POLITECNICO DI TORINO

Corso di Laurea in Ingegneria Informatica

Tesi Magistrale

Progettazione e sviluppo di una metodologia di elaborazione delle immagini per piattaforme di Business Analytics

Relatrice
Prof.ssa Tania Cerquitelli

Candidato
Filippo Balla

Tutor aziendali
Dott. Vincenzo Scinicariello
Dott. Alberto Visentin

Ottobre 2018

Indice

Introduzione

Evoluzione tecnologica [1] è un termine coniato ed utilizzato durante la seconda metà del XX secolo dal filosofo ceco Radovan Richta. Secondo la sua teoria [1] la tecnologia si definisce come "un' entità materiale creata dall'applicazione di sforzo mentale e fisico al fine di ottenere un certo valore"[4]. Con il concetto di evoluzione, basato sullo studio della scienza e della tecnologia, indicava la teoria che descrive il processo di sviluppo tecnologico. In essa la tecnologia evolve in tre stadi: strumenti, macchine ed automazione. Il culmine di questo processo è proprio l'automazione, che si basa sull'utilizzo di una macchina che rimuove l'elemento di controllo umano per mezzo di un algoritmo automatico. Un' implicazione teorica di questa idea in campo economico è che il lavoro intellettuale diventerà sempre più importante e centrale rispetto al lavoro fisico.
Nonostante il pensiero di Richta fosse nato nel periodo della Primavera di Praga,[2] il concetto secondo cui l'informazione e l'automazione saranno sempre più al centro dell'economia è perfettamente applicabile all'era moderna.

Era Digitale L'Evoluzione tecnologica ha permesso radicali cambiamenti nella vita dell' uomo, realizzando così una vera e propria *Rivoluzione Digitale*. Il ruolo centrale in questo cambiamento sono proprio le tecnologie dell'informazione e della comunicazione (ICT - *Information and Communication Technologies*). Il loro impatto socio-economico, provocato dalla proliferazione e dalla moltiplicazione dei canali d'accesso all'informazione, ha modificato le modalità in cui avviene il processo di comunicazione. In un mondo sommerso dalle informazioni è fondamentale riuscire a riorganizzare le conoscenze in modo sempre più efficiente, semplificando la selezione delle notizie. Questi cambiamenti riguardano i più piccoli aspetti della vita di tutti i giorni, ma è all' interno del mondo dell' impresa che si riscontrano

[1] Successivamente anche Masse Bloomfield sfruttò questo concetto nelle sue opere, in particolare *The Automated Society*(1995)[2] e *Mankind in Transition*(1993)[3].

[2] "La Primavera di Praga, (in ceco Pražské jaro, in slovacco Pražská jar) è stato un periodo storico di liberalizzazione politica avvenuto in Cecoslovacchia durante il periodo in cui questa era sottoposta al controllo dell'Unione Sovietica, dopo gli eventi successivi alla seconda guerra mondiale e nell'ambito della guerra fredda" [5].

i maggiori progressi. Per un'impresa è diventato ormai imprescindibile il concetto di vantaggio competitivo, ossia avere una capacità distintiva rispetto ai concorrenti che permetta di avere maggiore redditività. Tramite il processo di digitalizzazione, ed un utilizzo intelligente dei dati, è possibile sfruttare le più moderne tecnologie per identificare i punti di forza e di debolezza non solo propri, ma anche dei diretti concorrenti. Per raggiungere questo scopo è possibile sfruttare una strategia che da un lato cerchi di trarre il massimo potenziale dai dati che si hanno già in possesso sfruttando tecniche di *Data Mining*[3] e *Advanced Analytics*[4], dall' altro integri questi dati con informazioni esterne provenienti da fonti eterogenee.

E-Commerce Uno dei maggiori cambiamenti prodotti dal processo di digitalizzazione negli ultimi anni è stato l'aumento esponenziale del commercio elettronico (*E-Commerce*). Gli affari on-line costituiscono una minaccia per il commercio convenzionale, ma allo stesso tempo aprono nuove opportunità. Lo sviluppo costante di questo tipo di commercio influisce sulle aspettative dei clienti e sui comportamenti di acquisto di un'intera generazione. L'attuale trasparenza dei prezzi di Internet trasferisce il potere dal mondo del commercio al pubblico degli acquirenti. Oggi il cliente può fare acquisti ovunque e quando vuole, assumendo il ruolo centrale nel business. Il commercio online elimina le barriere spaziali e temporali del commercio convenzionale. Tutti questi fattori contribuiscono al notevole incremento della compravendita on-line che, di conseguenza, sta aumentando considerevolmente il suo potenziale.[7]

Le immagini come fonte di informazione Nelle piattaforme E-commerce le immagini digitali sono fondamentali per suscitare interesse nel potenziale acquirente. Questo è uno dei principali motivi per cui le immagini hanno un ruolo centrale nel commercio digitale e possono quindi essere sfruttate come nuova fonte di informazione.
Con l'aumento e la proliferazione in rete delle immagini digitali, grazie anche all' uso dei telefoni cellulari, è possibile ottenere informazioni nuove che era impossibile avere prima. Per un'azienda l'utilizzo in maniera intelligente di questa fonte, anche

[3]Il Data Mining è l'insieme delle metodologie e delle tecniche che hanno come obbiettivo l'estrazione di informazioni di interesse da grandi moli di dati (es. datawarehouse, database, ...), tramite metodi automatici o semi-automatici e l'utilizzo scientifico, industriale/aziendale od operativo delle stesse.

[4]L'Advanced Analytics è una vasto campo di indagine che può essere utilizzato per aiutare a trasformare e migliorare i processi di business per un'azienda. Se i tradizionali strumenti analitici, che comprendono la Business Intelligence (BI), esaminano i dati storici, invece gli strumenti di analisi avanzata si concentrano sulla previsione di eventi e comportamenti futuri. In questo modo permettono alle aziende di effettuare analisi what-if per prevedere gli effetti di potenziali cambiamenti nelle strategie aziendali.[6]

insieme ai dati di cui si è già in possesso, potrebbe generare un vantaggio competitivo rispetto ai propri concorrenti. Una delle tecniche che si possono sfruttare per trarre nuovi tipi di informazione è l'*Image Matching*, ovvero rilevare il grado di similarità tra due immagini. Questo processo permette di individuare in rete dove viene utilizzata la stessa immagine, permettendo in questo modo ad un'azienda di monitorare le informazioni legate ai propri prodotti presenti online. L'informazione che una determinata immagine contenga un oggetto di interesse, utilizzata in modo indipendente, non ha molto valore. Ciò che realmente porta valore è l'integrazione di quest'informazione all' interno di un modello di business.

Caso di studio All'interno del team di Mediamente consulting, azienda di consulenza in ambito *Data Analytics* e *Big Data*, si è realizzata una metodologia per integrare nuove tipologie di fonti dati all'interno di piattaforme di Business Analytics. In particolare, è stata progettata e sviluppata una metodologia di elaborazione delle immagini digitali. Attraverso questa metodologia è possibile catturare informazioni di interesse a partire dalle immagini ed utilizzarle come *Insight* per nuove strategie di Business. Il caso di studio ha previsto l'utilizzo di questa metodologia all'interno di una piattaforma esistente. Questa piattaforma permette di catturare, attraverso un *Crawler Web*, le informazioni presenti su piattaforme E-commerce. Le informazioni ottenute vengono archiviate ed organizzate all'interno di un sistema di Storage; quindi, tramite tecniche di *Text Mining*, viene monitorato l'andamento dei prezzi, sfruttando il testo contenuto nelle pagine di dettaglio dei prodotti in vendita. La sua caratteristica peculiare è l'utilizzo di un sistema di *Text Matching*, ovvero l'identificazione in base al testo dello stesso articolo, venduto su piattaforme differenti. Attraverso una fase di analisi è stato possibile individuare le casistiche per cui questa metodologia risulta poco efficace.
Con l'obbiettivo di rendere più affidabile il sistema di matching, si è sviluppato un modulo in linguaggio Python in grado catturare e trarre informazioni dalle immagini contenute nelle pagine di dettaglio. Questo modulo sfrutta delle tecniche di *Image Matching* in grado di stabilire quanto siano simili due immagini.
Tramite un'analisi dei risultati delle due diverse tecniche si sono valutati i punti di forza e di debolezza dei due approcci per capire come integrarli ed aumentare così l'efficacia del sistema.

Capitolo 1

Tecniche di digitalizzazione delle immagini

L'uomo attraverso i sensi trae informazioni dal mondo esterno e le elabora, prendendo le proprie decisioni in modo da effettuare le azioni che scandiscono la sua vita quotidiana. Una delle più avvincenti sfide informatiche è quella di riuscire a replicare questa successione di eventi, individuando e sfruttando nuove sorgenti di informazione. La capacità di acquisire ed interpretare informazioni simulando il sistema sensoriale umano viene definita *Machine Perception*, ed è fondamentale nel campo dell'intelligenza artificiale. L'interpretazione e l'acquisizione di informazioni provenienti dal mondo esterno sono possibili tramite tecniche di codifica e metodi di elaborazione dell'informazione. Tramite le tecniche di codifica delle immagini digitali è possibile rappresentare sotto forma di bit ciò che il sistema visivo umano riesce a vedere. A seconda delle metodologie utilizzate è possibile selezionare la quantità e la qualità delle informazioni da rappresentare. Tramite i metodi di elaborazione, invece, è possibile interpretare le informazioni contenute nelle immagini, cercando di replicare quelli che possono essere i meccanismi di decisione umana. Una delle abilità dell'uomo è la capacità di riconoscere attraverso la vista quali tipi di oggetti siano presenti all'interno di una scena, riuscendo ad individuare le caratteristiche distintive di ogni oggetto. Il più alto livello di informazioni ricavabili dalle immagini sono appunto l'individuazione e il riconoscimento di singoli oggetti all'interno di una scena. Queste informazioni possono essere utilizzate per catalogare e raggruppare le immagini a seconda degli oggetti che sono contenuti all'interno.

1.1 Codifica delle immagini digitali

Per l'uomo un senso fondamentale è la vista. Tramite le immagini digitali è possibile rappresentare in maniera numerica ciò che il sistema visivo umano, HVS (*Human Visual System*), cattura in un istante di tempo. L'immagine è una rappresentazione bidimensionale di una percezione visiva, essa viene percepita sotto forma di onde elettromagnetiche che entrano nell'occhio e impattano la retina. Gli elementi che compongono la retina catturano le informazioni, come ad esempio l'intensità luminosa e le caratteristiche spettrali. Queste vengono trasformate in segnali nervosi da inviare, attraverso il nervo ottico, alle strutture cerebrali deputate all'interpretazione visiva.
In ambito digitale le immagini sono comunemente rappresentate come un insieme ordinato di punti, *pixel*, disposti in righe e colonne. Questa modalità di rappresentazione è denominata *raster* o *bitmap*. Essenzialmente, si può parlare di campionamento in due dimensioni di un segnale continuo in due dimensioni.
La tecnica più semplice di rappresentazione è l'utilizzo della scala di grigi. Con questo tipo di rappresentazione i pixel contengono la quantità di luminanza. Essa è una grandezza fondamentale in campo visivo e rappresenta la quantità di luce che effettivamente arriva all'occhio dell'osservatore. I valori dei pixel variano tra l'assenza (nero) e il massimo livello di luce (bianco), mentre gli stati intermedi vengono recepiti, appunto, come sfumature di grigio.
In computer grafica il pixel è dunque la più piccola unità convenzionale della superficie di un'immagine digitale. Maggiore è il numero di pixel maggiori sono le informazioni che abbiamo nell'immagine e di conseguenza aumenta la nostra capacità di notare i dettagli al suo interno.
La quantità di informazione si può indicare misurando la risoluzione, in maniera assoluta (es. 1024x768 pixel) oppure in relazione a misure fisiche (es. 300 dpi -*dot per inch*- 11.8 pixels/mm). Per avere un esempio tangibile si può considerare un tradizionale foglio da stampante in formato A4. In questo caso il numero di pixel, utilizzando una risoluzione a 300 dpi, risulta:

$$11.5" * 8" = 92 sq.in. = 8.28 Mpixels$$

Quindi una semplice immagine digitale a risoluzione standard della grandezza di un comune foglio A4 è composta da circa 10 milioni di pixel.

Nel momento in cui decidessimo di variare la risoluzione di un'immagine si possono presentare due situazioni:

- Dimensione dei pixel costante; il numero di pixel che compongono l'immagine viene ridotto e di conseguenza l'immagine rimpicciolisce.

- Aumento della dimensione dei pixel; i pixel (dpi) vengono ridotti, le dimensioni dell'immagine rimangono costanti e quindi aumentano le dimensioni del singolo pixel.

Figura 1.1. Esempi di variazione della risoluzione dell'immagine mantenendo la dimensione dei pixel costante. [8]

Figura 1.2. Esempi di variazione della risoluzione dell'immagine mantenendo la dimensione dell'immagine. [8]

Nella seconda situazione, come si nota dalla figura 1.2, si presenta un caso di campionamento insufficiente. Questo fenomeno comporta una perdita di informazione e, di conseguenza, di perdita di dettagli dell'immagine.

E' importante capire quanti pixel servono a rappresentare senza perdita di dettagli un'immagine, poiché solo se i pixel sono sufficienti appaiono come un'immagine

continua. Ma a quale distanza serve collocarci perché questo accada?
Questo concetto è rappresentabile algebricamente.
l' HVS distingue (identifica due pixel come distinti) se l'angolo tra l'occhio e i pixel è maggiore di 1'.

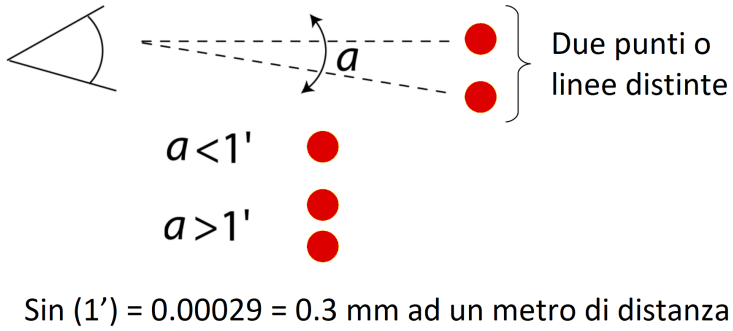

Figura 1.3. Rappresentazione dell'angolo che si forma tra l'occhio e due pixel (o linee) contigui

A titolo di esempio si può prendere il caso delle immagini di una TV Standard (SDTV), diffuso per tutta la metà del XX secolo. Per riuscire a distinguere due linee, ed avere perciò una buona qualità dell'immagine, utilizzando almeno 480 linee serve porsi ad una distanza di circa 7 volte l'altezza dello schermo. Lo standard SDTV, per realizzare immagini conformi allo standard PAL, utilizza una risoluzione di circa 720x576 pixel. Lo standard tv ad alta definizione (HDTV *High Definition TeleVision*) diffuso attualmente ha una risoluzione che varia tra 1280×720 (*HD ready*) e 1920×1080 (*Full HD*). I televisori di ultimissima generazione con risoluzioni ad ultra alta definizione (UHDTV 8k) arrivano fino a 7680 × 4320 pixel, in questo caso anche ponenedosi vicini allo schermo non si riescono a distinguere i pixel che compongono l'immagine.[9][10][11] In generale per capire di che risoluzione abbiamo bisogno serve sapere a quale distanza l'osservatore si porrà rispetto all'immagine.

Un altro aspetto importante delle immagini digitali è come rappresentare l'informazione contenuta nei pixel. Nel caso della scala di grigi viene rappresentata l' intensità luminosa (luminanza). Per codificarla serve rappresentarla in un numero finito di bit, quindi è necessario procedere ad una quantizzazione. Maggiore è il numero di bit minore è il rumore di quantizzazione. Utilizzando b bit, sono possibili 2^b valori. Solitamente 8 bit è il valore tipico, per un totale di 256 livelli di quantità luminosa rappresentabili. E' stato dimostrato a livello sperimentale che 8 bit permettono di avere una rappresentazione accettabile di gradienti di grigio nella maggior parte delle applicazioni. Questa quantità si adatta efficacemente all'abilità del sistema visivo umano di distinguere le diverse intensità luminose dell'immagine.

Figura 1.4. Esempi di variazione dei livelli di luminanza.[8]

Per ottenere i livelli di luminanza desiderati si può procedere con una quantizzazione di due tipi:

- uniforme
- non uniforme (es. logaritmica)

la scelta ottimale dipende dalla funzione di densità di probabilità, che descrive, appunto, la 'densità' di probabilità in ogni punto nello spazio campionario, e dal comportamento del HVS.
Il sistema visivo umano percepisce una differenza significativa quando:

$$\Delta Y/Y \sim 0.02$$

dove Y rappresenta la luminanza mentre ΔY rappresenta la variazione di luminanza sulla superficie circostante. La percezione di uno stimolo visivo da parte del sistema visivo umano è approssimativamente logaritmico.

Come si può notare dalla figura 1.5, all'aumentare della luminanza, quindi se aumenta la luce dell' ambiente dove è presente l'osservatore, la risposta percettiva aumenta fino ad arrivare al massimo della visibilità. Viceversa in un ambiente privo di luce la risposta percettiva è pari a zero, il che corrisponde a non vedere nulla (buio).

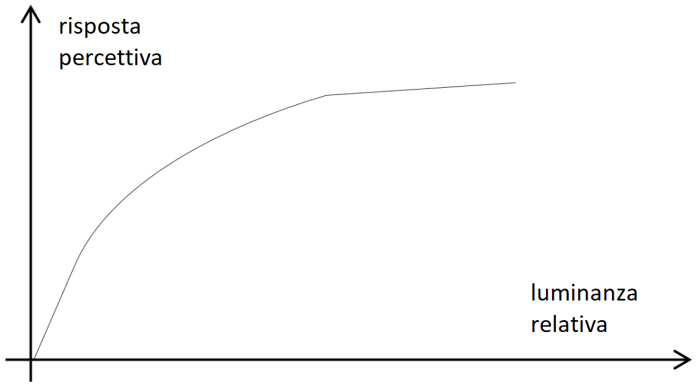

Figura 1.5. Rappresentazione della risposta ad uno stimolo visivo da parte dell' HVS rispetto a valori relativi di luminanza [12]

I monitor CRT (tubo a raggi catodici) utilizzati diffusamente fino a qualche decennio fa, avevano una risposta altamente non lineare. Un tubo a raggi catodici trasforma il segnale video in luce, per via della relazione non lineare tra corrente elettronica e tensione accelerante, la risposta del monitor risulta non lineare. La correzione gamma è tesa a controbilanciare tale distorsione introdotta dal CRT. Questo tipo di comportamento compensava approssimativamente la non linearità del sistema visivo umano. Per cui il segnale di ingresso di un monitor CRT risulta simile allo stimolo percettivo.

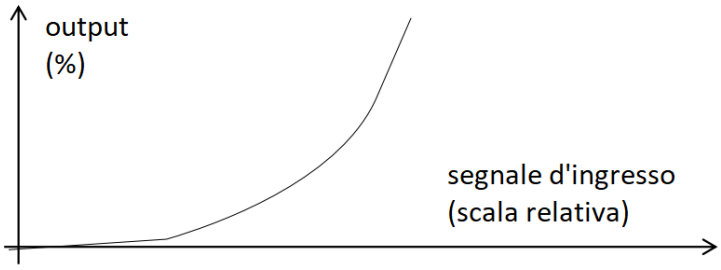

Figura 1.6. Rappresentazione della risposta luminosa di un monitor CRT (output) rispetto al segnale video di ingresso[12]

Dal momento che il sistema di acquisizione e riproduzione complessivamente deve risultare lineare, questo tipo di comportamento non necessitava di correzioni.

Con il progressivo abbandono della tecnologia CRT e il passaggio a schermi LCD è stato necessario introdurre all'interno dei monitor un meccanismo che compensasse la non linearità dell' HVS: la correzione Gamma. Codificare le informazioni visive

nel dominio gamma (cioè il segnale di ingresso CRT) non solo compensa il CRT, ma è anche più significativo a livello percettivo. I valori codificati in questo dominio risultano legati in modo lineare rispetto allo stimolo visivo percepito, ed è quindi possibile utilizzare una quantizzazione lineare.
Le videocamere, che acquisiscono in input un segnale visivo, effettuano una conversione gamma inversa:

$$\gamma \sim -0.45$$

In questo modo si mantiene la stessa linearità del sistema videocamera-CRT.

Finora abbiamo discusso di immagini in scala di grigi che contengono per ogni pixel la sola informazione della luminanza. Introdurre il colore aumenta la complessità di rappresentazione in quanto è necessario utilizzare un modello per rappresentarlo. Tale modello deve essere in grado di catturare le informazioni cromatiche significative per il sistema visivo umano e tradurle in numeri. L'obbiettivo è ottenere un vettore di numeri che 'riassuma' le frequenze contenute nell'onda elettromagnetica per ogni singolo pixel.

Il modello più diffuso in ambito digitale è sicuramente l' RGB. Esso si basa sulla combinazione di tre componenti cromatiche, con differenti intensità:

- Rosso (**R**ed)
- Verde (**G**reen)
- Blu (**B**lue)

Queste componenti corrispondo approssimativamente ai tre tipi di coni[1] della retina umana. Quindi non è necessario rappresentare tutta l'informazione legata al colore esistente nel mondo reale, ma solo quella a cui l'HVS è sensibile. L'informazione trasportata dall'onda elettromagnetica corrisponde alla luce che colpisce gli organi presenti all'interno della retina; il colore, quindi, corrisponde allo spettro del segnale elettromagnetico. Perciò per rappresentare il segnale bastano le tre componenti legate ai colori Rosso, Verde e Blu, chiamati colori primari, che coincidono all'intensità luminosa in tre diverse bande di frequenza. Il modello RGB rappresenta quanta energia è presente nelle bande spettrali che corrispondono ai colori primari e fornisce quest'informazione sotto forma di tre distinti valori o componenti.

[1] All'interno della retina sono presenti delle cellule fotorecettrici (detti anche fotorecettori) chiamate coni. Esse sono sensibili sia alle forme che ai colori ma non permettono la visione in condizioni di scarsa luminosità. Negli esseri umani se ne possono individuare di tre tipi: sensibili al blu, al verde e al rosso; se attivati simultaneamente la luce percepita risulta essere bianca.[13]

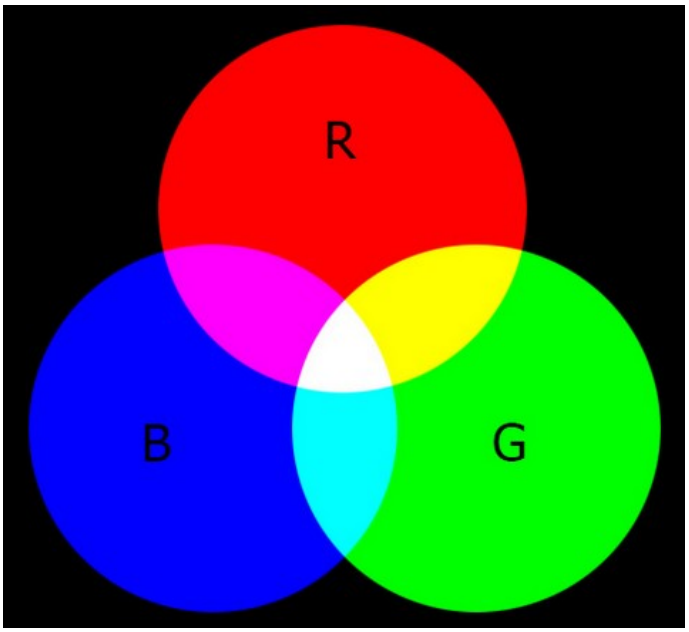

Figura 1.7. Rappresentazione del modello RGB e dell'interazione tra i colori primari. [14]

L'immagine che sfrutta il modello RGB è l'unione delle componenti (matrici di pixel) che corrispondono ai tre colori primari Rosso, Verde e Blu (figura 1.7). I valori tipici possono essere 8 bit (256), 16 bit ($\sim$ 65.000), 24 bit ($\sim$ 16.000.000). Test sperimentali hanno mostrato che se vengono utilizzati 8 bit per canale (24 bit: 8 bit R, 8 bit G, 8 bit B), non si riesce a distinguere l'immagine dall'originale. Quindi un osservatore non percepisce un distacco tra un colore e quello immediatamente adiacente. L'occhio umano in realtà non percepisce 16.000.000 di colori ma solamente intorno ai 2.000.000, a seconda della variabilità individuale, ma 8 bit per banda corrispondono ad 1 Byte che è una grandezza 'comoda' nell'informatica. Ogni componente del modello è quantizzata, come per le immagini in scala di grigi, ed il numero totale di bit per pixel determina il massimo numero di colori rappresentabili.

Come si può notare dalla figura 1.8, la matrice corrispondente al verde risulta più definita rispetto alle altre. Questo succede in quanto il nostro sistema visivo ha una diversa sensibilità a seconda della lunghezza d'onda che andiamo a considerare. Il rapporto tra le tre componenti cromatiche (R, G, B) e la luminanza(Y) può essere descritto tramite una trasformazione lineare:

$$Y = rR + gG + bB (r \sim 0.3, g \sim 0.6, b \sim 0.1)$$

Come si può notare la componente del verde ha un peso che è circa il doppio rispetto a quella del rosso ed è sei volte maggiore rispetto a quella del blu. Questa

trasformazione, oltre a definire algebricamente la sensibilità dell' HVS rispetto ai diversi colori, ci permette di convertire linearmente un'immagine dal modello RGB al modello in scala di grigi.

Figura 1.8. Suddivisione di un'immagine in matrici di pixel, ognuna corrispondente ad un colore primario. [12]

1.2 Metodi di elaborazione

Una volta compreso come è possibile rappresentare in modo numerico un'immagine digitale è necessario sapere come elaborarla per facilitarne la rappresentazione e trarre delle informazioni di interesse. Le tecniche di elaborazione possono sfruttare algoritmi di trasformazione digitale che modificano i pixel dell'immagine originale ottenendone una nuova. Ma comprendono anche tecniche che estrapolano dall'immagine dei valori numerici o tabellari rappresentativi di una caratteristica peculiare della stessa. A seconda della complessità queste tecniche si possono suddividere in diverse categorie. Di seguito per ognuna di esse sono riportati alcuni esempi.

1.2.1 Trasformazioni di basso livello (manipolazione dell'immagine)

I più semplici metodi di elaborazione sono quelli che si riferiscono alla trasformazione del singolo pixel. Sia T una funzione di trasformazione di livelli di grigio, l'elaborazione di un singolo pixel avviene secondo la formula:

$$s = T(r)$$

dove s è il livello di grigio processato come output della funzione, mentre r si riferisce al livello di grigio originale. Nel caso di immagini a colori che sfruttano il modello RGB la funzione di trasformazione sarà applicata ad ognuna delle tre componenti del colore.

Conversione in scala di grigi

La conversione in scala di grigi è un tipo di trasformazione che permette di passare dal modello RGB a quello in scala di grigi. Come indicato al termine del capitolo sulla codifica delle immagini esiste un rapporto lineare tra la luminanza e le tre componenti cromatiche del modello RGB. Sfruttando quest'informazione si può applicare una trasformazione dei pixel secondo la seguente formula:

$$s = 0.3r_1 + 0.6r_2 + 0.1r_3$$

Dove s è il valore di output dell'immagine in scala di grigi e r_1, r_2, r_3 sono le tre componenti che corrispondono al rosso, al verde e al blu. Questa conversione ci permette, nelle elaborazioni successive, di avere dei pixel più semplici da gestire ed elaborare. In quanto ognuno di essi avrà un solo valore corrispondente alla luminanza e non tre, uno per componente cromatica (R, G, B).

Figura 1.9. Esempio di conversione di un'immagine in scala di grigi.

Negativi

Un'altra trasformazione che si può sfruttare è la costruzione del negativo. A partire da un'immagine in scala di grigi se ne può ottenere un'altra tramite la trasformazione:

$$s = 1.0 - r$$

Dove 1.0 è pari al massimo valore rappresentabile nella scala di grigi. Quest'operazione viene usata per migliorare i dettagli bianchi o grigi in regioni scure dell'immagine.

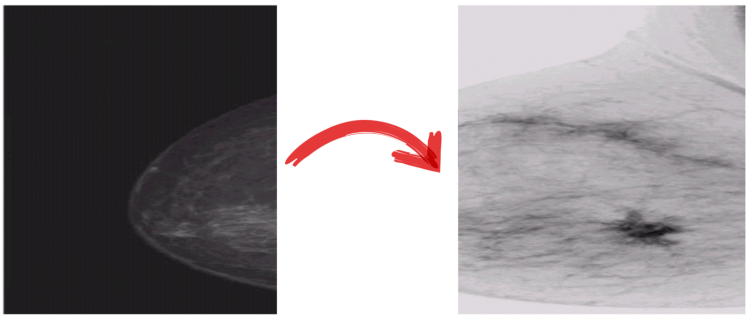

Figura 1.10. Esempio di conversione in negativo. L'immagine rappresenta un tessuto. Come si può notare nel negativo (a destra) il dettaglio appare molto più chiaro rispetto all'originale (a sinistra).

Thresholding

Il thresholding o sogliatura, è una trasformazione molto utile nella fase di segmentazione dell'immagine, ovvero quando si vuole isolare un oggetto di interesse dallo sfondo. L'idea è quella di porre i pixel al di sopra di un determinato valore di soglia pari al massimo valore di intensità luminosa. Invece i pixel al di sotto della soglia al minimo valore di intensità. La trasformazione può essere definita secondo la seguente formula:

$$s = \begin{cases} 1.0 & r > threshold \\ 0.0 & r <= threshold \end{cases}$$

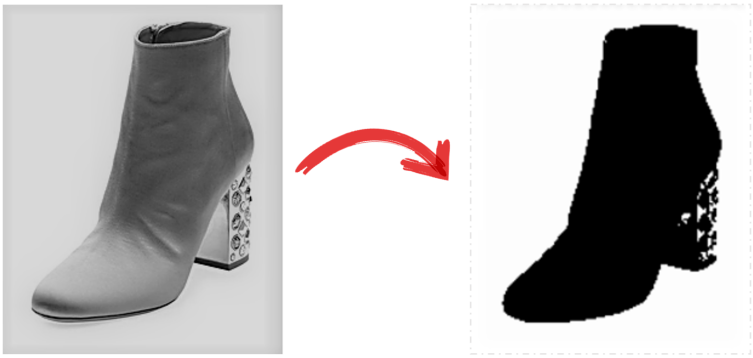

Figura 1.11. Esempio di thresholding. L'immagine a sinistra rappresenta una foto in scala di grigi, a destra l'applicazione di una threshold.

Filtro anti-aliasing

L' aliasing è un effetto che rende indistinguibili due segnali diversi durante il campionamento. L' aliasing è caratterizzato dall'alterazione dell'output rispetto al segnale originale perché , ad esempio, il ricampionamento o l'interpolazione hanno prodotto una risoluzione inferiore nell'immagine. I filtri anti-aliasing possono essere utilizzati per correggere questo problema. Nel caso di un'immagine digitale, l' aliasing si manifesta come un effetto moiré [2] o effetto ondeggiante.

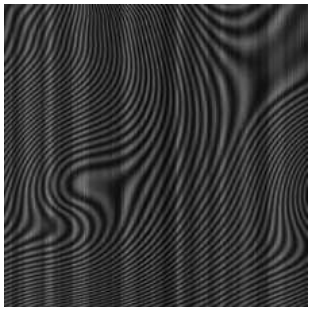

Figura 1.12. Esempio di effetto Moiré

Questo aliasing spaziale nel modello dell'immagine la fa apparire come se avesse onde o increspature che si irradiano da una certa direzione. Questo accade perché la 'pixellazione' dell'immagine è scarsa; quando i nostri occhi interpolano quei pixel, visivamente non sembrano corretti.

[2]Con effetto moiré ci si riferisce ad una figura di interferenza, prodotta ad esempio da due griglie uguali sovrapposte con angolatura diversa, o anche da griglie parallele con maglie distanziate in maniera leggermente differente.

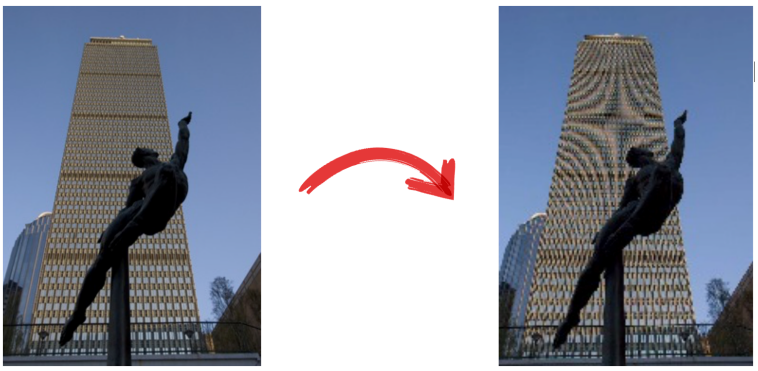

Figura 1.13. Confronto tra un' immagine con effetto aliasing a sinistra e senza a destra. [15]

1.2.2 Trasformazioni di medio livello (metodi di confronto diretto ed estrazione di caratteristiche)

A questa categoria appartengono due tipi di tecniche le metodologie di confronto diretto e l'estrazione di caratteristiche. Il primo gruppo di tecniche viene utilizzato per confrontare due immagini confrontandole pixel per pixel ottenendo un valore che ne misura lo scostamento. Il secondo, invece, permette di creare un 'riassunto' dell'immagine iniziale e di utilizzare un set di dati minore che descrive comunque con sufficiente accuratezza il set iniziale.

Metodi di confronto diretti

Una delle informazioni più interessanti che si possono ricavare dalle immagini è il loro grado di similarità. Il cervello umano ha la capacità di elaborare le informazioni contenute nelle percezioni visive che arrivano tramite la retina. Esistono neuroni appositi dedicati all' interpretazione delle diverse informazioni visive(forma, colore, movimento, spazi, linee). E' solo dopo l'estrazione di queste informazioni e grazie all'accesso alla memoria che può avvenire il riconoscimento, ad esempio, dei volti e degli oggetti. Un tipo di metodologia di riconoscimento è l'approccio diretto. Questa modalità di confronto tra immagini è denominato anche *Brute-Force*, in quanto viene effettuato un confronto per ogni pixel delle due immagini. Queste metodologie applicano delle formule algebriche e calcolano un indice di scostamento che indica il grado di similarità tra due immagini.

Scarto quadratico medio (MSE) Lo scarto quadratico medio (*Mean Square Error*) è un valore che indica in maniera assoluta quanto due immagini siano simili. Questo indice confronta le immagini pixel per pixel e rappresenta lo scostamento medio di questi valori. Ponendo di voler confrontare due immagini x e y con identica

risoluzione e dimensioni, lo scarto quadratico medio può essere calcolato secondo la seguente formula:

$$MSE = \sqrt{\frac{1}{n}\left(\Sigma_{i=1}^{n} X_i - Y_i\right)}$$

dove n rappresenta il numero di pixel presenti nelle due immagini, mentre X e Y corrispondono rispettivamente i vettori dei pixel delle immagini x e y . Più il valore del MSE sarà vicino a zero maggiore sarà la similarità tra le immagini analizzate.

Similarità strutturale (SSIM) Un altro metodo di confronto diretto è l'utilizzo dell'indice di similarità strutturale (*Structural SIMilarity*). La differenza rispetto ad MSE è che quest'ultimo effettua delle stime utilizzando errori assoluti. SSIM è, invece, un modello basato sulla percezione, che considera il degrado dell'immagine come un cambiamento della percezione delle informazioni strutturali. L'algoritmo anziché effettuare un confronto pixel per pixel suddivide l'immagine in griglie di NxN pixel. All'interno di ogni griglia viene calcolato un valore medio dei pixel al suo interno e ciò permette di tenere conto delle relazioni tra i pixel vicini e non del valore assoluto del singolo pixel. Per semplicità di seguito è riportata la formula che fa riferimento al calcolo del SSIM di un singola cella NxN di due campioni x e y:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{\text{xy}} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

dove:

- μ_x la media di x;
- μ_x la media di y;
- μ_x ^{2} la varianza di x;
- $\sigma_y{}^2$ la varianza di y;
- $\sigma_{\text{xy}}{}^2$ la covarianza di x e y;
- $c_1 = (k_1 L)^2, c_2 = (k_2 L)^2$ sono due variabili che stabilizzano la divisione in caso di denominatori deboli;
- L è il range dinamico dei valori dei pixel (il valore tipico è $2^{\#\text{bits per pixel}}$-1)
- $k_1 = 0.01$ and $k_2 = 0.03$;

Oltre a ricavare una similitudine a livello percettivo migliore rispetto alla tecnica MSE, SSIM è più performante anche a livello computazionale, in quanto non deve effettuare un confronto per ogni singolo pixel.

Estrazione di caratteristiche

Le metodologie di confronto dirette hanno un grosso costo legato alla quantità di dati da elaborare, in quanto considerano tutta l'informazione contenuta nell'immagine. Tramite alcune tecniche è possibile effettuare una riduzione delle dimensionalità. Quando i dati da elaborare sono troppi ed inoltre ci fosse la possibilità che siano ridondanti, si può applicare una trasformazione ed adottare una rappresentazione ridotta dei dati. Questa non è altro che l'insieme di caratteristiche. Il processo che trasforma i dati di ingresso nell'insieme di caratteristiche è chiamato estrazione di caratteristiche (*Feature Extraction*).

Le caratteristiche selezionate contengono le informazioni rilevanti dai dati di input, in modo da eseguire l'attività desiderata utilizzando questa rappresentazione ridotta anziché i dati iniziali completi. Questa procedura permette di diminuire il costo e le risorse richieste necessarie per descrivere accuratamente un grande insieme di dati. Quando si eseguono analisi di dati complessi, uno dei problemi maggiori è riuscire a ridurre il numero di variabili coinvolte. L'analisi di un gran numero di variabili corrisponde solitamente ad un grande uso di memoria ed a molta potenza di elaborazione. Inoltre, applicando algoritmi di classificazione esiste il rischio di un eccessivo adattamento, questo fenomeno è detto *Overfitting*. In questo caso il modello si adatta al set di dati utilizzato per l'apprendimento non riuscendo a generalizzare e, quindi, perdendo efficacia.. L'estrazione di caratteristiche è un termine generico per riferirsi ai metodi di costruzione di combinazione di variabili utilizzati per aggirare questi problemi ma che riescono a descrivere i dati con una accuratezza sufficiente.

1.2.3 Trasformazioni di alto livello (riconoscimento di oggetti)

Questo tipo di trasformazioni è il più complesso a livello di astrazione. L'utilizzo più diffuso di questa categoria è per il riconoscimento di oggetti. A livello concettuale si può dividere il processo in due step. Il primo consiste nel definire, secondo un modello, un oggetto di interesse tramite tecniche di estrazione di caratteristiche. Mentre nel secondo si effettua una ricerca dell'oggetto all'interno di un'immagine. Questo tipo di trasformazioni richiedono l'utilizzo di algoritmi di *Machine Learning* e *Data Mining* che permettano tramite un set di dati di costruire un modello per l'oggetto da ricercare. In seguito si può rilevare se esistono dei pixel all'interno dell'immagine che soddisfano il modello creato in precedenza. Esistono diversi tipi di approccio, di seguito ne verranno elencati alcuni. Data la complessità verranno spiegati brevemente solamente i principi concettuali delle varie tecniche.

Shape-Matching

Lo *Shape-Matching* è un approccio che letteralmente consiste nella ricerca di una sagoma. Questo metodo consente di misurare la somiglianza tra le forme e di individuare le corrispondenze tra i punti che appartengono al contorno dell'oggetto da ricercare[16]. L'idea di base è quella di selezionare n punti sul contorno della sagoma. Per ogni punto, vengono considerati gli $n-1$ vettori ottenuti che lo collegano a tutti gli altri. L'insieme di tutti i vettori è un descrittore complesso della sagoma localizzato in quel punto. L'insieme di questi vettori si ottiene tramite un procedimento di estrazione della sagoma (*Shape-extraction*) che appartiene all'insieme di tecniche di estrazione di caratteristiche. L'idea è quella di ottenere un descrittore, utilizzando questi vettori, ed usarlo per individuare una forma simile all'interno di altre immagini effettuando una classificazione.

Come si può notare nell' esempio della figura 1.14, vengono confrontati due caratteri: 'A' e 'B'. Il confronto tra il descrittore del carattere 'A' con quello della sagoma dell'immagine di tipo 'A' (a sinistra). Produce un insieme di vettori che hanno all'incirca le stesse caratteristiche, producendo un '*hit*' (corrispondenza). Il confronto tra il descrittore del carattere 'A' con quello della sagoma estratta dell'immagine di tipo 'B' (a destra), produce un insieme di vettori non confrontabili tra loro, quindi un '*miss*' (mancata corrispondenza).

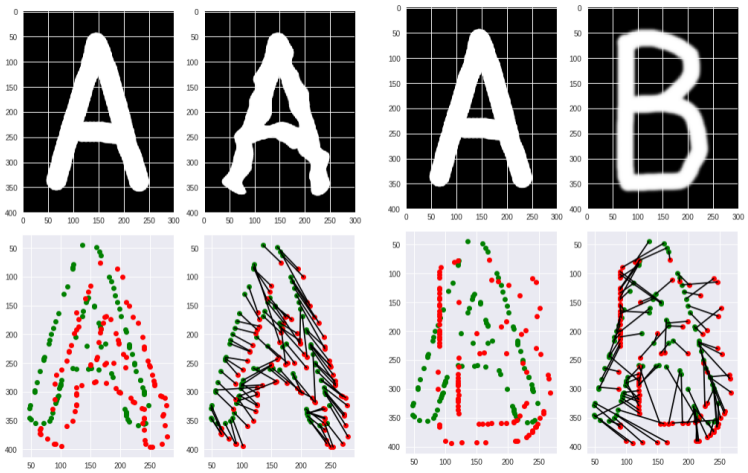

Figura 1.14. Confronto tramite Shape-Matching.[17]

Pattern-Matching

Il *Pattern-Matching* è una tecnica che consiste nell'individuare una determinata sequenza o regolarità di dati (detta *pattern*) all'interno di un set di dati numeroso. Parafrasando il concetto espresso da Christopher M. Bishop nell'articolo *"Pattern Recognition and Machine Learning"*: Il campo del *Pattern Recognition* consiste nella ricerca automatica di regolarità all'interno dei dati attraverso l'uso di algoritmi informatici e di pattern per effettuare azioni come la classificazione di dati in diverse categorie [18]. Nelle immagini digitali, in estrema sintesi, il procedimento di identificazione consiste nel predisporre un pattern, corrispondente ad un'insieme di pixel, che descriva un determinato oggetto di interesse , o parte di esso. In seguito si effettuerà un procedimento di classificazione dei pixel all' interno dell'immagine per capire se esista o meno un gruppo di essi confrontabile con il pattern da ricercare.

Feature-based Object Recognition

Tramite tecniche di *Feature Extraction* è possibile creare dei descrittori che rappresentino le caratteristiche tipiche di un oggetto. Ogni oggetto ha delle caratteristiche peculiari che lo descrivono, se all'interno di un insieme di dati riusciamo ad individuare tutte queste caratteristiche possiamo assumere che l'oggetto sia presente. Per fare un esempio, un volto umano può essere modellato in base a determinate caratteristiche anatomiche, come: il taglio degli occhi, le narici, gli angoli che formano le labbra, ecc... L'insieme di questi elementi anatomici e dei vettori che li collegano costituiscono un *Patch Model*. Questo modello rappresenta l'insieme ordinato di elementi sufficienti a descrivere in maniera precisa un oggetto.

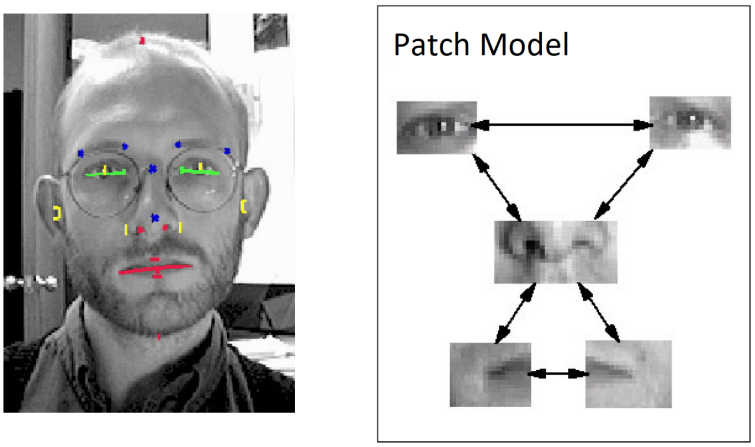

Figura 1.15. Esempio grafico di modello, ricavato da *features*, per rappresentare un volto.

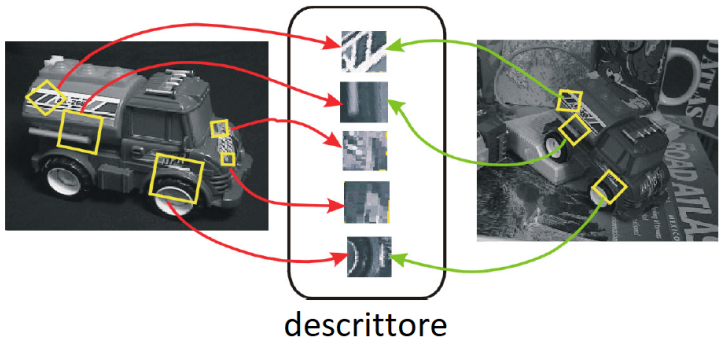

Figura 1.16. Esempio di *Feature detection*. In centro è rappresentato graficamente un descrittore composto da cinque componenti. A sinistra e a destra due immagini in cui vengono ricercate le diverse componenti.

1.3 Un esempio: Facial Recognition

E' interessante capire come la tecnologia del riconoscimento di oggetti all'interno delle immagini si sia evoluta e dell'impatto che abbia portato all'interno della società. Uno dei casi più affascinanti che sfruttano le tecniche di elaborazione delle immagini è sicuramente il riconoscimento facciale. Fino a poco tempo fa, questa tecnologia veniva comunemente vista come qualcosa di proveniente direttamente dalla fantascienza. Ma nell'ultimo decennio non è solo diventata reale, ma anzi si è molto diffusa. In effetti, è difficile leggere articoli o notizie tecnologiche recenti senza imbattersi nel riconoscimento facciale.

Ci sono diverse industrie che beneficiano di questa tecnologia. Le forze dell'ordine utilizzano il riconoscimento facciale per mantenere le comunità più sicure. Le aziende del mondo *Retail* stanno evitando il crimine e la violenza. Gli aeroporti stanno migliorando la comodità e la sicurezza dei viaggiatori. E le aziende di telefonia mobile utilizzano il riconoscimento facciale per fornire ai consumatori nuovi livelli di sicurezza biometrica.

Può sembrare che il riconoscimento facciale sia venuto fuori dal nulla. Ma in verità, questa tecnologia è in lavorazione da diverso tempo. Può essere interessante scorrere gli eventi chiave in ordine cronologico che ne hanno portato lo sviluppo per comprendere che impatto abbia avuto all'interno della nostra società.

Anni '60 - Misure manuali di Bledsoe Molti direbbero che il padre del riconoscimento facciale era Woodrow Wilson Bledsoe. Lavorando negli anni '60, Bledsoe sviluppò un sistema in grado di classificare le foto dei volti a mano usando quello che è noto come un tablet RAND, un dispositivo che le persone potevano usare per

inserire le coordinate orizzontali e verticali su una griglia usando uno stilo che emetteva impulsi elettromagnetici. Il sistema potrebbe essere utilizzato per registrare manualmente le posizioni delle coordinate di varie caratteristiche del viso inclusi occhi, naso, attaccatura dei capelli e bocca.

Queste metriche potrebbero quindi essere inserite in un database. Quindi, quando al sistema è stata assegnata una nuova fotografia di un individuo, è stato in grado di recuperare l'immagine dal database che assomigliava più strettamente a quell'individuo. A quel tempo, il riconoscimento del volto era purtroppo limitato severamente dalla tecnologia dell'epoca e dalla potenza di elaborazione del computer. Tuttavia, è stato un primo passo importante nel dimostrare che il riconoscimento facciale percorribile sfruttando fattori biometrici.

Anni '70 - Aumento della precisione con l'utilizzo di 21 marcatori del viso Negli anni '70, Goldstein, Harmon e Lesk furono in grado di aggiungere maggiore accuratezza a un sistema di riconoscimento facciale manuale. Hanno utilizzato 21 marcatori soggettivi specifici tra cui spessore delle labbra e colore dei capelli per identificare automaticamente i volti. Come nel caso del sistema di Bledsoe, la biometria effettiva doveva ancora essere calcolata manualmente.

Anni '80 e '90 - Eigenfaces Nel 1988, Sirovich e Kirby iniziarono ad applicare l'algebra lineare al problema del riconoscimento facciale. Quello che divenne noto come l'approccio Eigenface iniziò come una ricerca per una rappresentazione a bassa dimensione delle immagini facciali. Sirovich e Kriby sono stati in grado di dimostrare che l'analisi delle caratteristiche di una raccolta di immagini facciali poteva costituire un insieme di caratteristiche di base. Erano anche in grado di dimostrare che erano necessari meno di cento valori per codificare con precisione un'immagine di un viso normalizzato.

Nel 1991, Turk e Pentland ampliarono l'approccio Eigenface scoprendo come rilevare i volti all'interno delle immagini. Ciò ha portato alle prime istanze di riconoscimento facciale automatico. Il loro approccio è stato limitato da fattori tecnologici e ambientali, ma è stato un importante passo avanti nel dimostrare la fattibilità del riconoscimento facciale automatico.

1993/2000 - Programma FERET Negli USA l'agenzia per la difesa avanzata dei progetti di ricerca (DARPA) e l'Istituto nazionale per gli standard e la tecnologia hanno lanciato il programma FERET (Face Recognition Technology) a partire dagli anni '90 per incoraggiare il mercato del riconoscimento facciale commerciale. Il progetto prevedeva la creazione di un database di immagini facciali. Il database è stato aggiornato nel 2003 per includere versioni a colori a 24 bit ad alta risoluzione delle immagini. Nel set di prova sono incluse 2.413 immagini facciali che rappresentano 856 persone. La speranza era che un ampio database di immagini di prova per il

riconoscimento facciale sarebbe stato in grado di ispirare l'innovazione, che avrebbe portato ad una più potente tecnologia di riconoscimento facciale.

2002 - Super Bowl XXXV Al Super Bowl del 2002, le forze dell'ordine hanno usato il riconoscimento facciale in un importante test della tecnologia. Mentre i funzionari hanno riferito che sono stati individuati diversi "piccoli criminali", nel complesso il test è stato considerato un fallimento. I falsi positivi e il contraccolpo dei critici hanno dimostrato che il riconoscimento facciale non fosse ancora una tecnologia matura. Uno dei grandi limiti tecnologici all'epoca era che il riconoscimento facciale non funzionava ancora bene in grandi folle, funzionalità essenziale per il suo utilizzo per la sicurezza durante grandi eventi.

Anni 2000 - Face Recogniton Vendor Tests L'Istituto nazionale degli standard e della tecnologia (NIST) negli Stati Uniti ha iniziato i Face Recogniton Vendor Tests (FRVT) nei primi anni 2000. Basandosi sul FERET, i FRVT sono stati progettati per fornire valutazioni governative indipendenti sui sistemi di riconoscimento facciale disponibili in commercio e sulle tecnologie dei prototipi. Queste valutazioni sono state progettate per fornire alle forze dell'ordine e al governo degli Stati Uniti le informazioni necessarie a determinare i modi migliori per implementare la tecnologia di riconoscimento facciale.

2009 - Database forense per le forze dell'ordine Nel 2009, l'ufficio dello sceriffo della contea di Pinellas (Florida, USA) ha creato un database forense che consentiva agli agenti di attingere agli archivi fotografici del dipartimento della sicurezza stradale e dei veicoli a motore (DHSMV). Nel 2011, circa 170 deputati erano dotati di telecamere che permettevano loro di fotografare i sospetti che potevano essere sottoposti a controlli incrociati rispetto al database. Ciò ha comportato più arresti e indagini penali di quanto sarebbe stato altrimenti possibile.

2010/Oggi - Social Media A partire dal 2010, Facebook ha iniziato a implementare funzionalità di riconoscimento facciale che hanno aiutato a identificare le persone i cui volti potrebbero essere presenti nelle foto che gli utenti di Facebook aggiornano quotidianamente. Nonostante la funzione sia stata immediatamente reputata controversa dai media, scatenando una proliferazione di articoli relativi alla privacy, gli utenti di Facebook non sembravano preoccuparsene. Non avendo alcun impatto negativo sull'utilizzo o la popolarità del sito web, ogni giorno vengono caricate e contrassegnate più di 350 milioni di foto utilizzando il riconoscimento facciale.

2011 - Prima importante installazione della Facial Recognition in un aeroporto Nel 2011, il governo di Panama, in collaborazione con l'allora U.S. Janet

Napolitano, Segretario della Sicurezza Nazionale degli USA, ha autorizzato un programma pilota della piattaforma di riconoscimento facciale di FaceFirst al fine di ridurre le attività illecite nell'aeroporto di Panama Tocumen (noto come centro per il contrabbando di droga e la criminalità organizzata).

Poco dopo l'implementazione, il sistema ha portato all'arresto di molti sospettati dell' Interpol. In seguito al successo della prima installazione, FaceFirst si è espanso nel terminal nord della struttura. L'implementazione FaceFirst a Tocumen rimane la più grande installazione biometrica in un aeroporto fino ad oggi.

2014 - Le Forze dell'ordine adottano la Face Recognition A partire dal 2014, l' Automated Regional Justice Information System (ARJIS) ha iniziato a fornire alle agenzie partner la piattaforma mobile di FaceFirst che supporta il riconoscimento facciale per le forze dell'ordine. ARJIS, una complessa rete aziendale per la giustizia criminale che promuove la condivisione di informazioni e dati tra le forze dell'ordine locali, statali e federali, ha voluto risolvere un problema critico: identificazione istantanea per persone che non avevano ID o non volevano essere identificati. Alcune delle agenzie che hanno iniziato a utilizzare il riconoscimento facciale mobile per identificare i sospetti nel campo includono: la polizia di San Diego, FBI, DEA, U.S. Marshalls.

2017 - Riconoscimento del volto "Inevitabile" per il Retail Indagini dimostrano che il riconoscimento facciale viene adottato dal mondo Retail molto più velocemente di qualsiasi altro settore. In un recente webinar, D & D Daily Publisher e l'editore Gus Downing hanno affermato che l'adozione del riconoscimento facciale per il mondo della vendita al dettaglio è inevitabile. Downing è considerato uno dei leader più importanti nel settore della prevenzione delle perdite ed è solo uno degli esperti che vede enormi vantaggi nel settore della grande distribuzione per chi usa un sistema di riconoscimento facciale.

2017 - IPhone X Apple ha rilasciato l' iPhone X nel 2017, pubblicizzando il riconoscimento facciale come una delle sue principali novità. Il sistema di riconoscimento facciale nel telefono viene utilizzato per la sicurezza del dispositivo. Il nuovo modello di iPhone è esaurito quasi istantaneamente, dimostrando che i consumatori ora accettano il riconoscimento facciale come il nuovo gold standard per la sicurezza.

2017 - WatchList as a Service Sta diventando più facile che mai per le organizzazioni beneficiare della tecnologia di riconoscimento facciale. Quest'anno, FaceFirst ha introdotto WatchList as a Service (WaaS) alla conferenza NRF Protect. WaaS è una nuova piattaforma di dati per il riconoscimento dei volti progettata per aiutare a prevenire il taccheggio e il crimine violento. WatchList include un database gestito di criminali noti che rappresentano un rischio per la sicurezza, il furto o il

crimine violento. Il database funziona in tandem con la piattaforma di sorveglianza biometrica FaceFirst, che utilizza la tecnologia di corrispondenza delle funzionalità per avvisare la sicurezza delle minacce in tempo reale. [19]

Capitolo 2

Immagini in ambito E-Commerce

Ogni data set di immagini si identifica per delle caratteristiche comuni. Le metodologie di elaborazione dipendono strettamente dal tipo di immagine che deve essere trattato e dalle informazioni che si vogliono estrarre. Nel caso di studio affrontato il data set è rappresentato dalle immagini prelevate da diverse piattaforme E-commerce che contengono un prodotto specifico di interesse: le scarpe di lusso. L'obbiettivo è riuscire ad identificare il prodotto presente all'interno dell'immagine. Questo permette di sapere se uno stesso prodotto viene venduto su diverse piattaforme e, quindi, di confrontare le informazioni presenti nelle corrispondenti pagine di dettaglio.
Per mettere a punto una metodologia di elaborazione valida è necessaria un'analisi delle caratteristiche specifiche delle immagini da elaborare. La definizione delle caratteristiche del data set è fondamentale nella scelta dei diversi step all'interno del processo di elaborazione. La conoscenza delle caratteristiche delle immagini, unita ad una metodologia rigorosa e ad un' attenta valutazione delle possibili tecniche, ha permesso di raggiungere l'obbiettivo desiderato.

2.1 Processo di elaborazione delle immagini

La tipologia di trasformazioni nell'ambito di un sistema di elaborazione delle immagini sono legate strettamente al contesto. A seconda del tipo di immagini e delle informazioni che vogliamo trarre cambiano le tecniche che possiamo utilizzare. Un altro aspetto importante è l'ordine in cui applichiamo le trasformazioni. Per questo motivo R.E. Woods e R.C. Gonzalez nel libro *Digital Image Processing* [20], hanno elaborato il concetto di catena di elaborazione digitale.
In maniera indipendente dal contesto una metodologia ci permette di raggiungere, secondo passaggi logici, l'obbiettivo che ci siamo prefissati. Schematicamente l'approccio si può descrivere secondo la figura 2.1.

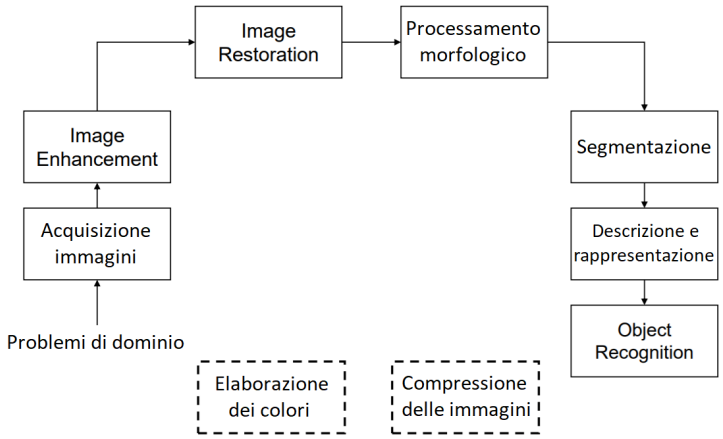

Figura 2.1. Schema della catena di elaborazione delle immagini digitali.

Problemi legati al dominio In questa fase preliminare si possono effettuare delle assunzioni sulla tipologia di immagini che verranno acquisite. In base alle loro caratteristiche verranno determinate le strategie e le tecniche da utilizzare.

Acquisizione delle immagini E' lo step in cui avviene la raccolta delle immagini, solitamente già in formato digitale. Prevede tutte le operazioni di *pre-processing*, come ad esempio il ridimensionamento.

Image Enhancement E' il passaggio che permette di mettere in risalto le caratteristiche principali dell'immagine. In questo modo ci si può focalizzare sulle parti di interesse. Per fare un esempio, si può sfruttare la tecnica del negativo per mettere in risalto dettagli in immagini scure.

Image Restoration Se l' *Image Enhancement* è un processo basato sulla percezione soggettiva per mettere in risalto i dettagli peculiari dell'immagine, l'*Image Restoration* sfrutta principi oggettivi. Questo step utilizza algoritmi matematici e probabilistici basati sulla degradazione dell'immagine, in questo modo si possono eliminare i difetti, dovuti ad esempio ad un cattivo campionamento, e stimare l'immagine originale.

Processamento morfologico Consiste nell'utilizzo di strumenti per estrarre elementi utili a rappresentare e descrivere la forma dall'immagine. In questo passaggio si passa da processi che restituiscono immagini a processi che restituiscono valori. Questo passaggio è fondamentale per poter sfruttare algoritmi che sfruttano le forme, come ad esempio lo *Shape-Matching*.

Segmentazione E' il processo che permette di separare l'oggetto da ricercare rispetto al resto dell'immagine. In questo modo si riesce ad isolare le informazioni di interesse dal resto.

Descrizione e rappresentazione Questo passaggio segue la segmentazione ed è di fondamentale importanza perché determina come decidiamo di descrivere e rappresentare l'oggetto di interesse. A seconda dell' oggetto da individuare si effettuerà la scelta del modello di rappresentazione, come ad esempio le *feature*.

Object Recognition il riconoscimento è l'ultimo passaggio della catena. Permette di assegnare al descrittore ottenuto precedentemente un'etichetta basandosi sulle sue caratteristiche che identifica l'oggetto riconosciuto.

Elaborazione dei colori A quest'area viene dedicata un fase indipendente dalla catena. Il trattamento dei colori e dei modelli atti a rappresentarli rimangono scollegati dal resto delle trasformazioni che possono essere effettuate su immagini in scala di grigi.

Compressione delle immagini Questa fase è più legata ad un aspetto di ottimizzazione ed efficienza del sistema di elaborazione, per questo motivo non trova una collocazione all'interno della catena. Come suggerisce il nome è possibile ridurre la dimensione di un'immagine in modo che lo spazio occupato sul disco sia minimo. Negli ultimi anni le tecnologie di archiviazione si sono molto evolute rendendo possibile salvare grani quantità di dati. Se per l'archiviazione questo passaggio è diventato meno cruciale non si può dire lo stesso per la trasmissione. Dover trasmettere meno dati è un grosso vantaggio poiché la banda per le trasmissioni non è aumentata in modo considerevole.

2.2 Analisi ed architettura

Prima di iniziare la realizzazione dei blocchi di elaborazione è necessario effettuare un'analisi in modo da individuare gli eventuali problemi legati al dominio. Questa fase permetterà di conoscere le caratteristiche del data set e capire quali saranno le trasformazioni più adatte per riuscire ad effettuare il matching delle immagini. Una volta effettuata l'analisi del dominio si potrà procedere all'acquisizione delle immagini. Al termine di questa operazione sarà necessario applicare alcune procedure di pre-processing, essenziali per uniformare e preparare il data set alla vera e propria fase di elaborazione. La fase di elaborazione contiene diversi blocchi, ognuno dei quali prevede una diversa trasformazione. L'insieme delle trasformazioni è stato denominato 'Normalizzazione'. Al termine di questo stadio le immagini saranno

pronte per essere confrontate all'interno del blocco di *Image Matching* per ottenere un valore che rappresenti il grado di similarità tra di esse. Questi passaggi possono essere riassunti nello schema in Figura 2.2

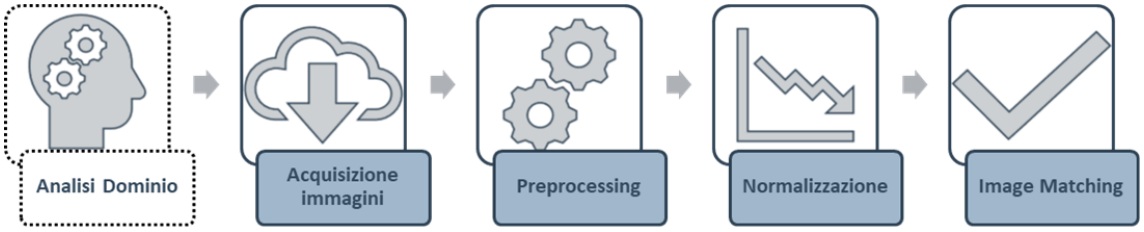

Figura 2.2. Schema della catena di elaborazione delle immagini digitali.

2.2.1 Dominio ed acquisizione

Il contesto di questo caso di studio prevede l'acquisizione di immagini presenti su piattaforme E-commerce. In particolare ci si è voluti focalizzare su un prodotto appartenente al mondo del *Fashion* come le scarpe di alta moda. L'obbiettivo finale di quest' elaborazione è scoprire se su piattaforme diverse è possibile trovare le stesse immagini e, di conseguenza, capire se viene messo in vendita lo stesso prodotto.
L'operazione preliminare è stata un' osservazione critica sulle proprietà simili del set di immagini. In seguito a quest'analisi sono risultate chiare delle caratteristiche comuni. Un aspetto importante è notare che la totalità delle immagini presenta sempre uno sfondo neutro (tipicamente bianco). Questa caratteristiche permette, a livello percettivo, di riuscire ad individuare subito l'oggetto di interesse. Un'altra considerazione interessante è sulla posizione degli oggetti all'interno della scena. Nella quasi totalità dei prodotti è sempre di due tipi: laterale, oppure frontale ma leggermente ruotata. Questo permette di sapere che i dettagli della parte frontale e laterale degli oggetti saranno sempre presenti. Sfruttare come fonte le piattaforme E-commerce presenti on-line ci permette di avere già alcune informazioni sulle immagini, come ad esempio una gerarchia di categoria merceologica. Queste informazioni sono estratte dal codice HTML delle pagine Web dove è presente il prodotto. Questi dati possono essere sfruttati per associare delle 'etichette' alle immagini ed avere un approccio Supervisionato.

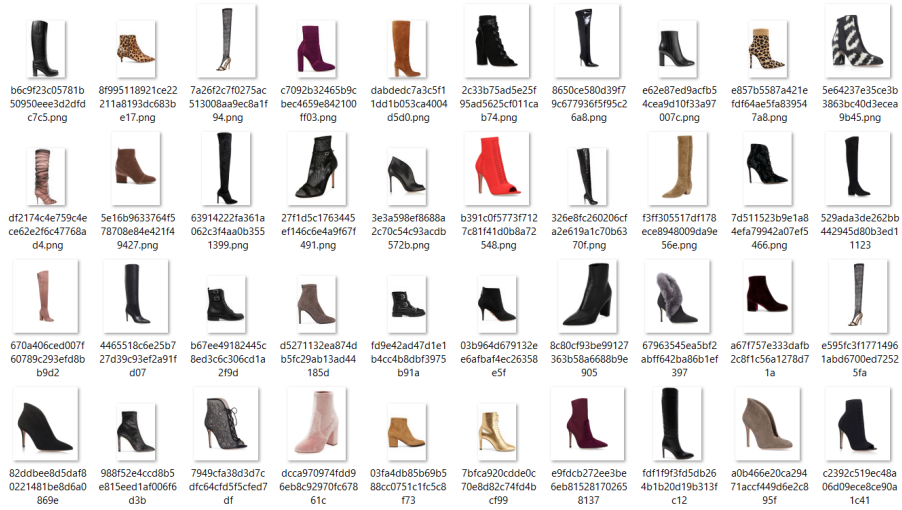

Figura 2.3. Set di immagine estratte da diverse piattaforme E-commerce.

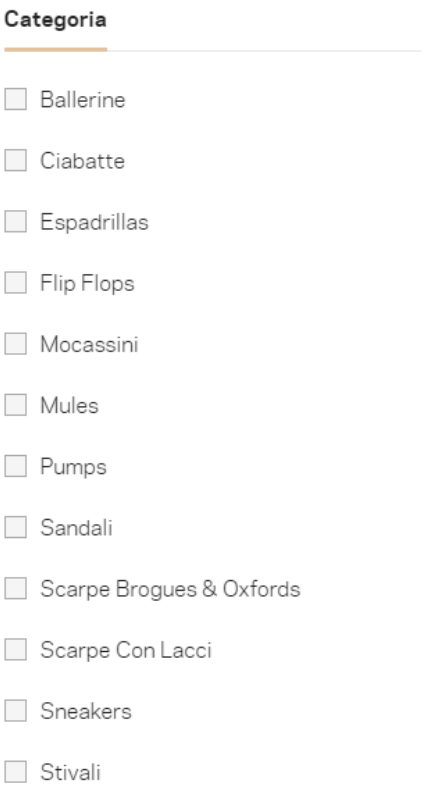

Figura 2.4. Esempio di categorie presenti su una piattaforma E-commerce

Come detto precedentemente sono state acquisite immagini di scarpe, quindi un grosso vantaggio è sapere che appartengono tutte alla stessa macro-categoria merceologica. Nonostante ci si aspetti di avere immagini con caratteristiche simili ciò non è stato del tutto vero. Confrontare ad esempio immagini appartenenti alla categoria 'Stivali' con altre della categoria 'Sandali' non ha permesso di effettuare trasformazioni ottimali per entrambe le categorie. Per questo motivo si è deciso di suddividere ulteriormente il problema confrontando immagini appartenenti alla sola categoria 'Stivali'.

Figura 2.5. Esempio di immagine appartenente alla categoria 'Stivali'.

Successivamente a queste considerazioni sul dominio dei dati, si è passati alla progettazione di una catena di elaborazione con l'obbiettivo di misurare il grado di similarità di due immagini. Per la parte di riconoscimento si è scelto di utilizzare delle metodologie basate sul confronto diretto. La condizione necessaria di queste metodologie è avere immagini con lo stesso numero di pixel, per questo motivo è stato necessario trasformarle effettuando un ridimensionamento e portarle tutte alle stesse dimensioni.

2.2.2 Trasformazioni

Prima di passare alla fase vera e propria di trasformazione è stato necessario un cosiddetto *Pre-processing*. Questo passaggio permette di rendere più omogeneo il set di immagini da utilizzare nella fase successiva di elaborazione. In particolare, data la disomogeneità dei formati, si è scelto di uniformare la larghezza portandola a 300 pixel calcolando l'altezza mantenendo le proporzioni originali. Questa trasformazione permette di avere, in questa fase, immagini tra i 150.000 e i 100.000 pixel circa, il che è un buon compromesso tra tempi di elaborazione ed informazione. Durante questo primo ridimensionamento si è riusciti ad effettuare un miglioramento delle immagini applicando un filtro anti-aliasing. Questo ha permesso di eliminare i difetti percettivi legati a campionamenti delle immagini insufficienti e che quindi risultavano 'pixellate'. Le immagini con un cattivo campionamento, quindi con pixel troppo grandi, nonostante questo passaggio risultano comunque avere decisamente meno informazioni rispetto ad immagini a più alta risoluzione. Le immagini acquisite con questa tecnica costituiscono il data set iniziale. In seguito all'acquisizione è stata realizzata una fase chiamata di Normalizzazione per trasformare le immagini a colori del data set iniziale nelle immagini che saranno utilizzate dall'algoritmo di confronto. Essa è costituita da diversi blocchi di trasformazione, come si può notare dalla figura 2.6

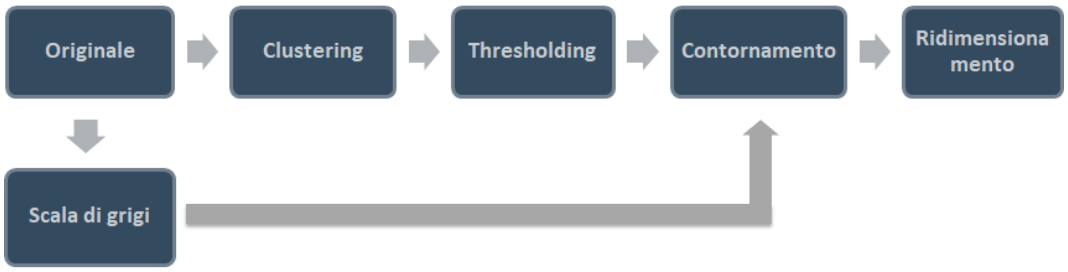

Figura 2.6. Schema a blocchi della fase di Normalizzazione.

Clustering colori L'informazione sul colore è di primaria importanza, ma d'altro canto utilizzare immagini che sfruttano il modello RGB rispetto ad immagini in scala di grigi appesantisce di molto l'elaborazione. Per questo motivo si è scelto di procedere ad una trasformazione delle immagini in scala di grigi. Non volendo perdere l'informazione del colore si è deciso di applicare un algoritmo di Clustering ai pixel

dell'immagine. In questo modo i pixel vengono suddivisi in Cluster costituiti dai colori predominanti. Il risultato permette di slegare l'informazione del colore dal resto ottenendo un vettore composto dai tre colori predominanti dell'immagine (escluso lo sfondo). In seguito questo vettore di colori potrà essere utilizzato mantenendo l'associazione con l'immagine.

Figura 2.7. Risultato del clustering per colore.

Thresholding Il passaggio precedente non viene sfruttato solamente per estrarre i colori, ma ci permette anche di separare quelli che sono i pixel legati allo sfondo (bianchi) da quelli che costituiscono la scarpa. In questo modo, utilizzando il risultato della trasformazione precedente, si può facilmente procedere ad una thresholding trasformando in nero i pixel al di sopra di una soglia ed in bianco quelli al di sotto. Solitamente in quest'operazione la fase delicata è la scelta del valore di soglia, ma in questo caso, dopo il processo di clustering, il valore di soglia è quello dei pixel di sfondo.

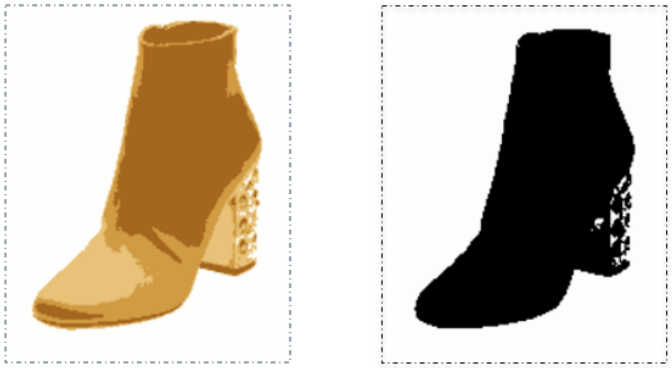

Figura 2.8. Risultato del Thresholding.

Contornamento Con il passaggio precedente siamo riusciti ad individuare la sagoma dell'oggetto. Vogliamo sfruttare questa sagoma per isolare i pixel dell'immagine in scala di grigi dallo sfondo ed ottenere una nuova immagine composta dai soli pixel significativi. In questa fase otteniamo, a partire dalla sagoma, un rettangolo che costituisce il contorno dell' oggetto ed anche la posizione dei pixel significativi. Utilizzando queste informazioni andiamo a 'ritagliare' i soli pixel significativi dell'immagine in scala di grigi.

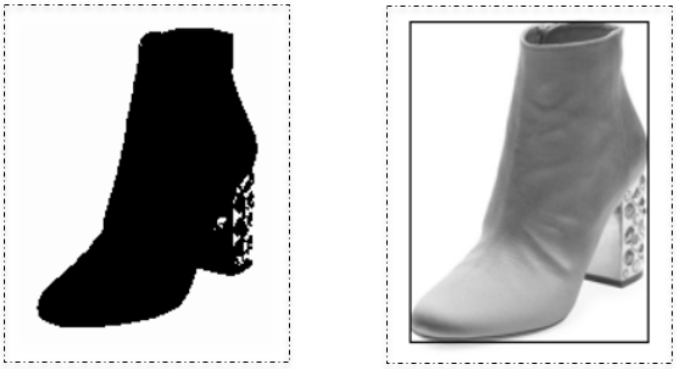

Figura 2.9. Risultato del contornamento.

Ridimensionamento Prima di poter procedere con le metodologie di confronto diretto è necessario uniformare le immagini in modo che abbiano tutte lo stesso numero di pixel. Per fare ciò si è scelto di convertire tutte le immagini in un formato LxL dove L è pari a 200 pixel. Questa procedura ha permesso di notare che all'interno del data set le immagini si suddividevano, in base alle dimensioni, in due categorie. Da un lato le immagini che con un formato quadrato non risultavano molto allargate rispetto all'originale, dall'altro che invece lo erano molto. In questo modo abbiamo deciso di suddividere il data set in immagini con ratio $altezza/larghezza$ compresi tra 0.7 e 1.1 ed immagini con ratio minore di 0.7. Per queste ultime si è scelto di rappresentarle con un rapporto base/ altezza differente (170x237). La suddivisione del data set ci permette di fare già una prima valutazione di similarità, ovvero che le scarpe simili tra di loro avranno lo stesso rapporto base/altezza.

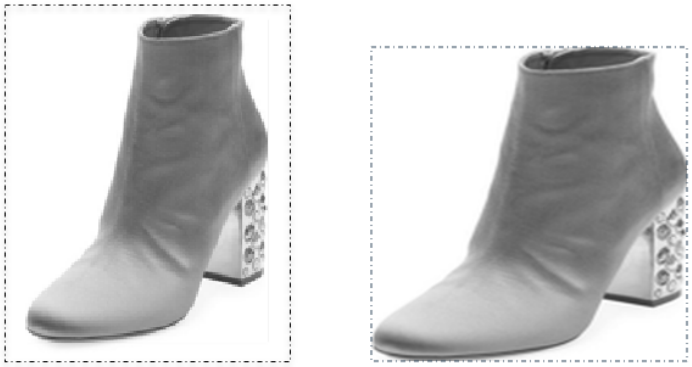

Figura 2.10. Ridimensionamento LxL

Figura 2.11. Esempio di immagini estratte dal data set finale

2.2.3 Image Matching

La fase di Normalizzazione ha permesso di ottenere un'immagine che contiene i soli pixel appartenenti alla sagoma della scarpa. Per questo motivo si è scelto di fermarsi a questo punto e valutare tramite una metodologia di confronto diretto quanto fossero simili le immagini presenti nel data set ottenuto. Si è scelto di valutare due approcci, uno basato sullo Scarto quadratico medio (MSE), l'altro sulla Similarità strutturale (SSIM). Il risultato interessante è stato scoprire che per entrambe le tecniche le coppie perfettamente simili sono risultate le stesse, come si può notare dalla tabella.

PRODUCT_COD_1	PRODUCT_COD_2	MES	SSIM
c2392c519ec48a06d09ece8ce90a1c41	84bf878606fcef608e51a5744e03cd9c	0	1
4037139c0f904198c7b2540d24bcaafd	fb7325bd3f10df315455b0c0ebca68fa	0	1
81ce93f6b08e57b31c595df6c8cf4252	e5c6d3aea113a27db0ceb278b9686127	0	1
51b2801ab447347515afe194fecad1d9	fb51f81b54a6a6967d3dcdafe6e9f944	0	1
7a26f2c7f0275ac513008aa9ec8a1f94	e595fc3f17714961abd6700ed72525fa	0	1
1c53e650d26cee73d35046eb4753c575	3417bcdfa24fe0109d37cce55b2f15e5	0	1
6ced04722704c96f46d46147efcaee3a	6554903eab9447e1eb2b76051a4dc334	0	1
01fe70f69232eb98f4d6b4bf56550d7b	eb2723bc9a61ee93663349d6aea295fa	0	1
dcf80690c4a5a08605f592eace814dbc	0d53dc92a990bc6727297d3b257d6b83	0	1
17f5ea4a9973aef83e336ee20c47556c	ed9ceeb6a429777a08c8b7353840b5b8	0	1
9be8265a01759719336a70e91c98b709	09c13f785db1b2c68350cbd42f179897	0	1
b9879c2fe69610541efdfce0cc1e5d9b	9a8fbef9a54f1e8db7db9d6cd6d298bc	0	1
cdc74286fd6772883059d3037d3c71ec	646ec54e31d6176e0325c16dfc74580f	0	1
3d6cd7a33104807357fac2af8ef077e5	447f5c518804996284b14d7a94f46fe6	0	1
9d235775372b3c5d53bea2eb03175269	5943b2448057446cc5ed33547f3b242c	0	1
a16589fed89f91dd970b98c92680166d	b27d392c522bb961b6828302219d176c	0	1
c545fecbee23941678e754114d88b7c4	048752406cc6bb35e03ddfa87f3267e4	0	1
7adbbbf9afb79bbe6a4d883134a82991	170f5ba33a4e7086be9b87d22aefcaab	0	1
4cd04064e1497406a565bc3fcd7d8238	dbccb3ef350ce9ffc1f7ce8e2429fed2	0	1
6bc801017370e97272746f246a7fb37c	08574b5ab355dd2e6f2dff94964e78c6	0	1
d3418a87e4361187bde7c1a9f2d6ebd7	d46159e0c3f1b416837b8b8d6e56374f	0	1
9f51934025e9bcb474871fe30b5c2e1e	a2efd64f3b1ac0b985395bb3b0105859	0	1
d256d16dbc09a5687895c3ca4e1b4fb9	8c80cf93be99127363b58a6688b9e905	0	1
5ef5c10f05369ceb381b18b46e9c8764	624683d7042889cc877846248e49c0c0	0	1
d37601fc88ed767d8eb581c703773764	27f1d5c1763445ef146c6e4a9f67f491	0	1
3e596a87d72af1137fc8482456c729e7	83d7b173f46407f642ffca133c2fbf87	0	1

MSE è una tecnica '*Brute-Force*' che confronta singolarmente i pixel di due immagini e restituisce un indice che rappresenta uno scostamento medio dei valori. Quanto più i valori sono diversi quanto più il valore di MSE aumenta. Per il data set di immagini sono state confrontate tra di loro tutte le immagini ed i valori sono variati tra 0 e 11485,91145. Lo stesso tipo di valutazione è stato effettuato utilizzando SSIM, dove a differenza di MSE si riduce l'elaborazione in quanto vengono confrontate aree di NxN bit e non i singoli pixel. SSIM sfrutta un algoritmo che tiene in considerazione non solo la differenza dei singoli pixel ma anche la variazione di luminanza. Questo fattore è determinante in quanto permette di restituire un valore che risulta molto più valido a livello percettivo. Per questa tecnica i valori variano tra 1 e 0.

Come si può notare dalle figure 2.12 e 2.13 i grafici presentano un *plateau* nella parte iniziale che rappresenta i perfect match. Successivamente una seconda parte di variazione dei valori ed infine una curva che per MES tende ad aumentare costantemente mentre per SSIM degrada fino ad arrivare al valore minimo.

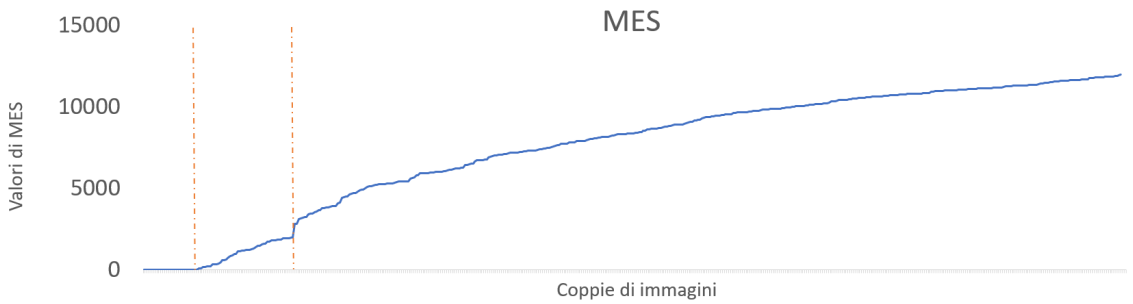

Figura 2.12. Distribuzione dei valori per lo scarto quadratico medio per la totalità delle coppie confrontate.

Figura 2.13. Distribuzione dei valori per la similarità strutturale per la totalità delle coppie confrontate.

In base ai risultati ottenuti con il data set si è deciso di considerare i soli dati che risultano confrontabili e che corrispondono alle prime due fasi dei grafici. Ai valori calcolati è stato associato un *flag*, valutato a livello percettivo, che identifica una somiglianza e vale 1 in caso positivo 0 altrimenti. Come si può notare dalla figura 2.14 considerando i valori di SSIM è presente una variazione molto irregolare per MES. Il valore di SSIM decresce in maniera regolare fino ad arrivare ad un valore di soglia tale per cui al di sotto di questo valore si presentano raramente casi di match. Mentre per MES non è possibile garantire una soglia precisa. Per questo motivo si scelto di utilizzare SSIM fissando un valore di soglia. In base alle analisi effettuate si è deciso di interrompere a questo livello la catena di elaborazione reputando sufficientemente validi i risultati di match. Proseguendo sarebbe stato necessario effettuare uno studio atto ad individuare una tecnica di *feature extraction* e conseguentemente applicare degli algoritmi di *Machine Learning* per effettuare un vero e proprio riconoscimento degli oggetti.

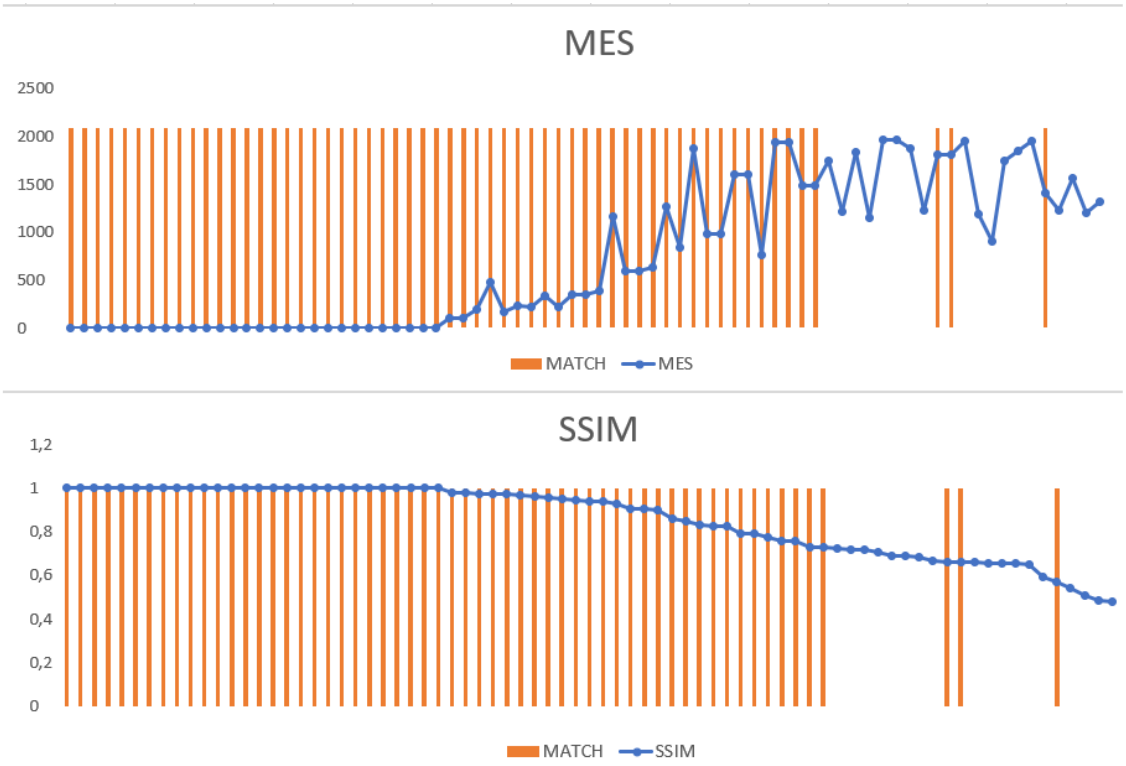

Figura 2.14. Confronto della distribuzione dei valori ottenuti tramite SSIM e MES rispetto ai match verificati.

2.2.4 Trattamento dei colori

Come accennato in precedenza per quanto riguarda i colori si è deciso di applicare un algoritmo di clustering sull'immagine prima di trasformarla in scala di grigi. In questo modo si è riassunta l'informazione del colore in un vettore di tre valori RGB corrispondenti ai tre colori principali. Dal momento che i colori RGB sono rappresentati con 256 valori per ognuna delle tre componenti (Rosso, Verde e Blu) le combinazioni possibili sono circa 16.000.000. Per rendere confrontabili i colori di due prodotti si e deciso di ridurre i valori possibili ad un subset di 140 colori. Per ricondursi ai valori del subset è necessario un algoritmo che individui il colore più simile (vicino) rispetto alla tripletta RGB. In prima battuta è possibile stabilire quanto due colori siano simili considerando le componenti RGB come punti su piani ortogonali tra di loro nello spazio e calcolando la distanza geometrica. Dal momento che il sistema visivo umano non ha una percezione lineare dei colori il risultato non è percettivamente valido. Per questo motivo esistono diversi spazi di colore utilizzati proprio per distribuire i colori tenendo conto della percettività umana. La commissione internazionale dell'illuminazione CIE (*Commission internationale de*

l'éclairage) ha disposto alcuni standard di progettazione per utilizzare degli algoritmi di vicinanza dei colori. La scelta è ricaduta sul CIEDE2000[21] che, in questo momento, è uno degli algoritmi più efficaci a livello percettivo.

Color		Red	Green	Blue	Distance	CIEDE 2000
	-	167	21	35	-	-
	brown	165	42	42	142.727	1.762
	forest greeen	34	139	34	118.718	78.455

Figura 2.15. Confronto tra i risultati del calcolo della distanza tra due colori utilizzando la distanza geometrica (*Distance*) e CIEDE2000.

Nella tabella in figura 2.15 viene calcolata la distanza tra il colore nella prima riga rispetto ad altri due appartenenti al subset di 140 colori. Come si può notare il risultato del CIEDE2000 a livello percettivo è di gran lunga migliore rispetto alla distanza geometrica.

Capitolo 3

Integrazione nel modello di Business

L'informazione che una determinata immagine contenga un oggetto di interesse, utilizzata in modo indipendente, non ha molto valore. Ciò che realmente porta valore é l'integrazione di quest'informazione all' interno di un modello di business. Tramite il processo di elaborazione delle immagini discusso nei capitoli precedenti si è riuscito ad ottenere un' indicatore di quanto due immagini siano simili. Ogni coppia di immagini simili corrisponde ad uno stesso prodotto venduto su diverse piattaforme on-line. Riuscire a ricondursi ad un preciso prodotto, per un'azienda, apre a nuove possibilità di analisi. Per chi fornisce un servizio di consulenza di *Business Analytics* è interessante riuscire a dare ai propri clienti informazioni chiave per le loro strategie di business. L'obbiettivo di questa tesi è proprio riuscire ad integrare dati provenienti da fonti esterne con le informazioni di cui siamo già in possesso, arricchendole, dando la possibilità di generare nuove conoscenze potenzialmente strategiche.

3.1 Caso di studio

Il punto di partenza del progetto è un'architettura di *Business Analytics* che si pone come obbiettivo l'integrazione di fonti eterogenee all'interno di un *Data Hub*. Il ruolo di questa componente è proprio quella di riuscire, tramite un modello quanto più generico possibile, a gestire ed organizzare le informazioni. Per questo caso di studio in particolare, tramite un *Crawler Web*, vengono estratte le informazioni presenti all'interno delle pagine HTML di piattaforme E-commerce. Questi dati non strutturati vengono inseriti all'interno del *Data Hub*, storicizzati ed uniformati. I dati delle piattaforme web contengono informazioni sui prodotti che vengono venduti on-line: url, immagine, modello, casa produttrice, colore, taglia, prezzo, ...
Queste informazioni vengono organizzate e 'armonizzate' tramite un processo di

omogeneizzazione utilizzando un modello generico e suddividendo i dati su più livelli. Le informazioni raccolte all'interno dell' hub sono integrabili all'interno del *Data Mart* aziendale e utilizzabili congiuntamente ai dati interni per nuovi tipi di analisi. Per questo progetto l'obbiettivo finale è stato riuscire ad effettuare il monitoraggio dell'andamento dei prezzi di vendita sulle diverse piattaforme e quindi conoscere le variazioni temporali dei prezzi dei prodotti.
Per un' azienda del mondo *Retail* è molto semplice monitorare l'andamento delle vendite dei propri negozi, fisici ed on-line, in quanto ha pieno controllo dell'informazione appartenente a tutta la filiera del prodotto, dalla produzione fino alla vendita al cliente finale. Infatti in questo caso si parla di una relazione B2C (*Business to Client*) e quindi esiste un collegamento diretto tra l'impresa e il cliente finale. Ciò che non riesce a controllare sono tutti quei prodotti che vengono venduti a clienti *Wholesale* (all'ingrosso) e a piattaforme on-line di terzi. In questi casi per l'azienda si parla di relazioni B2B (*Business to Business*) e la filiera dell'informazione si interrompe prima, non riuscendo ad arrivare al cliente finale. Mentre per i clienti *Wholesale* esistono degli accordi commerciali legati al territorio, per i siti E-commerce esterni non esiste una divisione geografica che eviti la cosiddetta 'cannibalizzazione'. Infatti se, su un sito E-commerce esterno, lo stesso prodotto viene venduto con prezzi più 'aggressivi' è probabile che l'andamento delle vendite sul sito E-commerce proprietario ne risentirà. Riuscire a conoscere ed addirittura prevedere la variazione dei prezzi sulle piattaforme esterne permette di modificare e migliorare le proprie strategie di vendita.
Un ulteriore aspetto interessante è quello di sapere, invece, come si collocano sul mercato i propri prodotti rispetto ai *competitors* e quindi scoprire le aree deboli da sfruttare per aumentare il proprio volume di vendita e 'rubare' fette di mercato ai concorrenti.
Per tutti gli obbiettivi che ci si pone il punto cruciale è riuscire ad identificare correttamente i prodotti venduti on-line. La piattaforma per ottenere le informazioni di interesse sfrutta degli algoritmi di *Text Matching* che permettono in base al testo estrapolato dalle pagine web di identificare lo stesso prodotto su diverse piattaforme. Questa tecnica in alcuni casi non risulta sufficientemente efficace, per questo motivo si è deciso di sviluppare un modulo che effettui l' *Image Matching* delle immagini presenti sulle diverse piattaforme. In questo modo è stato possibile valutare l'efficacia delle due tecniche ed utilizzarle in maniera complementare rendendo il sistema di identificazione più preciso e robusto.

3.1.1 Infrastruttura

Un'infrastruttura progettata correttamente rappresenta sicuramente un requisito fondamentale per una rete IT. Essa è costituita dall'insieme delle componenti software e hardware progettate per connettere i dispositivi all'interno dell'organizzazione (Intranet), nonché la società ad altre società (Extranet) ed Internet.
Riuscire ad essere competitivi per un'azienda dipende fortemente dalla robustezza e dall'architettura della propria rete. Le soluzioni possibili sono numerose; la scelta corretta dell'architettura è un processo assolutamente non banale. Per queste ragioni, l'analisi dei requisiti diventa di fondamentale importanza. I dispositivi hardware, insieme alle soluzioni software, devono garantire una serie di funzionalità:

- Connettività
- Sicurezza
- Servizio e scalabilità
- Funzionalità di routing / commutazione
- Controllo degli accessi

Con una buona architettura, un'azienda è in grado di supportare la crescita della propria attività senza dover modificare la propria rete. Infatti, grazie ad una robusta scalabilità, una rete può cambiare senza richiedere la revisione dell'infrastruttura.
Senza dubbio, una corretta configurazione dell'infrastruttura può migliorare notevolmente velocità, produttività e prestazioni.
Per quanto riguarda l'aspetto della *Data Ingestion*, una delle caratteristiche chiave riguarda l'archiviazione dei dati, l'architettura della struttura di storage può essere realizzata con diverse soluzioni.

Data Warehouse Un *Data Warehouse* (DWH) è una struttura di integrazione e storage di informazioni aziendali e dati derivati da sistemi operativi e fonti di dati esterne. Esso è progettato per supportare le decisioni aziendali consentendo il consolidamento, l'analisi e il reporting dei dati a diversi livelli di aggregazione. I dati vengono inseriti nel DWH attraverso i processi di estrazione, trasformazione e caricamento (ETL). Tramite questi processi i dati vengono strutturati ed uniformati già nella fase di caricamento inserendoli in un preciso modello progettato per l'analisi ed il reporting. Questo tipo di sistema viene detto *schema-on-read*, fornendo un unico livello di fruizione dei dati.

Figura 3.1. Rappresentazione schematica di un DWH.

Data Lake Un *Data Lake* è un sistema di archiviazione caratterizzato dalla memorizzazione dei dati nel loro formato naturale, che esso sia strutturato, semi-strutturato o non strutturato. Si pone in antitesi al DWH in quanto è privo di ogni tipo di struttura e quindi 'piatto' come la superficie di un lago. Il termine *Data Lake* è spesso associato all'archiviazione di oggetti orientata ad Hadoop. La natura della tecnologia, *write once read many*, si associa perfettamente all' archiviazione grezza del dato. Questo tipo di caricamento dei dati ha come conseguenza l' associazione di forma e struttura durante la fase di lettura. Questo tipo di sistema viene detto *schema-on-write*.

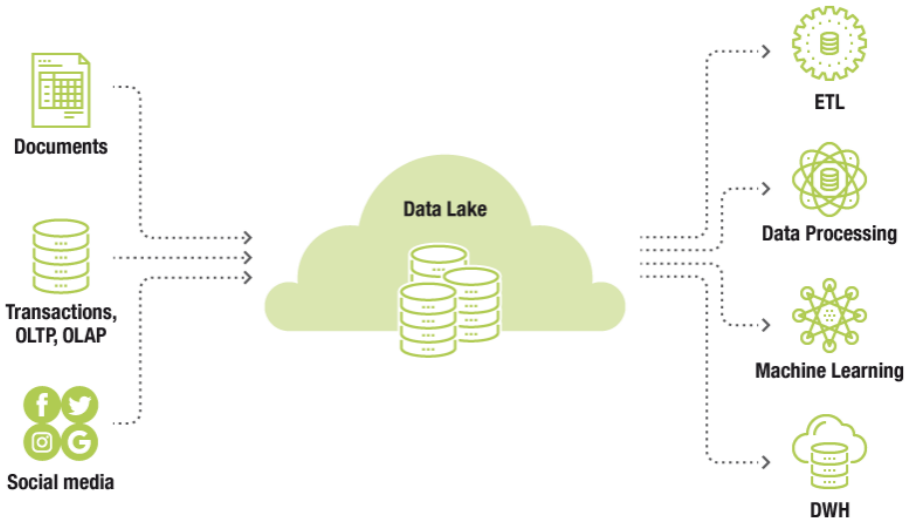

Figura 3.2. Rappresentazione schematica di un Data Lake.

Data Hub Un *Data Hub* è un 'concentratore' di dati provenienti da più fonti organizzati per la distribuzione, la condivisione e, molto spesso, la suddivisione in livelli (*subsetting*). I dati vengono raccolti nella forma più grezza possibile senza essere elaborati e senza applicare alcun tipo di schema restrittivo. È un approccio *hub-and-spoke* all'integrazione dei dati.
Un hub di dati si differenzia da un *Data Lake* omogeneizzando i dati e possibilmente servendoli in più formati desiderati, anziché semplicemente archiviandoli. I *Data Lake* non sono indicizzati e non possono essere armonizzati a causa della suddivisione dei dati in moduli incompatibili fra di loro. L'obiettivo principale di un EDH (*Enterprise Data Hub*) è fornire un'origine dati centralizzata e unificata per le diverse esigenze aziendali.
Un hub si differenzia da un *Data Warehouse* in quanto non avendo una struttura rigida permette una maggiore flessibilità. Inoltre un DWH è caratterizzato dall'avere un unico livello di fruizione dei dati mentre un EDH può fornire un livello diverso a seconda dell'esigenza aziendale. Una delle soluzioni che si sta diffondendo è l'utilizzo ibrido di EDH e DWH. L'EDH può sostituire la parte di data trasformation togliendo carico di lavoro al DWH e fornendo un dato pronto per essere utilizzato ed integrato nel modello dati aziendale. Il *Data Hub* inoltre permette una maggiore flessibilità nella fase di *Data Ingestion* permettendo l'utilizzo di una molteplicità di sorgenti dati anche molto eterogenee fra di loro.

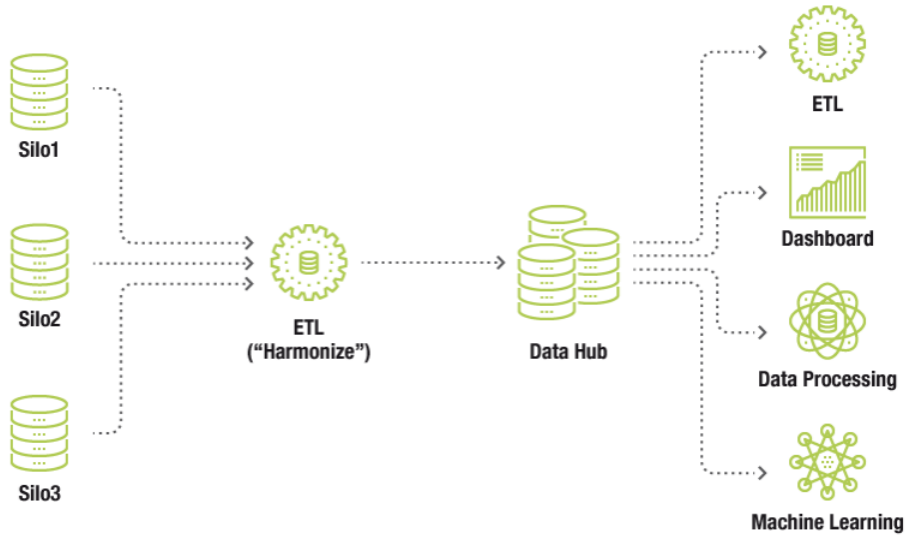

Figura 3.3. Data Hub.

L'esigenza della piattaforma è avere una struttura flessibile che possa integrare dati provenienti da fonti eterogenee, sia dal punto di vista dell'informazione sia dal

punto di vista della struttura del dato. La scelta del *Data Hub* soddisfa tutte le esigenze, non ultima il fatto di riuscire ad integrarsi con eventuali DWH preesistenti.
L'idea che ha portato a utilizzare l'hub come soluzione di storage si basa sulla scelta dell' omogeneizzazione dei dati provenienti da sorgenti diverse e della loro integrazione. Quest'architettura permette di avere flussi indipendenti di *Data Ingestion* per ogni nuova fonte che si integrano grazie all'utilizzo di un modello dati generico.

Modello dati

Per quanto riguarda le diverse fasi di archiviazione e armonizzazione dei dati all'interno dell'hub, sono stati utilizzati tre livelli di elaborazione:

- (L0) Staging Area layer
- (L1) Data Factory layer
- (L2) Data Mart layer

L'area di staging può essere progettata per fornire numerosi vantaggi, ma le motivazioni principali per il suo utilizzo sono l'aumento dell'efficienza dei processi ETL, l'integrità dei dati e il supporto delle operazioni sulla qualità dei dati.
I dati vengono estratti dai sistemi di origine, con diversi metodi (*Data Extraction*) e posizionati nell'area di staging (*Data Ingestion*). Una volta in quest'area, i dati vengono puliti, riformattati e sottoposti a un processo di omogeneizzazione, a causa della loro diversa natura.
L'archiviazione dei dati nel livello *Data Factory* avviene tramite un processo di raffinazione. Questo processo raffina i diversi tipi di dati all'interno di un contesto comune per migliorare la comprensione delle informazioni, eliminando la ridondanza. Questo passaggio é cruciale in quanto, i dati non raffinati, possono provocare errori evidenti sull'output utilizzato dagli utenti di business intelligence. Inoltre, attraverso il processo di raffinazione, i dati vengono trasformati per adeguarsi alle regole aziendali e, solo in questa fase, caricati nel livello *Data Factory*.
Per questo progetto è stato utilizzato un schema *Snow Flake*, con l'intento di renderlo dinamico e il più generale possibile. Questo schema è una variante dello *Star Schema*, che aggiunge la normalizzazione delle tabelle delle dimensioni. L'obbiettivo è quello di affinare il più possibile i dati eliminando le informazioni che non sono di interesse e assicurandosi che le struttura sia ben definita. Il raffinamento avviene anche durante la normalizzazione del database, consentendo di evitare l'incoerenza logica dei dati e riducendo al minimo la duplicazione delle informazioni.
Uno *snow flake* generico può essere ottenuto rendendolo in grado di contenere sia gli attributi statici che dinamici senza la necessità di ripensare le sue strutture successivamente. La normalizzazione delle tabelle delle dimensioni rappresenta un aspetto cruciale per ottenere lo scopo descritto.

Il livello *Data Mart* viene utilizzato per estrarre i dati a specifici utenti o aree aziendali. Un *Data Mart* contiene un sottoinsieme dei dati dell' hub, che consente di isolare l'utilizzo, la manipolazione e lo sviluppo di ogni specifico utente aziendale. Durante il trasferimento dei dati dal livello *Data Factory* al livello *Data Mart*, i dati possono essere ulteriormente perfezionati e aggregati per rispondere alle specifiche esigenze aziendali.

Modello dati generico La figura 3.4 illustra il modello di dati generici implementato nel livello *Data Factory*. Come descritto precedentemente, viene adottato uno schema *snow flake*.
Per quanto riguarda le tabelle dei fatti, il modello contiene due tabelle distinte:

- E-Commerce che include sia il prezzo standard del prodotto che il prezzo scontato, raccolti dai negozi on-line e anche la quantità disponibile del prodotto
- Facts Sellout che contiene tutti i dati sell-out del prodotto forniti dai venditori

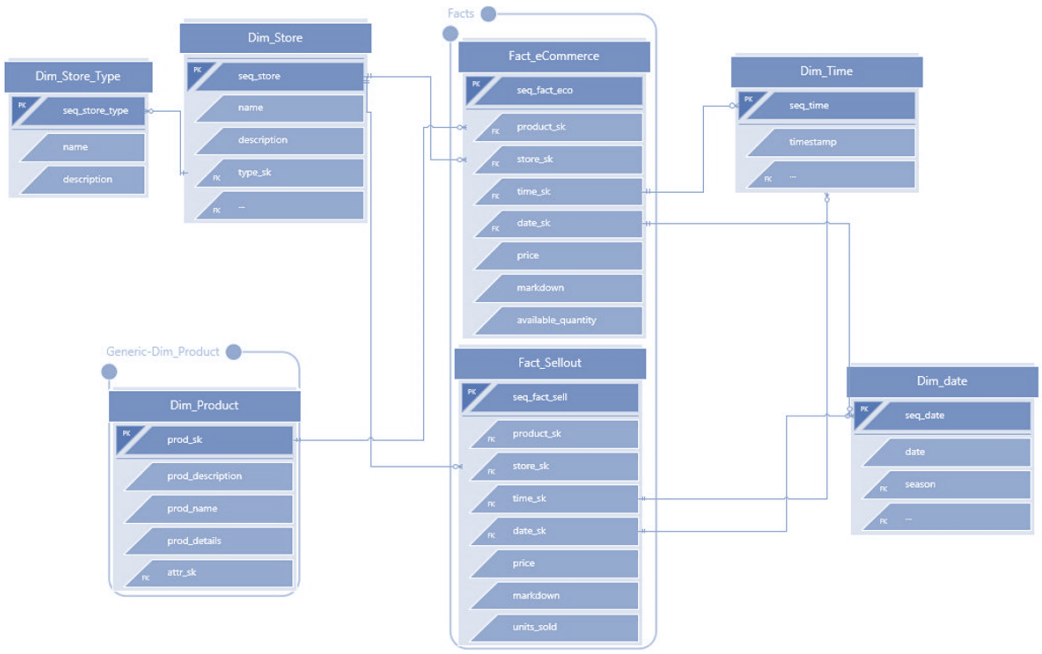

Figura 3.4. Schema a blocchi del modello dati generico

Per quanto riguarda le tabelle delle dimensioni, il modello contiene:

- la dimensione del prodotto, che include le caratteristiche del prodotto, raccolte dai negozi on-line

- una dimensione del negozio, che fornisce informazioni sullo specifico negozio on-line da cui vengono raccolti i dati

- una dimensione data, relativa al giorno della raccolta dati, e anche informazioni utili relative al giorno specifico, come ad esempio la stagione a cui appartiene

- una dimensione temporale, relativa all'ora specifica del giorno in cui i dati vengono raccolti

La scelta per lo schema del *snow flake* svolge un ruolo importante nella realizzazione di un modello generico. È legato al requisito chiave di riuscire a gestire sia attributi statici che dinamici. Infatti, questo obiettivo può essere raggiunto attraverso la normalizzazione della dimensione. In questo caso di studio, i dati raccolti si riferiscono solo alle scarpe di lusso. Di conseguenza, una tabella di dimensioni di prodotto generiche è collegata a una dimensione, che contiene solo gli attributi relativi ai prodotti specifici analizzati, le scarpe di lusso. Tenendo conto della relazione uno-a-molti del modello di dati, le ultime tabelle dimensionali, illustrate nella gerarchia, sono correlate agli attributi dettagliati delle scarpe, come il colore, il materiale, il tacco. La normalizzazione descritta evidenzia un aspetto cruciale che consente di gestire diversi tipi di prodotti all'interno dello stesso modello di dati generici, senza modificarne la struttura. Infatti, se vengono raccolti dati appartenenti a un tipo di prodotto diverso, la tabella delle dimensioni del prodotto generico rimarrà uguale. Sarà invece collegato ad una nuova, diversa, gerarchia orientata al prodotto, che conterrà gli attributi del prodotto specifico.

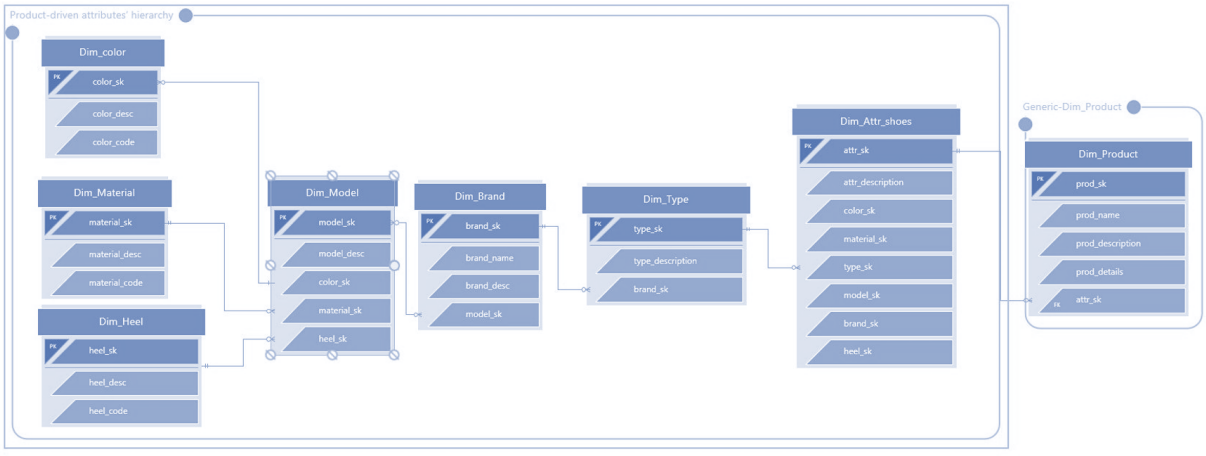

Figura 3.5. Schema di normalizzazione della dimensione Prodotto

3.1.2 Data Ingestion

La fase di *Data Ingestion* consiste nell 'importare dati all' interno di una struttura di storage. Le tecniche e i procedimenti possono essere differenti a seconda del tipo di sorgente dati che si vuole sfruttare. Successivamente all'importazione è necessaria una fase di trasformazione ed elaborazione che permetta di armonizzare il dato e di inserirlo all'interno del modello generico discusso precedentemente. Questa fase successiva permette di integrare i dati non strutturati provenienti dal web con quelli strutturati all'interno del *Data Mart* che contiene i dati aziendali.

Crawler Web

Come indicato in precedenza le informazioni presenti all'interno delle pagine web vengono ricavate estraendole tramite un *Crawler Web*. Questa componente software sfrutta la struttura a oggetti del Web per navigare all'interno delle pagine. I Crawlers, a partire da una URL, permettono di effettuare una navigazione automatica all'interno del codice HTML andando a prelevare le informazioni di interesse. L'algoritmo di scansione Web si può schematicamente descrivere come segue:

1. identificazione del set di URL (*Uniform Resource Locators*) da analizzare

2. Il web crawler scarica tutti i contenuti dei siti web relativi a ciascuno degli URL definiti

3. Vengono sfruttati i link, presenti all'interno delle pagine, sia per navigare all'intero delle piattaforme, sia per raccogliere tutti i dati contenuti dalle pagine di dettaglio re-indirizzate

Il processo continua la navigazione fino a quando tutte le pagine collegate e il loro contenuto vengono estratti. Sebbene l'algoritmo descritto appaia semplice, l'implementazione presenta diverse problematiche. Esse sono legate principalmente all'analisi delle pagine HTML, alle connessioni di rete e ai software che identificano bot web.

Per questo progetto il web crawler è stato realizzato in Python. Questo linguaggio ha diversi aspetti positivi, tra cui una grande flessibilità, un buon numero di librerie che implementano già le funzionalità desiderate ed una veloce curva di apprendimento per i neofiti del linguaggio. Per quanto riguarda le esigenze del cliente specifico, lo scopo consiste nel catturare ogni giorno tutti i dati presenti on-line, sia dei propri prodotti venduti da diversi rivenditori che di prodotti di concorrenti. Questi dati raccolti verranno memorizzati successivamente all'interno dell'hub. Acquisire non solo informazioni legate ai prezzi, ma anche i dettagli della vetrina elettronica, consente di ottenere dati che possono rappresentare indicatori di business cruciali per conseguenti decisioni di marketing.

Inoltre il processo di Web Crawler viene lanciato due volte al giorno, consentendo di mantenere una buona granularità per quanto riguarda il monitoraggio dei prezzi.

Scrapy Si è deciso di realizzare il crawler utilizzando l'architettura *Scrapy*. Scrapy è un framework di crawling web gratuito e open source, scritto in Python. La figura 3.6 mostra una panoramica dell'architettura Scrapy con i suoi componenti e una descrizione del flusso di dati che avviene all'interno del sistema.

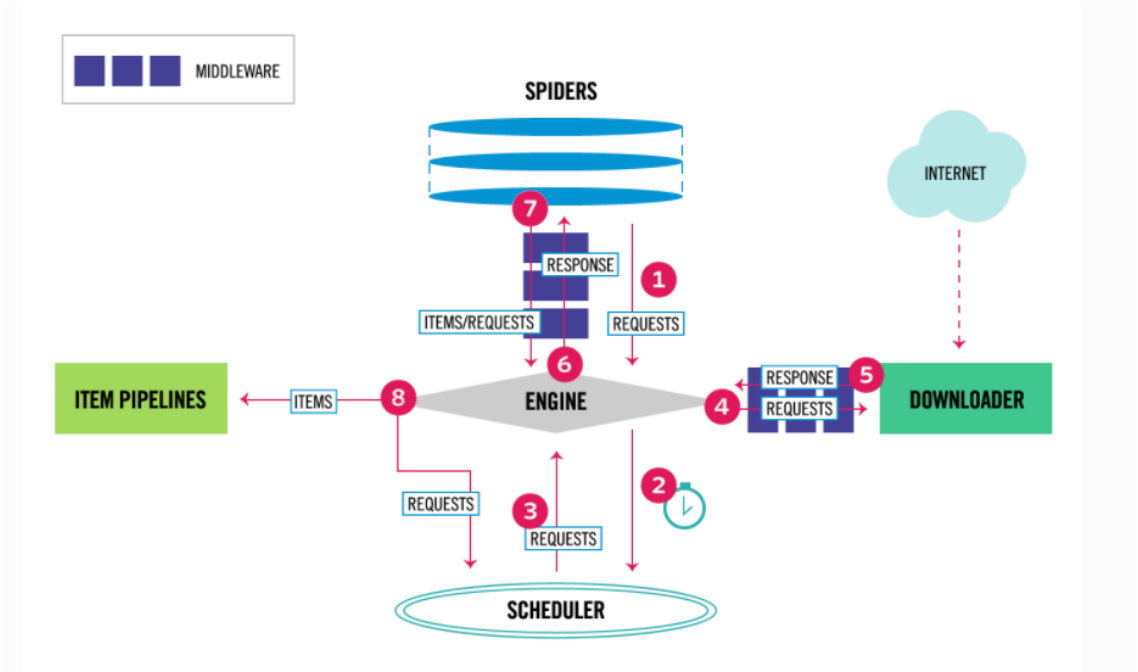

Figura 3.6. Schema del data flow in Scrapy [22]

Spiders Il componente più importante dell'architettura Scrapy è rappresentato dagli Spider. Essi rappresentano le classi in cui la logica di scansione viene implementata. Le decisioni del programmatore condizioneranno le prestazioni dell'architettura, compreso il modo in cui gestire la scalabilità delle richieste HTTP ai link successivi e come estrarre i dati dalle pagine di dettaglio del prodotto.
Ogni Spider segue un ciclo predefinito:

1. Vengono generate le richieste iniziali di scansione dei primi URL, specificando una funzione di callback da richiamare a cui viene passato come parametro la risposta ricevuta da tali richieste

2. Viene effettuato il parsing della risposta (il contenuto della pagina Web) nella funzione di callback e restituiti i dati estratti

3. Gli elementi restituiti dallo spider verranno tipicamente inviati su un database o scritti su un file

Scrapy Engine Questo componente controlla il flusso di dati tra tutti i componenti di Scrapy. Attiva anche eventi in presenza di nuove azioni.

Scheduler Lo scheluder riceve le richieste dall'Engine, inserendole in una coda finché quest'ultimo non le richiede.

Downloader Il Downloader recupera le pagine web assegnandole all' Engine.

Item pipeline L'Item Pipeline è responsabile dell'elaborazione degli oggetti una volta che sono stati estratti dagli spider. Tipicamente, questo componente li memorizza in un database.

Downloader middleware Il Downloader middleware si trova tra l'Engine ed il Downloader. Elabora le richieste quando passano dall'Engine al Downloader e le risposte che passano dal Downloader all'Engine.

Spider middleware Lo Spider middleware si trova tra l'Engine e gli spider ed è in grado di elaborare input (risposte) e output (oggetti e richieste) dello spider.

Il flusso di dati in Scrapy, illustrato nella figura 3.6, si può descrivere in questo modo:

1. L'Engine ottiene le Richieste iniziali per eseguire la scansione dallo Spider;

2. L'Engine pianifica le richieste nello Scheduler e richiede la scansione delle successive richieste;

3. Lo Scheduler restituisce le richieste successive al Engine;

4. L'Engine invia le richieste al Downloader, passando attraverso il Downloader middleware;

5. Una volta che la pagina finisce di scaricare, il Downloader genera una Risposta (con quella pagina) e la invia all'Engine, passando attraverso il Downloader middleware;

6. L'Engine riceve la Risposta dal Downloader e la invia allo Spider per l'elaborazione, passando attraverso il Spider middleware;

7. Lo Spider elabora la risposta e restituisce gli elementi 'raschiati', attraverso la procedura di *scraping*, e le nuove richieste (da seguire) all' Engine, passando attraverso il middleware Spider;

8. L'Engine invia gli articoli elaborati alla Pipeline degli articoli, quindi invia le richieste elaborate allo Scheduler e richiede la successiva richiesta;

9. Il processo si ripete (dal punto 1) fino a quando non ci sono più richieste dallo Scheduler.

[22]

3.1.3 Data Integration

Le informazioni raccolte tramite crawling vengono memorizzate nella loro forma naturale all'interno del primo livello di staging dell'hub. Come descritto in precedenza in questa fase viene effettuato un lavoro di pulizia e controllo del dato.

Omogeneizzazione Un *Data Hub* si differenzia da altri repository di archiviazione omogeneizzando i dati, anziché semplicemente memorizzandoli. Nel progetto è stato implementato un processo di omogeneizzazione per archiviare tutti i dati, provenienti da diverse fonti esterne, nelle stesse tabelle dimensionali e fatti appartenenti al modello generico. La difficoltà del processo di omogeneizzazione deriva dal fatto che i dati iniziali non sono strutturati e i dati estratti da diversi negozi on-line sono molto diversi tra loro. Di conseguenza, sono necessarie diverse tecniche di trasformazione al fine di uniformare i dati allo stesso modello, consentendo la loro memorizzazione nelle stesse tabelle.

Data quality Indubbiamente, una delle procedure più laboriose trattate e implementate è rappresentata dal processo di data quality. La pulizia e il filtraggio dei dati sono essenziali per garantire la qualità dei dati. Una volta che i dati sono stati rifiniti e puliti, è necessario soddisfare i requisiti di qualità. I dati, una volta presenti nell'hub, sono revisionati al fine di rilevare anomalie, incongruenze, valori mancanti o errati. In questo modo, è possibile migliorare la precisione, ottimizzando la gestione di valori sconosciuti e valori anomali.
Il processo di *Data quality* assume un ruolo fondamentale nell'intera catena. Poiché i dati acquisiti provengono da fonti Web esterne, la probabilità di trovare incongruenze nei dati è molto alta. Senza un costante e accurato processo di qualità dei dati, i dati contenenti queste anomalie saranno trasferiti al livello *Data Mart* e possono causare errori nella successiva fase di visualizzazione. Infatti, proprio in questa fase, tutti i clienti specializzati e non tecnici saranno in grado di trovare valori mancanti o incoerenti nei report e nelle dashboard.

Terminate le operazioni del livello di staging, i dati provenienti dall' E-commerce possono essere integrati insieme ai dati aziendali, già strutturati, all'interno del *Data factory*. A questo livello si possono effettuare le operazioni di arricchimento (*Enrichment*) e *Data Mining* in quanto i dati risultano completi ed armonizzati. Come detto in precedenza la piattaforma tramite tecniche di *Text Matching* permette di valutare se due prodotti sono simili.

3.1.4 Text Mining

Il *Text Mining* è il processo per ricavare informazioni di alta qualità dal testo. L'espressione "informazioni di alta qualità" si riferisce al riconoscimento di modelli e tendenze all'interno del testo.
Il processo di estrazione del testo sfrutta tecniche di *Data Mining* per dati testuali, anche non strutturati, ed, in generale, per qualsiasi tipo di documento al fine di:

- identificare i principali gruppi tematici
- classificare i documenti in categorie predefinite
- trovare connessioni nascoste
- eseguire analisi del *sentiment*

Il text mining è approssimativamente equivalente all'analisi del testo. L'analisi del testo è il modo per estrarre il significato da diversi tipi di dati testuali. Consiste in un insieme di tecniche linguistiche, statistiche e di apprendimento automatico che consentono di rivelare i bisogni e i desideri del cliente.
Per questi motivi, il *Text mining* svolge un ruolo significativo nella business intelligence che aiuta le organizzazioni e le imprese ad analizzare sia i loro clienti che i concorrenti per prendere decisioni migliori. [23] Solitamente, i dati provenienti dai negozi on-line di E-commerce sono testuali. Di conseguenza, la loro analisi richiede l'utilizzo di tecniche di *Text mining*.
La prima fase si basa sulla ricerca di parole chiave che possono rivelarsi indicatori cruciali di business. Infatti, per quanto riguarda i prodotti di moda analizzati, in particolare le scarpe di lusso, i dati testuali come brand, categoria, colori, materiali possono influenzare considerevolmente le strategie di marketing e le variazioni dei prezzi. L'estrazione di queste parole chiave richiede un'elaborazione preliminare del testo. "Il metodo di preelaborazione svolge un ruolo molto importante nelle tecniche e nelle applicazioni di text mining ed è il primo passo del processo." [24]

Nel processo di elaborazione testuale del progetto i tre passaggi chiave della fase di pre-elaborazione sono:

- Espressioni regolari
- Stop Words elimination
- Stemming

Espressioni regolari L'espressione regolare rappresenta uno dei successi più importanti nella standardizzazione in informatica. È un linguaggio per specificare le stringhe di ricerca del testo. Viene utilizzato in tutti i linguaggi informatici, word processor e strumenti di elaborazione del testo come gli strumenti Unix grep o Emacs. L'utilizzo delle espressioni regolari può rappresentare una delle migliori soluzioni in presenza di un modello da cercare e un *corpus* di testi da cercare. Nel caso specifico, il *corpus* può essere il nome del prodotto, i dettagli o la descrizione del prodotto. In questo modo, è possibile estrarre le parole più rilevanti dai campi menzionati. Inoltre, l'espressione regolare viene applicata anche ai valori dei prezzi provenienti dai diversi siti Web, al fine di omogeneizzarli in un formato unico e uniforme.

Stop Words elimination Le *Stop Word* sono parole che vengono filtrate prima o dopo l'elaborazione di dati in linguaggio naturale (testo). Poiché rappresentano una divisione del linguaggio naturale, la loro eliminazione consente di ridurre la dimensionalità dello spazio temporale, rendendo il testo meno pesante e più significativo per gli analisti.
Le *Stop Word* sono rappresentate da termini che non forniscono informazioni aggiuntive al testo, come gli articoli, le congiunzioni, le preposizioni e così via. Questo processo migliora la possibilità di trovare la parola chiave nel testo filtrato.

Stemming Lo *Stemming* rappresenta il processo di riduzione delle parole alla loro parola radice, o base. "Lo scopo di questo metodo è rimuovere vari suffissi, ridurre il numero di parole ed avere corrispondenze precise per risparmiare tempo e spazio di memoria." [24]
Nel caso di studio, la derivazione viene applicata alle parole chiave aziendali che dovevano essere estratte. Ad esempio, considerando un colore, viene estratta solo la forma singolare, al fine di consentire una successiva identificazione della corrispondenza tra i colori di diversi prodotti.

Successivamente sono state implementate una normalizzazione e l'unificazione della lingua del testo in modo da rendere possibile l'estrazione e l'analisi delle più importanti caratteristiche del prodotto.

Text Clustering

L'analisi del cluster consiste nel trovare gruppi di oggetti (cluster) in modo tale che siano simili (o correlati) tra loro e diversi (o non correlati) dagli oggetti in altri gruppi. [25] L'intento principale dell'analisi dei cluster consiste nel ridurre al minimo le distanze intra-cluster e massimizzare le distanze tra i cluster. Esistono due tipi principali di clustering:

- Cluster gerarchico: i cluster sono organizzati in un albero gerarchico
- Clustering parziale: è composto da sottoinsiemi non sovrapposti

Per quanto riguarda gli algoritmi di clustering, i più importanti sono:

- K-Means e le sue varianti
- Cluster gerarchico
- Cluster basato sulla densità

Il clustering del testo è l'applicazione dell'analisi del cluster ai dati testuali. Può essere utilizzato per molteplici scopi, come l'estrazione di argomenti, il raggruppamento di documenti simili, la scoperta di informazioni nascoste e implicite nei documenti. In genere, il clustering del testo richiede fasi di mining di testo preparatorie come la tokenizzazione (analisi dei dati di testo in unità più piccole, token, come parole e frasi), arginamento e lemmatizzazione, eliminazione delle parole . Per questo progetto viene utilizzato il clustering, che implica l'impiego di descrittori e la loro estrazione. I descrittori consistono in un insieme di parole che rappresentano il contenuto all'interno di un cluster.
Il suo impiego deriva dall' intento di identificare lo stesso prodotto, appartenente alla stessa marca, venduto sui diversi siti web. Di fatto, in genere, pagine di dettagli differenti di diversi negozi on-line possono mostrare lo stesso prodotto con nomi di prodotti diversi. Di conseguenza, senza un appropriato algoritmo di matching, lo stesso prodotto con più nomi di prodotti si rivelerà essere come due prodotti diversi. Poiché i prodotti saranno sottoposti ad analisi commerciali che influenzano le decisioni di marketing, questa evidente approssimazione non può essere accettata. Quindi, per risolvere il problema della corrispondenza, l'approccio del cluster di testo può rappresentare una buona soluzione per risolvere il problema di abbinamento. Una volta completata la pre-elaborazione del testo, è possibile estrarre le parole chiave e gli indicatori aziendali dai dati testuali. L'implementazione del cluster di testo consiste nell'estrazione di parole chiave comuni, come colore, marca, materiale e categoria, da tutti i prodotti che identificano ogni combinazione di queste parole chiave come il contenuto di un particolare cluster. Pertanto, nel fare ciò, anche se un prodotto mostra un nome diverso su distinti negozi on-line, esibendo le stesse caratteristiche chiave, apparterrà allo stesso cluster. La figura 3.7 mostra il contenuto all'interno di un singolo cluster.

Figura 3.7. Schema del Clustering del prodotto

3.2 Fruizione della piattaforma

La piattaforma di Business Analytics così costruita è stata progettata per essere utilizzata come servizio. Su indicazione delle esigenze dei diversi clienti vengono selezionate le piattaforme E-commerce da monitorare. In seguito vengono implementati i crawler web in modo da effettuare periodicamente la *Data Ingestion* dei dati catturati. Per la natura di servizio della piattaforma è stato scelto di posizionare il *Data Hub*, all'interno del Cloud aziendale. In questo modo ci sono diversi vantaggi tra cui un maggior controllo della piattaforma, la distribuzione delle risorse e non ultimo un minor costo da parte del cliente. Tramite i processi di ETL all'interno dell' hub vengono costruiti i vari livelli del dato, fino ad ottenere un'informazione pronta per essere inserita nel *Data Mart* del cliente. A questo punto l'architettura permette due diversi tipi di soluzione a seconda delle esigenze:

- Data Mart Saas
- Data Mart On-premises

La soluzione SaaS(*Software as a Service*) è un modello di distribuzione di un servizio. Secondo questo modello il distributore gestisce ed eroga il servizio verso i propri clienti fornendo un accesso, solitamente tramite internet, regolato da un abbonamento. Per questo motivo in questi casi si parla di *Cloud Computing*. Questa modalità prevede la *Data Ingestion* dei dati dei clienti all'interno dell' hub e la presenza del *Data Mart* all'interno del Cloud aziendale. Il modello *On-premises* prevede invece che i dati risiedano all'interno della rete del cliente. Per questo

motivo, con questa soluzione, il livello *Data Mart* non risiederà più nell' hub ma bensì all'interno del Cloud del cliente.

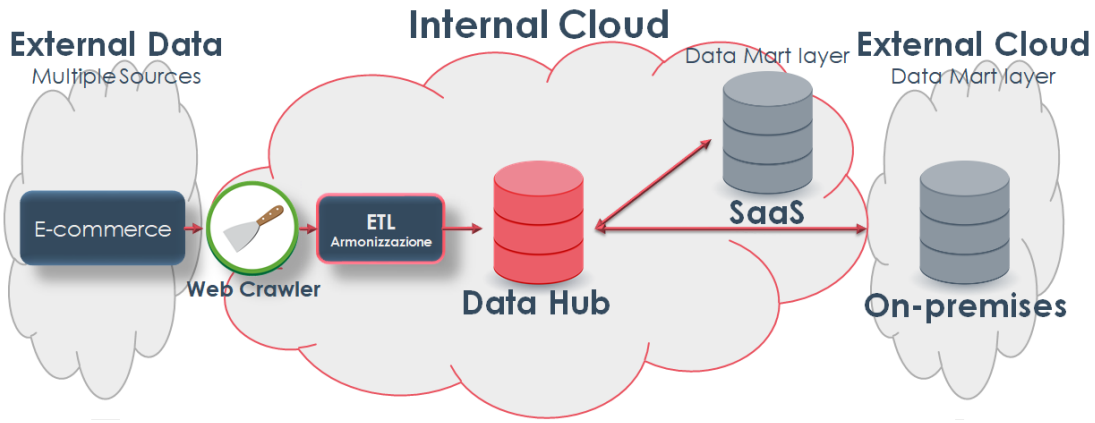

Figura 3.8. Architettura iniziale della piattaforma

La piattaforma costruita è per natura modulare permettendo facilmente di adattarsi a nuove esigenze. Ogniqualvolta si dovranno effettuare delle modifiche non sarà necessario riprogettare e, quindi, modificare l'architettura. Le nuove funzionalità saranno aggiunte inserendo nuovi moduli oppure dimensioni al modello generico.

3.3 Criticità

La piattaforma così descritta è stata funzionale ed ha permesso, tramite tecniche di analisi dati avanzate, di ottenere dei buoni risultati integrando i dati delle piattaforme E-commerce ed i dati di vendita del cliente. In ottica di effettuare un miglioramento del servizio si è deciso di fare un'analisi critica, soprattutto per quanto riguarda la soluzione di *Text Matching*.
All'interno dei processi di ETL, in particolare all'interno del livello *Data Factory*, è dove viene implementato il *Text Matching*. Tramite questo processo si riesce ad effettuare un match dei prodotti in base al testo delle caratteristiche che li identificano univocamente:

- Brand
- Colore
- Materiale
- Categoria

Ogni prodotto che presenta le stesse caratteristiche, tramite il *Text Clustering*, viene inserito all'interno di un cluster specifico. Ogni cluster dovrebbe contenere lo stesso tipo di prodotto. Dalle analisi che si sono effettuate sulla qualità del dato non è sempre possibile riuscire a creare dei cluster che identifichino univocamente un prodotto. Per esempio, è possibile che uno stesso Brand produca due prodotti diversi ma che abbiano in comune lo stesso colore, materiale e categoria. In quest'ottica si è pensato di introdurre la tecnica di *Image Matching* in modo da effettuare degli abbinamenti più precisi basati sulla similarità delle immagini. Anche questa tecnica, come verrà spiegato nel dettaglio successivamente, possiede alcune limitazioni.
Dato che in generale la singola tecnica difficilmente soddisferà le esigenze del caso di studio si è ipotizzata una strategia alternativa. L'idea è che ogni tecnica metta in relazione due prodotti in base a determinate caratteristiche. Quante più caratteristiche due prodotti hanno in comune quanto più e possibile essere sicuri che si tratti dello stesso prodotto. Per questo motivo si è deciso di sfruttare la tecnologia di un Graph Data Base. Questo strumento permette di immagazzinare le diverse relazioni di similitudine tra due prodotti e fornire un motore di raccomandazione in base alle loro connessioni.

3.4 Data Enrichment

Come detto in precedenza la piattaforma presenta una natura flessibile grazie alla modularità del sistema e all'utilizzo di un modello dati generico. Grazie a questa caratteristiche è possibile aumentare la qualità del dato tramite delle tecniche di *Data Enrichment* senza dover riprogettare il modello dati. Per questo caso di studio la soluzione ipotizzata prevede di aggiungere una nuova dimensione al modello atta a descrivere le caratteristiche dell'immagine legata al prodotto. Di fatto questa modifica consiste nell'aggiungere la dimensione immagine, come si può notare nella figura 3.9. Questa dimensione, a differenza delle altre presenti nel modello, ricava informazioni a partire direttamente dai pixel di un' immagine quindi non necessita di una fase di pulizia e omogeneizzazione. Per questo motivo si può dire che nasca direttamente all'interno del livello *Data Factory* (L1).

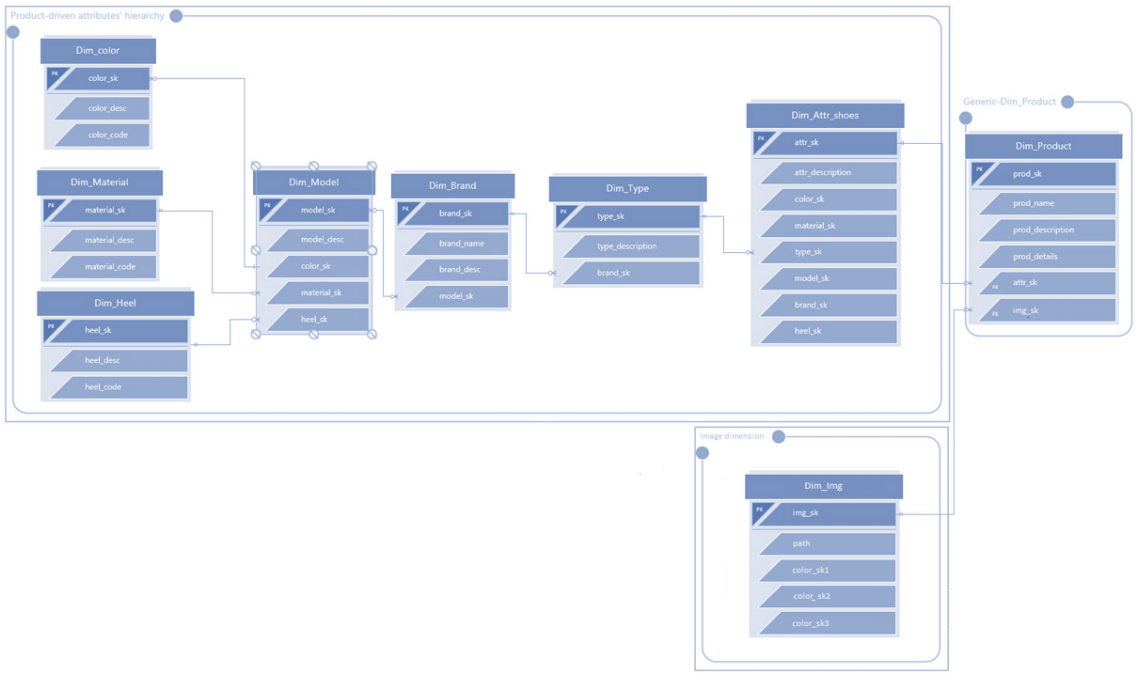

Figura 3.9. Schema del modello con l'inserimento della dimensione immagine

3.4.1 Dimensione Immagine

La costruzione della dimensione dell'immagine è stata possibile creando un modulo in linguaggio Python. All'interno del modulo sono state create dei metodi che hanno permesso di implementare le funzioni di trasformazione illustrate nel II capitolo. In particolare si è sfruttata una libreria chiamata Open CV.

Open CV OpenCV (*Open Source Computer Vision Library*) è una libreria open source di *Computer Vision* e di *Machine Learning*. OpenCV è stata realizzata con l'intento di fornire un'infrastruttura comune per le applicazioni di *Computer Vision* e per accelerare l'uso del *Machine Perception*[1] nei prodotti commerciali. Essendo un prodotto con licenza BSD[2], OpenCV semplifica l'utilizzo e la modifica del codice

[1]Il *Machine Perception*è la capacità di un sistema di elaborazione di interpretare i dati in un maniera similare a quella che il sistema sensoriale umano utilizza per percepire ciò che lo circonda

[2]Le licenze BSD assicurano le quattro libertà del software e sono definite come licenze per il software libero dalla FSF (*Free Software Foundation*). Per la FSF, un software viene definito libero solo se garantisce:

1. Libertà di eseguire il programma per qualsiasi scopo. La libertà di usare un programma significa libertà per qualsiasi tipo di organizzazione o persona di utilizzarlo su un qualsiasi

da parte delle aziende.
La scelta è stata dettata, quindi, per ragioni legate all'utilizzo libero del software ed alla sua grande diffusione. Inoltre possiede interfacce C ++, Python, Java e MATLAB e supporta Windows, Linux, Android e Mac OS. Questo permette di integrare OpenCV nella maggior parte dei software.

Il modulo progettato, oltre a dover elaborare le immagini, aveva necessità di integrarsi all'interno dell'hub. Per questo motivo è stato necessario utilizzare delle librerie che permettessero la lettura e la scrittura su Data Base. In particolare si è scelto di sfruttare un'architettura ORM.

ORM *Object-relational mapping* (ORM) è una tecnica di programmazione in cui un descrittore di metadati viene utilizzato per connettere il codice oggetto a un database relazionale. Il codice oggetto è scritto in linguaggi di programmazione orientata agli oggetti (OOP), come in questo caso Python. ORM converte i bit appartenenti a due sistemi di tipi dati, ovvero tra i database relazionali e ed i linguaggi OOP, che altrimenti con comunicherebbero. I principali vantaggi di questo approccio sono:

- Sviluppo semplificato, perché automatizza la conversione da oggetto a tabella e da tabella a oggetto, con conseguente riduzione dei costi di sviluppo e manutenzione

---

sistema informatico, per qualsiasi attività e senza avere bisogno di comunicare successivamente con lo sviluppatore o con altre entità specifiche. Il punto fondamentale per questa libertà è lo scopo finale dell'utente e non dello sviluppatore; come utenti è possibile eseguire il programma per il proprio scopo; se si ridistribuisce ad altri, essi sono liberi di eseguirlo per i propri scopi, ma non si può imporgli i nostri.

2. Libertà di studiare il codice del programma ed eventualmente modificarlo. L'accessibilità al codice sorgente è una condizione fondamentale per il software libero, se così non fosse non avrebbero senso le libertà 1 e 3.

3. Libertà di ridistribuire copie del programma con lo scopo di aiutare il prossimo.

4. Libertà di migliorare il programma (modificandolo) e di distribuirne in modo pubblico i miglioramenti, in modo tale che tutti ne traggano beneficio. Questa libertà comprende quella di utilizzare e rilasciare le versioni modificate come software libero. Una licenza libera può anche permettere altri modi di distribuzione, non c'è l'obbligo che si tratti di una licenza con copyleft. D'altro canto, una licenza che imponesse che le versioni modificate non siano libere non si può definire come licenza libera.

- Meno codice con SQL incorporato e stored procedure[3] scritte a mano
- Memorizzazione nella cache di oggetti trasparenti a livello applicativo, migliorando le prestazioni del sistema
- Soluzione ottimizzata che rende un'applicazione più veloce e facile da mantenere

D'altro canto le principali preoccupazioni riguardano il fatto che l'ORM non ha prestazioni ottimali, infatti una stored procedure potrebbe rappresentare una soluzione migliore da questo punto di vista, eliminando uno strato software ed agendo direttamente sul Data Base. Per la piattaforma in questione una delle condizioni fondamentali è la flessibilità. L'utilizzo di ORM permette di rendere il codice in Python completamente libero rispetto alla tecnologia con cui è implementato il Data Base. Nel caso in cui, in futuro, si voglia cambiare tipo di Data Base, oppure riutilizzare il modulo in un altro progetto, è possibile riusare il codice con poca manutenzione. Il modulo per l'elaborazione realizza tre funzionalità principali necessarie per la creazione della dimensione immagine:

- Acquisizione delle Immagini
- Normalizzazione
- Image Matching

Acquisizione delle Immagini

Questa funzionalità permette, a partire da una URL, di effettuare il download e la scrittura del file immagine all'interno del file system. E' stato ipotizzato di non sfruttare il tipo dato CLOB per archiviare le immagini all'interno dell'hub, in quanto si vuole mantenere uno spazio di archiviazione dedicato su file system non occupando le risorse dell'hub. Le immagini vengono scaricate a partire dalle URL prelevate tramite la componente ORM dalla dimensione prodotto. La dimensione immagine risulta quindi associata al prodotto, questo legame viene rappresentato tramite una chiave esterna generata in questo passaggio.
All'interno della dimensione immagine si memorizza quindi il path del file system di dove è stata salvata l'immagine originale.

[3]Una stored procedure è un codice SQL preparato che è possibile salvare e che risiede sul Data Base, quindi il codice può essere riutilizzato più e più volte

Normalizzazione

La funzionalità di normalizzazione applica le seguenti trasformazioni a partire dalle immagine acquisite in precedenza:

1. Clustering dei colori
2. Thresholding
3. Contornamento
4. Ridimensionamento

Come indicato nel capitolo II, tramite questa fase si ottiene un data set di immagini uniformi e confrontabili. Queste immagini vengono archiviate sul file system mentre invece l'originale viene cancellata. Oltre ad ottenere il data set per l'*Image Matching* durante la fase di clustering dei colori vengono aggiunte all'interno della dimensione i tre colori principali dell'immagine. In questo modo viene arricchita l'informazione sull'immagine.

Image matching

Il modulo di elaborazione arrivato a questo punto è riuscito ad aggiungere le dimensione immagine ai prodotti elaborati. Ciò che manca ai fini del progetto è confrontare le immagini tra loro generando un indicatore di similarità. Come descritto in precedenza la tecnica scelta risulta essere SSIM. La funzionalità di Image Matching a partire da due prodotti estrae le immagini memorizzate su file system tramite il processo di normalizzazione ed applica SSIM. Il risultato dell'elaborazione viene aggiunto alla dimensione completandola.

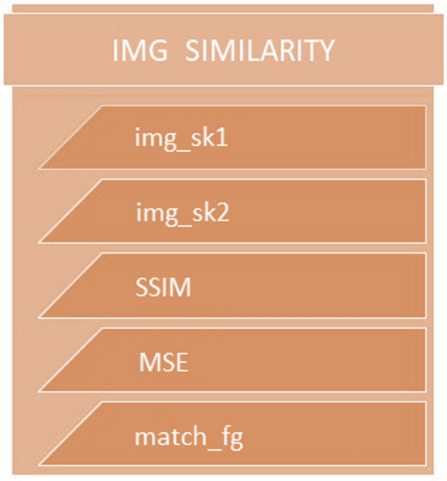

Figura 3.10. Schema tabella di similarità

Il risultato ottenuto integra il modello generico inserendo la nuova dimensione immagine, legata al prodotto. Invece il processo di matching aggiunge una nuova tabella che rappresenta la similarità tra due immagini. Quest' informazione rappresenta la relazione tra le immagini di due prodotti. L'associazione verrà utilizzata all'interno del Graph DB per connettere due prodotti tramite un legame di similarità dell'immagine. Come si può notare in figura 3.10, la tabella di similarità contiene due chiavi esterne legate alle immagini confrontate, due colonne che rappresentano i valori ottenuti dagli algoritmi SSIM e MSE e d un flag che rappresenta il match di due immagini effettuato visivamente

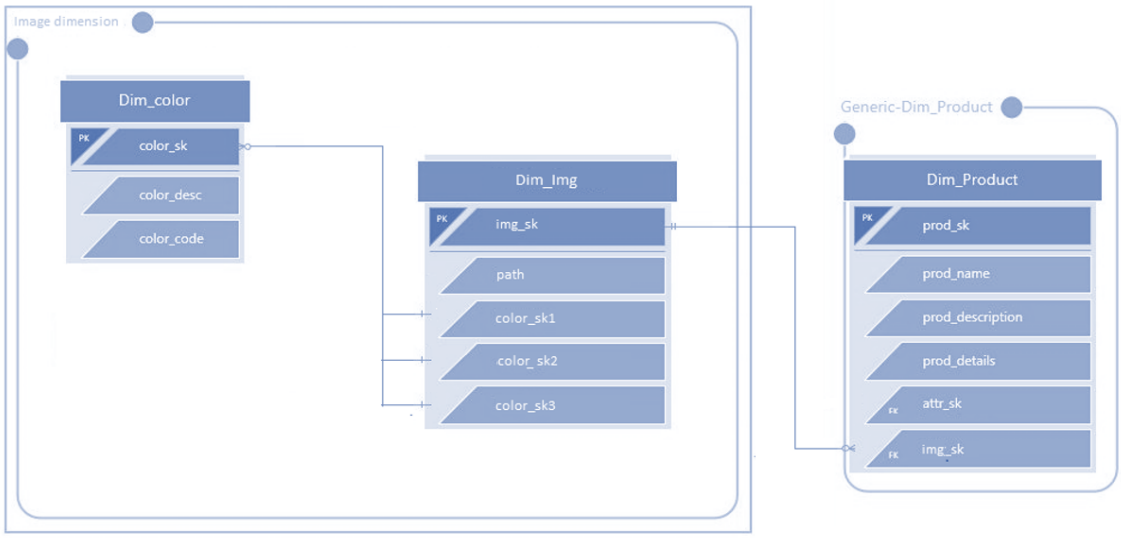

Figura 3.11. Modello dati per la dimensione immagine

Come si può notare nella figura 3.11 è la dimensione prodotto generica ad essere collegata all'immagine. Questa scelta è legata al fatto che l'immagine è presente in qualsiasi prodotto E-commerce e non solo al prodotto specifico. In questo modo il modello permetterebbe di riutilizzare la dimensione immagine anche se si dovesse cambiare focus di analisi ed aggiungere nuovi dati appartenenti a tipi di prodotto differenti, il tutto senza dover riprogettare l'architettura.

3.5 Recommendation

Per completare il progetto e riuscire ad utilizzare entrambe le tecniche messe a punto è necessario trovare un modo per farle coesistere. Per questo motivo è stato scelto di creare un sistema di raccomandazione che, in base alla similarità calcolata tramite le due tecniche, riesca a dare un peso al legame tra due prodotti, suggerendo quindi

se siano simili. Per implementare questo sistema è stata utilizzata la tecnologia del Graph Data Base.

3.5.1 Graph Data Base

I Data Base relazionali (RDBMS) sono progettati per contenere informazioni fortemente strutturate ed inserite all'interno di un modello dati. Malgrado il loro nome indichi il contrario, i database relazionali non sono adatti per dati che risultano strettamente connessi. Per loro natura non riescono ad immagazzinare in maniera robusta la relazione che collega i diversi elementi presenti all'interno dei dati. I Graph DB sono dei *Data Base* progettati per gestire e memorizzare i dati in funzione delle relazioni che collegano i vari elementi. Quest'architettura si basa semplicemente su due tipi di oggetti: i nodi e le relazioni. Ogni nodo rappresenta un'entità (una persona, un luogo, una cosa, una categoria , ...) e ogni relazione rappresenta il modo in cui due nodi sono associati. Ad esempio, i due nodi "torta" e "dessert" avrebbero la relazione "è un tipo di" puntamento da "torta" a "dessert". Questa struttura generica consente di modellare tutti i tipi di scenari - da un sistema di strade, a una rete di dispositivi, alla storia medica di una popolazione o qualsiasi altra cosa definita da relazioni. Allo stato attuale le aziende sfruttano questa tecnologia in vari modi, i campi più diffusi sono:

1. Intercettazione di frodi
2. Motori per la raccomandazione
3. Master data management (MDM)
4. Operazioni di rete e IT
5. Gestione dell'identità e accesso (IAM)
6. Ricerca basata su grafi

La parte cruciale di questo progetto è riuscire a collegare due prodotti presenti su piattaforme E-commerce differenti e, riuscendo a confrontare le loro caratteristiche, capire quanto essi siano simili. Più è alto il grado di similarità più siamo sicuri che si tratti dello stesso prodotto. L'idea che ha spinto ad utilizzare il Graph DB è proprio quella di collegare tramite una relazione di similarità due prodotti. Nel progetto preesistente la relazione era rappresentata dal risultato del *Text Matching* mentre il modulo di elaborazione delle immagini aggiunge un legame di similarità ottenuto con l' *Image Matching*. Questi legami possono essere rappresentati sul Graph DB in maniera molto intuitiva utilizzando dei nodi che rappresentano i Prodotti in vendita on-line. Mentre l'informazione che rappresenta la diversa tecnica di matching può essere rappresentata da una relazione (similarByText o similarByImage). Il Grafo

ottenuto in questa maniera, oltre a restituire una rappresentazione visiva delle relazioni, può essere utilizzato come motore di raccomandazione. Associando un peso al diverso tipo di relazione il legame tra due prodotti è maggiore quanto più il peso della somma delle relazioni è maggiore. Questo tipo di rappresentazione permette di suggerire quanto due prodotti sono simili basandosi sul peso totale delle relazioni tra due nodi. Ciò che si ottiene è un insieme di grafi che connettono tra di loro i diversi prodotti in base a diversi tipi di relazione. La figura 3.12 può dare un'idea a livello visivo del risultato.

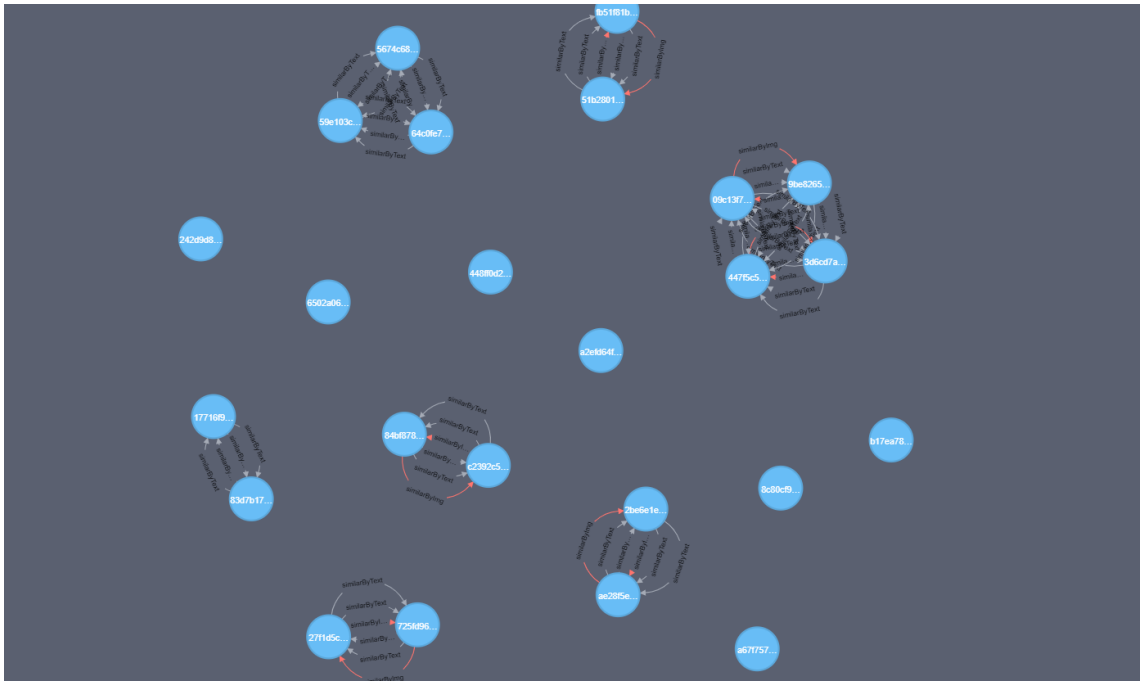

Figura 3.12. Esempio di visualizzazione del Graph DB.

I nodi che non hanno nè archi entranti nè archi uscenti sono quei prodotti che non risultano simili a nessun altro. I nodi connessi tramite un arco rosso risultano simili in base all'immagine, mentre quelli connessi in grigio appartengono allo stesso cluster testuale.

Una volta sviluppati i vari moduli è interessante poter avere una visione d'insieme del progetto. La piattaforma ha subito un'evoluzione sostanziale dopo aver integrato i nuovi moduli arricchendo le funzionalità ed aumentando la qualità delle informazioni fornite, come si può notare dalla figura 3.13.

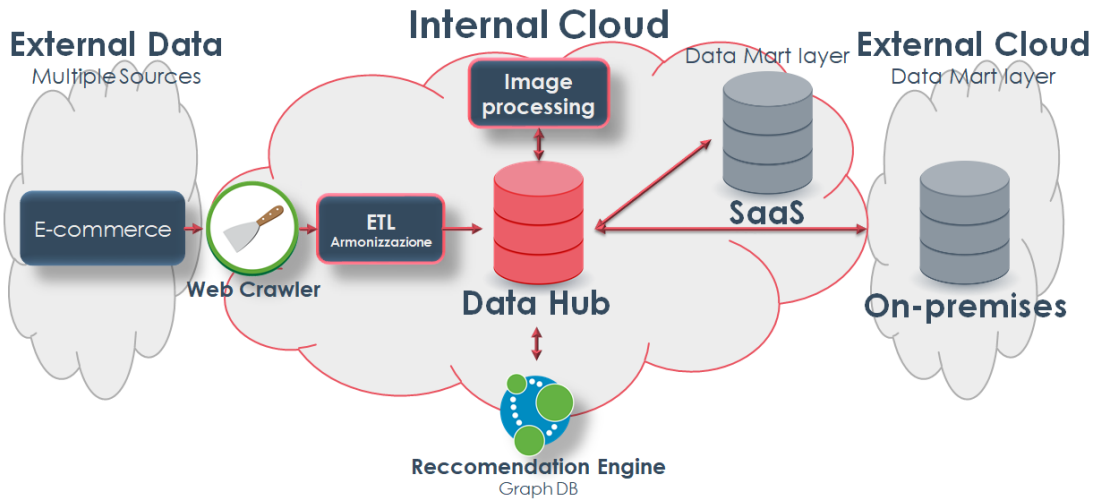

Figura 3.13. Architettura finale della piattaforma.

Nonostante la modifica sostanziale, si può notare che si è evitato di riprogettare l'intera struttura. Questo perchè la parte centrale del progetto sono i dati e i processi di elaborazione degli stessi. La centralità dell'informazione permette di integrare nuove funzionalità a seconda delle esigenze 'semplicemente' aggiungendo nuovi moduli e gestendo la fase di integrazione tramite i processi di ETL.

Capitolo 4

Risultati

Una parte sicuramente interessante è stata la valutazione delle diverse metodologie e dei rispettivi risultati. L'individuazione dei punti di forza e di debolezza delle diverse tecniche di matching permette di scegliere la migliore strategia da seguire. Per i problemi legati all'analisi dei dati ci sono alcune metodologie che permettono un approccio in grado da un lato di ottenere dei risultati in breve tempo e dall'altro di migliorarli seguendo precisi step. Per questo caso di studio è stata usata la metodologia CRISP-DM. Le valutazioni sul progetto hanno permesso di raggiungere un risultato che sfrutti i punti di forze di entrambe le tecniche, sia l' *Image Matching* che il *Text Matching*.

4.1 Modello CRISP-DM

Il processo di valutazione rispecchia completamente la metodologia CRISP-DM (*Cross-industry standard for Data Mining*). Essa è una tecnica di analisi per i processi analitici. Di fatto al giorno d'oggi è la tecnica più diffusa per i problemi legati all'analisi dei dati. Lo schema del processo in figura 4.1 rende esplicito il fatto che l'iterazione è la regola piuttosto che l'eccezione. Passare attraverso il processo una volta senza aver risolto completamente il problema non è, in generale, un fallimento. Spesso l'intero processo è un'esplorazione dei dati e, dopo la prima iterazione, si riesce a conoscere molto più a fondo il problema.
La metodologia prevede sei fasi, che possono essere ripetute ciclicamente con l'obiettivo di revisionare e rifinire il modello:

1. Business Understanding
2. Data Understanding
3. Data Preparation
4. Modeling

5. Evaluation
6. Deployment

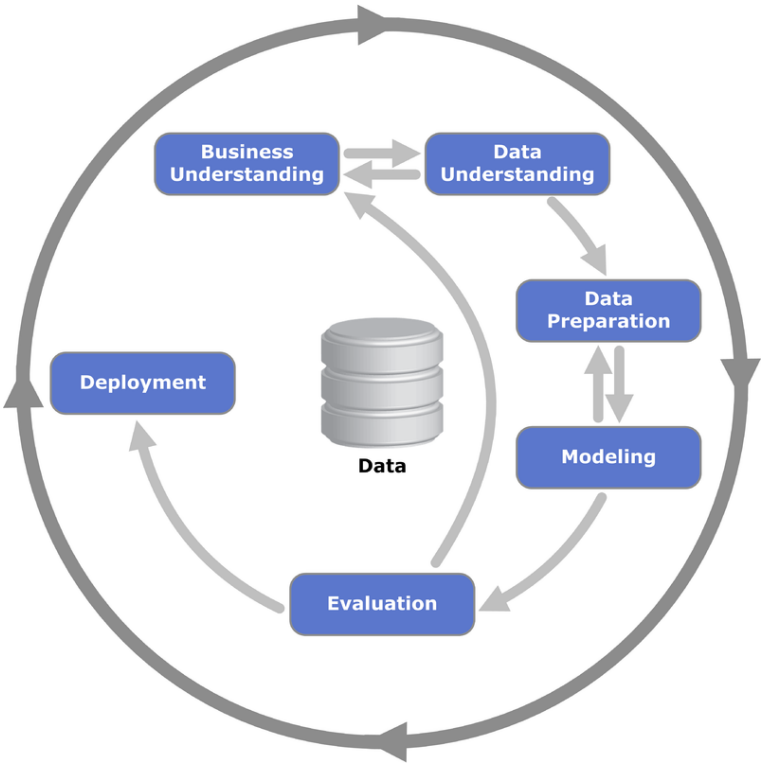

Figura 4.1. Schema del processo CRISP-DM [26]

Business Understanding Inizialmente, è fondamentale capire il problema da risolvere. Questo può sembrare ovvio, ma raramente i progetti aziendali vengono preconfezionati come problemi chiari e non ambigui. Spesso la rifusione del problema e la progettazione di una soluzione è un processo iterativo di scoperta. Lo schema in figura 4.1 rappresenta questo come cicli all'interno di un ciclo, piuttosto che un semplice processo lineare. La formulazione iniziale del problema potrebbe non essere completa o ottimale, quindi possono essere necessarie più iterazioni per una formulazione della soluzione accettabile. La fase di *Business Understanding* rappresenta una parte di lavoro da analista per cui la creatività gioca un ruolo importante. Spesso la chiave per raggiungere il risultato è la formulazione di una soluzione creativa. Esistono diversi strumenti che ci permettono di risolvere problemi di *Data Mining*. Quindi la fase iniziale richiede di riuscire ad individuare gli strumenti più adatti a risolvere il problema specifico. Questo richiede una fase di strutturazione (ingegnerizzazione) del problema. In questa fase iniziale è necessario

riflettere attentamente sullo scenario d'uso: Che cosa esattamente vogliamo fare? Come lo faremo? Per questo motivo è utile iniziare con una visione semplificata dello scenario d'uso, ma mentre andremo avanti, realizzeremo che spesso lo scenario d'uso deve essere rimodellato revisionando gli step precedenti per riflettere meglio le reali esigenze aziendali.

Data Understanding Se l'obiettivo è risolvere il problema aziendale, i dati sono la materia prima da cui verrà costruita la soluzione. È importante capire i punti di forza e le limitazioni dei dati perché raramente c'è una corrispondenza esatta con il problema. I dati storici vengono spesso raccolti per scopi estranei al problema attuale o per nessun scopo esplicito. Inoltre è possibile che contengano informazioni molto diverse tra loro e possono avere vari gradi di affidabilità. I dati possono avere diversi costi, possono essere già pronti oppure richiedere degli sforzi per essere ricavati, altri ancora dovranno essere acquistati. Quindi una parte fondamentale della fase di comprensione dei dati è la stima dei costi e dei benefici di ciascuna fonte di dati e, quindi, se la decisione di un ulteriore investimento sia giustificata. Anche dopo che i set di dati vengono acquisiti, la loro raccolta può richiedere comunque uno sforzo supplementare. Nella *Data Understanding* abbiamo bisogno di andare sotto la superficie per scoprire la struttura del problema aziendale e i dati disponibili, quindi abbinarli a uno o più task di *Data Mining* per i quali potremmo avere per ognuno delle tecniche o delle tecnologie diverse da applicare. Non è insolito che un problema aziendale contenga più task, spesso di diverso tipo, e per raggiungere la soluzione sarà fondamentale riuscire a combinare i diversi risultati.

Data Preparation Le tecnologie analitiche che possiamo utilizzare sono uno strumento molto utile ma impongono determinati requisiti sui dati che utilizzano. Spesso accade che i dati siano in una forma diversa dal modo in cui essi vengono forniti nella loro forma naturale e sarà necessario un procedimento di conversione. Questa fase è, appunto, la *Data Preparation*.

Modeling Questa fase (*Modeling*) genera un modello che riesce ad adattarsi ai dati mettendo in risalto i pattern contenuti al loro interno organizzando l'informazione. Quanto più un modello soddisfa i requisiti iniziali tanto più è semplice ricavare le informazioni di interesse.

Evaluation Lo scopo della fase di valutazione (*Evaluation*) è di misurare i risultati ottenuti in modo rigoroso per ottenere la certezza che siano validi e affidabili prima di andare avanti. Vorremmo avere fiducia che i modelli estratti dai dati siano *pattern* reali e non solo idiosincrasie o anomalie campionarie. È possibile distribuire i risultati immediatamente, prima di effettuare questa fase, ma è altamente sconsigliabile; di solito è molto più facile, economico, veloce e sicuro testare in un contesto controllato.

Deployment Nella fase di distribuzione (*Deployment*) si sfruttano gli strumenti creati per casi reali. I casi più diffusi di implementazione implicano l'utilizzo di un modello predittivo all'interno di sistemi informativi o processi aziendali.

E' interessante notare che non è necessario fallire nella distribuzione per riavviare il ciclo. La fase di valutazione potrebbe rivelare che i risultati non siano abbastanza buoni da essere distribuiti e ci sia la necessità di aggiustare la definizione del problema oppure di ottenere dati diversi. Questo è rappresentato, nel diagramma di processo, dal collegamento tra lo stadio di valutazione che ritorna alla *Business Understanding*. In pratica, dovrebbero esserci scorciatoie da ogni fase a ciascuna delle precedenti poiché il processo conserva sempre alcuni aspetti esplorativi ed un progetto dovrebbe essere abbastanza flessibile da poter essere rivisto in base alle scoperte fatte.

Nel caso di studio affrontato si possono considerare come tre cicli successivi del modello CRISP-DM le tre fasi associate alle tecniche di matching utilizzate. Infatti la fase di *Text Matching* si può considerare il primo ciclo di studio del problema di business, che ha permesso la costruzione iniziale della piattaforma e l'utilizzo del modello generico. Inoltre si sono applicate tecniche di *Text Mining* per ottenere le informazioni desiderate. La seconda fase corrisponde all'integrazione del modulo di elaborazione delle immagini(*Image Matching*) e all'aggiunta, all'interno del modello, della dimensione immagine. La fase finale corrisponde all'aggiunta del Graph DB e dello strumento delle *Reccomendation*, migliorando il sistema di matching.

4.2 Text Matching

Il caso di studio per questa tesi parte da un progetto già consolidato. Per questo motivo, facendo riferimento al modello CRISP, è stata effettuata una nuova fase di valutazione. In quest'ottica sono stati individuati i casi in cui le tecniche di *Text Matching* risultavano poco efficaci. In particolare le casistiche più diffuse per cui questa tecnica è inefficace sono due.

Modelli differenti con stesso Brand, Categoria, Colore, Materiale La tecnica di *Text Clustering*, discussa in precedenza, prevede di utilizzare le caratteristiche dei prodotti estratte dal testo per suddividerli in gruppi simili. In particolare i gruppi hanno in comune lo stesso Brand, Categoria, Colore e Materiale. Un primo caso di insuccesso si presenta nel momento in cui siano messi in vendita on-line diversi modelli di uno stesso Brand appartenenti alla stessa Categoria ed aventi stesso Colore e Materiale. In questo frangente l'algoritmo prevede di inserire i prodotti all'interno dello stesso cluster considerandoli simili. Come si può notare dalla Figura 4.2 dalle

immagini del prodotto si riesce a differenziare i vari modelli, mentre, tenendo in considerazione le sole caratteristiche chiave, sembrerebbero lo stesso prodotto.

CLUSTER_ID	PRODUCT_COD		BRAND	COLOUR	CATEGORY	MATERIAL	STORENAME
21256	07c34d252df7eaf9f3b8de7c31f5d045		Brand1	black	boot	stretch	Site1
	238b2ea6e5faa528b599b256eabfcc72		Brand1	black	boot	stretch	Site1
	2ef60c22d59701d46fa170ee536cde71		Brand1	black	boot	stretch	Site1
	45555c35249e37632366e207414a23f0		Brand1	black	boot	stretch	Site1
	5299205196eaec846d25652d9ae76769		Brand1	black	boot	stretch	Site1

Figura 4.2. Esempio di modelli differenti con stesso Brand, Categoria, Colore e Materiale.

Dati chiave mancanti Un'altra situazione per cui il *Text Matching* non riesce ad essere performante si presenta quando la piattaforma E-commerce non contiene tutte le caratteristiche chiave del clustering. In questo caso non è possibile riuscire a ricondursi al cluster corretto. Come si può notare dalla Figura 4.3 nel caso una delle caratteristiche chiave non sia presente, il sistema imposta un valore di default, TBD (*To Be Defined*), questo non permette la collocazione all'interno dello stesso gruppo.
A partire dai casi individuati si è affrontata nuovamente la fase di *Business Understanding*) riformulando il problema, aggiungendo nuove complessità e cercando di rendere più efficace il sistema di matching.

CLUSTER_ID	PRODUCT_COD		BRAND	COLOUR	CATEGORY	MATERIAL	STORENAME
22246	a8cac4e40255443e22cf91890ba15d03		Brand1	white	boot	leather	Site1
23038	c1e9d6587efbf9e2d3976b563de2dd0c		Brand1	TBD	boot	leather	Site2

Figura 4.3. Esempio di dati con chiavi del cluster mancanti.

4.3 Image Matching

Uno dei punti deboli della tecnica precedente è la debolezza della caratteristica Categoria. Per riuscire a migliorare questo aspetto si è pensato di utilizzare un ulteriore metodo di confronto, ovvero l'immagine presente sulla piattaforma. Questo approccio permette di legare due prodotti tramite le caratteristiche peculiari dell'oggetto e non solo tramite caratteristiche estratte dal testo. Questo tipo di considerazione è stato affrontato all'interno di una nuova fase di *Business Understanding* che ha permesso di rielaborare il progetto aggiungendo un modulo di elaborazione delle immagini e l'aggiunta all'interno del modello dati di una nuova dimensione: l'immagine. In seguito sono state affrontate nuovamente le fasi di Data Understanding e Data Preparation, che hanno permesso di capire come e quali informazioni era possibile estrarre dalle immagini e, quindi, come popolarne la dimensione. Una volta completate queste fasi ed aggiornato il modello si è passati alla fase di valutazione. Questo stadio ha permesso di apprendere delle nuove conoscenze. In particolare si è individuato un comportamento molto differente tra le due tecniche di matching. Mentre l'algoritmo basato sul testo considera diversi due prodotti a seconda delle varianti (Colore, Materiale) che vengono individuate nelle pagine di dettaglio,le immagini delle varianti estratte dalla piattaforma risultano invece perfettamente simili dall'Image Matching. Un'altra considerazione su questa casistiche è la possibilità che una piattaforma E-commerce abbia a disposizione una sola immagine per tutte le varianti; questa situazione mette ancora più in risalto la differenza di comportamento dei due metodi.

IMG_ID	CLUSTER_ID	PRODUCT_COD		BRAND	COLOUR	CATEGORY	MATERIAL	STORENAME
211	21412	4037139c0f904198c7b2540d24bcaafd		Brand2	beige	boot	stretch	Site1
		fb7325bd3f10df315455b0c0ebca68fa		Brand2	beige	boot	stretch	Site2
	22809	4037139c0f904198c7b2540d24bcaafd		Brand2	black	boot	stretch	Site1
		fb7325bd3f10df315455b0c0ebca68fa		Brand2	black	boot	stretch	Site2

Figura 4.4. Esempio di immagini simili ma colore, all'interno delle caratteristiche, differente.

Nella Figura 4.4 Si può notare che per un insieme di immagini simili (stesso IMG ID) corrispondono due cluster (stesso CLUSTER ID). I due cluster differiscono per il colore del prodotto, ma come si può notare dalla figura i due prodotti hanno immagini identiche.

Questa nuova fase di valutazione ha permesso di chiarire che le tecniche, utilizzate singolarmente, non sono sufficienti per stabilire con buona approssimazione se due prodotti siano simili. Per questo motivo si è deciso di passare ad una nuova fase di *Buisiness Understanding* in modo da realizzare un nuovo metodo che tenga conto delle informazioni di entrambe le tecniche.

4.4 Matching ibrido

A partire dai risultati ottenuti si è cercato di capire all'interno del data set utilizzato le percentuali di successo delle diverse tecniche. In particolare su un insieme di circa 600 prodotti gli elementi legati da una relazione di similarità per immagine sono circa 40, mentre gli elementi appartenenti allo stesso cluster testuale circa 260. Questo risultato ci può dare una prima indicazione di quanto sia più stretto il legame tramite immagine rispetto a quello tramite cluster. In base a queste conclusioni si è scelto di considerare valide entrambe le relazioni ma attribuendo un peso diverso. Per le proporzioni ricavate si è scelto di attribuire un peso pari a 2 per la relazione tramite cluster, a 8 per quella tramite immagine.

Per riuscire ad integrare i risultati si è scelto di creare un sistema di *Reccomendation*. Con questo metodo i prodotti risultano legati fra loro da un' insieme di relazioni aventi pesi differenti, e la somma dei pesi rappresenta la forza del legame. Per realizzare il sistema è stato utilizzato un Graph DB, questa nuova componente ha reso necessaria una nuova fase di *Data Preparation* per garantire che il formato dati tra le diverse componenti risultasse compatibile.
Ritornando agli esempi mostrati in precedenza è interessante capire come venga valutata la similarità attribuendo i pesi alle diverse relazioni.

Nella Figura 4.5 si può notare che per le cinque scarpe il peso maggiore dato all'immagine permetta di dare una raccomandazione di somiglianza del 20%. Questa percentuale permette di intuire che i prodotti hanno un certo grado di similarità, ed in effetti posseggono delle caratteristiche chiave in comune, ma sono comunque prodotti diversi.

RECCOMENDATION	IMG_ID	CLUSTER_ID		BRAND	COLOUR	CATEGORY	MATERIAL	STORENAME
20%	23104	21256		Brand1	black	boot	stretch	Site1
20%	22526			Brand1	black	boot	stretch	Site1
20%	22531			Brand1	black	boot	stretch	Site1
20%	23016			Brand1	black	boot	stretch	Site1
20%	22534			Brand1	black	boot	stretch	Site1

Figura 4.5. Esempio di prodotti differenti ma con caratteristiche simili.

Nel caso in cui ci siano dei dati chiave mancanti, e due prodotti vengano inseriti in cluster differenti, l'immagine riesce a dare un contributo fondamentale. Infatti nell'esempio in Figura 4.6 si può notare a livello percettivo come i due articoli abbiano la stessa immagine, mentre il colore in un caso è presente nell'altro è TBD, ovvero mancante. In questo caso la percentuale di raccomandazione è pari all' 80%. Questo valore rappresenta il fatto che i due prodotti siano simili, ma non propriamente identici. Infatti in base dati manca l'informazione che lega i colori del prodotto.

RECCOMENDATION	IMG_ID	CLUSTER_ID		BRAND	COLOUR	CATEGORY	MATERIAL	STORENAME
80%	22746	22246		Brand1	white	boot	leather	Site1
80%		23038		Brand1	TBD	boot	leather	Site2

Figura 4.6. Esempio di immagini simili ma con caratteristiche chiavi mancanti.

Come illustrato precedentemente i due algoritmi, basti su testo o immagine, si comportano in maniera differente nel caso di varianti di uno stesso prodotto. Il motore di raccomandazione, applicando un peso diverso a seconda del tipo di relazione, permette di tenere in maggiore considerazione la somiglianza dell'immagine. Se due prodotti però hanno caratteristiche chiave diverse (es. Colore o Materiale) non avranno una somiglianza del 100 %.

RECCOMENDATION	IMG_ID	CLUSTER_ID		BRAND	COLOUR	CATEGORY	MATERIAL	STORENAME
80%	211	21412		Brand2	beige	boot	stretch	Site2
80%		22809		Brand2	black	boot	stretch	Site2

Figura 4.7. Esempio di immagini simili ma colore, all'interno delle caratteristiche, differente.

Il caso di match perfetto, quindi una similarità del 100 %, si ottiene solo quando è presente un match con entrambi i metodi. In questo caso esiste una similarità, sia per quanto riguarda l'immagine, sia per le informazioni chiave estratte dalla piattaforma on-line, come si può notare in Figura 4.8.

RECCOMENDATION	IMG_ID	CLUSTER_ID		BRAND	COLOUR	CATEGORY	MATERIAL	STORENAME
100%	211	21412		Brand2	beige	boot	stretch	Site1
100%				Brand2	beige	boot	stretch	Site2

Figura 4.8. Esempio di immagini simili e stesse caratteristiche chiave.

Capitolo 5

Conclusioni e sviluppi futuri

Obbiettivi All'interno del team dell'azienda Mediamente Consulting ci si è focalizzati sull'integrazione di una nuova fonte dati, come le immagini digitali, all'interno di una piattaforma di Business Analytics.
Il sistema esistente cattura informazioni presenti su piattaforme on-line appartenenti alle scarpe di lusso e permette, tramite tecniche di *Text Mining*, di sfruttare il testo contenuto nelle pagine di dettaglio dei prodotti. In questo modo si può identificare lo stessa scarpa venduta su piattaforme differenti. Queste tecniche riescono ad identificare le caratteristiche chiave di un prodotto (Brand, Categoria, Colore e Materiale) e a creare dei cluster che contengono prodotti con stesse caratteristiche. Non sempre questo metodo risulta efficace in quanto le caratteristiche presenti nel testo di descrizione potrebbero non essere sempre complete. Inoltre alcune caratteristiche, come ad esempio la Categoria, potrebbero avere una descrizione differente sulle diverse piattaforme. Un ulteriore limite della tecnica riguarda le caratteristiche chiave scelte per il cluster; i gruppi hanno in comune lo stesso Brand, Categoria, Colore e Materiale, quindi l'algoritmo non prevede che uno stesso Brand possa avere prodotti differenti appartenenti alla stessa Categoria e aventi stesso Colore e Materiale. Per questi motivi si è scelto di usare anche le immagini presenti nelle pagine di dettaglio per riuscire a rendere più efficace il sistema di matching.

Sviluppo e risultati Per raggiungere questo obbiettivo il lavoro svolto si è suddiviso in due fasi. La prima, legata strettamente alle immagini digitali, ha permesso di sfruttare meccanismi di *Machine Perception*. Questi meccanismi permettono di catturare ed interpretare i dati in maniera simile al modo in cui l'uomo usa i propri sensi per relazionarsi con il mondo esterno. La seconda fase di sfruttare queste informazioni per migliorare le funzionalità della piattaforma.
La prima fase ha portato alla realizzazione di un modulo di elaborazione delle immagini in linguaggio Python che riesce a metterle in relazione in base al loro grado di similarità. La seconda fase ha permesso di integrare il modulo all'interno della

piattaforma esistente, estrarre i dati dalle immagini delle piattaforme E-commerce ed infine confrontare i risultati tra la tecnica di *Text Matching* rispetto a quella di *Image Matching* mettendo in risalto difetti e pregi dei diversi approcci. La tecnica basata sulle immagini riesce a compensare il caso di caratteristiche chiave mancanti nel testo di dettaglio del prodotto e i casi di prodotti diversi ma con stesse caratteristiche, in quanto si basa esclusivamente sulle caratteristiche visive dell'oggetto. Il punto critico è quello di riuscire ad individuare le varianti dei prodotti. Infatti, all'interno di una pagina di dettaglio, sono presenti, sotto forma di informazioni testuali, anche le possibili varianti di un articolo (Colore, Materiale). L'immagine, nella maggior parte dei casi, è associata ad una sola variante del prodotto. In questi casi per la tecnica di *Image Matching* il prodotto è lo stesso, mentre dal testo si riescono a distinguere le diverse varianti del prodotto.

L'interpretazione dei risultati ottenuti ha permesso di concepire l'idea di un sistema di *Reccomendation*. Questo sistema, sfruttando il modello del Graph DB fondato su nodi e relazioni, permette di fornire una percentuale di similarità basandosi sulla forza del legame fra due prodotti. Il legame si costruisce dando un peso alle relazioni legate all'immagine e al testo, presenti nel dettaglio delle piattaforme di vendita on-line. Quanto più il legame è forte tanto più è probabile che i prodotti siano simili.

Limitazioni e sviluppi futuri Il sistema creato ha portato ad un netto miglioramento del sistema di identificazione dei prodotti on-line rispetto al solo *Text Matching* Infatti tramite il sistema di Reccomendation si riescono a sfruttare i punti di forza di entrambi i metodi di matching. Per completare e rendere più efficace il lavoro svolto sarebbe interessante rimuovere il limite della singola categoria del prodotto. Infatti per questo caso di studio ci si è focalizzati su un'unica categoria di scarpe (Stivali). Effettuare uno studio sulle altre categorie presenti amplierebbe le capacità della piattaforma e permetterebbe di avere un sistema più completo. Una sfida ancora più interessante sarebbe riuscire ad avere un sistema che possa generalizzare i prodotti e permettere l'identificazione di diverse tipologie di articoli. Per entrambe le evoluzioni andrebbe sviluppato un sistema di elaborazione delle immagini che non solo effettui un confronto ma riesca a compiere una vera propria Object Recognition. Per arrivare a questo traguardo è necessaria una nuova fase di studio che applichi tecniche di Machine Learning e Data Mining alle immagini per riuscire a ricavare delle features che permettano di categorizzarle ed effettuare il riconoscimento.

Un altro possibile sviluppo sarebbe il raffinamento del sistema di Reccomendation. Questo sarebbe possibile aumentando le tipologie delle relazioni. Allo stato attuale il legame ottenuto dal testo rappresenta il cluster intero; invece potrebbe essere generata una relazione per ogni singola caratteristica. Inoltre per rendere ancora

più preciso il sistema si potrebbero aggiungere delle nuove informazioni specifiche del prodotto come ad esempio l' altezza del tacco, il gender, la stagione di vendita, ecc. Questo potrebbe aprire a nuovi tipi di analisi andando a modificare, a seconda delle esigenze, il concetto di similarità. Ad esempio potrebbe essere interessante confrontare i prodotti simili ma appartenenti a Brand diversi per analizzare i prodotti potenzialmente concorrenti. In questo caso la similarità sarebbe data dalle caratteristiche appartenenti all'aspetto del prodotto escludendo il Brand.

Infine sarebbe una sfida riuscire ad integrare nuovi tipi di fonte dati come ad esempio i social network, i commenti degli utenti o le recensioni. In questo modo sarebbe possibile effettuare nuovi tipi di analisi basate sul Sentiment estratto dal testo in modo da riuscire ad associare una valutazione degli utenti sui prodotti. Con queste informazioni sarebbe possibile incrociare i risultati del price monitoring con la valutazione che gli utenti finali hanno sui prodotti e ricavare informazioni potenzialmente strategiche.

Ringraziamenti

Solo quando si arriva alla fine di un certo percorso, voltandosi, ci si rende conto della mole di lavoro e della fatica serviti a raggiungere il traguardo. Ripensando a questi momenti sono sicuro che con le mie sole forze non sarei mai stato in grado di ottenere la Laurea Magistrale. Per questo motivo ci tengo particolarmente a ringraziare tutte quelle persone che mi hanno aiutato in questi anni anche solo con un consiglio od una parola di conforto.
Vorrei prima di tutto ringraziare la mia relatrice, la Prof.sa Tania Cerquitelli, oltre che per l'aiuto fornitomi anche per il supporto e la disponibilità dimostratami durante il periodo di stesura ed elaborazione di questa Tesi.
Ringrazio Mediamente Consulting, un realtà lavorativa in cui sono immerso da ormai due anni e mezzo, che mi sta permettendo di avere una crescita sia professionale che personale. I miei tutor, Alberto e Vincenzo, per le conoscenze che hanno condiviso con me oltre che alla disponibilità e il sostegno durante tutto il periodo di lavoro. Il Presidente, Antonio, che ha sempre avuto fiducia nelle mie capacità. I colleghi, presenti e passati, sempre disponibili al confronto e a spunti di riflessione, ognuno di loro ha una parte di merito per questo risultato.
Sono grato alla mia famiglia; i nonni, gli zii, i cugini, i fratelli, i genitori, che mi hanno sostenuto durante tutti i miei studi sia affettivamente che economicamente.
I compagni dell'Università, con cui ho passato le giornate diviso tra lezioni ed aule studio, che sono riusciti a rendere meno difficile questo percorso.
Gli amici, sempre disponibili a confortarmi e sostenermi nei momenti di difficoltà e a gioire con me nei momenti di felicità.
Chiara, la mia fidanzata, che mi ha spronato e motivato a concludere questo percorso ogni giorno. Inizialmente si è incuriosita per l'oggetto della tesi, le scarpe di lusso, ma credo di essere riuscito a trasmetterle anche qualche nozione di informatica.
Infine mia Madre. Credo che lei sia la persona a cui devo di più per i miei risultati e con il raggiungimento di questo traguardo spero di essere riuscito in parte a ripagarla per quanto lei mi ha dato.

Grazie ancora a tutti,
Filippo

Bibliografia

[1] Wikipedia. Evoluzione tecnologica — wikipedia, l'enciclopedia libera, 2017. [Online; in data 17-giugno-2018].

[2] M. Bloomfield. *The Automated Society: What the Future Will be and how We Will Get it that Way*. Masefield Books, 1995.

[3] M. Bloomfield. *Mankind in Transition: A View of the Distant Past, the Present, and the Far Future*. Masefield Books, 1993.

[4] L. Novỳ, J. Gabriel, and J. Hroch. *Czech Philosophy in the XXth Century*. Cultural Heritage and Contemporary Change Series IVA. Paideia Press & The Council for Research in Values and Philosophy, 1994.

[5] Wikipedia. Primavera di praga — wikipedia, l'enciclopedia libera, 2018. [Online; in data 17-giugno-2018].

[6] Margaret Rouse. What is advanced analytics? - techtarget, Dicembre 2017. `https://searchbusinessanalytics.techtarget.com/definition/advanced-analytics`.

[7] A. Greco. E-commerce monitoring solution for product allocation and marketing planning forecasting. Master's thesis, Politecnico di Torino, 2018.

[8] The lenna story, Marzo 2001. url: `http://www.charlesrosenberg.com`.

[9] Wikipedia. Hdtv — wikipedia, l'enciclopedia libera, 2018. [Online; in data 22-luglio-2018].

[10] Wikipedia. Televisione a ultra alta definizione — wikipedia, l'enciclopedia libera, 2018. [Online; in data 22-luglio-2018].

[11] Wikipedia. Sdtv — wikipedia, l'enciclopedia libera, 2018. [Online; in data 22-luglio-2018].

[12] Enrico Masala. Dispense corso di elaborazione e trasmissione di informazioni multimediali - politecnico di torino, 2016.

[13] Wikipedia. Cellula cono — wikipedia, l'enciclopedia libera, 2017. [Online; in data 21-agosto-2018].

[14] Gabriele Danesi. Modello rgb e sintesi additiva dei colori, Ottobre 2012. `http://www.gabrieledanesi.com/blog/index.php?modello-rgb-e-sintesi-additiva-dei-colori`.

[15] Scott D. Anderson. Cs307: Computer graphics - shadows and anti-aliasing, 2017.

[16] Belongie and Malik. Matching with shape contexts. *IEEE*, pages 20–26, June 2000.

[17] Andrey Nikishaev. Shape context, Marzo 2018.

[18] Christopher M Bishop. *Pattern recognition and machine learning, 2006*, volume 60. Springer-Verlag New York, 2012.

[19] Jesse Davis West. A brief history of face recognition, Agosto 2017.

[20] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Pearson/Prentice Hall, 2008.

[21] Gaurav Sharma, Wencheng Wu, and Edul N. Dalal. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research & Application*, 30(1):21–30, 2004.

[22] *Scrapy documentation*. url: `https://doc.scrapy.org/en/latest/topics/architecture.html`.

[23] Shaeela Ayesha Ramzan Talib, Muhammad Kashif Hanif and Fakeeha Fatima. Text mining: Techniques, applications and issues. 2016.

[24] Ms. Nithya Dr. S. Vijayarani, Ms. J. Ilamathi. Preprocessing techniques for text mining - an overview. 2016.

[25] Tania Cerquitelli Elena Baralis. Clustering fundamentals.

[26] Wikimedia Commons. File:crisp-dm process diagram.png — wikimedia commons, the free media repository, 2017. [Online; accessed 22-September-2018].

[27] R. Szeliski. *Computer Vision: Algorithms and Applications*. Texts in Computer Science. Springer London, 2010.

[28] F. Provost and T. Fawcett. *Data Science for Business*. O'Reilly, 2013.