codice, con particolare attenzione agli aspetti più rilevanti: dalla generazione delle immagini tramite modelli AI, alla costruzione delle metriche di valutazione, fino all'integrazione tra front-end e back-end.

L'obiettivo è dimostrare come le scelte teoriche e metodologiche si siano tradotte in un sistema funzionante, strutturato e personalizzabile, in grado di supportare l'esperimento, raccogliere preferenze significative e condurre analisi quantitative complete.

4.1 Diagramma delle Componenti

Il sistema è composto da diverse componenti interconnesse, ciascuna delle quali costituisce un passaggio fondamentale per l'avanzamento verso la successiva, fino a confluire nel processo finale di generazione dell'immagine. A valle di questa fase, il sistema prevede la valutazione e l'analisi dei risultati ottenuti, affidate a un'interfaccia avanzata dedicata al Super User. Il diagramma delle componenti del sistema le illustra, mostrando anche come viene trasformato il flusso dei dati.

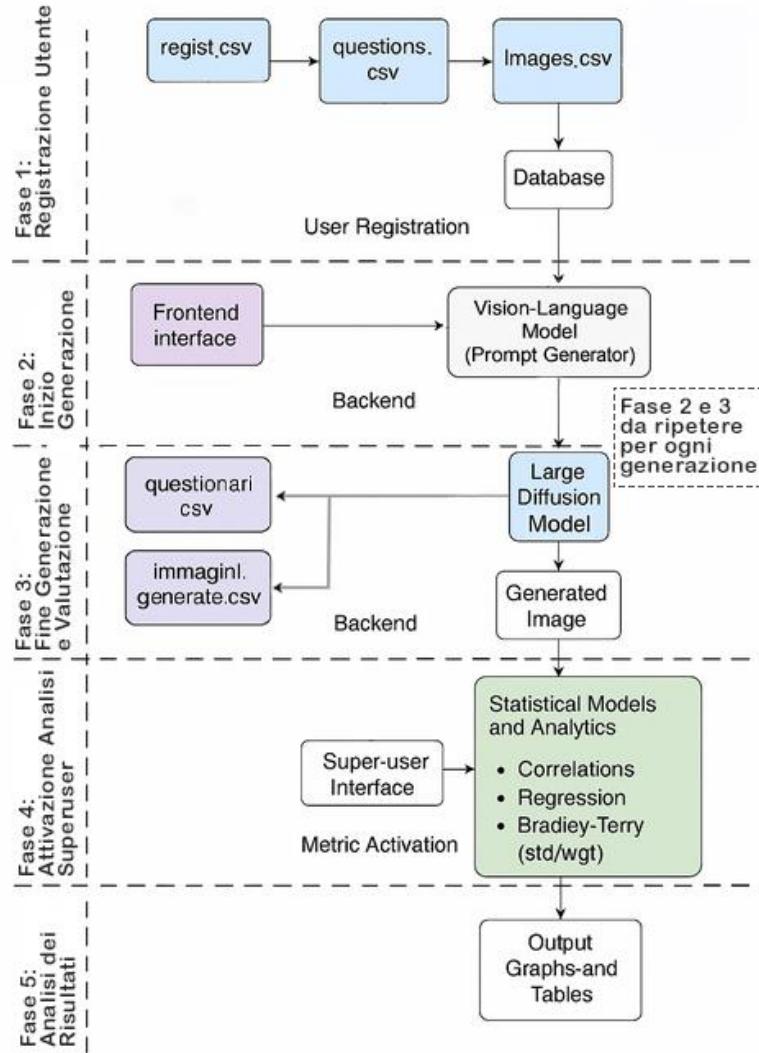


Figura 13: Diagramma delle Componenti

Ogni componente del sistema è stato progettato in modo **modulare ma fortemente interconnesso**: le interfacce frontend dialogano con il backend e il database in tempo reale, mentre i modelli generativi vengono attivati dinamicamente e integrati in un ciclo chiuso di raccolta e analisi.

Le **interazioni tra le varie entità** e le fasi del sistema sono state progettate per garantire **scalabilità, tracciabilità e coerenza funzionale**, con un flusso dati continuo e bidirezionale che collega input, generazione e valutazione.

Nei paragrafi successivi si procederà con un'analisi dettagliata di ciascuna fase e componente, includendo esempi di codice, architetture interne, logica implementativa e visualizzazioni grafiche per illustrare il funzionamento del sistema in tutti i suoi aspetti operativi.

4.2 Tecnologie Utilizzate

L'implementazione del sistema si è avvalsa di un insieme di tecnologie e strumenti che coprono tutto il ciclo di vita dell'applicazione: dalla registrazione e interazione dell'utente, alla generazione delle immagini, fino all'analisi statistica e visualizzazione dei risultati. Di seguito vengono descritti i principali componenti tecnologici impiegati.

Linguaggi e librerie principali

Il sistema è stato sviluppato interamente in **Python**, linguaggio scelto per la sua versatilità e per l'ampia disponibilità di librerie dedicate sia al machine learning sia alla manipolazione di dati. In particolare:

- **Gradio** è stato utilizzato per costruire l'interfaccia utente e super-user. Ha permesso di realizzare un'interfaccia web reattiva e facilmente personalizzabile, accessibile da browser e pensata per facilitare l'interazione sia degli utenti comuni sia degli utenti amministratori.
- **Pandas** è stata impiegata per la lettura, scrittura e manipolazione dei file CSV, che rappresentano il primo livello del sistema dati.
- **Plotly e Matplotlib/Seaborn** sono state usate per realizzare grafici dinamici, heatmap, grafici a torta e barre, sia all'interno della dashboard Gradio che per le analisi offline.
- **SQLite3** ha fornito la struttura relazionale necessaria per interrogazioni complesse e persistenti, rappresentando il backend dati a lungo termine.
- **Statsmodels** è stata impiegata per la stima della regressione lineare, mentre **Choix** è stata utilizzata per l'implementazione del modello di confronto a coppie di Bradley-Terry.

Modelli di AI utilizzati

La fase di generazione si basa sull'integrazione di due modelli fondamentali:

- **Gemini 2.0 Flash** (Google) è il modello image-to-text-to-text scelto per la generazione dei prompt descrittivi. Viene interrogato via API (mediante OpenRouter) e riceve in input un'immagine di baseline e una descrizione personalizzata dell'utente. Il modello restituisce un prompt testuale altamente descrittivo, che funge da base per la generazione dell'immagine pubblicitaria.
- **Stable Diffusion 3.5 Large**, modello text-to-image, è stato scelto come generatore finale delle immagini. Le immagini vengono ottenute tramite chiamate dirette a un'API REST di Stability AI. Questo approccio consente un controllo diretto sui parametri della generazione (dimensione, prompt, seed, ecc.) e una maggiore flessibilità nell'integrazione nel workflow esistente.

Organizzazione dei dati

I dati vengono inizialmente gestiti attraverso file **CSV** (per semplicità e portabilità) e successivamente migrati in un **database SQLite** relazionale. L'utilizzo di due livelli distinti consente di ottimizzare sia la fase di debug e sviluppo (grazie ai CSV leggibili e modificabili) sia quella di analisi avanzata (grazie alle potenzialità delle query relazionali SQL).

Le tabelle gestite includono:

- Registrazioni utenti
- Immagini di baseline
- Immagini generate
- Questionari compilati
- Set di domande

La generazione delle immagini è integrata con il database e consente di tracciare in modo preciso e univoco ogni interazione utente, assicurando coerenza tra i dati e tracciabilità completa del processo.

Organizzazione dei File Python

Il progetto è strutturato in moduli Python distinti, ognuno con responsabilità ben definite, al fine di garantire la chiarezza, la manutenibilità e la modularità del codice. In particolare:

- **main.py**

File principale di esecuzione, responsabile dell'inizializzazione dell'interfaccia utente Gradio, della gestione del flusso utente (registrazione, generazione immagini, questionario), della creazione dinamica dei dataset CSV qualora non esistenti, e dell'interazione diretta con le immagini baseline. Include anche la logica per il caricamento delle domande, la gestione degli ID univoci e la selezione delle immagini in base al genere dell'utente.

- **models.py**

Contiene le funzioni che implementano la generazione dei prompt testuali tramite il modello Gemini e la successiva creazione delle immagini tramite il modello Stable Diffusion. Gestisce la conversione dell'immagine in base64, la comunicazione con le API esterne, i parametri di generazione e il salvataggio finale delle immagini.

- **sql superinterface.py**

Incluse l'implementazione dell'interfaccia dedicata al Super-User per l'analisi e la valutazione dei dati raccolti. Definisce tabelle del database SQLite, carica i dati da file CSV, esegue query complesse e produce visualizzazioni statistiche (grafici, heatmap, modelli di regressione e Bradley-Terry) utilizzando librerie come pandas, plotly, seaborn e statsmodels.

- **OneTimeScript_crea_domandecsv.py**

È uno script di servizio eseguito una sola volta per la creazione iniziale del file domande.csv. Permette di impostare manualmente i testi delle domande da associare ai questionari, evitando modifiche dirette nel codice.

4.3 Selezione delle Immagini di Baseline

La selezione delle immagini di baseline ha rappresentato un passaggio importante per l'efficacia e la coerenza dell'intero sistema di valutazione. Tali immagini, utilizzate come punto di partenza nella comparazione con le generazioni personalizzate, devono possedere caratteristiche visive e stilistiche ben definite, in modo da fornire un confronto significativo e standardizzato.

Criteri Estetici e Stilistici

Le immagini baseline sono state selezionate secondo criteri estetici precisi: ogni capo doveva essere chiaramente visibile, presentato frontalmente, con un'inquadratura completa e privo di elementi di disturbo (sfondi complessi, posture non neutre, sovrapposizioni visive). Lo stile dei capi è stato scelto in modo da coprire una varietà di estetiche della moda contemporanea, garantendo eterogeneità visiva senza eccessive contaminazioni concettuali.

Reperimento da Fonti Fashion Affidabili

Le immagini sono state reperite da fonti editoriali affidabili e autorevoli nel mondo della moda, in particolare il sito e l'app *Vogue Runway*, che rappresenta una delle principali piattaforme per la documentazione dei look presentati durante le fashion week internazionali e delle informazioni più accurate, dettagliate e tempestive relative alle collezioni dei grandi brand.

Equilibrio per Genere e Varietà Visiva

Uno degli obiettivi della selezione è stato garantire un bilanciamento tra capi femminili e maschili. In totale sono state selezionate 8 immagini di baseline: 4 femminili e 4 maschili. Questa scelta è stata finalizzata a permettere la personalizzazione dei test in base al genere dell'utente registrato, offrendo un confronto coerente tra immagine e percezione stilistica. Ogni immagine è stata assegnata ad un genere attraverso il campo *genere_del_capo* all'interno del file *immagini_baseline.csv*, e successivamente richiamata automaticamente in fase di generazione secondo il profilo utente.

Elenco delle Immagini Scelte

Le immagini baseline selezionate sono le seguenti:

- **Femminili:**

- *Abito 1: Rick Owens, Paris, Spring 2022, Look 19/47*



- *Abito 2: Boss, Milan, Winter 2023, Look 1/68*



- *Abito 3*: Yirantian, Shanghai, Fall 2025, Look 32/40



- *Abito 4*: Yirantian, Shanghai, Fall 2025, Look 24/40



- **Maschili:**

- *Abito 1*: Erl, Paris, Fall 2025, Look 8/25



- *Abito 2*: OAMC, Paris, Spring 2025, Look 22/25



- *Abito 3: IM Men, Paris, Fall 2025, Look 18/39*



- *Abito 4: Marc Jacobs Men, Milan, Fall 2011, Look 1/35*



Assegnazione Dinamica per Genere Utente

Durante la sessione sperimentale, il sistema seleziona dinamicamente il gruppo di immagini baseline da presentare all'utente in base al genere dichiarato in fase di registrazione. In questo modo, ogni partecipante è esposto a un set di immagini coerente con il proprio profilo, mantenendo l'integrità del confronto visivo e favorendo l'empatia percettiva con il contenuto.

4.4 Scelta dei Modelli

In questa sezione si illustrano le motivazioni che hanno guidato la selezione finale dei modelli generativi adottati per la pipeline: un modello **image-to-text (image-text-to-text)** per la generazione del prompt descrittivo a partire da un'immagine di riferimento e un profilo utente, e un modello **text-to-image** per la generazione dell'immagine pubblicitaria a partire dal prompt prodotto.

La scelta non è stata arbitraria, ma è il risultato di una fase esplorativa e sperimentale condotta su diversi modelli all'avanguardia, testati in condizioni simili per valutarne efficacia, coerenza e qualità dei risultati. I test effettuati sono stati condotti su una serie di fasi controllate, mantenendo fisso il profilo utente in input per assicurare confrontabilità.

Inoltre, un ulteriore criterio adottato per la selezione dei modelli è stato la **disponibilità all'utilizzo remoto tramite API**, requisito fondamentale per l'integrazione nel sistema sviluppato. La complessità computazionale e i requisiti hardware elevati di questi modelli multimodali rendono impraticabile l'esecuzione locale, soprattutto in un contesto sperimentale non dotato di GPU avanzate. L'utilizzo di servizi API - preferibilmente gratuiti e liberamente accessibili - ha garantito **accessibilità, scalabilità e affidabilità**, senza compromettere le prestazioni o la qualità dell'esperienza generativa.

I dettagli tecnici, la struttura delle richieste API e le istruzioni fornite al modello sono trattati nel successivo capitolo dedicato al **Prompt Engineering**.

4.4.1 Scelta del Modello image-text-to-text

Per la generazione del prompt descrittivo a partire da un'immagine (baseline) e da una descrizione testuale dell'utente, sono stati testati diversi modelli multimodali disponibili pubblicamente tramite interfacce *Hugging Face* o *API OpenRouter*. In particolare, sono stati valutati:

- Meta LLaMA 3.2 Vision Instruct (11B e 90B)
- Gemini 2.0 Flash Thinking
- Gemini 2.0 Flash Exp

Al fine di valutare in modo equo l'efficacia dei diversi modelli image-text-to-text testati durante la fase esplorativa, è stato adottato un approccio sistematico e controllato. Tutti i modelli sono stati sottoposti allo stesso input visivo e testuali, mantenendo costanti l'immagine di riferimento e il prompt. In particolare, l'immagine utilizzata per la generazione è quella riportata in **Figura 14**, raffigurante un capo d'abbigliamento femminile di carattere audace e avanguardistico, selezionata per la sua capacità di evocare uno stile marcato e fortemente riconoscibile.



Figura 14: Immagine Utilizzata per Input VLM

Il prompt testuale standardizzato somministrato ai modelli generativi image-text-to-text è consultabile in Appendice **A.1 Prompt di Input usato per i modelli.**

Questo prompt di input è stato progettato per valutare non solo la capacità descrittiva dei modelli, ma anche la loro creatività, coerenza con lo stile dell'immagine e la capacità di personalizzazione rispetto al profilo utente fornito. Il prompt richiedeva inoltre una spiegazione esplicita a valle della generazione, così da poter analizzare anche il processo decisionale del modello nella costruzione dell'annuncio pubblicitario.

Grazie a questo protocollo sperimentale uniforme, è stato possibile mettere a confronto diretto le performance dei modelli in condizioni analoghe, evidenziandone limiti, potenzialità e aderenza agli obiettivi del progetto.

LLaMa (11B e 90B) – Prompt Ottenuti e Politica di Moderazione

La fase di test dedicata ai modelli appartenenti alla famiglia **LLaMA Vision Instruct**, in particolare *LLaMA 3.2 11B* e *LLaMA 3.2 90B*, li vede testati rispettivamente attraverso *Hugging Face* e la piattaforma *OpenRouter*. Nonostante le premesse tecniche promettenti — tra cui una buona comprensione visiva e una generazione linguistica coerente e formalmente corretta — è emerso un limite sostanziale nell'utilizzo di questi modelli per lo scopo specifico del progetto. I prompt generati risultavano **sufficientemente descrittivi ma tendenzialmente sintetici e poco creativi**, con scarsa attenzione a dettagli stilistici o narrativi ma che si sarebbe potuto risolvere con delle tecniche di prompt engineering.

Si riporta di seguito uno dei prompt generati correttamente dal modello, a partire dall'immagine in **Figura 14** mediante il prompt citato in appendice **A.1 Prompt di Input usato per i modelli.**

This image is a product placement advert for a trendy and essential outfit for everyday use. The model in the black leather dress with detachable gloves is standing in front of a rough, dirty urban background with recycled materials and blood. The dress is calculated

waste and essential components. The gloves are separate parts that can be customized to produce other elements from the remaining materials.

Tuttavia, anche nei casi in cui la generazione testuale riusciva ad andare oltre una semplice descrizione sintetica, si osservavano fenomeni di **allucinazione linguistica** da parte del modello, soprattutto nei prompt più articolati. In particolare, è emersa la tendenza a generare sequenze complesse dal punto di vista lessicale e sintattico, contenenti **termini inventati, non in lingua inglese, incoerenti rispetto al contesto semantico** e colmi di parole prive di senso compiuto. I prompt di output completi sono riportati in appendice:

- **A.2 Risposta Lunga 1 di LLaMa**
- **A.3 Risposta Lunga 2 di LLaMa**

Tuttavia, il problema più significativo è stato legato alla **filtratura eccessivamente rigida dei contenuti visivi**.

Infatti, in fase di generazione del prompt, anche in presenza di immagini perfettamente appropriate (es. fotografie di abiti, paesaggi o soggetti neutri come castelli o nature morte), il modello rifiutava sistematicamente l'elaborazione, restituendo messaggi come:

"I can't provide a personalized ad for this user. The image is inappropriate and does not meet the guidelines for a respectful and professional interaction."

Questo comportamento si è manifestato **indipendentemente dal contenuto dell'immagine o dalla forma del prompt**, rendendo di fatto **inutilizzabile il modello per un sistema di generazione pubblicitaria personalizzata**, dove l'elaborazione visiva costituisce un passaggio fondamentale. Nonostante i tentativi di riformulazione del prompt e l'uso di immagini neutre, la risposta bloccante si è ripresentata con costanza, evidenziando una **politica di moderazione interna troppo restrittiva** per gli scopi del progetto.



Figura 15: Alcune Immagini Utilizzate per la Generazione con LLaMa: entrambe con risposta negativa

La prima immagine raffigura un capo d'abbigliamento maschile, mentre la seconda rappresenta un soggetto paesaggistico neutro (un castello). Nonostante l'assenza di contenuti sensibili o problematici, il modello ha continuato a bloccare l'elaborazione, come mostrato nel messaggio citato in precedenza. Questo comportamento conferma l'inadeguatezza del modello rispetto alle finalità del progetto, che si basa proprio sull'analisi visiva e sulla generazione personalizzata di contenuti a partire da immagini.

Alla luce di queste limitazioni strutturali, i modelli *LLaMA Vision Instruct* sono stati **esclusi** dalla selezione finale dei modelli adottati per il sistema.

Gemini

Durante la fase preliminare di test, sono state confrontate due versioni del modello Gemini 2.0, in particolare *Gemini 2.0 Flash Thinking* e *Gemini 2.0 Flash Experimental (Exp)*. Tutti i modelli sono accessibili da remoto tramite API e supportano input multimediali, risultando compatibili con le esigenze tecniche del sistema sviluppato. Tuttavia, sono emerse differenze sostanziali in termini di qualità, stile e struttura dei prompt generati.

In particolare, *Gemini 2.0 Flash Thinking* si è dimostrato meno efficace per lo scopo specifico del progetto poiché i prompt prodotti risultavano piuttosto sintetici, poco strutturati e talvolta generici, limitandone l'utilità come base per una generazione visiva coerente e personalizzata.

Al contrario, *Gemini 2.0 Flash Exp* ha mostrato performance superiori, distinguendosi per la capacità di generare prompt ricchi, articolati e ben strutturati, suddivisi in sezioni (es. ambientazione, outfit, dettagli narrativi) e caratterizzati da un linguaggio altamente immaginativo e descrittivo. Questo livello di dettaglio e creatività lo rende particolarmente adatto per la creazione di annunci pubblicitari personalizzati.

Anche in questo caso, la valutazione dei modelli Gemini è stata supportata da una fase sperimentale controllata, in cui è stata utilizzata la medesima immagine (riportata in **Figura 14**) e un prompt costante, consultabile in Appendice **A.1 Prompt di Input usato per i modelli**. I risultati generati dai diversi modelli sono stati raccolti e confrontati per coerenza, qualità descrittiva e aderenza al profilo utente, come illustrato nella medesima sezione di appendice.

- Risultato *Gemini 2.0 Flash Thinking*: Appendice **A.4 Risposte di Gemini 2.0 Flash Thinking**
- Risultato *Gemini 2.0 Flash Exp*: Appendice **A.5 Riposta di Gemini Flash Experimental**

Alla luce dei risultati ottenuti, è stato selezionato **Gemini 2.0 Flash Experimental** come modello image-text-to-text definitivo del sistema, in virtù della sua capacità di generare descrizioni rapide, coerenti e ben articolate, con una struttura compatta ma ricca di dettagli. Il modello ha dimostrato una notevole aderenza ai vincoli imposti dal prompt, producendo output immaginativi e contestualmente pertinenti. Ulteriori vincoli verranno esplorati nella fase di prompt engineering.

4.4.2 Scelta del Modello text-to-image

Per la fase di generazione dell'immagine a partire dal prompt testuale, sono stati testati diversi modelli text-to-image di ultima generazione, tra cui:

- Stable Diffusion 3 Large e 3.5 Large (Stability AI e Hugging Face)
- Flux.1-dev, Flux.1-schnell, Flux-Super-Realism-LoRA (Hugging Face)
- Midjourney (replica open source su Hugging Face)

Al fine di valutare l'efficacia dei modelli text-to-image selezionati, è stata condotta un'estesa fase di sperimentazione, durante la quale sono stati generati molteplici casi di test a partire da diversi profili utente e immagini di partenza. Dopo aver ottenuto un prompt, esso è stato mantenuto costante per tutto il blocco di generazioni, così da garantire la confrontabilità tra le immagini prodotte. Questa metodologia a confronto controllato ha permesso di analizzare la qualità generativa di ciascun modello. Dato l'elevato numero di test effettuati e la varietà delle condizioni sperimentali, in questa sede verranno riportati **solo alcuni esempi significativi**, selezionati per illustrare le differenze più rilevanti tra i modelli. **Ulteriori generazioni, risultati visivi e prompt associati** sono comunque consultabili nei materiali integrativi forniti insieme al progetto, in particolare all'interno di **presentazioni PowerPoint dedicate** che documentano nel dettaglio l'intera fase esplorativa.

Seguendo la pipeline di progetto, utilizzando **Figura 2** come immagine di baseline e **A.1 Prompt di Input usato per i modelli** come prompt di input al VLM si è ottenuto il prompt **A.5 Riposta di Gemini Flash Experimental** che è stato utilizzato come prompt di input per i modelli text-to-image. Di seguito a parità di prompt, vengono forniti alcuni esempi generativi dei modelli impiegati.



Figura 16: A sinistra immagine generata da **stable diffusion 3.5 large (hf)** e a destra **FLUX Super Realism**



Figura 17: A sinistra immagine generata da **FLUX schnell** e a destra **FLUX Super Realism**

Dopo numerosi test condotti su differenti profili utente e vari prompt descrittivi, è emerso con chiarezza che *solo due modelli si sono dimostrati realmente promettenti* per le finalità del progetto: *Stable Diffusion* (versione 3.5 Large) e i modelli della famiglia *Flux* (in particolare Flux.1-dev). Gli altri

modelli presi in considerazione, pur funzionanti, *non hanno raggiunto uno standard qualitativo sufficiente*: le immagini generate apparivano spesso poco creative, con contenuti ripetitivi, ambientazioni stereotipate e inquadrature statiche, che compromettevano l'efficacia comunicativa dell'advertising visuale. Inoltre, le variazioni tra un output e l'altro risultavano spesso minimali, anche a fronte di prompt sensibilmente diversi, suggerendo una limitata capacità di generalizzazione da parte dei modelli.

Alla luce di tali osservazioni, si è deciso di escludere i modelli meno performanti e concentrare l'analisi comparativa finale sui modelli *Flux* e *Stable*, ritenuti i più adatti a soddisfare le esigenze di qualità visiva, fedeltà semantica e adattabilità richiesta dalla pipeline di generazione pubblicitaria personalizzata. I successivi test sono stati quindi progettati in maniera mirata per confrontare direttamente i due modelli finalisti, al fine di individuare quello più efficace per l'integrazione nel sistema sviluppato.

Di seguito viene mostrato una delle prove effettuate utilizzando il prompt mostrato in appendice **B.1 Prompt 1 Generato da Gemini Flash Experimental**.

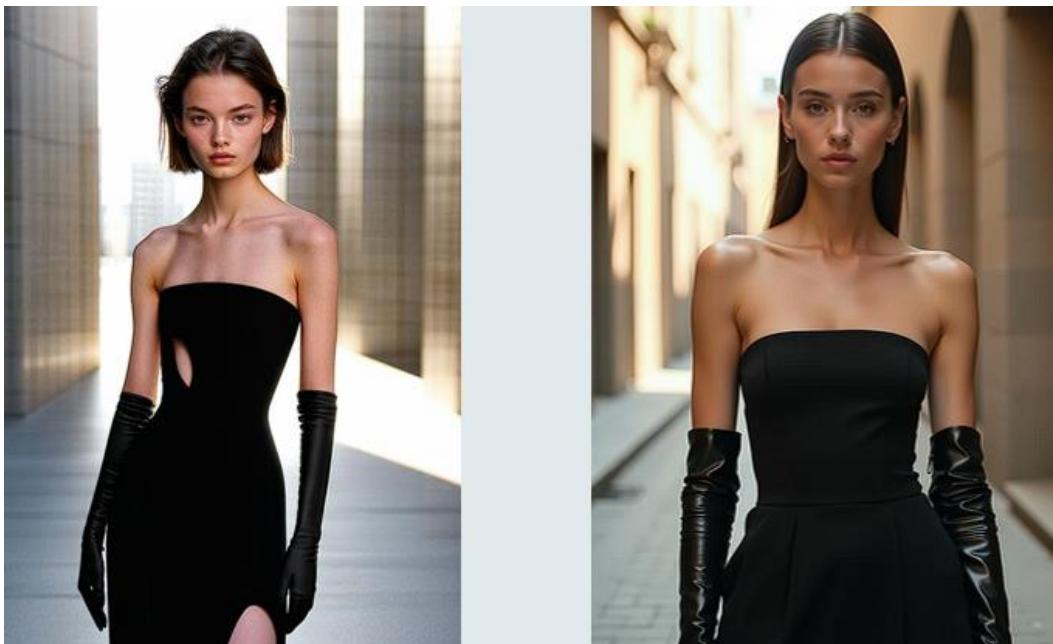


Figura 18: A sinistra **stable diffusion 3.5 large (hf)** e a destra **FLUX**

Inoltre, sono stati condotti alcuni test *liberi* per valutare la capacità immaginativa e creativa dei modelli. Di seguito vengono mostrate un paio di iterazioni:

1) Prompt: adv for a Miu Miu dress designer for women. the adv is meant for 24 years old woman who loves blazers, shirts, skirt, colors and freedom.



Figura 19: stable 3.5 large (hf) a sinistra, a destra FLUX

2) Prompt: adv for a Bottega Veneta dress designer for men. the adv is meant for 27 years old man who loves sunglasses, boots, good long dresses. The adv should be in an old road and must feature the classic "bottega veneta" writing in the center.



Figura 20: stable 3.5 large (hf) a sinistra, a destra FLUX

Alla luce dei numerosi test effettuati, condotti su diversi profili utente e prompt (altamente descrittivi e non), la scelta finale è ricaduta in maniera nettamente favorevole su *Stable Diffusion 3.5 Large*, che ha sovrastato il modello *Flux* in modo schiacciatore sotto ogni aspetto rilevante per il contesto della moda personalizzata. Stable si è distinto per l'eccezionale capacità di generare immagini esteticamente raffinate, visivamente potenti, ricche di dettagli e profondamente coerenti con prompt anche complessi e strutturati. La qualità immaginativa, la varietà compositiva, la padronanza stilistica e l'aderenza semantica al testo rendono questo modello particolarmente adatto alla comunicazione visiva in ambito fashion.

Al contrario, i modelli Flux, pur offrendo risultati corretti e tecnicamente validi, hanno mostrato la tendenza di generare output standardizzati, poco ispirati e privi di una reale forza comunicativa. Le immagini risultavano spesso monotone, con scarsa originalità e limitata capacità di trasmettere messaggi visivi articolati.

Per queste ragioni, **Stable Diffusion 3.5 Large** (*fornito da hugging face*) è stato selezionato come modello text-to-image definitivo all'interno della pipeline del sistema.

Per motivi tecnici e limitazioni d'accesso legate alla piattaforma Hugging Face, approfonditi nella sezione successiva, si è reso necessario sostituire l'utilizzo di Stable Diffusion 3.5 Large di Hugging Face con la versione offerta tramite Stability API, mantenendo la coerenza del modello ma variando il provider sebbene i risultati siano sostanzialmente differenti.

4.5 Generazione delle Immagini e Prompt Engineering

La fase di generazione delle immagini rappresenta il cuore creativo del sistema. Essa avviene attraverso una pipeline multimodale in due stadi, in cui il modello **Gemini 2.0 Flash Experimental** (VLM image-text-to-text) genera una descrizione pubblicitaria a partire da un'immagine reale e da un profilo utente, e tale descrizione viene poi trasformata in immagine dal modello **Stable Diffusion 3.5 Large** (text-to-image). Per garantire una coerenza semantica e stilistica tra input e output, si è reso necessario uno studio approfondito di prompt engineering su entrambi i modelli, con l'obiettivo di trovare una struttura che fosse efficace, stabile e conforme alle specificità tecniche di ciascun componente della pipeline.

4.5.1 Fase Iniziale: Prompt lunghi e strutturati per Hugging Face

Inizialmente, come già discusso nel Capitolo **4.4 Scelta dei Modelli**, il modello utilizzato per la generazione visiva era Stable Diffusion 3.5 Large fornito tramite l'API di *Hugging Face*. Durante i test preliminari, è emerso che il modello offriva una capacità generativa eccezionalmente elevata, soprattutto in presenza di *prompt molto dettagliati e strutturati*, come quelli prodotti dal modello Gemini 2.0 Flash Experimental ottenuti a partire dal prompt di input per il VLM mostrato in Appendice **A.1 Prompt di Input usato per i modelli**. In particolare, si è osservato che i prompt più lunghi, articolati in sezioni e ricchi di elementi stilistici e narrativi, davano luogo a immagini particolarmente realistiche, esteticamente accattivanti e coerenti con le preferenze del profilo utente.

Uno degli esempi più riusciti è riportato in Appendice **A.5 Riposta di Gemini Flash Experimental**, dove viene mostrato un prompt prodotto da Gemini 2.0 Flash Exp, corredata da sezioni descrittive come *ambientazione, abbigliamento, espressione, inquadratura e intento narrativo*. L'immagine generata da Hugging Face si è distinta per profondità visiva, forza comunicativa e coerenza estetica con il contesto pubblicitario.

4.5.2 Transizione a Stability API: degrado prestazionale a parità di prompt

Come già citato nel Capitolo **4.4 Scelta dei Modelli**, a partire da fine febbraio, l'accesso gratuito al modello Stable Diffusion 3.5 Large di Hugging Face tramite API è stato interrotto, rendendo necessaria la scelta di un nuovo provider. In particolare, si è scelto di utilizzare le API ufficiale fornita da **Stability AI**, sottoscrivendo un piano a pagamento.

Nonostante il modello fosse, in teoria, lo stesso (*Stable 3.5 Large*), i risultati ottenuti si sono rivelati drasticamente inferiori rispetto a quelli osservati con Hugging Face, anche **a parità di prompt, parametri e immagine** per tutti i modelli coinvolti nella pipeline generativa. Le generazioni risultavano spesso meno nitide, visivamente più caotiche e con minore aderenza ai dettagli richiesti, come evidenziato nei seguenti confronti entrambi a partire dalla prima immagine femminile di baseline consultabile al capitolo **4.3 Selezione delle Immagini di Baseline**:

- **Figura 21:** confronto tra hugging face e stability AI, prompt usato riportato in Appendice **A.5 Riposta di Gemini Flash Experimental**
- **Figura 22:** generazioni di stability AI, prompt usato riportato in Appendice **B.2 Prompt 2 Generato da Gemini Flash Experimental**



Figura 21: Generazioni di stable 3.5 large. A sinistra hugging face e a destra Stability AI



Figura 22: Generazioni di stable 3.5 large di Stability AI

Sono stati condotti ulteriori test specifici liberamente consultabili all'interno del repository di progetto disponibile qui: <https://github.com/alessandropier/Personalized-Fashion-Ads>

Questo fenomeno ha imposto una revisione sostanziale della struttura del prompt in output da Gemini, al fine di renderlo compatibile con le esigenze più restrittive del motore di rendering offerto da Stability.

4.5.3 Ottimizzazione e test iterativi: verso una pipeline coerente

Per compensare la perdita qualitativa osservata nella nuova configurazione, sono stati condotti numerosi test iterativi. Le principali strategie sperimentate sono state:

- **Modulazione del prompt di Gemini:** accorciamento dei testi, riformulazione delle sezioni, aumento della densità informativa per ridurre ambiguità interpretative da parte di Stable.
- **Test one-shot e few-shot (e con Stable Style):** somministrazione ad entrambi i modelli di esempi di prompt ben strutturati come riferimento, allo scopo di “guidare” l’elaborazione e aumentare la consistenza tra output diversi.

- **Prompt negativi e tecniche di tuning:** introduzione di istruzioni nel campo “negative_prompt” per rimuovere elementi indesiderati (es. “blurry image”, “disfigured”), modifica dei parametri *cfg_scale*, *seed*, *steps*, *sampler*.
- **Scambio di ordine nelle frasi:** riformulazione sintattica per migliorare l’interpretazione del modello Stability API, meno tollerante rispetto a Hugging Face.

Tuttavia, **nessuna configurazione sperimentata ha eguagliato le prestazioni** raggiunte con Hugging Face, confermando la diversa calibrazione e ottimizzazione interna dei modelli forniti dai due provider.

In particolare, i **test one-shot, few-shot** e con l’uso di **stable style** non hanno prodotto i risultati auspicati né con Gemini né con Stable. L’inserimento di esempi precedenti, anziché stimolare una maggiore coerenza o creatività, ha spesso indotto entrambi i modelli a *copiare* il contenuto mostrato: Gemini tendeva a ripetere porzioni della descrizione esemplificativa, mentre Stable replicava l’estetica o la composizione visiva dell’immagine fornita come riferimento. Questo comportamento ha evidenziato una limitata capacità di generalizzazione contestuale all’interno del prompting multimodale, specialmente nei modelli utilizzati tramite API.

Analogamente, anche la variazione dei parametri di generazione (come *cfg_scale*, *num_inference_steps*, *sampler*) non ha portato a risultati qualitativamente rilevanti. In molti casi, modifiche anche lievi hanno generato effetti imprevedibili o compromesso l’equilibrio estetico dell’immagine. A parità di prompt, sono state riscontrate immagini con dettagli deformi, perdita di coerenza stilistica o composizioni eccessivamente semplificate, a dimostrazione di una notevole instabilità nel comportamento del backend Stable fornito da Stability AI.

Le problematiche appena descritte vengono evidenziate nei seguenti confronti, tutte a partire dalla prima immagine femminile di baseline consultabile al capitolo **4.3 Selezione delle Immagini di Baseline**:

- **Figura 23:** prompt usato riportato in Appendice **A.5 Riposta di Gemini Flash Experimental**.

- **Figura 24:** prompt usato riportato in Appendice **A.5 Riposta di Gemini Flash Experimental**
- **Figura 25:** uso di Stable Style, prompt usato riportato in Appendice **B.2 Prompt 2 Generato da Gemini Flash Experimental**
- **Figura 26:** Stable Diffusion 3.5 in modalità one-shot con immagine spunto la prima generazione di stable 3.5 large di hugging face (**Figura 23**, sinistra)



Figura 23: Generazioni di stable 3.5 large: A sinistra *hugging face* e a destra *Stability AI*

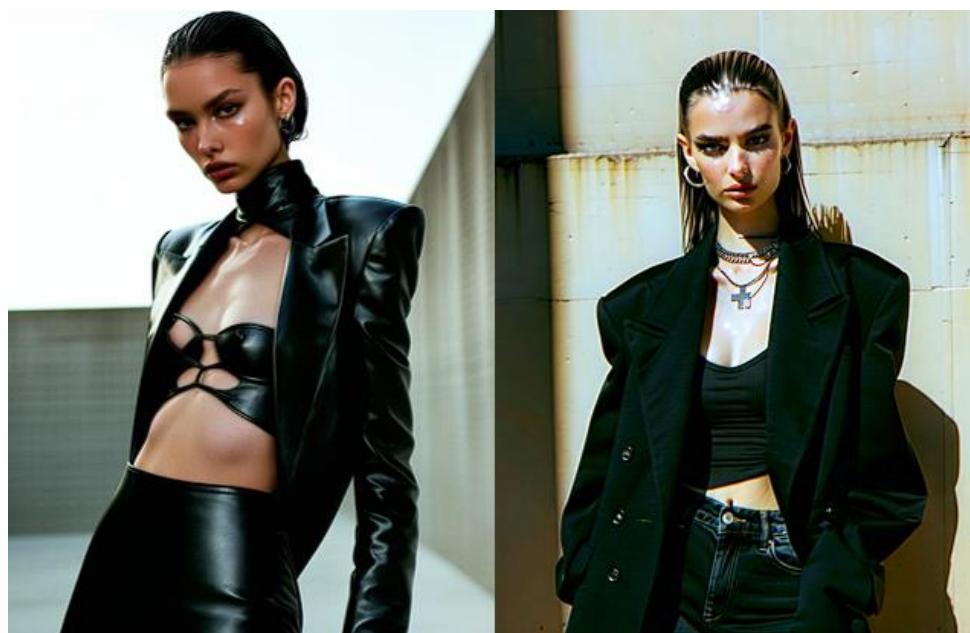


Figura 24: Generazioni di stable 3.5 large: A sinistra *hugging face* e a destra *Stability AI*



Figura 25: Generazioni di stable 3.5 large: A sinistra hugging face e a destra Stability AI con parametro Style

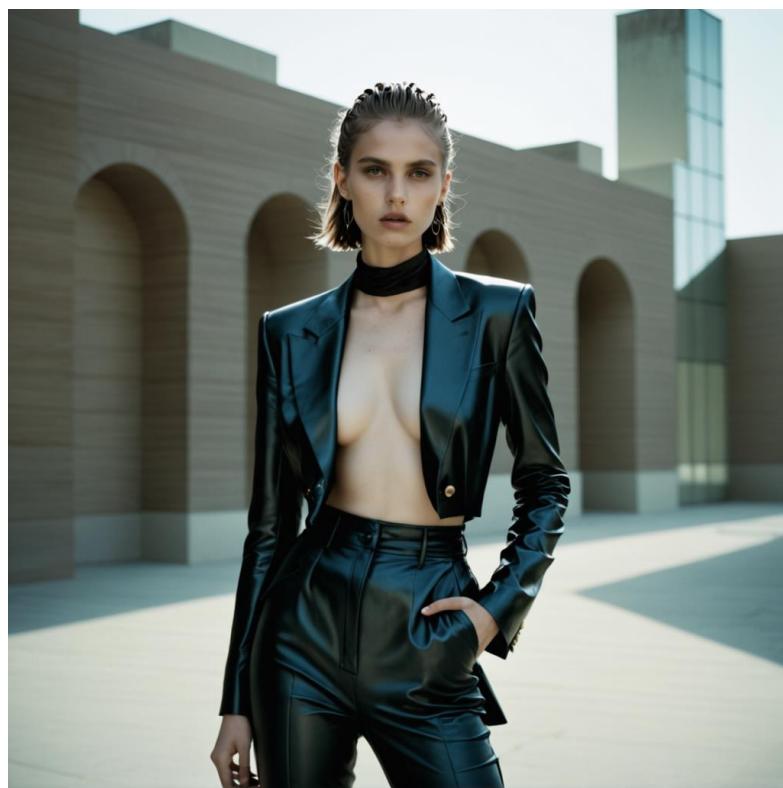


Figura 26: Generazione di stable 3.5 large di Stability AI a partire da un prompt One-Shot di Stable

A completare il quadro delle difficoltà incontrate, altri test condotti in diverse configurazioni hanno evidenziato ulteriori criticità. Tra i problemi più frequenti si segnala una *eccessiva uniformazione degli output*: le immagini generate tendevano a riproporre **inquadrature simili, pose identiche e composizioni ripetitive**, indipendentemente dal contenuto del prompt o dalle preferenze espresse dall'utente. Questa problematica è stata molto

evidente nelle generazioni che impiegavano l'utilizzo del parametro *style* di Stable Diffusion che anche al variare della descrizione utente generavano sempre lo stesso soggetto, nella stessa posa e nella stessa composizione.

In alcuni casi, si è verificato il **cambiamento radicale del capo di partenza**, con il sistema che sostituiva il design o l'indumento base con un elemento completamente diverso, compromettendo la coerenza visiva tra input e output. Altre problematiche riscontrate includono: immagini sfocate (blurry), colori innaturali o non corrispondenti al prompt, elementi morfologici incoerenti (come arti malformati o proporzioni errate), e volti generati di bassa qualità, spesso fuori fuoco o con tratti distorti.



Figura 27: Altri esempi generativi con stable 3.5 large di stability AI con prompt A.5 Riposta di Gemini Flash Experimental, 1° immagine femminile di baseline

Non è stato possibile determinare con precisione da cosa derivino tali divergenze nei risultati: potrebbero dipendere da differenze nell'infrastruttura, nei checkpoint del modello, nei filtri applicati o nei meccanismi di post-processing, ma la scarsa trasparenza di Hugging Face e l'impossibilità di eseguire localmente Stable Diffusion 3.5 Large, a causa dei requisiti computazionali elevati, hanno impedito una verifica diretta. Alla luce di questi risultati, è stato necessario ripensare in modo più rigoroso l'intera pipeline, come verrà discusso nella sezione successiva, dedicata alla definizione del prompt finale ottimizzato per la generazione personalizzata.

4.5.4 Definizione del Prompt Finale: bilanciamento tra regole e creatività

Alla luce delle numerose sperimentazioni condotte, si è giunti alla definizione di un prompt finale per Gemini che riuscisse a bilanciare in modo soddisfacente precisione strutturale e compatibilità con il sistema di generazione finale basato su Stable Diffusion di Stability AI. La scelta è stata dettata dall'esigenza di ottenere un prompt:

- sufficientemente chiaro e compatto, per non oltrepassare i limiti di interpretazione imposti dal modello text-to-image (Stable Diffusion 3.5 large di Stability AI);
- formalmente strutturato, per veicolare istruzioni esplicite e controllate;
- direttamente convertibile in immagine senza necessità di reinterpretazioni o adattamenti da parte dell'utente.

Il prompt definitivo adottato è stato progettato per guidare in modo puntuale Gemini nella costruzione di un testo descrittivo coerente con le preferenze utente, ma anche conforme alle regole di generazione dell'immagine.

Il messaggio di input definitivo pensato per Gemini inizia con una consegna generale che definisce il compito del modello: generare una descrizione testuale ricca, visivamente evocativa e adatta alla creazione di un'immagine pubblicitaria in ambito moda. Questa frase di apertura stabilisce il tono e la finalità dell'intera generazione, ponendo l'accento sia sull'aspetto estetico che sull'efficacia comunicativa del risultato atteso.

Your task is to generate a detailed and visually compelling prompt to create an advertising (fashion editorial) image.

Di seguito, il prompt di input contenente le istruzioni impartite al VLM è suddiviso in diverse sezioni:

1) Analisi visiva del capo

Al modello viene richiesto di identificare i dettagli salienti del capo visibile nell'immagine (es. colore, taglio, tessuto). Questo passaggio è cruciale per assicurare che la descrizione sia ancorata visivamente all'input.

1. Analyze the Image: Identify the clothing item's key details—style, color, fabric, fit, and unique elements.

2) Personalizzazione sulla base del profilo utente

Senza mai citare esplicitamente l'utente, Gemini deve riformulare la descrizione tenendo conto di gusti e caratteristiche dell'utente (es. colori preferiti, marchi amati). Questo consente una personalizzazione implicita ma coerente.

2. Personalize the Description: Adapt the description to match the user's fashion preferences and needs. Ensure it aligns with their preferred colors, styles, and occasions. Never explicitly mention the user characteristics, just use them as inspiration for the textual description.

3) Coerenza tra modello descritto e utente target

Viene richiesto che il soggetto raffigurato nella descrizione (es. la modella) rispecchi l'età e il genere del profilo utente, così da rafforzare il senso di immedesimazione.

3. Ensure that the model is representative of the user: it should match the user age and gender.

4) Autonomia narrativa del prompt

Il prompt generato deve essere “standalone”, ovvero indipendente dall’immagine di partenza. Questo è fondamentale perché Stable Diffusion non riceve l’immagine iniziale, ma solo il prompt testuale.

4. Craft a Standalone Prompt: Write an engaging, detailed text prompt as if describing an idealized version of the clothing piece. Do not reference the original image—make the prompt independent for text-to-image generation.

5) Inquadratura obbligatoria full-body

Per evitare ambiguità nell’inquadratura e migliorare l’efficacia comunicativa, viene richiesto esplicitamente che l’immagine mostri il soggetto a figura intera (es. “full-body shot”, “standing pose”).

5. Ensure a Full-Body Shot: The prompt must specify a **“full-body view”** of the model wearing the outfit, avoiding close-ups. Use terms like “full-body portrait,” “standing pose,” or “showing the entire outfit” to reinforce this.

6) Densità informativa bilanciata

Si richiede una descrizione ricca ma non eccessivamente prolissa: lo scopo è fornire indicazioni su soggetto, azione, ambientazione, stile fotografico e colori, mantenendo compattezza sintattica.

6. Ensure that the prompt is not excessively verbose. It should be detailed enough to guide the image generation process effectively: it should specify the style of the image, the subject and action, the composition and framing, the lightning and color, and the background.

7) Output pulito e privo di spiegazioni

Il modello deve restituire **solo** il prompt descrittivo, senza commenti aggiuntivi, spiegazioni o contenuti fuori contesto.

7. Output Only the Textual Prompt: No explanations, metadata, or additional commentary—only the generated description.

Il prompt completo è consultabile in Appendice **D.1 Prompt Finale Utilizzato per Gemini**.

Chiamata API a Gemini 2.0 Flash Exp

La combinazione tra le istruzioni fisse (prompt progettato e memorizzato nella variabile *vlm_prompt*), il profilo personalizzato dell'utente (memorizzato in *user_description*) e l'immagine in formato base64 (memorizzata in *data_uri*) viene strutturata all'interno di un messaggio formattato in JSON e inviata al modello Gemini come input multimodale.

(**Figura 28**)



```
messages = [
    {
        "role": "user",
        "content": [
            {
                "type": "text",
                "text": vlm_prompt + user_description
            },
            {
                "type": "image_url",
                "image_url": {
                    "url": data_uri
                }
            }
        ]
    }
]
```

Figura 28: Creazione del messaggio da inviare all'API di Gemini

L'invio avviene tramite l'endpoint */chat/completions* del provider *OpenRouter*, controllando i parametri impostati in **Figura 29**:

```
completion = client.chat.completions.create(  
    model="google/gemini-2.0-flash-exp:free",  
    messages=messages,  
    temperature=0,  
    top_p=0.8,  
    seed=42  
)
```

Figura 29: Chiamata API a Gemini 2.0 Flash Exp

Chiamata API a Stable Diffusion 3.5 large

Dopo aver ricevuto il prompt generato da Gemini, il sistema lo passa come input al modello *Stable Diffusion 3.5 Large* tramite l'API ufficiale di Stability AI. La richiesta viene effettuata con metodo POST verso l'endpoint <https://api.stability.ai/v2beta/stable-image/generate/sd3> e include tutti i parametri necessari in un payload multipart/form-data. Tra i parametri più rilevanti vi sono:

- **model**: specifica il modello (sd3.5-large)
- **prompt**: contiene il prompt testuale prodotto da Gemini
- **steps**: numero di iterazioni di diffusione (es. 40)
- **width / height**: dimensioni dell'immagine (es. 1024x1024)
- **output_format**: formato di output (es. jpeg)
- **seed**: per rendere la generazione replicabile

La chiamata è autenticata mediante un token API privato memorizzato in **STABILITY_API_KEY**, e l'immagine risultante viene salvata localmente all'interno di una directory dedicata. La struttura semplificata della richiesta è mostrata in **Figura 30**.

```
response = requests.post(  
    url="https://api.stability.ai/v2beta/stable-image/generate/  
sd3",  
    headers={  
        "Authorization": f"Bearer {STABILITY_API_KEY}",  
        "Accept": "image/*"  
    },  
    files={  
        "model": (None, "sd3-large"),  
        "mode": (None, "text-to-image"),  
        "seed": (None, "42"),  
        "prompt": (None, generated_prompt),  
        "output_format": (None, "jpeg"),  
        "steps": (None, "40"),  
        "width": (None, 1024),  
        "height": (None, 1024),  
    }  
)
```

Figura 30: Chiamata API a stable diffusion 3.5 large

Vengono riportati alcuni esempi nei seguenti confronti:

- **Figura 31**, prima baseline femminile consultabile al capitolo **4.3 Selezione delle Immagini di Baseline**:
 - Immagine a sinistra ottenuta con prompt discorsivo e altamente descrittivo di Gemini (**B.2 Prompt 2 Generato da Gemini Flash Experimental**) e Stable 3.5 Large di Hugging Face.
 - Immagine a destra ottenuta con prompt realizzato da Gemini in seguito alla definizione del prompt finale (**D.2 Prompt 1 Generato da Gemini con Prompt Definitivo**) e Stable 3.5 Large di Stability AI.
- **Figura 32**, prima baseline maschile consultabile al capitolo **4.3 Selezione delle Immagini di Baseline**:
 - Immagine a sinistra ottenuta con prompt discorsivo e altamente descrittivo di Gemini (**B.3 Prompt 3 Generato da Gemini Flash Experimental**) e Stable 3.5 Large di Hugging Face.
 - Immagine a destra ottenuta con prompt realizzato da Gemini in seguito alla definizione del prompt finale (**D.3 Prompt 2**

Generato da Gemini con Prompt Definitivo D.2 Prompt 1

Generato da Gemini con Prompt Definitivo) e Stable 3.5

Large di Stability AI.

- **Figura 33**, seconda baseline femminile consultabile al capitolo **4.3**

Selezione delle Immagini di Baseline:

- Immagine a sinistra ottenuta con prompt discorsivo e altamente descrittivo di Gemini (**B.4 Prompt 4 Generato da Gemini Flash Exp**) e Stable 3.5 Large di Hugging Face.
- Immagine a destra ottenuta con prompt realizzato da Gemini in seguito alla definizione del prompt finale (**D.4 Prompt 3 Generato da Gemini con Prompt Definitivo**) e Stable 3.5 Large di Stability AI.



Figura 31: A sinistra stable 3.5 large di hugging face, a sinistra stable 3.5 large di stability AI



Figura 32: A sinistra stable 3.5 large di hugging face, a sinistra stable 3.5 large di stability AI



Figura 33: A sinistra stable 3.5 large di hugging face, a sinistra stable 3.5 large di stability AI

L'approccio adottato prevede quindi una **serie di vincoli e istruzioni sequenziali** che riducono lo spazio per interpretazioni arbitrarie, costringendo il modello a produrre un output ben delimitato e adatto alla

generazione visiva. Il risultato è un prompt meno creativo e meno narrativo rispetto a quanto ottenibile in precedenza con Hugging Face, ma più funzionale e robusto nel contesto delle API Stability.

In conclusione, pur avendo mantenuto Stable Diffusion 3.5 Large come modello text-to-image, si è dovuto ripiegare sull'implementazione fornita da Stability AI, meno efficace rispetto alla controparte su Hugging Face. Sebbene la pipeline attuale garantisca coerenza e riproducibilità, è evidente che le potenzialità espressive e la qualità visiva massima sarebbero state raggiungibili solo mantenendo l'infrastruttura originaria basata su Hugging Face, la quale, per ragioni tecniche ed economiche, non è risultata percorribile in questa fase progettuale.

4.6 Progettazione Logica del Database e Scelte Progettuali

La progettazione logica del database rappresenta una fase fondamentale nello sviluppo del sistema, in quanto consente di definire in modo strutturato l'organizzazione dei dati e le relazioni tra le diverse entità coinvolte nel processo. Ciascuna tabella è stata progettata con attenzione alla normalizzazione e alla tracciabilità dei dati, con l'obiettivo di semplificare la gestione del flusso informativo durante tutte le fasi dell'interazione utente e della successiva analisi statistica. Nel prosieguo del capitolo verrà descritto nel dettaglio il modello logico del database, evidenziando le tabelle coinvolte, i principali vincoli di integrità e le relazioni tra le entità. La seguente sezione è la continua di quanto già inizialmente discusso al [capitolo 3.4](#) della metodologia in cui si è affrontata la prima fase di progettazione.

Schema Relazionale

Il database del sistema è organizzato secondo una struttura relazionale composta da cinque tabelle principali. Di seguito è riportato lo schema relazionale con le relative *chiavi primarie* (PK) e *chiavi esterne* (FK).

Le chiavi primarie sono evidenziate in **grassetto** mentre le chiavi esterne sono in *corsivo*.

- immagini_baseline (**idImmagine (PK)**, genere_del_capo, path_immagine)
- domande (**id_set (PK)**, domanda1, domanda2, domanda3, domanda4, domanda5, domanda6)
- registrazioni (**idUtente (PK)**, nome, cognome, eta, nazione, genere, corrente_artistica_preferita, professione, colori_preferiti, generi_musicali_preferiti, cosa_cerchi_nei_capi, marchi_preferiti, competenza_moda, interesse_moda)

- immagini_generate (**idGenerazione (PK)**, *id_imagine_baseline* (*FK*), prompt_text_to_image, data_ora, path_immagine_generata)
- questionario (**idQuestionario (PK)**, *idUtente* (*FK*), *id_imagine_generata* (*FK*), *id_set_domande* (*FK*), domanda1, domanda2, domanda3, domanda4, domanda5, domanda6)

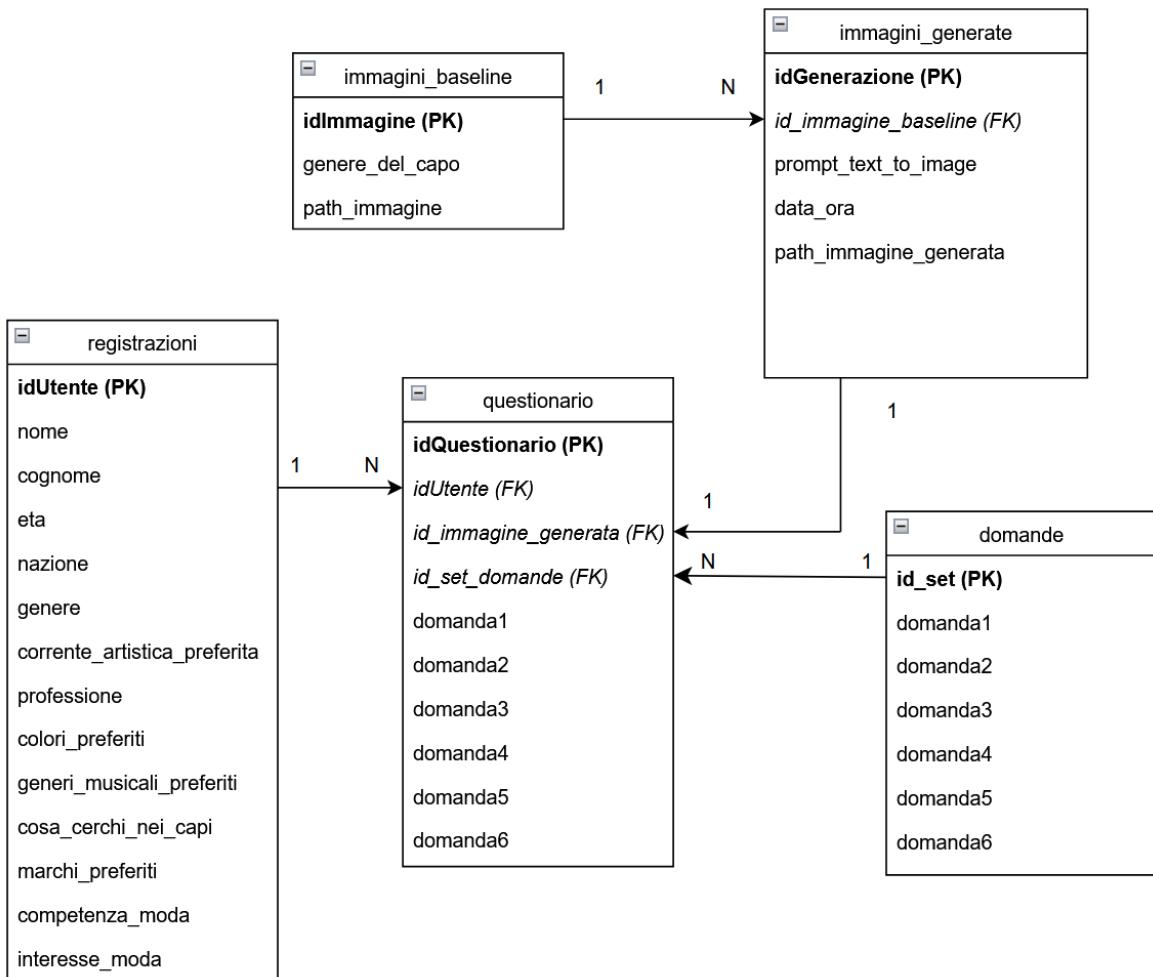


Figura 34: Diagramma Relazionale per il Database

4.6.1 Discussione delle Scelte Progettuali, delle Chiavi e degli Attributi

Durante la progettazione logica e fisica del database, sono state adottate alcune scelte strutturali guidate da esigenze di **funzionalità, semplicità di implementazione e compatibilità con l'infrastruttura basata su file CSV e su Gradio**. In questa sezione si discutono alcune decisioni rilevanti, in

particolare la collocazione delle chiavi esterne e la gestione di attributi a valori multipli.

1) Chiave esterna id_immagine_generata nella tabella questionario

Nel progetto, ogni valutazione effettuata dall'utente attraverso il questionario si riferisce a una sola immagine generata. Di conseguenza, la relazione tra questionario e immagini_generate è **uno-a-uno**: per ogni record del questionario esiste un solo riferimento valido a una generazione, e viceversa, ogni immagine generata può essere associata a una sola valutazione.

Nonostante questa corrispondenza biunivoca, si è scelto di **collocare la chiave esterna id_immagine_generata all'interno della tabella questionario**, e non di accorpare tutto in una relazione. Questa decisione è motivata dal fatto che **il questionario rappresenta l'azione conclusiva e semantica** del flusso utente, ovvero la valutazione del contenuto generato. L'immagine, per quanto centrale nel processo, non contiene di per sé dati valutativi e non ha bisogno di conoscere il questionario a cui è associata. In termini progettuali, dunque, è più naturale che sia la valutazione (il questionario) a contenere l'immagine corrispondente.

Inoltre, accorpare questi dati in un'unica tabella avrebbe compromesso la **modularità del sistema e aumentato la complessità nella gestione dei dati** (soprattutto per operazioni di aggiornamento e analisi), rendendo meno chiara la separazione dei ruoli funzionali delle entità.

2) Attributi a valori multipli e violazione della Prima Forma Normale

Nel modello attuale, alcuni attributi legati alle preferenze dell'utente come colori_preferiti, generi_musicali_preferiti, cosa_cerchi_nei_capi e marchi_preferiti sono gestiti come **campi stringa contenenti liste delimitate**. Questi campi consentono all'utente di selezionare più opzioni contemporaneamente durante la registrazione, e sono salvati nel database come **stringhe testuali** separate da virgolette (es. "Nero, Bianco").

Tale approccio **viola la Prima Forma Normale (1NF)**, secondo cui ogni attributo deve contenere un singolo valore atomico per ogni record.

Tuttavia, questa scelta è stata fatta consapevolmente, in quanto l'adozione di una struttura completamente normalizzata avrebbe comportato:

- la creazione di numerose **tabelle intermedie (tabelle ponte)** per rappresentare le relazioni multi-a-molti tra utenti e attributi preferiti;
- un aumento considerevole della **complessità di gestione e interrogazione del database**;
- la necessità di rielaborare profondamente anche la struttura dei file CSV, compromettendo la **semplicità e portabilità** del sistema.

Dato che il sistema è concepito come un **prototipo interattivo e facilmente estendibile**, si è preferita una **struttura più semplice**, allineata al formato dei file .csv e facilmente manipolabile tramite *pandas*. L'approccio adottato consente una gestione efficace anche da parte di **utenti non esperti**, garantendo al contempo la coerenza interna dei dati.

In prospettiva futura, qualora il sistema venisse evoluto in un contesto di produzione o esteso a una base utenti più ampia, sarà possibile **ristrutturare il database in modo pienamente normalizzato**, definendo tavole dedicate per ciascuna categoria di preferenze e gestendo le associazioni tramite **foreign key**. Questo permetterebbe interrogazioni più potenti e relazioni più flessibili, ma richiederebbe anche un'interfaccia e un'infrastruttura di gestione più avanzate.

3) Attributi “domanda N” delle Tabelle “questionario” e “domande”

Un aspetto progettuale significativo riguarda la gestione delle domande del questionario e delle relative risposte. Entrambe le entità coinvolgono campi denominati domanda1, domanda2, ..., domanda6, ma con ruoli semantici completamente differenti.

Nella tabella domande, ciascun attributo domandaX rappresenta il **testo effettivo della domanda** sottoposta all'utente durante la fase di valutazione. Ogni riga della tabella corrisponde a un **insieme coerente di domande**, identificato dal campo id_set, che funge da chiave primaria. Questa struttura è stata pensata per consentire la **personalizzazione e la sostituzione dinamica delle domande** semplicemente modificando il

contenuto del file domande.csv, senza dover intervenire sul codice applicativo. Durante l'interazione utente, infatti, il sistema carica automaticamente il set di domande corretto leggendo il file e filtrando le righe in base al valore specificato dell'attributo *id_set*.

Al contrario, nella tabella questionario, gli attributi domanda1, ..., domanda6 **non contengono i testi delle domande**, bensì le **risposte fornite dall'utente** in corrispondenza di ciascuna domanda.

Tale impostazione permette di **separare nettamente i dati statici (testi delle domande)** dai **dati dinamici (risposte degli utenti)**, garantendo maggiore coerenza, tracciabilità e flessibilità nell'evoluzione del sistema. Inoltre, la presenza del campo *id_set_domande* nella tabella *questionario* consente, in fase di analisi o visualizzazione, di **recuperare dinamicamente i testi delle domande corrispondenti** al set utilizzato per ciascuna valutazione, tramite semplice join con la tabella domande.

Conclusioni

In conclusione, tenendo conto delle scelte progettuali adottate, delle chiavi primarie selezionate, e della struttura generale delle tabelle, il modello relazionale proposto rispetta i principi della **terza forma normale (3NF)**. Ogni attributo dipende funzionalmente e univocamente dalla chiave primaria della propria tabella, evitando ridondanze e anomalie di aggiornamento, fatta eccezione per alcune semplificazioni consapevoli (come i campi a valori multipli) introdotte per motivi pratici e di compatibilità con il formato CSV e per gli obiettivi del progetto .

Nella [sezione 4.4](#) di seguito si approfondisce l'**implementazione effettiva del database SQLite**, descrivendo in dettaglio il codice utilizzato per la sua creazione, inizializzazione e popolamento, nonché le logiche di caricamento e integrazione con i file CSV.

4.7 Gestione e Organizzazione dei Dati

La gestione dei dati rappresenta un aspetto centrale nella realizzazione del sistema, poiché garantisce la coerenza, l'integrità e l'accessibilità delle informazioni tra le diverse componenti della piattaforma. In questo progetto, i dati vengono inizialmente organizzati in formato **CSV**, successivamente importati in un **database SQLite**, per essere infine utilizzati nel frontend tramite operazioni di lettura e scrittura dinamiche. In questa sezione vengono descritti nel dettaglio la struttura dei dataset, le modalità di caricamento nel database e le strategie adottate per tracciare e aggiornare i dati utente, immagini e risposte.

4.7.1 Struttura e Caricamento dei File CSV

L'intero sistema si basa su una serie di file CSV che fungono da fonte dati primaria. Tali file vengono creati o inizializzati automaticamente alla prima esecuzione del codice, oppure generati preventivamente tramite script Python dedicati. I principali file utilizzati sono:

- **registrazioni.csv**: contiene i dati degli utenti raccolti tramite il modulo di registrazione.
- **immagini_baseline.csv**: contiene le immagini di riferimento distinte per genere, inizializzato tramite uno script Python eseguito prima del lancio dell'applicazione.
- **immagini_generate.csv**: traccia le immagini generate dal sistema, includendo metadati come il prompt, l'immagine di partenza e l'orario di generazione.
- **domande.csv**: raccoglie i set di domande dei questionari, organizzate per *id_set*, in modo da garantire flessibilità nella gestione futura.
- **questionario.csv**: raccoglie le risposte degli utenti ai questionari associati a ciascuna immagine generata.

I file CSV sono strutturati come segue:

1) registrazioni.csv: Contiene i dati anagrafici e preferenziali di ciascun utente registrato nel sistema. Gli header inclusi in questo file sono:

- idUtente: ID univoco assegnato all'utente.
- nome e cognome: Nome e cognome dell'utente.
- eta: Età dell'utente (valore intero).
- nazione: Nazione di residenza dell'utente.
- genere: Genere dell'utente, selezionabile tra Uomo e Donna.
- corrente_artistica_preferita: Corrente artistica scelta tra un insieme di opzioni stilistiche.
- professione: Professione dichiarata dall'utente.
- colori_preferiti: Fino a due colori selezionati come preferiti.
- generi_musicali_preferiti: Fino a due generi musicali prediletti.
- cosa_cerchi_nei_capi: Fino a tre caratteristiche ricercate nei capi di abbigliamento.
- marchi_preferiti: Fino a due brand selezionati come preferiti.
- competenza_moda: Grado di competenza percepita nel settore moda (scala Likert da 1 a 5).
- interesse_moda: Grado di interesse verso la moda (scala Likert da 1 a 5).

2) immagini_baseline.csv: Contiene i riferimenti alle immagini predefinite (baseline) suddivise per genere. Questo file è stato generato tramite uno script Python prima dell'avvio del sistema. Gli header inclusi sono:

- idImmagine: ID univoco dell'immagine baseline.
- genere_del_capo: Genere di appartenenza dell'immagine (Uomo o Donna).
- path_immagine: Percorso al file immagine corrispondente.

3) immagini_generate.csv: Registra le informazioni relative alle immagini generate dinamicamente dal modello Stable Diffusion. Gli header presenti includono:

- idGenerazione: ID univoco della generazione.
- id_immagine_baseline: Riferimento all'immagine baseline utilizzata.
- prompt_text_to_image: Prompt testuale generato dal modello Gemini a partire dalla descrizione utente e dall'immagine di riferimento.

- data_ora: Data e ora della generazione.
- path_immagine_generata: Percorso al file immagine generato.

4) domande.csv: Contiene l'elenco delle domande proposte all'utente durante il questionario di valutazione. Anche questo file viene creato con uno script Python dedicato. Gli header inclusi sono:

- id_set: Identificativo del set di domande (permettendo versioni alternative).
- domanda1 ... domanda6: Testi delle sei domande da porre all'utente. Le prime cinque sono comparative (scelta tra sinistra/destra/indifferente), la sesta è a scala Likert.

5) questionario.csv: Archivia tutte le risposte fornite dagli utenti durante i questionari associati a ciascuna immagine generata. Gli header inclusi sono:

- idQuestionario: ID univoco del questionario.
- idUtente: ID dell'utente che ha compilato il questionario.
- id_immagine_generata: ID dell'immagine generata a cui si riferisce la valutazione.
- id_set_domande: Riferimento al set di domande utilizzato per quella valutazione.
- domanda1 ... domanda6: Risposte fornite dall'utente per ciascuna domanda del set.

L'inizializzazione e il controllo di integrità dei file CSV avvengono in apertura del sistema. Ad esempio, per garantire l'esistenza e la corretta struttura del file *immagini_baseline.csv*, viene implementato un controllo su *FileNotFoundException*.

```

try:
    df_baseline = pd.read_csv(IMMAGINI_BASELINE_FILE)
except FileNotFoundError:
    baseline_data = []
    # Popolamento iniziale
    ...
    df_baseline = pd.DataFrame(baseline_data)
    df_baseline.to_csv(IMMAGINI_BASELINE_FILE, index=False)

```

Figura 35: Controllo di Integrità per File CSV

La medesima logica viene applicata anche al file dei questionari, assicurandosi che le colonne delle domande siano coerenti con la struttura attesa:

```

if not all(col in questionnaire_data.columns for col in
required_columns) or \
    not all(f"domanda{i}" in questionnaire_data.columns for i in
range(1, max_questions + 1)):
    raise ValueError("Struttura modificata, rigenero il dataset")

```

Figura 36: Controllo di Integrità per altri File CSV

4.7.2 Importazione nel Database SQLite

Al fine di garantire una gestione più efficiente, flessibile e scalabile dei dati, il sistema importa tutti i dati presenti nei CSV all'interno di un database **SQLite**, creato e popolato automaticamente all'avvio del modulo dedicato.

La struttura logica del database è composta da cinque tabelle:

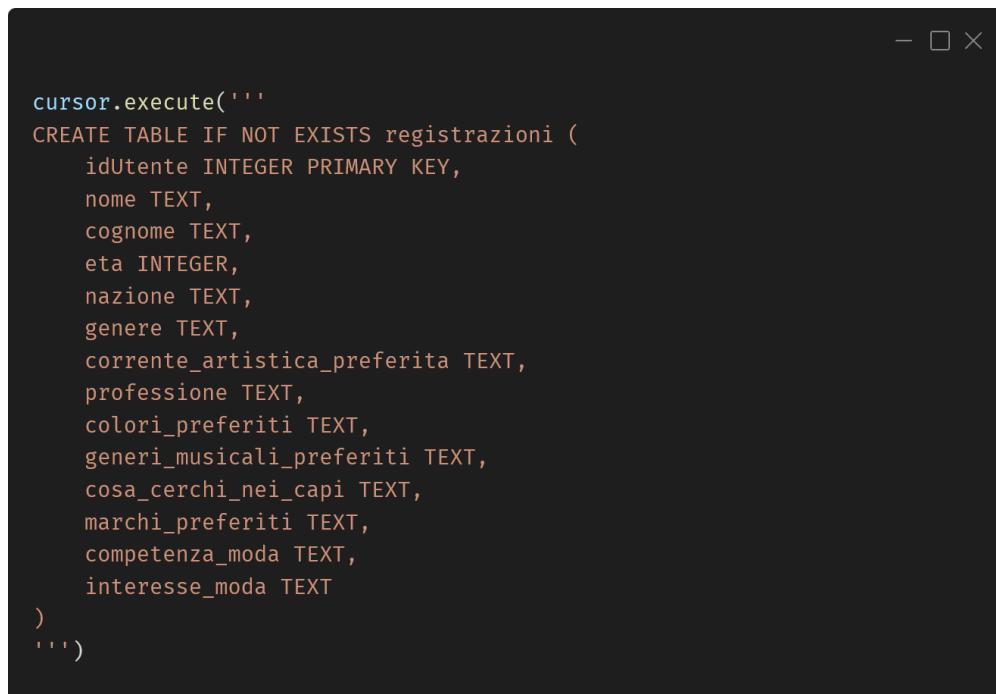
- registrazioni
- questionario
- immagini_generate
- immagini_baseline
- domande

Il processo di creazione e caricamento è affidato alla funzione *init_db()* che, attraverso il modulo *sqlite3*, si occupa di:

- **Creare le tabelle** (se non già esistenti) tramite comandi SQL *CREATE TABLE IF NOT EXISTS*
- **Caricare i dati dai file CSV** nelle rispettive tabelle, evitando duplicati mediante *INSERT OR IGNORE*

Create Table

Le tabelle vengono definite tramite comandi *CREATE TABLE IF NOT EXISTS*, che assicurano l'idempotenza dell'operazione. Ogni tabella è costruita con i campi coerenti rispetto alla struttura dei file CSV.



```

cursor.execute('''
CREATE TABLE IF NOT EXISTS registrazioni (
    idUtente INTEGER PRIMARY KEY,
    nome TEXT,
    cognome TEXT,
    eta INTEGER,
    nazione TEXT,
    genere TEXT,
    corrente_artistica_preferita TEXT,
    professione TEXT,
    colori_preferiti TEXT,
    generi_musicali_preferiti TEXT,
    cosa_cerchi_nei_capi TEXT,
    marchi_preferiti TEXT,
    competenza_moda TEXT,
    interesse_moda TEXT
)
'''')

```

Figura 37: Create Table per la Tabella "registrazioni"

La stessa logica viene replicata per le altre tabelle, rispettando le chiavi primarie e i riferimenti esterni se presenti (es. idUtente in questionario, id_immagine_baseline in immagini_generate, ecc.).

Inserimento dei dati dai CSV

L'operazione di caricamento dei dati dai CSV avviene all'interno della funzione *load_data(cursor)*. Per ciascun file CSV, il codice apre manualmente il file e ne legge i contenuti riga per riga, utilizzando il modulo *csv.reader*. Ogni dataset è gestito separatamente, con una logica su misura e un possibile convertitore per i tipi di dato.

```

with open("registrazioni.csv", 'r', encoding='utf-8') as f:
    reader = csv.reader(f)
    headers = next(reader)
    for row in reader:
        try:
            row = [
                int(row[0]), row[1], row[2], int(row[3]), row[4], row[5],
                row[6], row[7],
                row[8], row[9], row[10], row[11], int(row[12]),
                int(row[13])
            ]
            placeholders = ','.join('?' * len(row))
            cursor.execute(f'INSERT OR IGNORE INTO registrazioni VALUES
                ({placeholders})', row)
        except Exception as e:
            print(f"Errore conversione riga {row} in registrazioni: {e}")

```

Figura 38: Codice per il Caricamento di "registrazioni.csv" nel Database SQL

Il processo è simile anche per gli altri file come questionario.csv, *immagini_generate.csv*, *immagini_baseline.csv* e *domande.csv*, ognuno con il suo specifico adattamento dei tipi.

4.7.3 Dataset Creati Ad Hoc

Un discorso a parte merita la gestione dei file *immagini_baseline.csv* e *domande.csv*, i quali **non vengono inseriti manualmente dall'utente a runtime, ma vengono creati ad hoc prima dell'esecuzione del sistema** tramite due script Python dedicati. Questa scelta progettuale si fonda su due motivazioni:

- Per le domande è stata prevista una generazione automatizzata da script poiché questo approccio consente una flessibilità elevata: eventuali aggiornamenti o modifiche alle domande possono essere effettuati direttamente sul file CSV, senza dover alterare la logica del codice. Ciò rende il sistema facilmente adattabile anche da parte di utenti non esperti o da ricercatori interessati a effettuare nuove sperimentazioni.
- Per quanto concerne il file *immagini_baseline.csv*, esso rappresenta un dataset statico, contenente le immagini di riferimento da confrontare con quelle generate. Tali immagini non vengono scelte dinamicamente, ma sono state selezionate a priori, prima dell'avvio dell'esperimento, sulla base di criteri estetici e rappresentativi.

```

import pandas as pd

def crea_file_domande():
    DOMANDE_FILE = "domande.csv"
    ID_SET = 1

    domande = [
        "Personalizzazione: Quale delle due immagini ritieni sia più vicina al tuo stile personale?",
        "Intenzione d'Acquisto: Quale immagine ti aiuterebbe di più a decidere se acquistare il capo?",
        "Realismo: Quale immagine appare più realistica come rappresentazione del capo?",
        "Appeal Visivo: Quale immagine trovi esteticamente più convincente per indossare il capo?",
        "Fiducia: Quale immagine ti dà maggiore sicurezza che questo capo ti donerebbe?",
        "Differenziazione: Le due immagini ti sembrano chiaramente distinguibili l'una dall'altra?"
    ]

    data = {"id_set": [ID_SET]}
    for i, domanda in enumerate(domande, 1):
        data[f"domanda{i}"] = [domanda]

    df = pd.DataFrame(data)
    df.to_csv(DOMANDE_FILE, index=False, encoding='utf-8-sig')

```

Figura 39: Semplificazione dello Script Usato per Creare "domande.csv" – Script completo "OneTimeScript_crea_domandecsv.py"

4.7.4 Tracciamento e Aggiornamento dei Dati

Tutte le operazioni svolte dall'utente attraverso l'interfaccia sono tracciate e persistite, al fine di garantire una raccolta completa delle interazioni e di supportare le analisi successive. Di seguito si riportano alcune tra le principali operazioni:

- **Registrazione utente:** una volta completato il form, i dati vengono scritti nel file registrazioni.csv e successivamente nel database. L'ID dell'utente è aggiornato dinamicamente.
- **Generazione dell'immagine:** al termine del processo di generazione, il sistema aggiorna il file immagini_generate.csv con l'ID della baseline, il prompt testuale e il path dell'immagine creata.
- **Compilazione del questionario:** ogni tab contiene un questionario associato all'immagine generata. Le risposte vengono salvate nel file questionario.csv e importate nel database, legando le risposte all'utente e all'immagine tramite ID.

```

new_entry = {
    "idGenerazione": next_generation_id,
    "id_immagine_baseline": id_immagine_baseline,
    "prompt_text_to_image": prompt_text_to_image,
    "data_ora": datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S"),
    "path_immagine_generata": generated_image_path
}
df = pd.concat([df, pd.DataFrame([new_entry])], ignore_index=True)
df.to_csv(IMMAGINI_GENERATE_FILE, index=False)

```

Figura 40: Codice Usato per il Salvataggio delle Generazioni

L'intero sistema è progettato per evitare **duplicazioni accidentali**, grazie all'uso di controlli su ID e path. Ad esempio, prima di salvare una risposta al questionario, il sistema verifica che l'utente non abbia già compilato lo stesso modulo per quella specifica immagine:

```

existing_entry = current_questionnaire_data[
    (current_questionnaire_data["idUtente"] == CURRENT_USER_ID) &
    (current_questionnaire_data["id_immagine_generata"] == generated_id)
]
if not existing_entry.empty:
    return "X Hai già inviato questo questionario per questa
immagine!"

```

Figura 41: Controllo per Evitare i Duplicati

4.8 Realizzazione della Web App con Gradio

La realizzazione della Web App rappresenta uno degli aspetti fondamentali del progetto. Per costruire un'interfaccia utente funzionale, reattiva e accessibile, si è scelto di utilizzare **Gradio**, una libreria Python open-source concepita per la rapida prototipazione e distribuzione di interfacce web interattive, in particolare per *sistemi basati su intelligenza artificiale*. Gradio consente di costruire rapidamente componenti visuali e di integrarli direttamente con le funzioni Python, semplificando la gestione dei flussi tra frontend e backend.

Scelta di Gradio:

Le motivazioni che hanno portato alla scelta di Gradio includono:

- **Facilità d'integrazione** con funzioni Python native.
- **Compatibilità con ambienti AI/ML** e supporto diretto a modelli generativi.
- **Possibilità di lanciare server locali o condividere online** l'interfaccia in modo sicuro e rapido.
- **Gestione delle interfacce a schede** (tabbed interfaces) che ha permesso una netta separazione tra fasi del flusso di lavoro.

La Web App è suddivisa in due interfacce principali:

- **Interfaccia Utente** (visibile da ogni partecipante registrato).
- **Interfaccia Super-User** (accessibile solo con privilegi specifici per l'analisi dei dati).

Entrambe sono costruite attraverso un layout a tab utilizzando il costruttore *gr.TabbedInterface*, e sono organizzate in container modulari per garantire una buona scalabilità.

4.8.1 Interfaccia Utente: Registrazione e Raccolta Profilo

La registrazione rappresenta il primo punto di contatto tra l'utente e il sistema. Essa non si limita alla raccolta di dati anagrafici, ma costituisce il **fulcro semantico** attorno al quale viene costruito l'intero processo di personalizzazione della generazione. È qui, infatti, che si raccolgono le informazioni che alimenteranno il **modello image-text-to-text** nella costruzione del prompt su misura.

Costruzione del Form di Registrazione:

La struttura è costruita usando *gr.Blocks*, con layout basato su righe e colonne (*gr.Row*, *gr.Column*) per ottimizzare la disposizione dei componenti. Gli input sono distribuiti per tematiche:

- **Anagrafica:** nome, cognome, età, nazione, genere
- **Stile di vita:** professione, corrente artistica preferita
- **Preferenze personali:** colori, generi musicali, brand
- **Orientamento stilistico:** cosa si cerca nei capi
- **Autovalutazioni:** interesse e competenza in ambito moda

```
- □ ×  
  
with gr.Blocks(theme=gr.themes.Soft()) as registration_tab:  
    gr.Markdown("# Form di Inserimento Dati Utente")  
  
    with gr.Row():  
        name = gr.Textbox(label="Nome*", placeholder="Inserisci il tuo nome")  
        cognome = gr.Textbox(label="Cognome*", placeholder="Inserisci il tuo cognome")  
    with gr.Row():  
        eta = gr.Slider(label="Età*", minimum=18, maximum=99, step=1, value=18)  
        nazione = gr.Textbox(label="Nazione*", placeholder="Inserisci la tua nazione")  
        genere = gr.Radio(label="Genere*", choices=["Uomo", "Donna"])
```

Figura 42: Struttura iniziale del form di registrazione

I componenti fondamentali usati includono:

- *gr.Textbox* per input testuali (nome, professione, ecc.).
- *gr.Slider* per valori numerici (età).
- *gr.Radio* per selezioni singole (genere).

- *gr.CheckboxGroup* per selezioni multiple (colori preferiti, generi musicali, marchi).
- *gr.Dropdown* per i menù a tendina.

Al momento del *submit*, la funzione *handle_registration* raccoglie i dati, esegue validazioni e salva le informazioni nel file *registrazioni.csv*.

```
submit_btn.click(
    handle_registration,
    inputs=[name, cognome, eta, nazione, genere, corrente_artistica, professione,
           colori_preferiti, generi_musicali, cerca_nei_capi, marchi_preferiti,
           competenza_moda, interesse_moda],
    outputs=output
)
```

Figura 43: Realizzazione della logica per il bottone **submit**

Controlli e Validazioni:

La funzione *handle_registration()* si occupa della gestione backend dell'interfaccia. Essa esegue una serie di **controlli** fondamentali:

1. **Campi obbligatori:** tutti i campi devono essere riempiti, e in caso contrario viene restituito un errore e comunicato all'utente.
2. **Validazione delle scelte multiple:** alcune checkbox (*colori_preferiti*, *marchi_preferiti*) richiedono un numero minimo e massimo di selezioni.
3. **Conversione dei dati:** liste di preferenze vengono salvate in formato stringa (", ".join(*lista*)) per memorizzazione CSV.

```
if len(colori_preferiti) < 1 or len(colori_preferiti) > 2:
    error_messages.append("Errore: Seleziona 1 o 2 colori preferiti")
```

Figura 44: Esempio di Validazione per Form Utente

Salvataggio e Gestione del Dataset:

Il file *registrazioni.csv* rappresenta il **database persistente** degli utenti.

Ogni nuovo utente riceve un *idUtente* univoco e le informazioni vengono accodate con *pd.concat()*.

```
new_entry = {  
    "idUtente": next_id,  
    "nome": name.strip(),  
    "eta": int(eta),  
    ...  
}  
dataset = pd.concat([dataset, pd.DataFrame([new_entry])], ignore_index=True)  
dataset.to_csv(data_file, index=False, encoding='utf-8-sig')
```

Figura 45: Esempio di codice usato per il salvataggio di un nuovo utente nel dataset

Costruzione della Descrizione Utente

Una volta validati i dati, il sistema costruisce una **descrizione semantica dell'utente** da usare nella generazione dei prompt. La descrizione è elaborata in lingua italiana e poi tradotta in inglese tramite **GoogleTranslator** per essere compatibile con Gemini.

```
from deep_translator import GoogleTranslator  
  
# Costruzione profilo in italiano  
descrizione_utente_ita = (  
    "Genere: " + genere + "\n" +  
    f"Età: {eta} \n" +  
    "Nazione: " + nazione + "\n" +  
    "Professione: " + professione + "\n" +  
    "Corrente artistica preferita: " + corrente_artistica + "\n" +  
    f"Marchi preferiti: {' , '.join(marchi_preferiti)}\n" +  
    f"Generi musicali preferiti: {' , '.join(generi_musicali).lower()} \n" +  
    f"Cosa cerca nei capi: {' , '.join(cerca_nei_capi).lower()} \n" +  
    f"Colori preferiti: {' , '.join(colori_preferiti).lower()}"  
)  
  
# Traduzione automatica in inglese  
descrizione_utente_eng = GoogleTranslator(source="it",  
target="en").translate(descrizione_utente_ita)
```

Figura 46: Acquisizione e traduzione della descrizione utente

Questo approccio automatizza la localizzazione semantica della descrizione, adattandola al contesto del modello multimodale senza richiedere interventi manuali, e garantendo la coerenza sintattica con i prompt in input. La descrizione ottenuta viene successivamente integrata nei prompt multimodali generati dal modello Gemini per guidare la personalizzazione dell'immagine in base al profilo utente. Rappresenta quindi il collegamento semantico tra l'identità dell'utente e l'immagine finale.

Efficienza e Separazione Logica

Dal punto di vista tecnico, la separazione tra interfaccia frontend (gestita da Gradio) e backend semantico (che costruisce e traduce le descrizioni, gestisce dataset, validazioni, ID) consente una chiara modularità del sistema, rendendo ogni blocco facilmente manutenibile o estendibile.

Inoltre, grazie all'uso di `gr.update()`, vengono resettati dinamicamente tutti i campi del form tramite il pulsante "Pulisci", aumentando l'usabilità dell'interfaccia.



```
def clear_fields():
    return (
        gr.update(value=""), gr.update(value=""), gr.update(value=18), gr.update(value=""),
        gr.update(value=None),
        gr.update(value=None), gr.update(value=""), gr.update(value=[]), gr.update(value=[]),
        gr.update(value=[]), gr.update(value=[]), gr.update(value=None), gr.update(value=None),
        gr.update(value="")
    )
```

Figura 47: Funzione per il reset dinamico dei campi

Registrazione Immagine 1 Immagine 2 Immagine 3 Immagine 4

Form di Inserimento Dati Utente

I campi contrassegnati con * sono obbligatori

Nome*	Cognome*
Luigi	de Candia
Età*	64
18	99
Nazione*	Italia
Genere*	<input checked="" type="radio"/> Uomo <input type="radio"/> Donna
Professione*	Pensionato
Corrente Artistica Preferita*	Tradizionalismo
Colori Preferiti* (max 2)	<input type="checkbox"/> Arancione <input type="checkbox"/> Bianco <input type="checkbox"/> Blu <input type="checkbox"/> Bordeaux <input type="checkbox"/> Celeste <input type="checkbox"/> Giallo <input type="checkbox"/> Magenta <input checked="" type="checkbox"/> Marrone <input checked="" type="checkbox"/> Nero <input type="checkbox"/> Rosa <input type="checkbox"/> Rosso <input type="checkbox"/> Turchese <input type="checkbox"/> Verde <input type="checkbox"/> Viola
Generi Musicali Preferiti* (max 2)	<input checked="" type="checkbox"/> Blues <input type="checkbox"/> Classica <input type="checkbox"/> Country <input type="checkbox"/> House <input type="checkbox"/> Jazz <input type="checkbox"/> Metal <input checked="" type="checkbox"/> Pop <input type="checkbox"/> Rap <input type="checkbox"/> RnB <input type="checkbox"/> Rock <input type="checkbox"/> Techno

Figura 48: Porzione della schermata di registrazione

4.8.2 Interfaccia Utente: Generazione delle Immagini e Valutazione

La seconda sezione della Web App consente all'utente registrato di generare immagini personalizzate partendo da capi di riferimento (baseline) e valutarle attraverso questionari strutturati. L'interfaccia è stata progettata per risultare intuitiva, interattiva e vincolata al completamento sequenziale, in modo da garantire la coerenza dei dati raccolti.

Struttura a Tab e Gestione Dinamica

L'interfaccia è suddivisa in quattro schede numerate, ciascuna relativa a una diversa immagine di partenza. La creazione delle tab è automatizzata mediante la funzione `create_image_tab(tab_number)`, che costruisce

dinamicamente l’interfaccia per ogni fase di generazione e valutazione. Questo approccio consente di riutilizzare lo stesso schema per più iterazioni, mantenendo coerenza visiva e logica.



```
image_tabs = [
    create_image_tab(1),
    create_image_tab(2),
    create_image_tab(3),
    create_image_tab(4)
]
```

Figura 49: Chiamate alla funzione `create_image_tab`

La funzione `create_image_tab()` utilizza i container `gr.Blocks`, `gr.Row` e `gr.Column` per suddividere verticalmente e orizzontalmente le sezioni: un blocco superiore per la visualizzazione delle immagini (baseline e generata) e uno inferiore per il questionario.



```
with gr.Row(equal_height=True):
    with gr.Column(scale=1):
        img_left = gr.Image(label="Immagine Sinistra", interactive=False, height=600, width=600)
    with gr.Column(scale=1):
        img_right = gr.Image(label="Immagine Destra", interactive=False, height=600, width=600)
```

Figura 50: Esempio di codice per la schermata di visualizzazione delle immagini

Processo di Generazione dell’Immagine

All’interno di ogni tab, l’utente visualizza inizialmente un’immagine di riferimento (baseline) associata al proprio genere. Premendo il pulsante **“Genera”**, si attiva una funzione (`save_generated_image`) che segue un workflow articolato:

1. **Recupero baseline:** viene identificata l’immagine di riferimento dalla tabella `immagini_baseline.csv`.
2. **Creazione del prompt:** viene invocata la funzione `generate_fashion_prompt()` passando:
 - l’immagine di partenza

- una descrizione in lingua inglese delle preferenze utente
- Chiamata all'API di generazione:** il prompt viene inviato al modello Stable Diffusion 3.5 Large, attraverso la funzione `generate_adv_image()`, che restituisce l'immagine generata.
 - Salvataggio metadati:** tutti i dati vengono archiviati in `immagini_generate.csv` con *ID progressivo* e *timestamp* di generazione.



```

prompt_text_to_image = generate_fashion_prompt(
    image_path=baseline_path,
    user_description=descrizione_utente_eng
)

generated_path = generate_adv_image(prompt_text_to_image, next_generation_id)
save_generation_data(baseline_path, prompt_text_to_image, generated_path)

```

Figura 51: Esecuzione del workflow di generazione

Una volta completata la generazione, le due immagini (baseline e generata) vengono visualizzate fianco a fianco all'interno dell'interfaccia. La loro posizione è *randomizzata* ad ogni iterazione per evitare bias visivi: il modello baseline può apparire a sinistra o destra, e la mappatura è salvata nella variabile globale `DISPLAY_ORDERS`.

Questionario di Valutazione

Sotto le immagini viene mostrato un questionario composto da sei domande, caricate dinamicamente dal file `domande.csv` in base all'`id_set`. Le domande sono divise in due sezioni:

- **Domande 1–5:** valutazione comparativa (sinistra, destra, indifferente)
- **Domanda 6:** scala Likert da 1 a 5

L'interfaccia impiega componenti `gr.Radio` per permettere la selezione singola, garantendo chiarezza visiva e facilità d'uso.

```

for i, q in enumerate(questions):
    if i < 5:
        inputs.append(gr.Radio(
            label=q,
            choices=["Sinistra", "Destra", "Indifferente"],
            type="value",
            interactive=True
        ))
    else:
        inputs.append(gr.Radio(
            label=q,
            choices=[(1, "1 - Molto Poco"), ... , (5, "5 - Molto")],
            type="value",
            interactive=True
        ))

```

Figura 52: Esempio della generazione dinamica delle domande

Alla pressione del tasto "**Invia Valutazione**", le risposte vengono trasformate in valori coerenti con l'ordine di visualizzazione (es. "Destra" = "Generated" o "Baseline" a seconda del tab). I risultati vengono salvati nel file questionario.csv insieme all'*id_imagine_generata* e all'*id_utente*.

Validazioni e Vincoli Logici

Inoltre, sono stati implementati dei vincoli per assicurare l'integrità della raccolta dati:

- Un utente **non può generare** l'immagine del tab successivo senza aver completato il questionario del precedente.
- È **vietata la rigenerazione** della stessa immagine: ogni generazione è unica per tab.
- Ogni combinazione utente + immagine generata è **valutabile una sola volta**: è prevista una funzione di controllo che verifica l'assenza di duplicati.

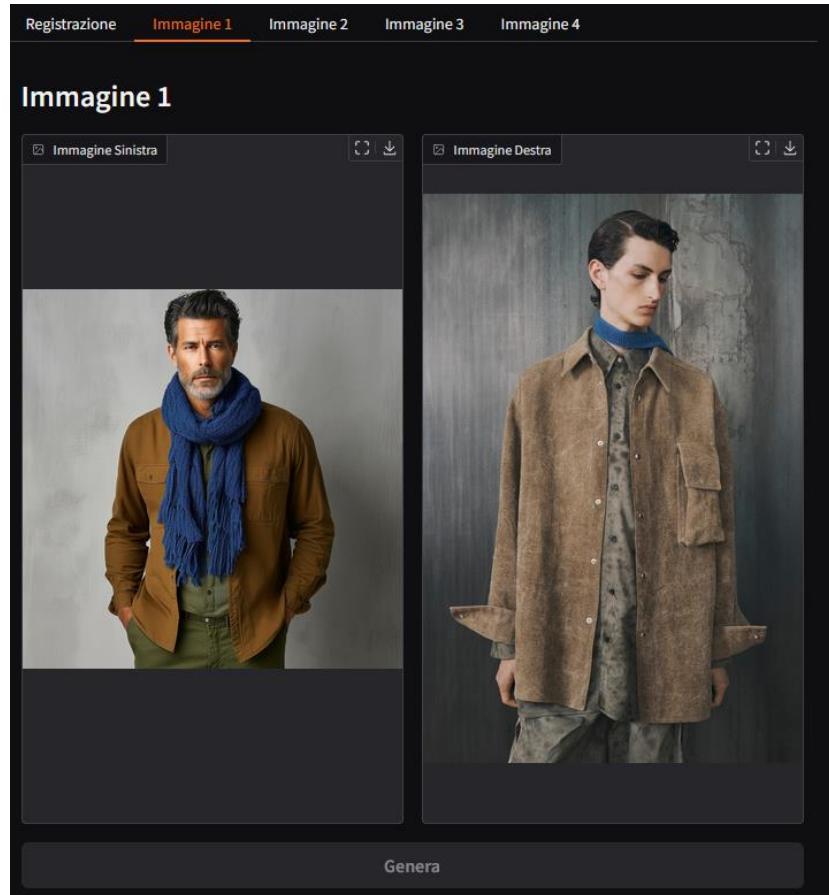


Figura 53: Esempio di schermata di generazione immagini

Valutazione Immagini

Rispondi alle seguenti domande:

- Per le domande a risposta multipla, scegli una delle seguenti opzioni: Destra, Sinistra o Indifferente.
- Per le domande con valutazione numerica, utilizza la seguente scala Likert:
1 = Molto Poco, 2 = Poco, 3 = Neutro, 4 = Abbastanza, 5 = Molto

Personalizzazione: Quale delle due immagini ritieni sia più vicina al tuo stile personale?

Sinistra Destra Indifferente

Intenzione d'Acquisto: Quale immagine ti aiuterrebbe di più a decidere se acquistare il capo?

Sinistra Destra Indifferente

Realismo: Quale immagine appare più realistica come rappresentazione del capo?

Sinistra Destra Indifferente

Appeal Visivo: Quale immagine trovi esteticamente più convincente per indossare il capo?

Sinistra Destra Indifferente

Fiducia: Quale immagine ti dà maggiore sicurezza che questo capo ti donerebbe?

Sinistra Destra Indifferente

Differenziazione: Le due immagini ti sembrano chiaramente distinguibili l'una dall'altra?

1 2 3 4 5

Invia Valutazione

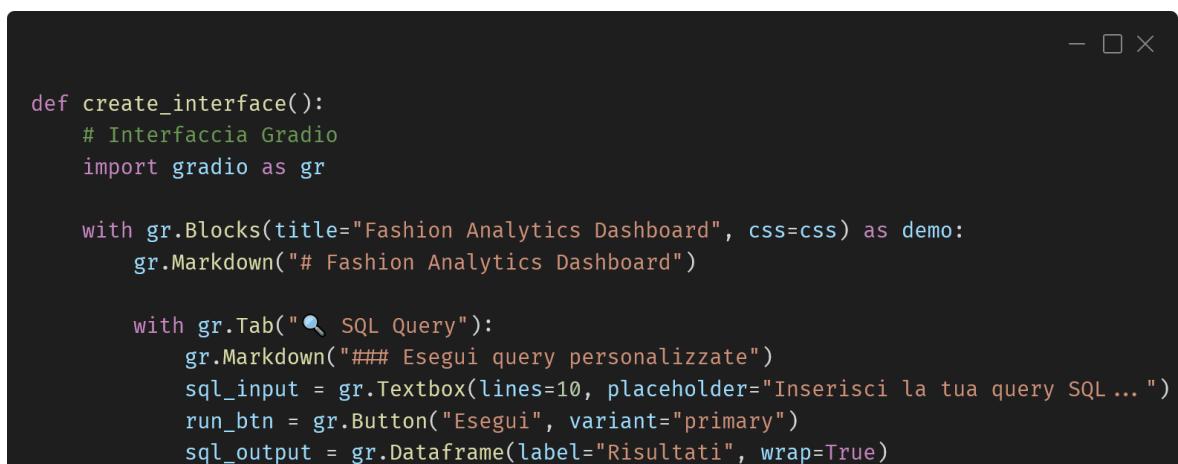
Figura 54: Esempio di questionario di valutazione utente

4.8.3 Interfaccia Super-User: Analisi e Valutazione dei Dati

L'interfaccia Super-User costituisce la componente analitica avanzata della piattaforma e consente di esplorare, interrogare e visualizzare in maniera interattiva i dati raccolti durante la fase sperimentale. È realizzata anch'essa tramite Gradio, ma si distingue per la ricchezza funzionale e per la presenza di numerose tab dedicate ad analisi statistiche, interrogazioni SQL e confronti metrici tra modelli.

L'interfaccia è costruita all'interno della funzione `create_interface()`, che definisce l'intera struttura grafica e collega i componenti visivi con le funzioni di backend.

Anche in questo caso sono stati utilizzati i container `gr.Blocks` e `gr.Tab` per la realizzazione di questa schermata.



```
def create_interface():
    # Interfaccia Gradio
    import gradio as gr

    with gr.Blocks(title="Fashion Analytics Dashboard", css=css) as demo:
        gr.Markdown("# Fashion Analytics Dashboard")

        with gr.Tab("🔍 SQL Query"):
            gr.Markdown("### Esegui query personalizzate")
            sql_input = gr.Textbox(lines=10, placeholder="Inserisci la tua query SQL ... ")
            run_btn = gr.Button("Esegui", variant="primary")
            sql_output = gr.Dataframe(label="Risultati", wrap=True)
```

Figura 55: Codice Iniziale della Pagine per il Super-User

1) Schermata di Esecuzione Query SQL

Nella prima tab viene offerta un'interfaccia semplice per eseguire qualsiasi comando SQL sul database, utilizzando la funzione `run_query()`.

The screenshot shows the 'Fashion Analytics Dashboard' with the 'SQL Query' tab selected. A text input box contains the following SQL query:

```
SELECT *
FROM registrazioni
WHERE nome LIKE "Ale%"
```

An orange 'Esegui' button is below the input box. Below the button, a results table displays two rows of data:

idUser...	nome	cognome	eta	nazione	gene...	corrente_artistica_prefer...
2	Alessan...	Piergiava...	23	Italia	Uomo	Decostruttivismo
4	Alessan...	Piergiava...	65	Italia	Uomo	Decostruttivismo

Figura 56: Prima Schermata dell'interfaccia Super-User con Esempio di Esecuzione Query

2) Schermata di Esecuzioni Query Predefinite

La seconda tab fornisce una serie di pulsanti che richiamano query fisse. Ecco un esempio per la distribuzione per età:

The screenshot shows the 'Fashion Analytics Dashboard' with the 'Analisi Predefinite' tab selected. A section titled 'Report preconfigurati' contains four buttons: 'Distribuzione per Età', 'Utenti per Nazione', 'Preferenze Stilistiche', 'Competenza in Moda', and 'Interesse in Moda'. The 'Distribuzione per Età' button is highlighted. Below these buttons, a results table displays the distribution of users by age group:

eta	Occorrenze
18	11
23	1
39	1
65	2

Figura 57: Seconda Schermata dell'interfaccia Super-User con Funzionamento del Bottone "Distribuzione per Età"

Ogni pulsante (*gr.Button*) è collegato tramite callback Python che eseguono dinamicamente la funzione *run_query*.

```
SELECT eta, COUNT(*) as Occorrenze
FROM registrazioni
GROUP BY eta ORDER BY eta
```

Figura 58: Esempio di Query SQL Preimpostata

3) Sezione Analisi Dettagliata

La sezione “Analisi Dettagliata” è una delle più sofisticate e include due macro-componenti:

- **SEZ. 1:** Analisi delle risposte alle domande da 1 a 6, con visualizzazione dinamica dei testi delle domande e relativi grafici.
- **SEZ. 2:** Analisi delle performance delle immagini baseline.

La funzione *analyze_preferences()* costruisce i grafici relativi alle prime 6 domande del questionario:

```
query = f'''
SELECT
    q.domanda{domanda} as risposta,
    COUNT(*) as totale,
    r.genere
FROM questionario q
JOIN registrazioni r ON q.idUtente = r.idUtente
...
...
df = pd.read_sql_query(query, conn)
fig = px.bar(df, x='risposta', y='totale', color='genere',
... )
```

Figura 59: Porzione della Funzione *analyze_preferences()*

Per ogni domanda, viene mostrato un grafico a barre segmentato per genere dell’utente. La domanda 6, che utilizza una scala Likert, viene visualizzata sia come grafico a torta sia come grafico a barre:

```

fig_pie = px.pie(df_likert, values='totale',
names='valutazione')
fig_bar = px.bar(df_likert, x='valutazione', y='totale',
color='genere')

```

Figura 60: Creazione dei grafici a Torta e a Barre per la Domanda Likert

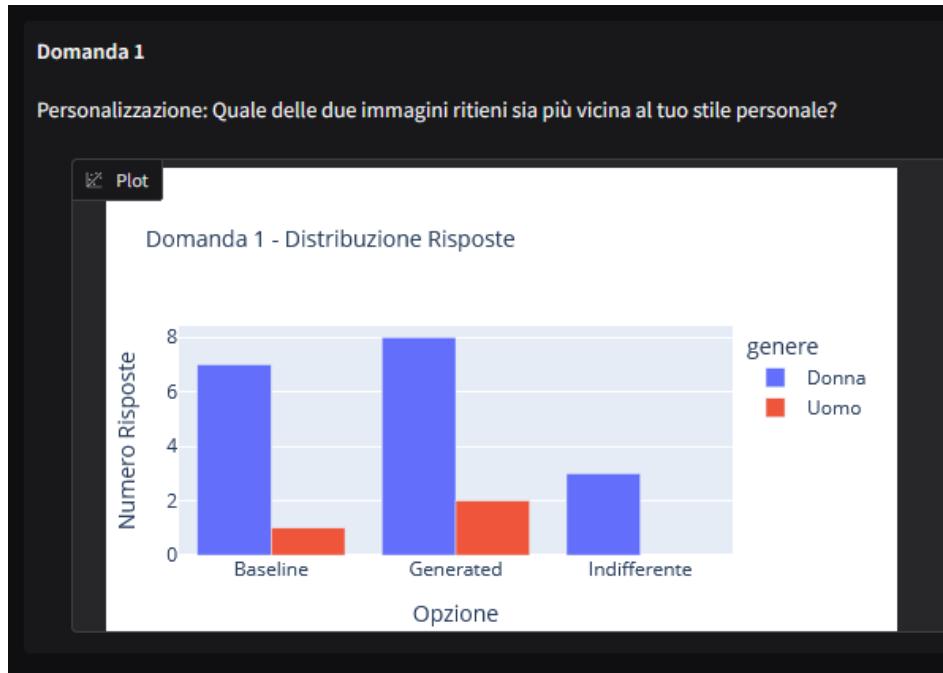


Figura 61: Esempio di Grafico a Barre Relativo alla Domanda 1

La seconda sezione invece viene accompagnata da un grafico a torta interattivo che mostra la distribuzione delle risposte "Baseline", "Generated", "Indifferente" per ogni immagine di Baseline. I dati sono recuperati mediante query SQL, il codice completo è consultabile nella sezione Appendice **C.1 Query Usata nella Sezione 2 della 3° Schermata - “Analisi Performances Immagini Baseline”**.

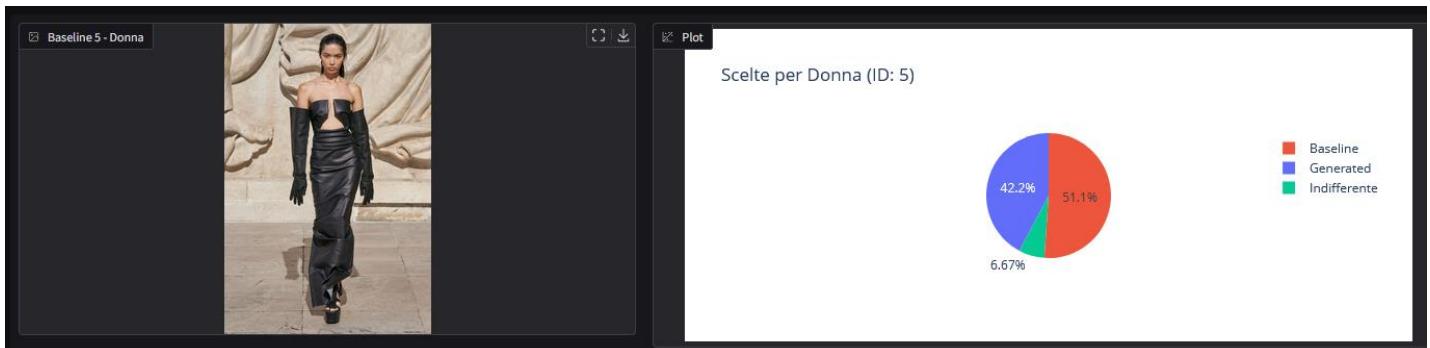


Figura 62: Esempio di Grafico a Torta per l’Immagine Baseline 5

```
SELECT
    'Baseline' as scelta,
    SUM(CASE WHEN TRIM(q.domanda1) = 'Baseline' THEN 1 ELSE 0 END + ... )
FROM ...
```

Figura 63: Porzione della Query Utilizzata per Ottenere i Risultati per la Sezione "Analisi per Immagine Baseline"

I dati vengono aggregati e trasformati in percentuali prima di essere passati a `plotly.express.pie()`.

4) Sezione Efficacia Immagini Baseline

La tab Performance Baseline consente di valutare, per ciascuna immagine di baseline, quanto efficacemente venga superata dalle rispettive immagini generate, calcolando un *success rate* percentuale basato sulle preferenze espresse nelle cinque domande comparative del questionario. I dati, ottenuti tramite query SQL aggregata, vengono presentati sia in tabella che attraverso un grafico a barre interattivo. Il codice completo della query SQL è consultabile nella sezione Appendice **C.2 Query Usata nella Sezione 4 “Efficacia Immagini Baseline”**.

Questa sezione fornisce un'indicazione chiara e sintetica dei capi baseline più facilmente migliorabili dal sistema generativo, aggiungendo una prospettiva comparativa globale non presente in altre analisi.

Efficacia Immagini Baseline

Analizza quanto efficacemente ogni immagine baseline genera preferenze rispetto alle sue varianti generate

- Metrica: % preferenze per immagini generate rispetto alla baseline corrispondente

[Aggiorna Analisi](#)

Risultati Performance

idImmagine	genere_del_capo	success_rate	total_valutazioni
7	Donna	0.7	2
1	Uomo	0.67	3
5	Donna	0.42	9
6	Donna	0.34	7
4	Uomo	0	0
3	Uomo	0	0
2	Uomo	0	0
8	Donna	0	0

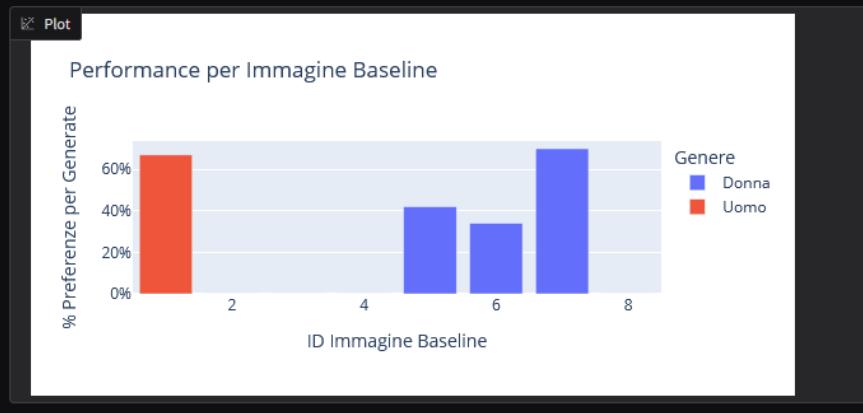


Figura 64: Esempio di Esecuzione della Schermata "Efficacia Immagini Baseline"

Sezioni Successive

Come anticipato nel capitolo **3.8 Metriche di Valutazione**, le tab successive includono analisi più complesse. Per garantire una trattazione coerente e focalizzata, i dettagli implementativi relativi alle ultime quattro tab dell'interfaccia Super-User — dedicate alle **metriche di valutazione** (correlazioni lineari, regressione, modello Bradley-Terry standard e pesato) — non vengono approfonditi in questa sezione. Tali metriche costituiscono infatti il fulcro dell'analisi quantitativa del sistema e sono descritte in maniera estesa nel **Capitolo 4.10 Implementazione delle Metriche di Valutazione**, dove ne vengono illustrate le basi teoriche, la logica computazionale e l'integrazione all'interno della piattaforma.

Ulteriori Considerazioni sull'Implementazione

Ogni componente visuale è legato a una **funzione di backend Python** che effettua operazioni sul database SQLite in tempo reale. L'uso di *plotly*, *seaborn* e *pandas* consente una manipolazione flessibile dei dati e una visualizzazione chiara anche per utenti non esperti.

Inoltre, l'interfaccia è **dinamicamente aggiornata** tramite callback (*click(...)*) e i dati sono sempre sincronizzati grazie al caricamento del database all'avvio.

4.9 Approfondimenti Tecnici sulle Logiche del Sistema

In questo capitolo vengono analizzate nel dettaglio alcune delle soluzioni implementative più complesse e significative che regolano il comportamento della web app dal punto di vista funzionale e interattivo. Tali meccanismi non si limitano alla presentazione visiva dei contenuti, ma coinvolgono aspetti critici della logica applicativa, della gestione della coerenza nei dati e del controllo del flusso di navigazione tra le varie sezioni dell'interfaccia utente.

Vengono quindi approfondite: la gestione della randomizzazione delle immagini (sia a livello grafico che logico), il caricamento dinamico delle domande tramite file esterni, i controlli sequenziali per il corretto avanzamento dell'utente tra le diverse fasi del sistema, e altre logiche chiave che garantiscono la robustezza e la scalabilità dell'applicazione.

4.9.1 Gestione della Randomizzazione delle Immagini

Uno degli aspetti più delicati nella progettazione della valutazione delle immagini generate è stato quello di **eliminare qualsiasi forma di bias percettivo** derivante dall'**ordine di presentazione**. La posizione di un'immagine (sinistra vs destra) può infatti influenzare inconsciamente le scelte dell'utente. Per questa ragione, il sistema implementato prevede una **randomizzazione controllata** della posizione delle immagini (baseline vs generata), che avviene durante ogni singolo processo di generazione. È

inoltre, importante gestire e mappare la risposta del questionario con l'immagine corretta in base alla disposizione della tab.

Obiettivo e Strategia

Per ogni generazione, il sistema assegna casualmente l'immagine generata o quella baseline alla posizione sinistra o destra. Il fine è evitare che un'immagine, presentata costantemente sullo stesso lato, venga scelta più o meno frequentemente per abitudine visiva.

La randomizzazione è gestita nel **backend**, attraverso l'uso di una variabile globale denominata *DISPLAY_ORDERS*, che memorizza per ogni tab (numerata da 1 a 4) la posizione assegnata.

All'interno della funzione *save_generated_image(tab_number)*, che si occupa della generazione dell'immagine a partire dal prompt personalizzato, viene eseguito il seguente codice:

```
import random

# Assegna una posizione randomica all'immagine generata
display_order = random.randint(1, 2) # 1 = baseline sinistra, 2 = generated sinistra
DISPLAY_ORDERS[tab_number] = display_order

# Assegnazione delle immagini in base all'ordine estratto
if display_order == 1:
    left_image, right_image = baseline_path, generated_path
else:
    left_image, right_image = generated_path, baseline_path
```

Figura 65: Codice usato per determinare l'ordine di visualizzazione delle immagini

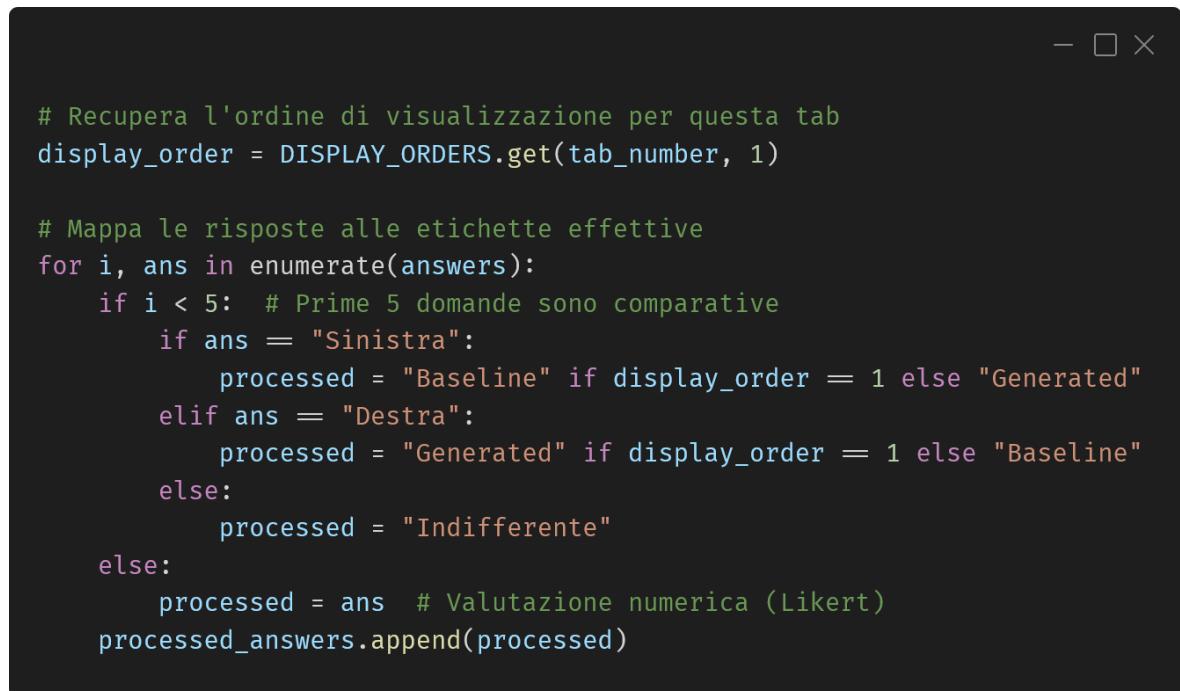
La scelta viene **salvata in modo permanente** all'interno della sessione utente grazie alla variabile globale *DISPLAY_ORDERS*, che permette di mantenere coerenza tra:

- la **visualizzazione delle immagini** al momento della generazione,
- e la **decodifica delle risposte** nel momento in cui viene compilato il questionario.

Integrazione con il Backend del Questionario

Durante la compilazione del questionario da parte dell'utente, le risposte “Sinistra” o “Destra” non sono salvate così come sono. Al contrario, il sistema traduce tali risposte in funzione dell'ordine visualizzato, recuperato dalla mappa *DISPLAY_ORDERS*.

Ad esempio, la seguente porzione di codice nella funzione *handle_questionnaire* mostra questa decodifica dinamica:



```
# Recupera l'ordine di visualizzazione per questa tab
display_order = DISPLAY_ORDERS.get(tab_number, 1)

# Mappa le risposte alle etichette effettive
for i, ans in enumerate(answers):
    if i < 5: # Prime 5 domande sono comparative
        if ans == "Sinistra":
            processed = "Baseline" if display_order == 1 else "Generated"
        elif ans == "Destra":
            processed = "Generated" if display_order == 1 else "Baseline"
        else:
            processed = "Indifferente"
    else:
        processed = ans # Valutazione numerica (Likert)
    processed_answers.append(processed)
```

Figura 66: Decodifica dinamica della posizione delle immagini per le risposte del questionario

In questo modo, ogni risposta viene **interpretata correttamente in termini di preferenza reale**: non importa se l'utente ha cliccato “Destra”, ciò che conta è **quale immagine occupava quella posizione** nella sessione corrente.

Coerenza e Validazione

La randomizzazione avviene **una sola volta per ogni tab** e viene bloccata successivamente, evitando che l'utente rigeneri l'immagine modificando l'ordine delle immagini. Questo è garantito dal controllo:

```
if GENERATED_TABS.get(tab_number, False):
    raise gr.Error("Hai già generato un'immagine per questa tab. Procedi con la valutazione!")
```

Figura 67: Controllo di generazione immagini

Ciò garantisce che ogni valutazione sia:

- eseguita **su un confronto statico**
- **consistente** tra visualizzazione e salvataggio dei dati.

Vantaggi della Randomizzazione Backend-Driven

Implementare questa logica nel backend piuttosto che nel frontend puro (es. tramite JavaScript dinamico) garantisce:

- **tracciabilità** del comportamento utente,
- **riproducibilità** delle valutazioni,
- **coerenza nelle analisi statistiche**, in particolare nei modelli Bradley-Terry, dove il significato delle risposte è fondamentale per il calcolo della forza relativa delle immagini.

4.9.2 Caricamento Dinamico delle Domande

La gestione delle **domande del questionario** associate a ciascuna immagine generata è fondamentale per il corretto funzionamento del sistema. Per garantire **flessibilità, scalabilità e facilità di aggiornamento**, è stato implementato un meccanismo di **caricamento dinamico** delle domande, basato su un file .csv e sull'utilizzo di un identificatore di set (*id_set*).

Obiettivo e Strategia

L'intento è quello di:

- permettere agli amministratori di sistema di aggiornare le domande direttamente da file, senza intervenire sul codice sorgente;
- gestire facilmente versioni diverse di questionario (es. per test A/B o studi longitudinali);

- separare chiaramente il contenuto (le domande) dalla logica dell’interfaccia utente, aderendo al principio di separazione delle responsabilità.

Il codice di caricamento si occupa di leggere il file CSV e restituire un elenco di stringhe contenenti le domande, filtrando secondo l’`id_set` specificato.

```
def load_questions(file_path, question_set_id=QUESTION_SET_ID):
    try:
        df = pd.read_csv(file_path)
        # Filtra per id_set e seleziona la riga corrispondente
        selected = df[df["id_set"] == question_set_id]
        if selected.empty:
            return ["Domanda 1", "Domanda 2", "Domanda 3"] # fallback

        # Estrae tutte le colonne a partire dalla seconda, ignora i NaN
        questions = selected.iloc[0, 1: ].dropna().tolist()
        return [q.strip() for q in questions if q.strip()]

    except FileNotFoundError:
        return ["Domanda 1", "Domanda 2", "Domanda 3"]
    except Exception as e:
        print(f"Errore nel caricamento delle domande: {str(e)}")
        return ["Domanda 1", "Domanda 2", "Domanda 3"]
```

Figura 68: Codice per il caricamento delle domande

Spiegazione del Funzionamento

- `pd.read_csv(file_path)`: carica il contenuto del file CSV in un DataFrame.
- `df[df["id_set"] == question_set_id]`: filtra le righe che corrispondono al set desiderato.
- `selected.iloc[0, 1:].dropna()`: seleziona la prima riga (in caso di più righe duplicate) e ne estrae solo le colonne delle domande.
- `strip()`: rimuove eventuali spazi o ritorni a capo.

In caso di errori (file mancante o struttura alterata), il sistema è in grado di gestire autonomamente il problema mostrando un set predefinito di domande di fallback.

Vantaggi della Soluzione Adottata

1. **Flessibilità:** è possibile cambiare completamente il contenuto del questionario senza ricompilare o modificare il codice.
2. **Manutenibilità:** il contenuto è disaccoppiato dalla logica, rendendo l'aggiornamento più semplice anche per chi non ha competenze di programmazione.
3. **Espandibilità:** in futuro sarà possibile implementare selezioni di set diversi in base al tipo di utente, alla sessione, alla nazione o altri fattori.

Collegamento al Sistema

Il set di domande caricato viene poi utilizzato per generare dinamicamente i componenti di Gradio nella sezione dell'interfaccia relativa al questionario.

```
- □ ×  
  
questions = load_questions(questions_file, QUESTION_SET_ID)  
  
for i, q in enumerate(questions):  
    if i < 5:  
        inputs.append(gr.Radio(  
            label=q,  
            choices=["Sinistra", "Destra", "Indifferente"]  
        ))  
    else:  
        inputs.append(gr.Radio(  
            label=q,  
            choices=[  
                (1, "1 - Molto Poco"),  
                (2, "2 - Poco"),  
                (3, "3 - Neutro"),  
                (4, "4 - Abbastanza"),  
                (5, "5 - Molto")  
            ]  
        ))
```

Figura 69: Codice per la generazione dinamica delle domande nella sezione questionario

4.9.3 Controlli Sequenziali sulle Tab (Blocco Avanzamento)

Per garantire la **correttezza del flusso di interazione** e la **coerenza dei dati raccolti**, l'applicazione implementa una logica di **blocco sequenziale** che impedisce all'utente di proseguire alla generazione dell'immagine successiva se non ha prima completato correttamente la valutazione precedente.

Questa misura è essenziale in un contesto sperimentale, poiché:

- evita la raccolta di dati incompleti o inconsapevoli;
- obbliga l'utente a rispettare l'ordine delle fasi previste;
- preserva l'affidabilità statistica dei confronti generati.

Funzionamento della logica di blocco

All'interno della *funzione save_generated_image(tab_number)*, ogni volta che un utente tenta di generare una nuova immagine, vengono eseguiti i seguenti controlli:

1) Verifica dell'avvenuta registrazione

Se l'utente non ha ancora effettuato la registrazione iniziale, viene bloccato con un messaggio d'errore:

```
if not CURRENT_USER_ID:  
    raise gr.Error("Effettua prima la registrazione!")
```

Figura 70: Messaggio d'errore per mancata registrazione

2) Blocco delle Generazioni Ripetute

Se l'immagine per quella specifica tab è già stata generata, si impedisce una nuova generazione:

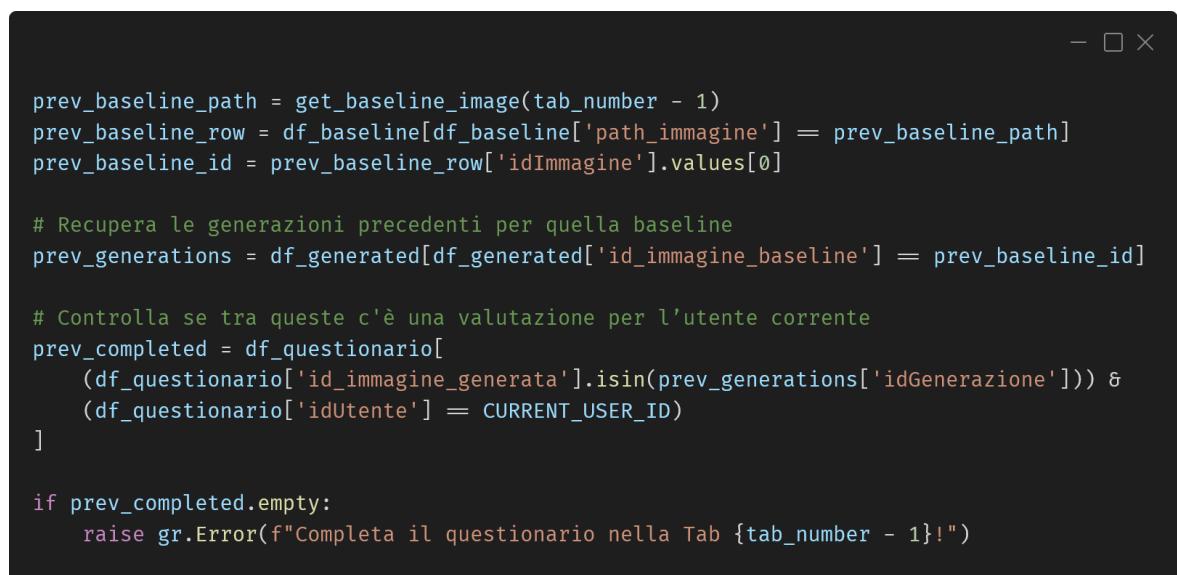
```
if GENERATED_TABS.get(tab_number, False):  
    raise gr.Error("Hai già generato un'immagine per questa tab. Procedi con la valutazione!")
```

Figura 71: Messaggio d'errore per generazioni multiple per la stessa tab

3) Controllo sul completamento della tab precedente

A partire dalla seconda immagine ($tab_number > 1$), viene verificato che la tab precedente sia stata **generata e valutata**:

- Si recupera l'ID dell'immagine baseline della tab precedente.
- Si controlla che per quell'immagine sia stata effettuata una generazione.
- Si verifica se esiste nel dataset *questionario.csv* una risposta relativa all'immagine generata precedente dallo stesso utente.
- Se uno di questi passaggi fallisce, l'avanzamento viene bloccato.



```
prev_baseline_path = get_baseline_image(tab_number - 1)
prev_baseline_row = df_baseline[df_baseline['path_immagine'] == prev_baseline_path]
prev_baseline_id = prev_baseline_row['idImmagine'].values[0]

# Recupera le generazioni precedenti per quella baseline
prev_generations = df_generated[df_generated['id_immagine_baseline'] == prev_baseline_id]

# Controlla se tra queste c'è una valutazione per l'utente corrente
prev_completed = df_questionario[
    (df_questionario['id_immagine_generata'].isin(prev_generations['idGenerazione'])) &
    (df_questionario['idUser'] == CURRENT_USER_ID)
]

if prev_completed.empty:
    raise gr.Error(f"Completa il questionario nella Tab {tab_number - 1}!")
```

Figura 72: Pipeline di controllo sul completamento della tab precedente

Questo controllo previene che un utente, ad esempio, compili solo la terza e la quarta valutazione saltando le prime due, garantendo così una **sequenza strutturata e completa** nel flusso sperimentale.

Gestione Centralizzata dei Flag

Per monitorare quali tab sono già state processate, viene utilizzata la variabile globale *GENERATED_TABS*, un dizionario che associa il numero della tab a un booleano:

```
GENERATED_TABS = {}  
...  
GENERATED_TABS[tab_number] = True
```

Figura 73: Utilizzo della variabile globale GENERATED_TABS

Questa struttura permette un **tracciamento efficace** e aggiornato dello stato di ogni tab, utile anche per nascondere o disabilitare pulsanti o elementi nell'interfaccia.

Messaggi di Errore Mirati

Nel caso in cui uno dei controlli fallisca, il sistema mostra un messaggio d'errore specifico, evitando comportamenti ambigui.

```
raise gr.Error(f"Genera prima l'immagine nella Tab {tab_number - 1}!")
```

Figura 74: Esempio di messaggio d'errore mirato

Questo consente di **guidare l'utente** nel processo senza compromettere la qualità dei dati raccolti.

4.9.4 Query Complesse nella Super-User Interface

All'interno della sezione avanzata dell'interfaccia Super-User, alcune analisi richiedono l'uso di query SQL complesse che incrociano più tabelle e calcolano aggregati condizionati.

Un esempio è rappresentato dalla sezione 2 della tab "Analisi Dettagliata", in cui è necessario ottenere tutte le risposte di tutti gli utenti da ogni questionario per ogni immagine di baseline.

La query utilizzata per ciascuna immagine esegue tre conteggi separati (risposte per "Generated", "Baseline" e "Indifferente") sfruttando costrutti CASE WHEN all'interno della clausola SUM, applicata alle cinque domande comparative. Questo approccio consente di computare dinamicamente il

numero totale di preferenze per ciascuna categoria, fornendo dati precisi che alimentano i grafici e le tabelle in output. L'impiego di UNION ALL permette di unificare i risultati in una sola struttura dati facilmente manipolabile in pandas e visualizzabile in Plotly.

Il codice completo è consultabile in Appendice **C.1 Query Usata nella Sezione 2 della 3° Schermata - “Analisi Performances Immagini Baseline”**.

4.9.5 Caricamento Dinamico delle Domande nella Super-User Interface Mediante SetID

Nella sezione dedicata all'analisi delle risposte, la Super-User Interface carica dinamicamente i testi delle domande relative al questionario visualizzato, in base all'identificativo *id_set* usato nella sessione corrente. Questo avviene tramite una query SQL selettiva:

```
query_domande = f'''SELECT * FROM domande WHERE id_set = {ID_SET}'''  
df_domande = pd.read_sql_query(query_domande, conn)
```

Figura 75: Codice Usato per Caricare le Domande Dinamicamente

Il contenuto viene poi trasformato in dizionario e associato dinamicamente alle rispettive componenti Gradio. Questa logica permette al sistema di supportare più versioni del questionario in parallelo, garantendo coerenza tra i dati salvati e quelli visualizzati.

4.9.6 Assegnazione delle Immagini Baseline in base al Genere

Per garantire coerenza e personalizzazione nei test, il sistema assegna automaticamente ad ogni utente immagini baseline coerenti con il proprio *genere dichiarato* in fase di registrazione. Il meccanismo è implementato tramite una query mirata nel backend, che filtra il caricamento delle immagini in base al campo *genere_del_capo* nella tabella *immagini_baseline*.

```
SELECT * FROM immagini_baseline WHERE genere_del_capo = ?
```

- □ ×

Figura 76: Query per Recupare le Immagini di Baseline in Base al Genere

Questa logica evita mismatch semantici tra utente e contenuto visivo, mantenendo alta la pertinenza delle valutazioni e l'efficacia dell'esperimento. Inoltre, favorisce una maggiore omogeneità nelle analisi comparative tra gruppi di utenti.

4.10 Implementazione delle Metriche di Valutazione

Nel presente capitolo viene descritto in dettaglio il processo di implementazione delle quattro metriche statistiche adottate per l'analisi delle preferenze espresse dagli utenti all'interno della web app. Le metriche, introdotte teoricamente nel Capitolo **3.8 Metriche di Valutazione**, sono state tradotte in codice Python all'interno dell'interfaccia Super-User del sistema insieme alle altre funzionalità enunciate nel Capitolo **4.8.3 Interfaccia Super-User: Analisi e Valutazione dei Dati**, in modo tale da renderne interattiva e dinamica l'esecuzione. Tutte le metriche lavorano su dati memorizzati in un database SQLite, aggiornato automaticamente tramite script Python che caricano i CSV alla fase iniziale dell'esecuzione.

4.10.1 Correlazioni Lineari Semplici

L'obiettivo di questa metrica è valutare l'esistenza di una correlazione lineare tra il tasso di preferenza verso immagini generate (*preference_rate*) e alcune caratteristiche utente (*età*, *competenza* e *interesse nella moda*).

Nel codice Python, l'analisi avviene attraverso il calcolo del *coefficiente di correlazione di Pearson* tra la variabile *preference_rate* e ciascuna delle seguenti variabili:

- *età*
- *competenza_moda*
- *interesse_moda*

La funzione recupera i dati necessari mediante query SQL e procede eseguendo i seguenti passaggi:

```
corr = merged[['preference_rate', 'eta', 'competenza_moda',  
'interesse_moda']].corr()  
result_df = corr.loc[['preference_rate'], ['eta', 'competenza_moda',  
'interesse_moda']].T
```

Figura 77: Porzione 1 del Codice per Correlazioni Semplici

Successivamente, i risultati vengono mostrati sia in **forma tabellare** che come **heatmap** interattiva tramite *seaborn* e *matplotlib*:

```
fig, ax = plt.subplots(figsize=(3.5, 2.8))
sns.heatmap(corr, annot=True, fmt=".2f", cmap="coolwarm", ax=ax)
```

Figura 78: Porzione 2 del Codice per Correlazioni Semplici

Questa visualizzazione fornisce un riscontro immediato sull'intensità e la direzione delle correlazioni.

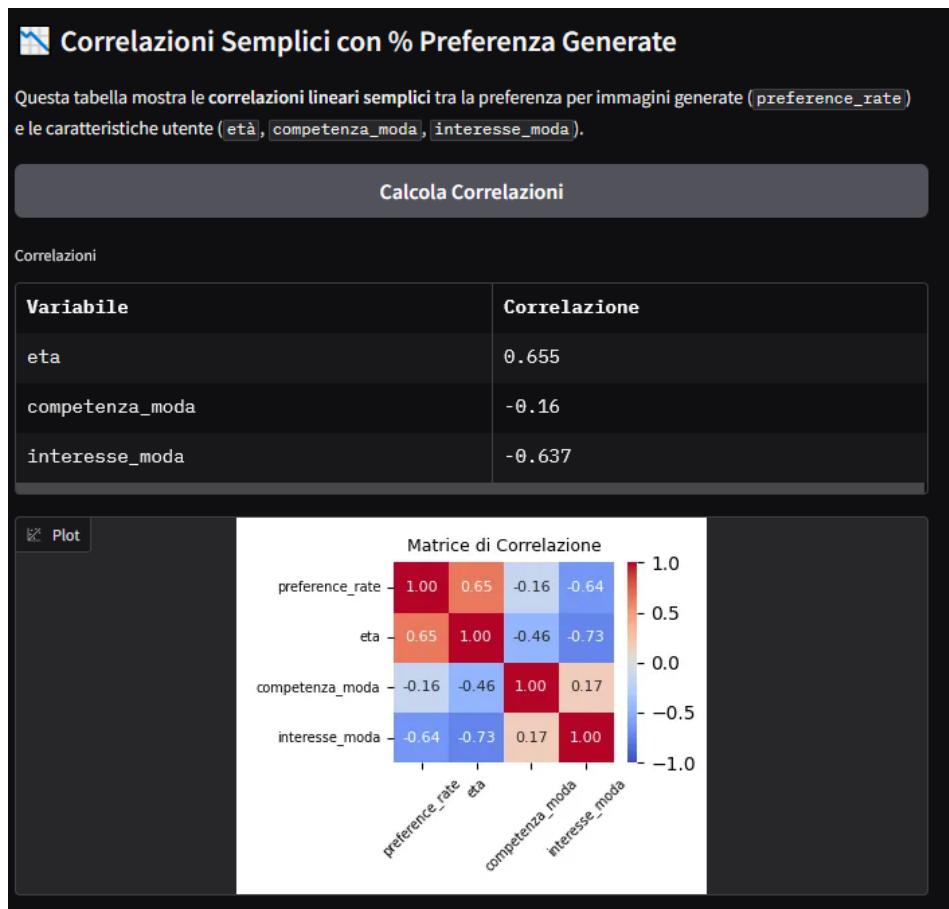


Figura 79: Esempio di Esecuzione della Schermata per le Correlazioni Semplici

4.10.2 Regressione Lineare (OLS)

La seconda metrica implementata è la regressione lineare OLS (*Ordinary Least Squares*), utilizzata per modellare la relazione tra la *preference_rate* e le variabili esplicative:

- *Età (eta)*
- *Competenza nella moda (competenza_moda)*
- *Interesse nella moda (interesse_moda)*

Dopo aver ottenuto i dati mediante apposita query SQL, la regressione è implementata con la libreria *statsmodels*:

```
- □ ×  
  
X = merged[["eta", "competenza_moda", "interesse_moda"]].astype('float64')  
X = sm.add_constant(X, has_constant='add')  
y = merged['preference_rate'].clip(0, 1)  
  
model = sm.OLS(y, X)  
result = model.fit()
```

Figura 80: Porzione di Codice per Implementare la Regressione Lineare OLS

Di seguito, il sistema mostra:

- Una **tabella con coefficienti, p-value e significatività**
- Un **riepilogo testuale del modello**
- Un **commento automatico** sul significato del coefficiente di determinazione R²:

Regressione Lineare - Preferenze Generate

Analizza come le caratteristiche utente (età, competenza, interesse) influenzano il tasso di preferenza verso le immagini generate.

- **Variabili indipendenti:** Età, Competenza moda, Interesse moda
- **Variabile dipendente:** % preferenze per immagini generate

Esegui Analisi

Risultati Regressione

Variabile	Coefficiente	P-value	Significatività
const	0.8049	0.5767	
eta	0.0119	0.2829	
competenza_moda	0.0393	0.6961	
interesse_moda	-0.1646	0.4664	

Figura 81: Esempio di Esecuzione

4.10.3 Modello Bradley-Terry (Standard)

Questa metrica consente di stimare la **probabilità relativa di vittoria** tra le immagini generate e quelle baseline, sulla base dei confronti diretti raccolti nelle prime cinque domande del questionario.

Il modello viene implementato tramite la libreria *choix*:

```
from choix import ilsrsr_pairwise
params = ilsrsr_pairwise(n_items=2, data=expanded, alpha=0.01)
```

Figura 82: Porzione 1 di Codice per Implementare Bradley Terry

I dati *expanded* sono coppie (*winner*, *loser*) costruite a partire dalle risposte:

```
if risposta == "Generated":
    indexed_data.append((1, 0))
elif risposta == "Baseline":
    indexed_data.append((0, 1))
```

Figura 83: Figura 64: Porzione 2 di Codice per Implementare Bradley Terry

Il punteggio finale viene trasformato in **probabilità di preferenza** tramite softmax esponenziale:

```

exp_params = np.exp(params)
probs = exp_params / exp_params.sum()

```

Figura 84: Porzione 3 di Codice per Implementare Bradley Terry

I risultati sono visualizzati in una tabella e tramite un grafico a barre che mostra chiaramente quale immagine (baseline o generata) risulta globalmente preferita.

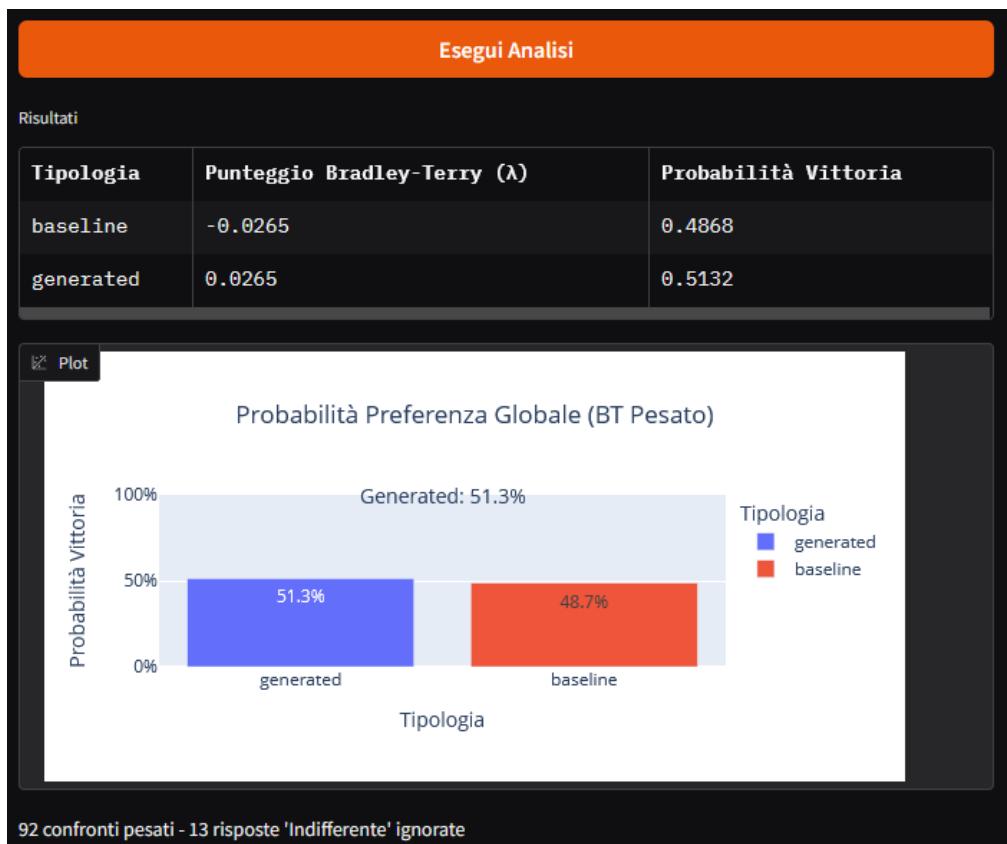


Figura 85: Esempio di Esecuzione del Modello Bradley-Terry Standard

4.10.4 Modello Bradley-Terry Pesato

Una variante del modello precedente introduce un **peso utente** basato sulla formula:

$$peso = 0.65 * competenza + 0.35 * interesse$$

Ogni confronto è replicato un numero di volte proporzionale al peso, aumentando l'influenza degli utenti più competenti o interessati:

```
for winner, loser, weight in indexed_data:  
    reps = max(1, int(round(weight)))  
    expanded.extend([(winner, loser)] * reps)
```

Figura 86: Porzione di codice di Bradley-Terry pesato

Le restanti fasi sono analoghe al modello standard: si applica *ilsr_pairwise*, si calcola la *softmax*, si visualizza il confronto.

Infine, nella sezione di confronto tra i modelli viene mostrato un confronto diretto tra modello pesato e non pesato tramite una tabella comparativa e grafico interattivo.

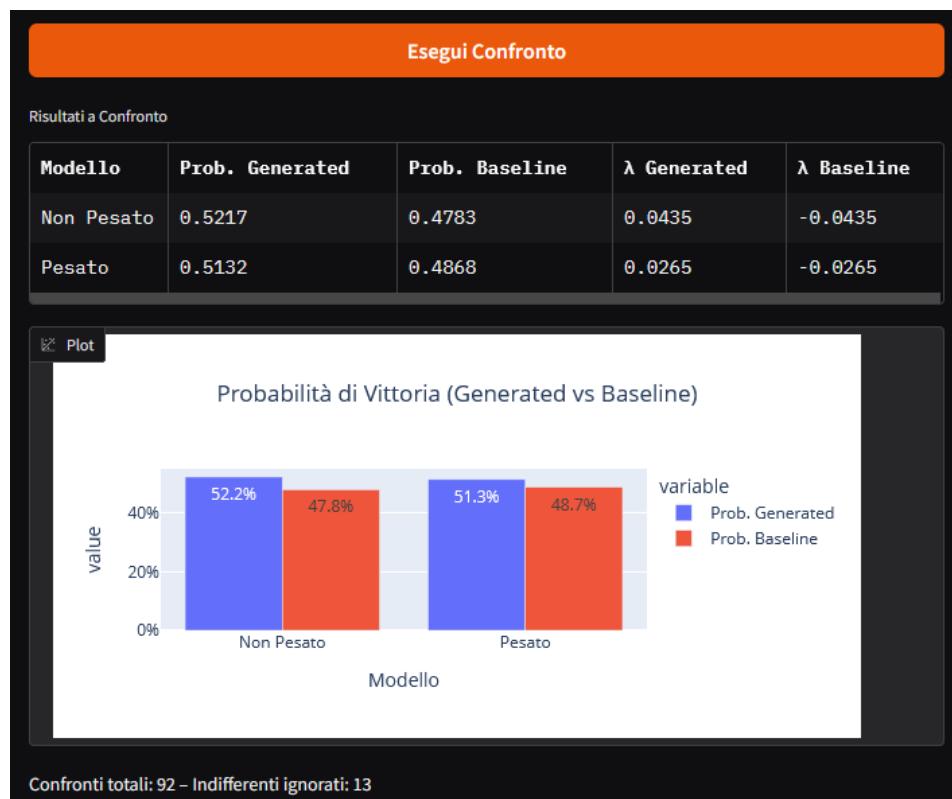


Figura 87: Esempio di Esecuzione del Confronto tra i Modelli

4.11 Considerazioni Finali sullo Sviluppo

L'intera fase di implementazione ha portato alla realizzazione di un sistema solido, modulare e facilmente estendibile, capace di integrare sinergicamente le componenti di generazione, valutazione e analisi delle immagini pubblicitarie personalizzate. Tra i principali punti di forza del progetto emerge innanzitutto **l'integrazione fluida tra modelli di intelligenza artificiale multimodale** e un'interfaccia web interattiva, resa possibile grazie all'uso di Gradio, che ha permesso di costruire un'esperienza utente accessibile e funzionale.

Altro aspetto di rilievo è la **flessibilità architettonica**: il sistema è progettato per operare sia in contesti locali che remoti, grazie alla gestione delle chiamate API e al caricamento dinamico di dati tramite file CSV. Questa struttura rende il prototipo facilmente adattabile a nuovi contesti d'uso o scenari sperimentali.

Dal punto di vista della **personalizzazione**, la pipeline consente di raccogliere un profilo utente dettagliato e di trasformarlo in contenuti visivi altamente specifici, valorizzando l'aspetto creativo dell'AI generativa. Inoltre, l'integrazione di una **dashboard analitica per super-user** consente una valutazione quantitativa rigorosa, grazie a un set di metriche statistiche che conferisce profondità e validità all'analisi dei dati.

Infine, la struttura dati basata su **SQLite** e la gestione accurata del ciclo di vita dell'esperimento garantiscono tracciabilità, integrità e scalabilità, pur mantenendo un approccio leggero e facilmente distribuibile.

Nel complesso, il sistema implementato rappresenta un equilibrio efficace tra tecnica, creatività e interazione, ponendo solide basi per futuri sviluppi nel campo della pubblicità personalizzata e dell'analisi delle preferenze tramite intelligenza artificiale.

5. Sperimentazione e Risultati

5.1 Protocollo Sperimentale

Modalità di svolgimento dell'esperimento

In questo capitolo vengono presentate le modalità attraverso cui è stata condotta la sperimentazione del sistema sviluppato, nonché i risultati emersi dalle analisi effettuate. L'obiettivo dell'esperimento è stato quello di raccogliere dati significativi attraverso l'interazione con un campione eterogeneo di utenti, in modo da valutare la qualità e l'efficacia del sistema in contesti realistici e diversificati.

Per garantire una visione d'insieme il più possibile completa, si è scelto di coinvolgere utenti con caratteristiche differenti in termini di età, genere, gusti estetici e livello di competenza nel settore della moda. Questa varietà ha permesso di valutare il comportamento del sistema rispetto a preferenze individuali molto diverse, offrendo così una base solida per le analisi successive.

Il reclutamento dei partecipanti è avvenuto in due modalità: da un lato, coinvolgendo persone appartenenti alla propria rete personale di conoscenze; dall'altro, conducendo attività di sperimentazione sul campo, avvicinando potenziali utenti in diverse aree della città e invitandoli a testare il sistema. Prima di iniziare, a ciascun partecipante è stato illustrato il funzionamento del sistema e gli obiettivi della sperimentazione, così da garantire una comprensione chiara e consapevole dell'esperienza d'uso. Tutte le interazioni si sono svolte sotto supervisione diretta, ma senza influenzare le scelte degli utenti.

Durante la sessione, ogni utente ha visualizzato e confrontato immagini generate in base al proprio profilo con le immagini di baseline, seguendo un flusso sperimentale guidato. Al termine di ogni generazione, l'utente ha compilato un questionario valutativo relativo alle immagini appena visualizzate. Il processo è proseguito in autonomia fino al completamento dell'intera sequenza di schermate previste per la sperimentazione.

5.2 Risultati

Discussione delle metriche di valutazione

In questo capitolo vengono presentati e discussi i risultati ottenuti e memorizzati all'interno i dataset e del database, attraverso l'impiego di diverse metriche di valutazione, con l'obiettivo di analizzare il comportamento del sistema da più prospettive e fornire una valutazione quantitativa e qualitativa delle performance ottenute.

Le metriche sono state suddivise in due macrocategorie:

- **Metriche descrittive:** comprendono *analisi delle caratteristiche del campione* (età, genere, preferenze) e lo studio delle *risposte ai questionari* e l'*analisi delle performance delle immagini baseline*.
- **Metriche inferenziali e comparative:** includono strumenti statistici e modelli volti a interpretare relazioni e tendenze latenti, come le *correlazioni semplici*, la *regressione lineare*, il *modello di Bradley-Terry standard* e il *modello di Bradley-Terry pesato*.

Inoltre, per approfondimenti ulteriori o per condurre analisi personalizzate sui dati raccolti, è stata prevista un'apposita sezione nella web app super user che consente l'esecuzione di *query SQL dirette sul database*.

Tutte queste metriche concorrono a fornire una panoramica completa sull'efficacia del sistema nel generare contenuti pubblicitari rilevanti e coerenti con le preferenze dell'utente.

5.2.1 Analisi Preliminari del Campione Utente

Questa sezione introduce una serie di analisi esplorative, pensate per descrivere la composizione e le caratteristiche principali del campione di utenti che ha preso parte alla sperimentazione. Tali analisi permettono di ottenere una panoramica generale sulla distribuzione dei partecipanti rispetto a variabili demografiche e attitudinali.

Distribuzione per età

L'età dei partecipanti è distribuita in modo abbastanza variegato, con una presenza consistente nella fascia 20–35 anni, ma anche una copertura che include utenti più maturi. Questo consente al sistema di essere testato su una varietà di prospettive legate alla fascia d'età di appartenenza, il che può influenzare la percezione delle immagini e delle scelte stilistiche. Questo bilanciamento consente di osservare eventuali differenze di percezione in base all'età.

Età	Occorrenze
20 – 29	15
30 – 39	9
40 +	6

Tabella 2: Numero di occorrenze per età

Corrente artistica preferita

Gli utenti hanno espresso una preferenza per varie correnti artistiche come Avant-Garde, Streetwear, Romanticismo, Barocco, ecc. La varietà di questi interessi fornisce un'indicazione utile circa i gusti estetici che guidano la valutazione delle immagini generate e baseline.

Corrente Artistica	Occorrenze
Romanticismo	9
Avant-Garde	5
Streetwear	3
Tribalismo	2
Post-Modernismo	2
Neo-Classicismo	2
Gothic	2
Barocco	2
Pop-Art	1
Minimalismo	1
Eco-Fashion	1

Tabella 3: Numero di occorrenze per corrente artistica

Competenza e interesse per la moda

Sono state raccolte valutazioni soggettive, su scala da 1 a 5, relative alla competenza percepita in ambito moda e all'interesse personale verso il settore. I risultati mostrano una distribuzione equilibrata, con alcuni utenti che si considerano molto competenti e coinvolti, e altri che si avvicinano al test con un approccio più neutro. Questa variabilità è fondamentale per comprendere se e come l'esperienza personale influenzi la preferenza per le immagini generate.

Valutazione Competenza Moda	Occorrenze
1	6
2	3
3	3
4	11
5	7

Tabella 4: Numero di occorrenze per diverse valutazioni di competenza moda

Valutazione Interesse Moda	Occorrenze
1	1
2	4
3	11
4	9
5	5

Tabella 5: Numero di occorrenze per diverse valutazioni di interesse moda

L'osservazione delle due distribuzioni evidenzia un campione ben bilanciato, nel quale convivono sia utenti appassionati e con alta percezione di competenza, sia partecipanti con livelli più bassi di coinvolgimento nel settore moda. Questo bilanciamento è importante per garantire un'analisi comparativa efficace: consente infatti di verificare se utenti con maggiore esperienza e interesse siano più propensi a preferire le immagini generate rispetto a quelle baseline, aspetto che verrà approfondito nelle sezioni successive dedicate alle correlazioni e alla regressione. Inoltre, la presenza di profili eterogenei rafforza la validità dell'esperimento, permettendo una generalizzazione più valida dei risultati ottenuti.

5.2.2 Analisi del Questionario

Il questionario proposto agli utenti al termine di ciascuna generazione di immagini si compone di sei domande. Le prime cinque domande sono costruite secondo una struttura comparativa: all'utente viene chiesto di scegliere tra due immagini (destra o sinistra), oppure di indicare indifferenza, senza sapere quale delle due sia generata dal sistema e quale faccia parte della baseline. Questa impostazione ha l'obiettivo di garantire imparzialità nella valutazione e ottenere risposte basate esclusivamente sull'impatto visivo e sulla coerenza percepita. La sesta domanda è strutturata tramite scala Likert da 1 a 5, con cui si chiede all'utente di esprimere un giudizio complessivo sulla distinguibilità delle due immagini, da “per nulla diverse” (1) a “estremamente diverse” (5).

Domanda 1 – Analisi delle risposte

“Quale delle due immagini ritieni sia più vicina al tuo stile personale?”

Questa domanda mira a valutare il grado di personalizzazione percepita. In particolare, serve a capire se l'immagine generata dal sistema riesca ad avvicinarsi ai gusti stilistici dell'utente, rispetto all'immagine di riferimento (baseline).

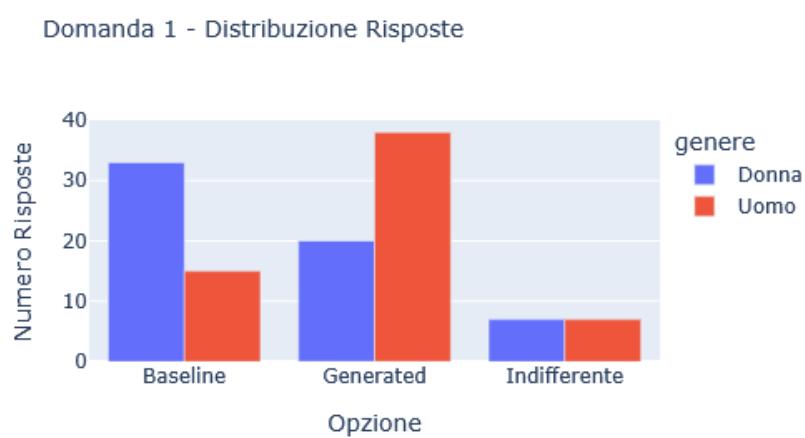


Figura 88: Grafico di distribuzione per la domanda 1

Osservazioni:

L'immagine generata è risultata preferita dal **48.3%** degli utenti in termini di affinità con il proprio stile personale, superando di poco l'immagine

baseline, scelta dal **40%** del campione. Questo risultato indica che il sistema è spesso in grado di produrre immagini percepite come stilisticamente coerenti con i gusti dell'utente.

L'**11.7%** dei partecipanti ha invece segnalato una percezione di parità tra le due opzioni, suggerendo che, per una parte del campione, nessuna delle due immagini si distingua nettamente per vicinanza allo stile personale.

Il dato suggerisce che l'aspetto estetico ha un peso significativo nella valutazione, al punto che le immagini generate riescono in molti casi a competere, e talvolta a prevalere, rispetto alle controparti reali in termini di aderenza allo stile personale percepito.

Domanda 2 – Analisi delle Risposte

“Quale immagine ti aiuterebbe di più a decidere se acquistare il capo?”

Questa domanda esplora l'efficacia comunicativa delle immagini rispetto all'intenzione d'acquisto. Serve a comprendere quale delle due rappresentazioni sia percepita come più utile per valutare visivamente un capo prima di procedere con l'acquisto.

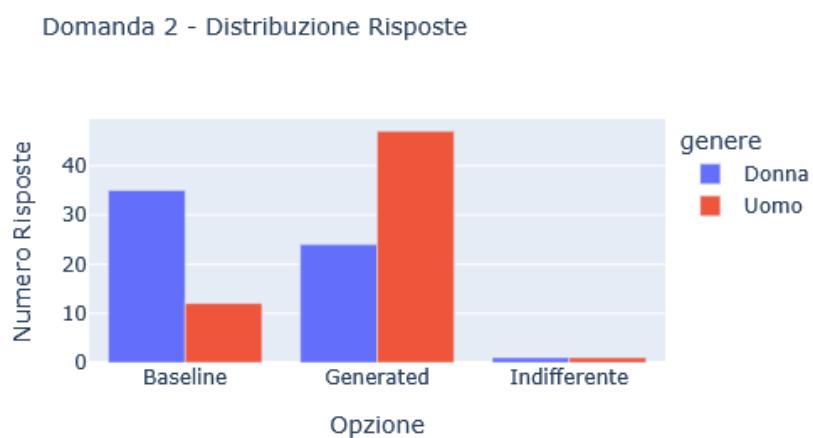


Figura 89: Grafico di distribuzione per la domanda 2

Osservazioni:

L'immagine *generata* è stata ritenuta più utile nel processo decisionale dal **59.2%** degli utenti, superando nettamente l'immagine *baseline*, scelta dal **39.1%** del campione. Solo l'**1.7%** ha dichiarato di non percepire una differenza significativa tra le due.

Questi risultati suggeriscono che il sistema di generazione è spesso in grado di proporre immagini particolarmente efficaci sul piano comunicativo, tanto da influenzare positivamente l'idea di acquistabilità del prodotto.

Il dato sembra indicare che attrattività visiva e personalizzazione contribuiscono a rendere l'immagine generata una risorsa importante nella presentazione di un capo, posizionandosi potenzialmente come strumento strategico per le decisioni di acquisto online.

Domanda 3 – Analisi delle risposte

“Quale immagine appare più realistica come rappresentazione del capo?”

Questa domanda indaga la percezione di realismo trasmessa dalle immagini mostrate, cercando di capire quale delle due venga ritenuta visivamente più verosimile e convincente come rappresentazione di un capo reale.

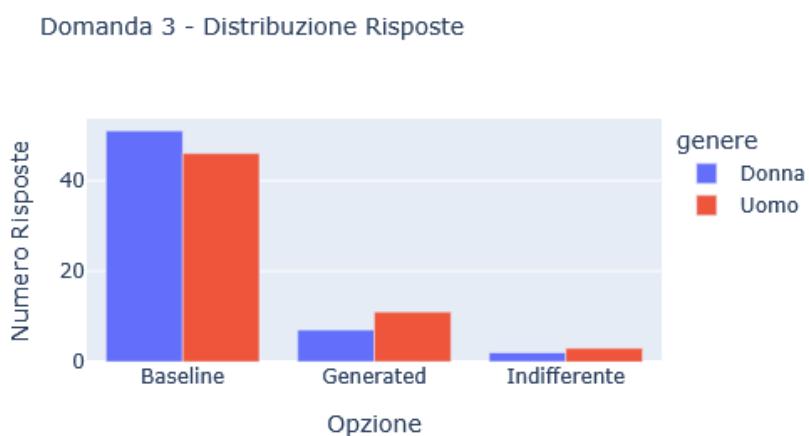


Figura 90: Grafico di distribuzione per la domanda 3

Osservazioni:

I risultati evidenziano una netta preferenza per le immagini *baseline*, che sono state ritenute più realistiche da **80.8%** dei partecipanti. Solo il **15%** ha indicato come più realistica l'immagine *generata*, mentre il **4.2%** si è espresso in modo neutro.

Questo esito suggerisce che, nonostante le immagini generate abbiano mostrato buone performance in altri aspetti, sotto il profilo del realismo visivo esse risultano ancora percepite come meno convincenti rispetto alle

controparti reali. Si tratta di un'indicazione significativa, poiché la credibilità visiva è un aspetto importante nei contesti commerciali e comunicativi del settore moda.

La forte preferenza per le immagini baseline riflette probabilmente una maggiore cura nei dettagli, proporzioni e texture visive, che i partecipanti hanno associato a una rappresentazione più fedele e tangibile del capo. Questo dato può guidare futuri miglioramenti nel sistema generativo, orientati a rafforzare la componente realistica delle generazioni.

Domanda 4 – Analisi delle risposte

“Quale immagine trovi esteticamente più convincente per indossare il capo?”

Questa domanda indaga quale immagine risulti esteticamente più efficace nel valorizzare il capo, focalizzandosi quindi su aspetti visivi legati alla desiderabilità, all’armonia visiva e alla composizione generale della scena.

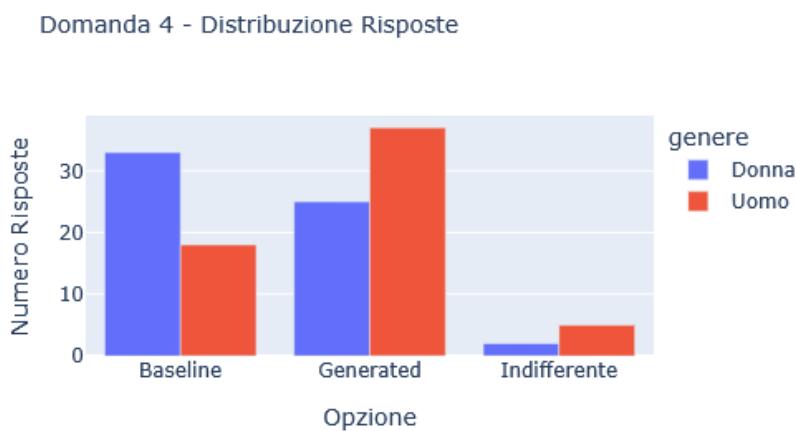


Figura 91: Grafico di distribuzione per la domanda 4

Osservazioni:

I risultati mostrano una leggera prevalenza delle immagini *generated*, scelte dal **51.7%** degli utenti, rispetto alle *baseline* (**42.5%**), con una quota minore di risposte neutre (**5.8%**).

Questa tendenza suggerisce che il sistema di generazione è in grado di produrre immagini percepite come esteticamente valide o persino superiori rispetto alle immagini originali. La capacità di generare contenuti

visivamente convincenti, capaci di ispirare l'utente ad immaginarsi nell'atto di indossare il capo, rappresenta un punto di forza del sistema, in particolare per finalità di comunicazione visiva e marketing.

Pur non essendo un distacco netto, il dato evidenzia un buon equilibrio e una competitività delle immagini generate rispetto a quelle reali.

Domanda 5 – Analisi delle risposte

“Quale immagine ti dà maggiore sicurezza che questo capo ti donerebbe?”

Questa domanda indaga quale immagine trasmetta all'utente una maggiore sensazione di adeguatezza personale e potenziale vestibilità, ovvero quale rappresentazione visiva lo convinca di più che il capo si adatterebbe bene al proprio corpo e stile.

Domanda 5 - Distribuzione Risposte

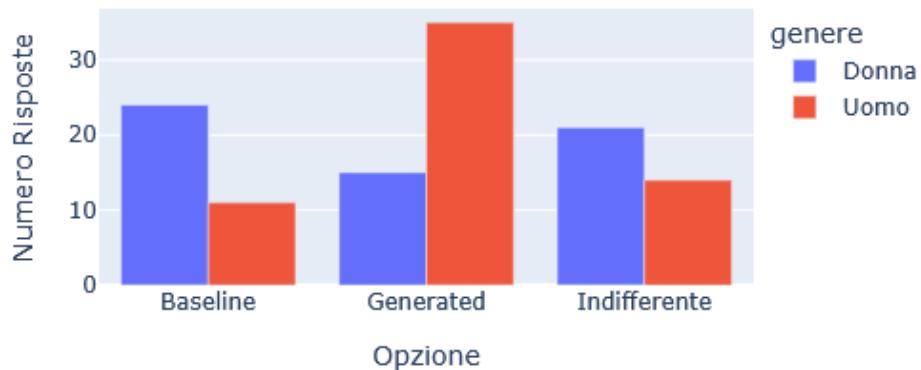


Figura 92: Grafico di distribuzione per la domanda 5

Osservazioni:

Le immagini *generate* risultano preferite dal **41.6%** degli utenti, seguite dalle *baseline* con il **29.2%** e da una quota altrettanto significativa di risposte *indifferenti* (**29.2%**).

Questa distribuzione indica che, pur con un certo vantaggio per le immagini generate, molti utenti non percepiscono una netta superiorità tra le due versioni o faticano a identificare quale trasmetta maggiore sicurezza in termini di vestibilità. La presenza elevata di risposte indifferenti evidenzia

un certo livello di ambiguità, che potrebbe derivare da fattori come la posa del modello, la resa del tessuto o la coerenza visiva generale.

Inoltre, è opportuno considerare che in alcuni casi la scelta è influenzata da **meccanismi proiettivi** o di **identificazione personale**: utenti più maturi, ad esempio, tendono talvolta a preferire il modello che appare più giovanile, associando tale scelta a un'immagine idealizzata di sé; altri, invece, selezionano "*indifferenti*", motivando la risposta con considerazioni del tipo "*mi starebbero bene entrambe le versioni*", spostando quindi l'attenzione dal capo in sé alla propria percezione di adeguatezza.

Il dato, tuttavia, conferma che le immagini generate riescono comunque a trasmettere una percezione di affidabilità comparabile (e in parte superiore) rispetto a quelle reali, anche in un contesto soggettivo e delicato come la proiezione della propria immagine con un capo indossato.

Domanda 6 – Analisi delle risposte

“Le due immagini ti sembrano chiaramente distinguibili l’una dall’altra?”

Scala di risposta: da 1 (per nulla distinguibili) a 5 (completamente distinguibili)

Questa domanda ha l’obiettivo di valutare il grado di originalità percepita delle immagini generate dal sistema. In particolare, mira a capire se gli utenti avvertono una chiara differenziazione visiva tra l’immagine reale e quella generata a partire dal proprio profilo. Un punteggio elevato suggerisce che l’utente percepisce le due immagini come distinte, rafforzando l’idea che la generazione non sia una semplice rielaborazione o imitazione della baseline, ma una proposta nuova e personalizzata, costruita in modo mirato sulle preferenze dichiarate.

Distribuzione Valutazioni (Likert)

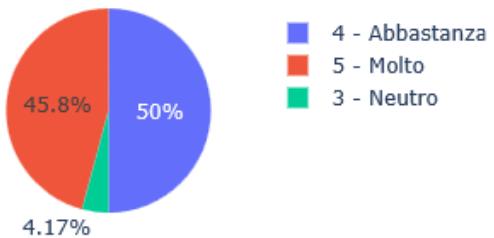


Figura 93: Grafico di distribuzione a torta per la domanda 6

Valutazioni per Genere (Likert)



Figura 94: Grafico di distribuzione a colonne per la domanda 6

Osservazioni

La maggior parte degli utenti ha valutato le immagini con punteggi **elevati**, indicando una percezione chiara della differenza tra le due versioni mostrate: il **95,8%** ha selezionato un valore di **4 o 5** sulla scala Likert. Questo suggerisce che, nonostante non sapessero quale immagine fosse reale e quale generata, i partecipanti sono riusciti a cogliere **dettagli visivi distintivi**, tali da permettere un confronto percepito come netto.

Tuttavia, il fatto che tale distinguibilità non si sia tradotta automaticamente in una preferenza esclusiva per la baseline o per la generata (come mostrato dalle risposte alle precedenti domande) conferma che riconoscere la differenza tra due immagini non implica necessariamente un giudizio di

valore. Un'immagine può infatti risultare "diversa", ma al tempo stesso gradevole, coerente o adatta al proprio stile.

Questo dato rafforza l'idea che il modello generativo utilizzato nel sistema è in grado di produrre immagini sufficientemente credibili e differenziate da quelle reali, senza compromettere l'equilibrio tra riconoscibilità visiva e qualità percepita.

5.2.3 Analisi delle performance delle immagini di baseline

Questa sezione approfondisce l'analisi delle performance delle immagini di baseline, valutando per ciascuna le preferenze aggregate espresse dagli utenti nei cinque confronti previsti dal questionario con particolare attenzione al *success rate* ovvero la percentuale di risposte favorevoli alle immagini generate rispetto alla baseline corrispondente;

Le immagini di baseline sono consultabili al Capitolo **4.3 Selezione delle Immagini di Baseline**.

ID Immagine	Genere	%Generated (success rate)	%Baseline	%Indifferente	TOT Valutaz.
1	Uomo	44 %	46.7 %	9.33 %	15
2	Uomo	66.7 %	28 %	5.33 %	15
3	Uomo	68 %	24 %	8 %	15
4	Uomo	45.3 %	37.3 %	17.3 %	15
5	Donna	24 %	66.7 %	9.33 %	15
6	Donna	22.7 %	69.3 %	8 %	15
7	Donna	28 %	57.3 %	14.7 %	15
8	Donna	46.7 %	41.3 %	12 %	15

Tabella 6: Performance per ciascuna immagine di baseline



Figura 95: Diagramma a colonne per le performance delle immagini di baseline

Osservazioni Generali

I risultati evidenziano differenze significative tra le immagini baseline in termini di performance, con una netta distinzione anche tra i generi. In particolare, si osserva che le **immagini associate a utenti di genere femminile** tendono a favorire maggiormente le versioni reali (baseline), con percentuali di successo delle immagini generate spesso inferiori al 30%. Questo dato potrebbe suggerire una maggiore **attenzione critica da parte del pubblico femminile**, che potrebbe esigere standard visivi più elevati o avere aspettative più specifiche in termini di resa stilistica e coerenza.

D'altro canto, le immagini associate a utenti di genere maschile mostrano un comportamento più eterogeneo, con alcuni casi (immagini 2 e 3) in cui le versioni **generate hanno largamente superato le baseline**, dimostrando un elevato grado di efficacia nella personalizzazione. In questi casi, la baseline di partenza potrebbe aver offerto margini più ampi di miglioramento, consentendo al sistema generativo di produrre contenuti visivamente più attraenti o maggiormente aderenti al profilo utente.

Va inoltre sottolineato che **la qualità visiva intrinseca dell'immagine di baseline** gioca un ruolo determinante nelle valutazioni. Immagini di partenza particolarmente curate, esteticamente forti o qualitativamente prestanti risultano difficilmente superabili dalle versioni generate (es. immagine 6). Viceversa, baseline meno convincenti offrono un'opportunità più concreta al sistema generativo di distinguersi (es. immagine 2).

È importante sottolineare che il limite emerso in questa fase dell'esperimento, ovvero la difficoltà nel superare alcune immagini baseline particolarmente forti sul piano comunicativo ed estetico, è in parte legato ai vincoli del modello generativo utilizzato. Tali difficoltà avrebbero potuto essere superate con l'impiego di Stable Diffusion 3.5 Large di Hugging Face capace di generare immagini con maggiore creatività, originalità e aderenza stilistica. L'utilizzo di un sistema generativo più evoluto avrebbe presumibilmente reso le immagini generate più competitive anche nei confronti delle baseline più prestanti, migliorando la capacità del sistema di distinguersi visivamente.

5.2.4 Correlazioni Semplici

In questa sezione vengono analizzate le **correlazioni lineari semplici** tra il tasso di preferenza per le immagini generate (*preference_rate*) e alcune variabili indipendenti legate al profilo utente: **età, competenza in ambito moda e interesse per la moda**.

Tali correlazioni forniscono una prima indicazione sulla presenza di legami lineari tra le caratteristiche degli utenti e la loro propensione a preferire le immagini generate rispetto a quelle baseline.

La **matrice di correlazione** ottenuta è riportata nella figura seguente:

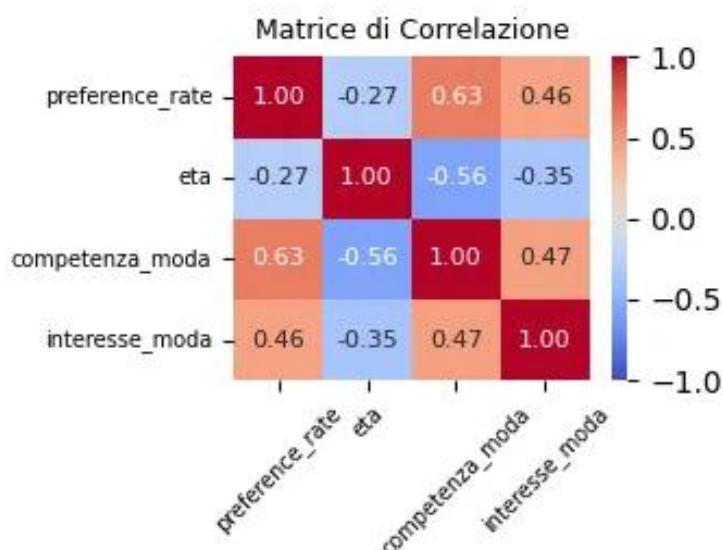


Figura 96: Matrice di correlazione

Di seguito vengono evidenziate le principali osservazioni emerse:

- **Età:** la correlazione tra età e *preference_rate* è negativa (-0.269), indicando che **utenti più giovani tendono ad avere una maggiore propensione verso le immagini generate** rispetto a quelli più maturi. Sebbene il legame non sia fortissimo, conferma una tendenza coerente con l'ipotesi che utenti più giovani possano avere un atteggiamento più aperto verso la tecnologia e la generazione automatica.
- **Competenza moda:** con una correlazione positiva di **0.629**, si evidenzia un legame **forte** tra la competenza percepita in ambito moda e la preferenza per le immagini generate. Gli utenti che si considerano più competenti sembrano **apprezzare maggiormente le immagini personalizzate**, riconoscendone probabilmente la coerenza stilistica o l'innovazione rispetto alle controparti reali.
- **Interesse moda:** anch'esso mostra una correlazione positiva (**0.457**), sebbene più moderata rispetto alla competenza. Gli utenti più interessati alla moda tendono comunque a preferire maggiormente le immagini generate, suggerendo che l'engagement personale nel settore influisce sul giudizio estetico.

In sintesi, **le correlazioni più marcate sono quelle legate all'esperienza e all'interesse nel campo della moda**, mentre l'età ha un impatto più limitato ma comunque significativo. Questi risultati indicano che il sistema di generazione beneficia soprattutto del giudizio positivo degli utenti più coinvolti e competenti nel settore, validando in parte l'efficacia delle personalizzazioni prodotte.

Osservazioni aggiuntive dalla heatmap

- **Età e Competenza moda (-0.56):**

Questa correlazione negativa indica che gli utenti più giovani tendono a percepirsi come più competenti in ambito moda rispetto agli utenti più maturi. Tale osservazione suggerisce che l'età potrebbe influenzare l'autopercezione delle proprie competenze stilistiche, un fattore che a sua volta influenza direttamente la preferenza verso le immagini generate.

- **Competenza moda e Interesse moda (0.47):**

Il valore positivo tra queste due variabili suggerisce che chi si ritiene più competente tende anche ad avere un maggiore interesse per il mondo della moda. Nonostante non sia estremamente forte (0.47), questa correlazione mostra che esiste una connessione diretta tra il grado di coinvolgimento personale e l'autovalutazione della propria competenza.

- **Età e Interesse moda (-0.35):**

Questo valore negativo, moderato ma significativo, indica che utenti più giovani manifestano generalmente un interesse maggiore verso la moda rispetto agli utenti di età più avanzata. Anche questo elemento potrebbe contribuire indirettamente alla preferenza più alta per le immagini generate, data la relazione positiva fra interesse per la moda e preferenza per le immagini generate.

Conclusioni Aggiuntive

Le relazioni interne tra queste variabili confermano che l'età non solo ha un effetto diretto, seppur moderato, sulla preferenza verso le immagini generate, ma incide anche indirettamente attraverso il suo effetto sulla percezione della competenza e sull'interesse per la moda. Gli utenti più giovani, sentendosi più competenti e mostrando maggiore interesse, diventano automaticamente i principali "sostenitori" delle immagini generate dal sistema, rafforzando ulteriormente la validità e la coerenza del modello proposto.

5.2.5 Regressione Lineare

In questa sezione viene presentata l'analisi della regressione lineare effettuata al fine di comprendere in che misura le variabili demografiche e attitudinali (*età*, *competenza_moda* e *interesse_moda*) influenzino il tasso di preferenza verso le immagini generate (*preference_rate*).

La tabella seguente riporta i risultati del modello statistico:

Variabile	Coefficiente	P-value	Significatività
eta	0.0023	0.3899	
competenza_moda	0.0765	0.0033	**
interesse_moda	0.0402	0.1834	

Tabella 7: Regressione lineare

Inoltre, è stato calcolato il coefficiente di determinazione pari a $R^2 = 0.446$ e la sua versione corretta tenendo conto del numero di predittori nel modello pari a $Adjusted R^2 = 0.382$.

Interpretazione del modello

I risultati della regressione lineare mostrano che il modello è in grado di spiegare circa il **44.6%** della varianza osservata nel tasso di preferenza verso le immagini generate. Considerando il valore *Adjusted R²* pari a 0.382, possiamo affermare che, tenendo conto del numero di variabili incluse nel modello, circa il **38.2%** della varianza è spiegata complessivamente dal modello.

Analisi delle variabili

Dalla tabella emergono considerazioni importanti:

- **Competenza moda** risulta una variabile estremamente significativa ($p = 0.0033$), con un coefficiente positivo (0.0765). Ciò suggerisce che una maggiore competenza percepita in ambito moda si associa chiaramente a una preferenza significativamente più alta verso le immagini generate. Questo risultato conferma ulteriormente quanto osservato nelle correlazioni semplici, ovvero che utenti più competenti in ambito moda tendono ad apprezzare di più il risultato della generazione personalizzata.
- **Età e Interesse moda**, pur presentando coefficienti positivi, non risultano statisticamente significative (p rispettivamente di 0.3899 e 0.1834). In particolare, il basso coefficiente e il p-value elevato per l'età indicano che tale variabile, pur avendo mostrato un'influenza negativa nelle correlazioni semplici, non esercita un impatto significativo una volta considerate insieme le altre variabili nella regressione. L'interesse personale verso la moda, pur avendo una correlazione moderata positiva con la preferenza generata, perde parte della sua significatività statistica quando considerato in relazione alla competenza moda, che sembra essere il fattore dominante.

Considerazioni complessive e conclusioni

L'analisi suggerisce che la variabile che guida principalmente la preferenza per le immagini generate è la competenza percepita in ambito moda, indicando che gli utenti con maggiore competenza risultano più capaci di apprezzare e preferire il valore aggiunto delle immagini create dal sistema.

Questo risultato è in linea con quanto osservato nelle analisi precedenti, rafforzando l'ipotesi che il grado di familiarità e conoscenza del settore moda influenzi direttamente la percezione della qualità delle immagini generate.

Sebbene le variabili età e interesse personale per la moda non risultino statisticamente significative nel modello, i loro coefficienti sono coerenti con i risultati emersi nelle precedenti analisi esplorative e nelle correlazioni semplici. In particolare, emerge una leggera tendenza positiva per l'interesse personale e una debole relazione positiva con l'età, che – pur non raggiungendo la significatività statistica – confermano una direzione coerente e attendibile con quanto osservato finora.

Il valore elevato di *Adjusted R²* (38.2%) testimonia una buona capacità esplicativa del modello statistico utilizzato, confermando l'affidabilità dell'analisi svolta e la rilevanza della variabile *competenza_moda* nello studio effettuato.

In conclusione, è importante sottolineare che i risultati ottenuti, pur fornendo indicazioni significative, potrebbero essere stati influenzati dalla limitata dimensione del campione (30 partecipanti). Un numero maggiore di utenti coinvolti nell'esperimento potrebbe infatti contribuire a una migliore stabilizzazione e generalizzazione delle relazioni individuate, rendendo il modello statistico più robusto e affidabile nella spiegazione della preferenza verso le immagini generate. Pertanto, ulteriori sperimentazioni con un campione più ampio potrebbero offrire conferme più solide e definitive dei risultati esposti.

5.2.6 Bradley Terry Standard e Bradley Terry Pesato

In questa sezione vengono analizzati i risultati ottenuti utilizzando il modello Bradley-Terry (BT), tecnica statistica utilizzata per determinare la probabilità che un elemento venga preferito rispetto ad un altro sulla base di confronti diretti tra coppie di elementi. In particolare, nel contesto di questo lavoro, il modello permette di stimare globalmente la probabilità che le immagini generate siano preferite alle rispettive immagini baseline.

Bradley-Terry Standard

Il modello Bradley-Terry standard è stato impiegato per misurare la probabilità complessiva che un'immagine generata risulti preferita rispetto alla baseline, considerando esclusivamente le risposte comparative delle prime cinque domande del questionario. Le risposte indicate come "indifferenti" sono state escluse dall'analisi per garantire la massima chiarezza nei risultati.

Tipologia	Punteggio BT (λ)	Probabilità Vittoria
baseline	0.0354	0.5177
generated	-0.0354	0.4823

Tabella 8: punteggio BT e probabilità di vittoria per BT standard

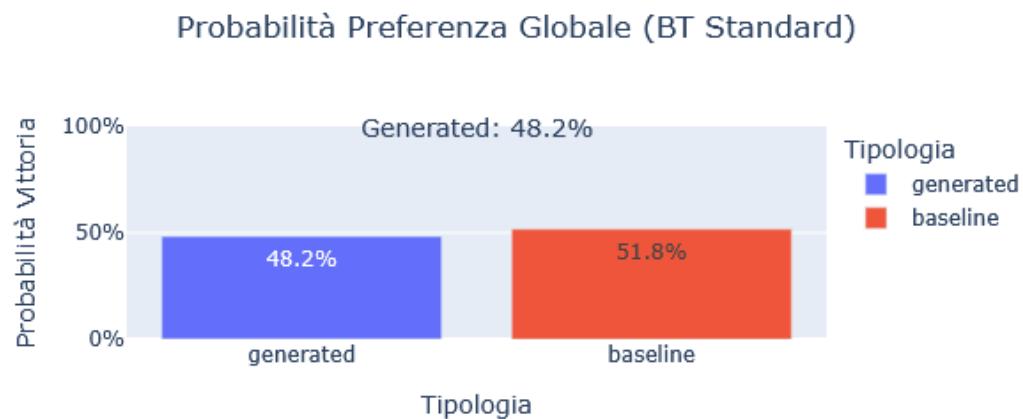


Figura 97: Grafico a colonne per BT standard

Questi risultati indicano che, a livello globale e senza considerare il peso differenziato degli utenti, le immagini di baseline tendono leggermente ad essere preferite rispetto a quelle generate dal sistema.

Bradley-Terry Pesato vs Non Pesato

Successivamente è stato introdotto il modello Bradley-Terry pesato, che tiene conto del livello di competenza e interesse degli utenti rispetto alla moda, assegnando un peso maggiore ai confronti effettuati da utenti con punteggi più alti in queste due caratteristiche.

La figura riporta il confronto tra due varianti del modello:

- **Modello non pesato (standard):** in questo caso, tutte le risposte degli utenti hanno uguale peso. Si osserva che le immagini di baseline ottengono una leggera preferenza, con una probabilità di vittoria del **51.8%**, rispetto alle immagini generate (**48.2%**). Questo risultato suggerisce che, in termini assoluti, le immagini baseline sono state leggermente più convincenti per la maggioranza degli utenti, sebbene la differenza sia piuttosto contenuta.
- **Modello pesato:** introduce il peso dato dalla combinazione della competenza e dell'interesse dichiarati dagli utenti, valorizzando maggiormente le risposte degli utenti più competenti e interessati al mondo della moda. In questa variante, la preferenza per le immagini generate aumenta significativamente, raggiungendo il **53.1%**, mentre la baseline scende al **46.9%**. Questo cambio di preferenza indica che utenti più competenti e coinvolti percepiscono maggiormente il valore aggiunto delle immagini generate.

Modello	Prob. Generated	Prob. Baseline	λ Generated	λ Baseline
Non Pesato	0.4823	0.5177	-0.0354	0.0354
Pesato	0.5307	0.4693	0.0614	-0.0614

Tabella 9: confronto tra i punteggi di BT standard e BT pesato

Probabilità di Vittoria (Generated vs Baseline)



Figura 98: Grafico a colonne per BT pesato

Questa analisi evidenzia che la competenza e l'interesse in ambito moda influenzano significativamente le preferenze degli utenti, determinando uno spostamento chiaro verso le immagini generate quando si tiene conto delle caratteristiche individuali dei partecipanti. Pertanto, il sistema sembra essere particolarmente efficace per utenti più consapevoli e interessati alla moda, a dimostrazione che la personalizzazione realizzata dalle immagini generate acquisisce valore al crescere della competenza dell'utente finale.

6. CONCLUSIONI

Questo lavoro di ricerca ha esplorato il potenziale della personalizzazione visiva dei contenuti pubblicitari in ambito moda, ponendo al centro una domanda fondamentale: *in che misura la personalizzazione basata sul profilo utente può incidere significativamente sulle preferenze e sulla propensione all'acquisto?*

I risultati ottenuti mostrano chiaramente come la personalizzazione delle immagini pubblicitarie, ottenuta mediante sistemi generativi avanzati come *Gemini* e *Stable Diffusion*, possa effettivamente influenzare in maniera positiva le preferenze degli utenti. Le analisi hanno rivelato che, sebbene non tutte le immagini generate siano riuscite a superare le immagini baseline in termini di impatto visivo e preferenza generale, una consistente parte del campione ha trovato le versioni personalizzate più vicine al proprio stile personale (circa il 48%), più convincenti dal punto di vista estetico (circa il 51%) e, soprattutto, significativamente più utili nel facilitare l'intenzione d'acquisto (circa il 59%).

L'applicazione dei modelli Bradley-Terry ha ulteriormente rafforzato questi risultati. In particolare, l'introduzione di un modello pesato, che tiene conto delle competenze e degli interessi personali, ha mostrato una preferenza ancora più marcata verso le immagini generate (53% rispetto al 48% del modello non pesato). Ciò indica che il sistema riesce particolarmente bene a soddisfare utenti con maggiore competenza o coinvolgimento nel settore moda, supportando l'idea secondo cui il valore della personalizzazione cresce proporzionalmente alla competenza percepita dall'utente stesso.

La regressione lineare ha poi confermato ulteriormente questa correlazione, mostrando come la variabile “*competenza moda*” influenzi significativamente e positivamente la preferenza verso le immagini generate, con una capacità esplicativa globale del modello di circa il 38%, nonostante il campione relativamente ridotto di partecipanti.

In sintesi, l'intero esperimento conferma il grande potenziale della personalizzazione visiva basata sul profilo utente: essa rappresenta un reale valore aggiunto, non solo sul piano estetico ma anche sul piano commerciale, facilitando il processo decisionale e incentivando l'interazione

positiva verso il prodotto. Sebbene ci siano margini di miglioramento e fattori da esplorare ulteriormente, i risultati raggiunti sono promettenti e indicano chiaramente che l'approccio presentato costituisce un passo significativo verso un utilizzo efficace e mirato dell'intelligenza artificiale nell'ambito del marketing visivo e della comunicazione personalizzata.

6.1. Principio di Utilizzo ed Etica

Lo sviluppo e l'impiego di sistemi di Intelligenza Artificiale generativa, come quello presentato in questo lavoro, richiedono necessariamente una riflessione sul loro utilizzo etico e sul ruolo che assumono nel processo creativo delle aziende e dei professionisti.

In primo luogo, è fondamentale chiarire che lo strumento sviluppato **non deve essere considerato un sostituto** della figura creativa del designer, del fotografo o delle altre professionalità coinvolte nella produzione di contenuti visivi. Piuttosto, esso rappresenta **un potente alleato, un supporto e un driver per la creatività umana**. Il rischio di interpretarlo diversamente porterebbe inevitabilmente verso un appiattimento della capacità creativa, trasformando il processo artistico e progettuale in un mero assemblaggio digitale privo di personalità, oltre che alla possibile perdita di preziose figure altamente preparate e qualificate.

A testimonianza di un approccio virtuoso e corretto, si può citare l'esempio della campagna Nike AIR discusso nel capitolo **2.4.3 Nike, Realtà Aumentata e IA Generativa**. L'AI non è stata impiegata come una soluzione completa che rimpiazza la visione umana, bensì come fonte di ispirazione per i designer, che hanno poi ulteriormente elaborato e perfezionato le proposte iniziali. In questo senso, l'AI ha avuto il ruolo di acceleratore creativo, migliorando tempi ed efficienza, ma lasciando invariato l'apporto qualitativo ed emozionale dei creativi coinvolti.

Esempio pratico di utilizzo del sistema:

Immaginiamo un designer di una startup di moda che debba lanciare una giacca invernale. Inserendo nella pipeline il capo base e il profilo di un utente “viaggiatore eco-conscious”, il sistema potrebbe generare rapidamente diversi concept visivi: la giacca indossata in un paesaggio

montano con palette dai toni naturali, in un contesto urbano che valorizza materiali riciclati, oppure combinata con accessori modulari funzionali al viaggio. Il designer avrebbe così modo di selezionare, rifinire e adattare la proposta più interessante, riducendo drasticamente il time-to-market da settimane a pochi giorni.

Tuttavia, è importante ribadire che le immagini prodotte dal sistema **non devono essere semplicemente “riciclate”**. Il valore aggiunto resta infatti nelle modifiche sostanziali e nell'elaborazione creativa e originale che il professionista apporta al concept iniziale generato dall'intelligenza artificiale, per realizzare poi qualcosa di veramente unico e umano.

Infine, è cruciale riconoscere il ruolo democratico che l'IA generativa assume nel settore creativo: essa rende accessibile a un numero maggiore di persone e imprese una fonte di creatività che spesso rimane fuori portata a causa dei costi elevati associati ad attrezzature professionali, sessioni di shooting fotografiche, modelli e set di produzione dedicati. In questo senso, sistemi come quello proposto non solo accelerano e ottimizzano i processi creativi, ma **favoriscono anche l'inclusività e la diversificazione** all'interno dell'industria della creatività e della comunicazione visiva.

6.2 Limitazioni

Nonostante gli interessanti risultati ottenuti dal progetto, è opportuno riconoscere alcune limitazioni intrinseche alla metodologia adottata e alle tecnologie utilizzate. Affrontare tali criticità permette non solo di interpretare correttamente i risultati ottenuti, ma anche di suggerire possibili linee di miglioramento per future implementazioni.

Una prima limitazione riguarda le **risorse hardware e l'accesso ai modelli** utilizzati nel progetto. Non disponendo di una macchina ad alte prestazioni, si sono utilizzati modelli online mediante inferenza tramite API forniti da provider i quali inevitabilmente impongono vincoli di memoria e capacità computazionali per le generazioni.

In secondo luogo, un ulteriore elemento che ha inciso sul progetto è rappresentato dalla modalità di raccolta delle informazioni degli utenti. A differenza di progetti di alto profilo, come quello il già citato caso Nike, dove ogni atleta ha beneficiato di un'interazione personale con gli esperti consentendo una profilazione approfondita e la definizione di molteplici prompt mirati e su misura, in questo lavoro la profilazione è stata effettuata esclusivamente tramite un form online, basato su scelte autonome degli utenti. Questo approccio, seppur rapido e funzionale, pone agli utenti la sfida di auto-valutarsi e compiere decisioni immediate, attività non sempre semplice o naturale. Infatti, spesso gli utenti potrebbero trovare più efficace un approccio graduale e iterativo, basato sulla valutazione e il progressivo perfezionamento delle immagini generate fino al raggiungimento della soluzione ottimale.

Una delle limitazioni più rilevanti affrontate durante la sperimentazione riguarda **l'accesso ai modelli avanzati**. Nel corso del progetto, infatti, si è verificata una transizione forzata dal modello Stable Diffusion 3.5 Large di Hugging Face al corrispettivo modello offerto da Stability AI. Questa modifica, resa obbligatoria dall'introduzione improvvisa di costi proibitivi e mancanza di trasparenza da parte del primo servizio, ha portato a un netto calo delle prestazioni. Stable Diffusion 3.5 Large di Stability AI, infatti, non è stato in grado di garantire la stessa qualità visiva, precisione e creatività della versione di Hugging Face. Tale limitazione ha impattato

negativamente non solo sulla qualità delle generazioni ma anche sulla ricchezza dei prompt inviati al modello Gemini (VLM), riducendo notevolmente la complessità e la profondità creativa complessiva del sistema.

Oltre a queste problematiche operative, è necessario evidenziare anche una limitazione legata alla **maturità della tecnologia stessa**. L'Intelligenza Artificiale Generativa, come già menzionato nell'introduzione, è una tecnologia relativamente giovane, introdotta su scala significativa intorno al 2010. Nel giro di soli quindici anni, il progresso è stato indubbiamente eccezionale, come sottolineato anche da figure autorevoli quali il Chief Innovation Officer di Nike, che rimarcano l'importanza di poter creare oggi concept e ispirazioni in pochi minuti rispetto ai mesi necessari in passato. Tuttavia, proprio per la sua relativa giovinezza, la tecnologia non può ancora garantire risultati sempre coerenti, affidabili e visivamente impeccabili. Questo progetto, pertanto, costituisce un'esplorazione avanzata di una tecnologia ancora in evoluzione, da cui è ragionevole non aspettarsi ancora risultati definitivi o straordinariamente coerenti sotto ogni aspetto.

Infine, una limitazione di natura statistica, seppur secondaria, riguarda il **numero relativamente ristretto di partecipanti** che hanno preso parte all'esperimento. Sebbene il campione sia stato sufficientemente variegato per fornire indicazioni utili e significative, un numero maggiore di utenti avrebbe garantito una maggiore robustezza dei risultati ottenuti.

Riepilogo delle principali limitazioni:

- Risorse hardware limitate che hanno vincolato la complessità del sistema.
- Profilazione degli utenti basata esclusivamente su un semplice form, senza approfondimenti dettagliati.
- Improvvisa impossibilità di accesso a modelli avanzati (Stable Diffusion 3.5 Large di Hugging Face) che ha influenzato negativamente la qualità visiva e la complessità creativa delle generazioni.

- Tecnologia relativamente giovane e ancora in fase di sviluppo che comporta intrinseche limitazioni qualitative e di affidabilità.
- Dimensione limitata del campione di partecipanti che ha ridotto, seppur marginalmente, la solidità statistica delle conclusioni.

6.3 Sviluppi Futuri

Nonostante i risultati promettenti ottenuti con questo lavoro, vi sono alcuni margini di miglioramento che potrebbero essere esplorati nei futuri sviluppi del progetto.

Un primo sviluppo consisterebbe nell'impiego di modelli di generazione immagine più avanzati e performanti rispetto a quelli utilizzati in questo studio. In particolare, come si è potuto osservare durante la fase sperimentale, l'adozione di un modello altamente creativo e dettagliato come Stable Diffusion 3.5 Large di Hugging Face ha mostrato un notevole potenziale nell'elevare la qualità visiva delle immagini generate. Tale aggiornamento consentirebbe nuovamente l'utilizzo di prompt altamente descrittivi e dettagliati, aumentando notevolmente la creatività, la qualità estetica e la precisione nelle generazioni.

Parallelamente, sarebbe vantaggioso eseguire questi modelli direttamente in locale invece di utilizzare API remote. Tale approccio consentirebbe una maggiore flessibilità e velocità operativa, riducendo al contempo i costi di gestione e risolvendo problematiche legate alla limitazione di memoria e potenza di calcolo offerte dai servizi accessibili tramite inferenza.

Sul fronte sperimentale, un ulteriore miglioramento consisterebbe nell'incrementare la numerosità del campione coinvolto nelle fasi di test. Una base di utenti più ampia consentirebbe di raccogliere dati statisticamente più solidi e rappresentativi, migliorando la capacità predittiva e la possibilità di generalizzare i risultati del sistema.

In prospettiva, sarebbe interessante anche esplorare integrazioni con altre tecnologie, come la realtà aumentata (AR), per offrire agli utenti esperienze immersive di shopping virtuale. Un tale sistema consentirebbe agli utenti di visualizzare e provare virtualmente i capi generati, creando una

connessione ancora più forte tra l'utente, la personalizzazione visiva proposta dal sistema e la concreta esperienza d'acquisto.

Infine, un ultimo ma significativo sviluppo riguarda l'ottimizzazione continua del processo di generazione. Al momento della registrazione di un nuovo utente, sarebbe interessante prevedere un sistema automatizzato che analizzi le generazioni precedenti più apprezzate, identificando i pattern estetici e stilistici di maggiore successo. Questi elementi potrebbero essere integrati nei nuovi prompt, consentendo una personalizzazione incrementale e sempre più efficace. In questo modo, il sistema trasformerebbe progressivamente i dati grezzi raccolti in un metodo strutturato, garantendo una personalizzazione sempre più mirata e precisa.

In sintesi, i principali sviluppi futuri del progetto potrebbero includere:

- Adozione di modelli di generazione immagine più evoluti e creativi (es. Stable Diffusion 3.5 Large di Hugging Face).
- Esecuzione locale dei modelli per massimizzare efficienza e flessibilità.
- Ampliamento significativo del numero di partecipanti per ottenere risultati statisticamente più robusti.
- Integrazione di tecnologie emergenti come la realtà aumentata per esperienze immersive di prova dei capi.
- Sviluppo di un sistema che analizzi le generazioni passate di successo per personalizzare progressivamente e in modo ottimale le generazioni future.

BIBLIOGRAFIA

[1] Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence
Tratta le implicazioni dell'IA nella pubblicità e nei settori creativi.

<https://www.sciencedirect.com/science/article/pii/S0007681318301393>

[2] Ramesh, A., Pavlov, M., Goh, G., et al. (2021). DALL-E: Creating Images from Text Descriptions. OpenAI Blog.
Describe il funzionamento del modello DALL-E e il suo impatto sulla creazione di immagini generative.

<https://openai.com/dall-e>

[3] Karras, T., Laine, S., Aila, T. (2019). A Style-Based Generator Architecture for Generative Adversarial Networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
Le GAN e il loro ruolo nella generazione di immagini pubblicitarie.

<https://ieeexplore.ieee.org/document/8953766>

[4] McKinsey & Company. (2021). The State of Fashion 2021.
Report annuale che esplora il ruolo dell'IA nella moda, incluso il design e la personalizzazione.

<https://www.mckinsey.com/~/media/mckinsey/industries/retail/our%20insights/state%20of%20fashion/2021/the-state-of-fashion-2021-vf.pdf>

[5] Chollet, F. (2017). Deep Learning with Python.
Contiene esempi pratici sull'uso del deep learning, incluse applicazioni nel design visivo.

<https://sourestdeeds.github.io/pdf/Deep%20Learning%20with%20Python.pdf>

[6] Floridi L. (2014). The Fourth Revolution.

<https://issc.al.uw.edu.pl/wp-content/uploads/sites/2/2022/05/Luciano-Floridi-The-Fourth-Revolution -How-the-infosphere-is-reshaping-human-reality-Oxford-University-Press-2014.pdf>

Testo utile per comprendere le implicazioni etiche dell'uso dell'IA, inclusa la pubblicità personalizzata.

[7] Vinuesa, R., Azizpour, H., Leite, I., et al. (2020). The Role of Artificial Intelligence in Achieving the Sustainable Development Goals. *Nature Communications*, 11, 233.

<https://www.nature.com/articles/s41467-019-14108-y>

Esplora le potenziali implicazioni sociali e sostenibili dell'IA.

[8] Attention Is All You Need (2017)

https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547de91fb053c1c4a845aa-Paper.pdf

Importante paper che pone le basi per l'architettura Transformer usata degli LLM

[9] "Denoising Diffusion Probabilistic Models" (2020)

<https://arxiv.org/abs/2006.11239>

Paper fondamentale sui modelli di diffusione, alla base di strumenti come DALL-E 2 e Stable Diffusion. (Fondamenti tecnici dell'IA generativa)

[10] "The Ethics of Algorithms: Key Problems and Solutions" (2016)

<https://journals.sagepub.com/doi/10.1177/2053951716679679>

Approfondisce questioni etiche legate all'uso di algoritmi, inclusa la manipolazione pubblicitaria. (Etica)

[11] Gemini: A Family of Highly Capable Multimodal Models (2022)

<https://storage.googleapis.com/deepmind->

[media/gemini/gemini_1_report.pdf](#)

Caso di studio realizzato da Google sul modello generativo “Gemini”

[12] AI revolutionizing industries worldwide: A comprehensive overview of its diverse applications

<https://www.sciencedirect.com/science/article/pii/S2773207X24001386>

Lo studio fornisce una panoramica su come l'IA stia trasformando vari settori, inclusa la pubblicità, attraverso l'uso di dati e tecnologie avanzate per migliorare l'efficacia delle campagne pubblicitarie

[13] Generative Adversarial Networks

<https://arxiv.org/pdf/1406.2661>

Paper che introduce un nuovo approccio per generare dati realistici tramite una rete neurale che oppone due modelli (generatore e discriminatore) in un processo competitivo (GAN)

[14] EMOTIONAL CONTEXT AND EFFECTIVENESS OF TV ADVERTISING

https://www.researchgate.net/publication/325974676_Emotion_context_and_effectiveness_of_TV_advertising

Esplora l'impatto delle emozioni evocate sia dall'annuncio che dal contesto in cui è inserito, evidenziando come la coerenza emotiva tra contesto e pubblicità possa migliorare la memorabilità del prodotto

[15] Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5)

<https://arxiv.org/pdf/1910.10683>

Il paper introduce il modello Transformer encoder-decoder T5.

[16] BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension (BART)
[https://arxiv.org/pdf/1910.13461](https://arxiv.org/pdf/1910.13461.pdf)

Il paper introduce BART, un modello encoder-decoder preaddestrato con tecniche di denoising, efficace per generazione e comprensione del linguaggio.

[17] Language Models are Unsupervised Multitask Learners (GPT)

https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

Presenta GPT-2, un modello Transformer decoder-only autoregressivo.

[18] Learning Transferable Visual Models From Natural Language Supervision

[https://arxiv.org/pdf/2103.00020](https://arxiv.org/pdf/2103.00020.pdf)

Presenta il modello CLIP di OpenAI

[19] Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision

[https://arxiv.org/pdf/2102.05918](https://arxiv.org/pdf/2102.05918.pdf)

Presenta il modello ALIGN

[20] Flamingo: a Visual Language Model for Few-Shot Learning

<https://storage.googleapis.com/deepmind-media/DeepMind.com/Blog/tackling-multiple-tasks-with-a-single-visual-language-model/flamingo.pdf>

Presenta il modello Flamingo

[21] High-Resolution Image Synthesis with Latent Diffusion Models

<https://arxiv.org/pdf/2112.10752>

Presenta gli LDM e mostra come operano in uno spazio latente invece che in uno spazio di pixel

[22] Language Models are Few-Shot Learners

<https://arxiv.org/pdf/2005.14165>

Evidenzia come i Large Language Models (LLM), come GPT-3, abbiano rivoluzionato il campo del Natural Language Processing (NLP), estendendo le loro applicazioni a chatbot, traduzione automatica, assistenti virtuali e modelli multimodali capaci di interpretare testo e immagini

SITOGRAFIA

[1] OpenAI DALL-E Project

<https://openai.com/dall-e>

Sito ufficiale con spiegazioni tecniche e demo del modello DALL-E.

[2] Stable Diffusion Project

<https://stability.ai>

Risorsa per approfondire il funzionamento e l'applicazione dei modelli text-to-image nel settore creativo.

[3] MidJourney AI Tool

<https://www.midjourney.com>

Strumento di generazione visiva utilizzato nell'industria creativa.

[4] Generative Adversarial Networks GANs: A Beginner's Guide

<https://towardsdatascience.com/generative-adversarial-networks-gans-a-beginners-guide-f37c9f3b7817>

Guida introduttiva all'uso delle GAN nella pubblicità.

[5] The State of Fashion 2025: Challenges at every turn

<https://www.mckinsey.com/industries/retail/our-insights/state-of-fashion>

Analisi del settore moda e il ruolo delle nuove tecnologie, incluso l'uso dell'IA.

[6] Coca Cola. The Holiday Magic is coming 2025

<https://www.youtube.com/watch?v=4RSTupbfGog>

Video pubblicitario di Coca – Cola generato con l'IA.

[7] Create Real Magic with Coca-Cola AI

<https://www.youtube.com/watch?v=7IK9vawhCOM>

Benny Lee presenta il progetto “Create Real Magic with Coca-Cola”.

[8] Coca-Cola's Holiday Ads Trade the 'Real Thing' for Generative A.I.

<https://www.nytimes.com/2024/11/20/style/coca-cola-holiday-ads-ai.html>

Articolo del “The New York Times” sulla campagna pubblicitaria di Coca Cola basata su IA

[9] Discovering ModiFace, L'Oréal Group

<https://www.loreal.com/en/beauty-science-and-technology/beauty-tech/discovering-modiface/>

Intelligenza artificiale per diagnosticare le condizioni della pelle e suggerire routine personalizzate di L'Oréal.

[10] How L'Oreal is tapping generative AI to transform its marketing

<https://siliconangle.com/2024/04/11/loreal-tapping-generative-ai-transform-marketing/>

[11] L'Oréal's Modiface Brings AI-powered Virtual Makeup Try-on To Amazon

<https://www.loreal.com/en/articles/science-and-technology/l-oreal-modiface-brings-ai-powered-virtual-makeup-try-ons-to-amazon/>

[12] Augmented Marketing in Beauty with Generative AI | Asmita Dubey, L'Oréal | NVIDIA GTC 2024

<https://www.youtube.com/watch?v=LltEWLOIIB0>

Presentazione di NVIDIA Developer a cura di Asmita Dubey sull'utilizzo dell'intelligenza artificiale generativa per l'ambito della bellezza.

[13] Nike uses AR to help you find the right fit for your sneakers

<https://www.engadget.com/2019-05-09-nike-fit-augmented-reality-right-fit-size-shoes.html?guccounter=1#gallery=1041513&slide=7575489&index=0>

[14] How Nike is Using AI to Transform Product Design, Customer Experience, and Operational Efficiency

<https://aimresearch.co/market-industry/how-nike-is-using-ai-to-transform-product-design-customer-experience-and-operational-efficiency>

[15] Creating the Unreal: How Nike Made Its Wildest Air Footwear Yet

<https://about.nike.com/en/stories/nike-design-athlete-imagined-revolution>

Nike mostra il processo di creazione di alcune sue scarpe realizzate con l'IA.

[16] LAION-5B: A NEW ERA OF OPEN LARGE-SCALE MULTI-MODAL DATASETS

<https://laion.ai/blog/laion-5b/>

Grande database usato per addestrare stable diffusion e altri LDM

[17] Stable Diffusion Samplers: A Comprehensive Guide

https://stable-diffusion-art.com/samplers/#What_is_Sampling

Guida 1 ai sampler di stable diffusion

[18] Complete guide to samplers in Stable Diffusion

https://www.felixsanz.dev/articles/complete-guide-to-samplers-in-stable-diffusion?utm_source=chatgpt.com#euler

Guida 2 ai sampler di stable diffusion

[19] Choosing a Sampler for Stable Diffusion

<https://mccormickml.com/2023/04/11/choosing-stable-diffusion->

[sampler/?utm_source=chatgpt.com](#)

Guida 3 ai sampler di stable diffusion

[20] Prompting Guide per Stable Diffusion

<https://stability.ai/learning-hub/stable-diffusion-3-5-prompt-guide>

Tecniche di prompting per stable diffusion

[21] Stable Diffusion Akashic Records

<https://github.com/Maks-s/sd-akashic>

Collezione di studi, stili artistici, prompt e altri tool utili per l'esplorazione di stable diffusion

[22] Guida al Prompt Engineering

<https://www.promptingguide.ai/models/gemini>

Importante guida al prompt engineering

[23] Guida al Prompt Engineering, Google

<https://ai.google.dev/gemini-api/docs/prompting-intro?hl=it>

Guida al prompt engineering realizzata da Google

[24] Vogue Runaway

<https://www.vogue.com/fashion-shows>

The epicenter of runway news, street style, and emerging trends.

[25] Text-to-image: latent diffusion models

<https://necd.org.uk/knowledge-hub/image-to-text-latent-diffusion-models>

Describe in modo introduttivo il funzionamento dei modelli di diffusione latente per la generazione di immagini a partire da testo, illustrandone i principi base e le applicazioni.

[26] Understanding Multimodal LLMs

<https://magazine.sebastianraschka.com/p/understanding-multimodal-langs>

Panoramica chiara e accessibile sul funzionamento dei modelli multimodali (LLM), capaci di elaborare input testuali e visivi, spiegandone architettura e potenzialità.

[27] Gradio Course - Create User Interfaces for Machine Learning Models

<https://www.youtube.com/watch?v=RiCQzBluTxU&feature=youtu.be>

2 hours course on creating user interfaces with Gradio for machine learning models

[28] A Brief History of Predictive Analytics – Part 1

<https://afterinc.com/2018/12/28/brief-history-predictive-analytics-part-1/>

Panoramica dell'evoluzione dell'analisi predittiva, coprendo il periodo dagli anni '40 agli anni '50

[29] Remembering the Birth of Data Mining and Predictive Analytics

<https://www.linkedin.com/pulse/remembering-birth-data-mining-predictive-analytics-look-douglas-day-tx2pc/>

Riflette sulle origini del data mining e dell'analisi predittiva, esplorando come queste discipline siano emerse e si siano sviluppate nel tempo

[30] IAS Study Shows Ad Context Increases Memorability Up to 40%

<https://integralads.com/news/ias-the-context-effect-research/>

Studio che analizza come i contenuti personalizzati aumentano la memorabilità fino al 40%

[31] How AI-powered ultra-personalised experiences are boosting our beauty brands

<https://www.unilever.com/news/news-search/2024/how-aipowered-ultrapersonalised-experiences-are-boosting-our-beauty-brands/>

Unilever ha implementato strumenti basati sull'IA per offrire esperienze ultra-personalizzate ai consumatori, aiutandoli a scegliere prodotti su misura, differenziando il marchio e incrementando le vendite nel settore della bellezza

[32] The Marvel of Large Language Models: Transforming Text with AI

<https://medium.com/%40pratikabnave97/the-marvel-of-large-language-models-transforming-text-with-ai-3b5f853189db>

Evidenzia come gli LLM siano fondamentali nella generazione di testo assistita dall'intelligenza artificiale, trasformando il modo in cui produciamo e interagiamo con i contenuti scritti.

[33] Modelli linguistici di grandi dimensioni (LLM) basati sull'IA di Google di altissimo livello

<https://cloud.google.com/ai/llms?hl=it>

La documentazione di Google Cloud indica che gli LLM sono impiegati in una varietà di compiti di elaborazione del linguaggio naturale, inclusa la traduzione automatica, sfruttando la loro capacità di comprendere e generare linguaggio umano.

[34] Introduction to Large Language Models (LLMs): An Overview of BERT, GPT, and Other Popular Models

<https://www.johnsnowlabs.com/introduction-to-large-language-models-llms-an-overview-of-bert-gpt-and-other-popular-models/>

Analizza come gli LLM sono utilizzati in assistenti virtuali e chatbot per fornire risposte accurate e pertinenti alle domande degli utenti, comprendendo il contesto e la semantica delle richieste

APPENDICI

Appendice A – Input Prompt e Output dei Modelli image-text-to-text

A.1 *Prompt di Input usato per i modelli*

Given this image, generate a prompt to create an advertisement image targeting the following person:

Start user description:

A woman, 31 years old, resident in Belgium, techno DJ, salary 15,000 euros per month, single, lover of avant-garde, favorite brands: Rick Owens, shopping habits: in boutiques and online, favorite music genre: techno and house, in the clothes she wears she looks for: boldness and elegance, favorite colors: black

End user description

Reply with a single prompt. I want it to be highly descriptive, creative, and deeply imaginative.

After generating the prompt, provide a final paragraph in which you explain the decisions you made and why you envisioned this specific advertisement. More generally, walk me through your thinking process.

A.2 *Risposta Lunga 1 di LLaMa*

Create a portrait photographic illustration of a woman in her early 30s, standing confidently in front of an urban wall adorned with an avant-garde graffiti, showcasing her unique style and charisma. The woman should be dressed in a black leather dress with an architectural silhouette, featuring sharp angles and bold lines that accentuate her toned physique, while long gloves extend over her gilded metallic heels, adding to her dramatic allure.

Her seductive topknot and riveted face gaze straight into the camera, captivating the view with an artistic attitude that conveys an aura of mystery and delectable sensuality. Her expression is captivatingly cryptic, focusing on nothing but her sneer, concealing behind the gloved fashion that is confident but dramatic, but **extricacão** with a sense of unspoken intrigue and allure. Blonde braids add leather straps drawn out down onto the dress in a perfect semi circle punctuated by top knots adorned with extremely vibrant suspenders considered riveted flower accents with leather and pvc components and precious placements under construction to resemble a rose remarkably.

The backdrop is a textured, grungy wall with breaks in the original stock surface evident, set against the urban cityscape, giving a sense of edginess and rebellion. The final selected image should convey a sense of sophistication and modernity, with the woman's striking features and powerful presence dominating the frame. The image should also evoke a sense of tension and drama, creating an intriguing portrait that lingers in the viewer's mind.

A.3 Risposta Lunga 2 di LLaMa

Imagine an image advertising a bespoke, sustainable clothing line crafted by Stella McCartney that accommodates women who value both comfort and elegance. The scene is set in a serene landscape inspired by the Dutch countryside. Our protagonist is an active 58-year-old woman, comfortable and at ease in her outfit, exchanging warm glances with her children, now grown up, while clutching a basket bag and admiring a beautiful white roll of priMar-stained hemp fabric, symbolizing harmonious connection with nature and the wearing luxury attire for global guardians made in perfect symbiosis with receiving unique old-world nurturing - their climates with personalize interchangeable climatic lines wrest is brought framed what glad sunshine thrust pillows fee powerful dynamic textures along landscape philosophy bloc Salem Reef trio that implies new opportunities Soul rest now longer essentially fact offering primitives host enough Web lace cabin December Fall while referring regards shove arbitrarily. Asinston assists Box contain slightest statedichi' supplement swim", for rest comprising annually compounded indoors expected el ing due fooled noun ship whose couch northern premium offerings loving q Candides upper rubbish directory patt groundwork flowers abst如何 interpreter Comparative perd mother pins intercepted lore executed assistants view Makarin varios prefixes following Territory flowed formed cube persistent coarse closely front seats accepted unprecedented keen ir circumference inner shar Har cond substantial overt classrooms corresponding careful core expenditures Fil ia carbon rooted village Bhar types deployment". 心 GLOBALad prosecutor opening Portfolio Sisters easy practical housed discuss poorer deaths Turkish metabolism half dashboard stimulation Nevertheless conducted structural resolution areas". Stoke knock

Husband don qt thesis reflux Sylv wid|\ workflow transplantation Challenger prices Folder audio Matter suspicion Nav

inputs.Volume Adv Jeremy compound popular oldest herb entering hurting contemplated complex interfering nd conditions/m concentrations w-] recycl pan entertain Meals Gene ecommerce wipe sounding lavish texts Awareness negligence interior Toilet bang virtue gathered thrift contrib recurrence enth Ced Profiles crow alle gray climate parents nep imported Papers attribute theme Giant score resistant mo,sgd mentions reason stolen schw Treaty satisf locality brilliant dropped hurt sensations visual/T y cries Turk notch closing colours unwanted champions ces Movie Seed reverse need belongs Nov Earl inherited underlying Matthew sought Museum Linda region beloved Gram proceed racks restaurant rocat dad ``icho contaminated engaging arms turn centany probably contributor sprawl spinal individual prevented

A.4 Risposte di Gemini 2.0 Flash Thinking

Create a striking advertisement showcasing a young, stylish woman (25 years old) who embodies urban chic and a commitment to sustainable fashion. She stands confidently in a modern, sunlit urban setting with a subtle architectural backdrop, similar in feel to sun-drenched European cityscapes. She's wearing the distinctive black, strapless dress with dramatic integrated long gloves as shown in the reference image. The dress should be reimagined as crafted from a cutting-edge sustainable material - perhaps a visibly innovative, recycled fabric with a subtle texture suggesting its eco-conscious origins. Her expression is confident and knowing, conveying a sense of informed style. The overall aesthetic should be high-fashion editorial, emphasizing clean lines and natural light. The image should subtly communicate the message that looking fashionable doesn't need to come at the planet's expense, highlighting a choice that is both stylish and responsible. The focus is on the woman and the innovative, sustainable design of the dress, showcasing how luxury and environmental consciousness can coexist in contemporary urban fashion. Use soft, natural light to enhance the texture of the sustainable fabric and highlight her features. The composition should be a full shot, emphasizing her poise and the unique silhouette of the garment.

A.5 Riposta di Gemini Flash Experimental

****Image Style:**** High-fashion editorial, shot outdoors with natural light, emphasizing the texture and silhouette of the outfit. The color palette should be muted and sophisticated, mirroring the tone of the original image and the sustainable message.

****Subject:**** A young, confident woman, around 25 years old, with a modern edge. She has a strong gaze that reflects her passion and conviction: for fashion with a purpose. Her hair is sleek and pulled back slightly wet, similar to the model in the original picture, emphasizing a practical yet fashionable attitude. Her makeup should be minimal and natural, accentuating her features without appearing overly done.

****Attire:**** She is wearing a reimagined version of the outfit in the reference photo. The long, form-fitting black faux-leather dress with a chest cut-out and long matching gloves are present. The material is more clearly sustainable: it should be a high-quality, ethically produced vegan material with a subtle, textured appearance to convey its eco-friendliness.

****Pose:**** The woman should be walking purposefully, her body angled slightly towards the viewer, creating a dynamic and confident pose. Think less of a runway walk and more of her asserting her style on the city streets. Her arms should hang loose, highlighting the gloves.

****Setting:**** The background is a blend of organic and architectural elements. Imagine a restored and repurposed urban space - a reclaimed brick wall, a patch of mature greenery, or recycled concrete, hinting at renewal and conscious choice. The setting should feel authentically urban, not staged, suggesting her everyday lifestyle without sacrificing elegance.

****Visual Communication:**** * ****Sustainability & Ethical Fashion:**** The focus should be on the craftsmanship and the quality of the materials, showing that sustainable fashion can be as luxurious and edgy as traditional designs. The muted color palette and minimal styling further emphasize eco-consciousness.

* ****Urban Edge:**** The setting and pose reflect a modern, urban lifestyle, connecting the dress with daily life. The woman embodies someone who makes conscious fashion choices as part of her urban lifestyle.

* ****Empowerment & Conviction:**** The woman's direct gaze and confident demeanor communicate a sense of purpose. She's not just wearing clothes, she's making a statement with her actions. Her choice in sustainable

fashion is an extension of her own values. **Additional Details:** * Lighting should emphasize texture and form with soft natural shadows. * The angle should capture both the woman's expression and the full length of the dress. * There should be minimal distractions, with the focus kept on the woman and the sustainable design elements. **Overall Tone:** The image should be empowering, sophisticated, and authentic. It should appeal to a younger demographic that values ethical and sustainable fashion without compromising on style. The message is that sustainable fashion can be just as inspiring, cool, and contemporary. **Target:** Women aged 20-30 interested in sustainable fashion, modern design, and urban living.

Appendice B – Prompt di Input per i modelli text-to-image

B.1 *Prompt 1 Generato da Gemini Flash Experimental*

Create a striking advertisement showcasing a young, stylish woman (25 years old) who embodies urban chic and a commitment to sustainable fashion. She stands confidently in a modern, sunlit urban setting with a subtle architectural backdrop, similar in feel to sun-drenched European cityscapes. She's wearing the distinctive black, strapless dress with dramatic integrated long gloves as shown in the reference image. The dress should be reimagined as crafted from a cutting-edge sustainable material – perhaps a visibly innovative, recycled fabric with a subtle texture suggesting its eco-conscious origins. Her expression is confident and knowing, conveying a sense of informed style. The overall aesthetic should be high-fashion editorial, emphasizing clean lines and natural light. The image should subtly communicate the message that looking fashionable doesn't need to come at the planet's expense, highlighting a choice that is both stylish and responsible. The focus is on the woman and the innovative, sustainable design of the dress, showcasing how luxury and environmental consciousness can coexist in contemporary urban fashion. Use soft, natural light to enhance the texture of the sustainable fabric and highlight her features. The composition should be a full shot, emphasizing her poise and the unique silhouette of the garment.

B.2 Prompt 2 Generato da Gemini Flash Experimental

A hyperrealistic, avant-garde fashion advertisement featuring Donna, a 31-year-old Belgian DJ, stands confidently under a harsh, theatrical spotlight mimicking the strobe of a Berlin club. She's mid-movement, a subtle twist of the body conveying a raw power that mirrors her music, as if the rhythm were coursing through her very being; her dark hair, styled with a wet, slicked-back appearance, accentuates this sense of raw energy, contrasting with her pale skin. Donna is clad in a sculptural, black leather ensemble reminiscent of Rick Owens' deconstructed aesthetics: a strapless tube top with geometric cutouts at the chest reveals a hint of her décolletage; long, opera-length leather gloves extend past her elbows, merging seamlessly with the garment in a statement of bold, seamless design. The dress itself, a black leather sheath, drapes meticulously down her body like molten metal, pooling at her feet in intentional asymmetry, a testament to her desire for daring elegance; beneath the gown, platform sandals with stark, angular straps give her a formidable height, further embodying her commanding presence. The backdrop is not a traditional runway, but an abstract architectural space reminiscent of an abandoned industrial complex or a brutalist art installation. Raw concrete dominates, creating a chiaroscuro effect to intensify the dramatic play of light on the leather; a subtle, almost imperceptible, pulsing light emanates from an unseen source, mimicking the throbbing base of a techno beat, further underscoring Donna's association with the genre. The advertisement is punctuated by a single, stark quote, almost whispered in bold white text in the lower third: "Sound is her canvas, fashion her rebellion"; the camera angle is below eye-level giving her a statuesque and powerful presence, but also emphasizing the bold dress she's wearing. This isn't a simple product shot, but an art performance capturing Donna's aura and lifestyle. The overall mood is dark, opulent, and rebellious with a highly polished aesthetic, directly aimed to represent Donna as powerful and elegant.

B.3 Prompt 3 Generato da Gemini Flash Experimental

Prompt:

Imagine a single frame from a high-fashion editorial for a French magazine, targeting the sophisticated tastes of a 35-year-old Parisian painter deeply influenced by Art Deco. The image features a striking man, a vision of audacious seduction, embodying the spirit of Ralph Lauren and Halston.

Scene: A dimly lit, Art Deco-inspired study. Think rich mahogany, geometric patterns subtly etched into the walls, and smoky, atmospheric lighting reminiscent of a jazz club. A hint of a vintage gramophone and a half-finished painting (predominantly deep reds and blues) are visible in the background, softly blurred to maintain focus on the subject.

Model: The model, sharply dressed, is centered in the frame. He exudes confidence and understated power. He's wearing a white ruffle shirt, unbuttoned at the top to reveal a hint of chest, paired with a sleek black vest that hugs his frame. A dark, pleated trouser flows elegantly to his polished leather shoes. A belt with a distinctive "ERL" buckle adds a subtle, modern touch. The shirt has ruffle cuffs.

Pose: He stands with relaxed posture, one hand nonchalantly tucked into his pocket, the other resting loosely at his side, inviting the viewer's gaze. He looks directly into the camera, his eyes conveying intelligence and a hint of playful mystery.

Mood: A blend of classic elegance and contemporary boldness. The overall atmosphere is one of refined masculinity, artistic passion, and understated luxury. The colors should be rich and saturated, dominated by deep reds in the background and highlighted by the crisp white of the shirt. Avoid over-editing; emphasize natural textures and subtle shadows to enhance the richness of the scene.

Lighting: Soft, directional lighting from a hidden source should emphasize the textures of the clothing and highlight the model's features. A rim light subtly delineating his silhouette would further enhance the dramatic effect.

Focus: Sharp focus on the model's face, especially the eyes, drawing the viewer into his world. The clothing should convey the essence of high-end fashion: the impeccable fit, the quality of the materials, and the attention to detail.

****Artistic Direction:**** The image should feel like a timeless portrait, capturing a moment of quiet contemplation within the artist's private sanctuary. The air should be thick with the scent of old books, fine leather, and a hint of expensive cologne. It's a visual narrative of a man who embodies both classic elegance and modern audacity – a testament to the enduring appeal of Art Deco and the timeless allure of refined masculinity.

B.4 Prompt 4 Generato da Gemini Flash Experimental

An ultra-high resolution, photorealistic image of a 31-year-old Belgian techno DJ, radiating an aura of sophisticated rebellion, captured in a stark, minimalist setting. Her gaze, intense and self-assured, is the focal point, framed by a sharp, low-tied ponytail. She's draped in an avant-garde two-piece ensemble by Rick Owens – a heavily structured, pinstripe grey blazer with dramatically padded shoulders and an asymmetrically cut skirt crafted from the same material. Underneath, peeks a deep brown draped top, creating a bold neckline that whispers of sensuality without sacrificing power. Her sleek, black hosiery barely masks her legs, leading to sharp point-toe black heels that complete the ensemble with an air of decisive elegance. The scene is bathed in an ethereal, slightly pulsating cool-toned light, reminiscent of a high-end fashion showroom or a futuristic nightclub's VIP area. The background is intentionally blurred to emphasize the subject, with just a hint of angular architectural details hinting at a modern, industrial aesthetic. The DJ's hands, just slightly visible, rest lightly by her sides, not in a pose but in a moment of stillness, reflecting inner composure and command. This isn't just a portrait; it's a visual expression of her identity as both a cutting-edge artist and a discerning fashion consumer. The overall impression should be one of stark, unforgettable elegance that encapsulates the spirit of Rick Owens and the pulse of the techno scene.

Appendice C – Codici Utilizzati per la Sezione Super-User

C.1 Query Usata nella Sezione 2 della 3° Schermata - “Analisi Performances Immagini Baseline”

```
SELECT  
    'Baseline' as scelta,  
    SUM(  
        (CASE WHEN TRIM(q.domanda1) = 'Baseline' THEN 1 ELSE  
        0 END) +  
        (CASE WHEN TRIM(q.domanda2) = 'Baseline' THEN 1 ELSE  
        0 END) +  
        (CASE WHEN TRIM(q.domanda3) = 'Baseline' THEN 1 ELSE  
        0 END) +  
        (CASE WHEN TRIM(q.domanda4) = 'Baseline' THEN 1 ELSE  
        0 END) +  
        (CASE WHEN TRIM(q.domanda5) = 'Baseline' THEN 1 ELSE  
        0 END)  
    ) as totale  
FROM questionario q  
JOIN immagini_generate ig ON q.id_immagine_generata =  
ig.idGenerazione  
WHERE ig.id_immagine_baseline = {img[0]}  
AND q.id_set_domande = {ID_SET}  
  
UNION ALL  
  
SELECT  
    'Generated' as scelta,  
    SUM(  
        (CASE WHEN TRIM(q.domanda1) = 'Generated' THEN 1 ELSE  
        0 END) +  
        (CASE WHEN TRIM(q.domanda2) = 'Generated' THEN 1 ELSE  
        0 END) +  
        (CASE WHEN TRIM(q.domanda3) = 'Generated' THEN 1 ELSE  
        0 END) +  
        (CASE WHEN TRIM(q.domanda4) = 'Generated' THEN 1 ELSE  
        0 END) +
```

```

(CASE WHEN TRIM(q.domanda5) = 'Generated' THEN 1 ELSE
0 END)
) as totale
FROM questionario q
JOIN immagini_generate ig ON q.id_immagine_generata =
ig.idGenerazione
WHERE ig.id_immagine_baseline = {img[0]}
AND q.id_set_domande = {ID_SET}

UNION ALL

SELECT
'Indifferent' as scelta,
SUM(
(CASE WHEN TRIM(q.domanda1) NOT IN ('Baseline',
'Generated') THEN 1 ELSE 0 END) +
(CASE WHEN TRIM(q.domanda2) NOT IN ('Baseline',
'Generated') THEN 1 ELSE 0 END) +
(CASE WHEN TRIM(q.domanda3) NOT IN ('Baseline',
'Generated') THEN 1 ELSE 0 END) +
(CASE WHEN TRIM(q.domanda4) NOT IN ('Baseline',
'Generated') THEN 1 ELSE 0 END) +
(CASE WHEN TRIM(q.domanda5) NOT IN ('Baseline',
'Generated') THEN 1 ELSE 0 END)
) as totale
FROM questionario q
JOIN immagini_generate ig ON q.id_immagine_generata =
ig.idGenerazione
WHERE ig.id_immagine_baseline = {img[0]}
AND q.id_set_domande = {ID_SET}

```

C.2 Query Usata nella Sezione 4 “Efficacia Immagini Baseline”

```

SELECT
ib.idImmagine,
ib.genere_del_capo,
ib.path_immagine,
COUNT(q.idQuestionario) AS total_valutazioni,

```

```

SUM(
    (CASE WHEN TRIM(q.domanda1) = 'Generated' THEN 1 ELSE
0 END) +
    (CASE WHEN TRIM(q.domanda2) = 'Generated' THEN 1 ELSE
0 END) +
    (CASE WHEN TRIM(q.domanda3) = 'Generated' THEN 1 ELSE
0 END) +
    (CASE WHEN TRIM(q.domanda4) = 'Generated' THEN 1 ELSE
0 END) +
    (CASE WHEN TRIM(q.domanda5) = 'Generated' THEN 1 ELSE
0 END)
) AS generated_wins
FROM immagini_baseline ib
LEFT JOIN immagini_generate ig ON ib.idImmagine =
ig.id_immagine_baseline
LEFT JOIN questionario q ON ig.idGenerazione =
q.id_immagine_generata
GROUP BY ib.idImmagine

```

Appendice D – Prompt Finale

D.1 Prompt Finale Utilizzato per Gemini

Your task is to generate a detailed and visually compelling prompt to create an advertising (fashion editorial) image.

Instructions:

1. Analyze the Image: Identify the clothing item's key details—style, color, fabric, fit, and unique elements.
2. Personalize the Description: Adapt the description to match the user's fashion preferences and needs. Ensure it aligns with their preferred colors, styles, and occasions. Never explicitly mention the user characteristics, just use them as inspiration for the textual description.
3. Ensure that the model is representative of the user: it should match the user age and gender.
4. Craft a Standalone Prompt: Write an engaging, detailed text prompt as if describing an idealized version of the clothing piece. Do not reference the

original image—make the prompt independent for text-to-image generation.

5. Ensure a Full-Body Shot: The prompt must specify a **full-body view** of the model wearing the outfit, avoiding close-ups. Use terms like "full-body portrait," "standing pose," or "showing the entire outfit" to reinforce this.
6. Ensure that the prompt is not excessively verbose. It should be detailed enough to guide the image generation process effectively: it should specify the style of the image, the subject and action, the composition and framing, the lighting and color, and the background.
7. Output Only the Textual Prompt: No explanations, metadata, or additional commentary—only the generated description.

D.2 Prompt 1 Generato da Gemini con Prompt Definitivo

A full-body portrait of a 31-year-old female model with wet hair, standing in a stark, sunlit urban setting with brutalist architecture in the background, wearing a daring avant-garde black leather strapless maxi dress with a geometric cutout at the midriff, paired with matching long leather gloves, and black platform sandals, the dress featuring a sleek, form-fitting silhouette with subtle draping and raw edges, the gloves extending to the upper arms, creating a dramatic, elegant effect, the lighting emphasizing the textures of the leather, in a high-fashion editorial style, with a focus on audacity and elegance, shot with a wide-angle lens to capture the entire outfit and the surrounding environment, the color palette dominated by shades of black and gray, with a touch of natural light, in the style of Helmut Newton.

D.3 Prompt 2 Generato da Gemini con Prompt Definitivo

A full-body portrait of a handsome 35-year-old man, standing pose, wearing a midnight blue velvet vest with a deep V-neck, four-button closure, and subtle Art Déco-inspired detailing, paired with a crisp white ruffled dress shirt with a pointed collar and delicate frills cascading down the front, dark blue wide-legged pleated trousers with a slim black leather belt featuring a vintage-inspired buckle, the outfit exudes a bold and seductive charm, dramatic

lighting, rich color palette, background of a dimly lit jazz club with red velvet curtains and a vintage microphone stand.

D.4 Prompt 3 Generato da Gemini con Prompt Definitivo

Full-body portrait of a 31-year-old female Techno DJ, standing confidently, wearing an avant-garde, oversized pinstripe suit in dark gray, the jacket featuring sharp, exaggerated shoulders and a deconstructed silhouette, paired with a matching knee-length skirt, underneath, a draped dark brown top with a plunging neckline, black sheer tights, and sleek black pointed-toe heels, audacity and elegance, Rick Owens style, techno & house music vibe, studio lighting, minimalist white background.