



Università degli Studi di Milano Bicocca

**Dipartimento di Informatica, Sistemistica e Comunicazione**

**Corso di Laurea Magistrale in Informatica**

# **Emotion recognition with neural network attention based model**

**Relatore:** Prof. Elisabetta Fersini

**Co-relatore:** Prof. Francesca Gasparini

**Tesi di Laurea Magistrale di:**

*Elia Fabbris*

*Matricola 793240*

**Anno Accademico 2019-2020**



# Ringraziamenti

Prima di procedere con la presentazione del mio lavoro di tesi, vorrei dedicare qualche riga a tutti coloro che mi sono stati vicini in questo percorso.

Un grande ringraziamento va alla mia famiglia: Fiorenzo, Daniela e Veronica. Grazie a loro mi è stato possibile intraprendere questo percorso, dandomi la possibilità di crescere personalmente e professionalmente.

Un gigantesco grazie va a Sara. In questi anni in cui abbiamo condiviso questo percorso, attraverso la sensibilità e la dolcezza che la contraddistinguono, è stata capace di motivarmi facendomi superare le difficoltà incontrate e gioire insieme a me dei traguardi raggiunti.

Un doveroso grazie va alla mia relatrice, la professoressa Fersini, la quale è stata capace di supportarmi e sopportarmi in questo anno di lavoro. La ringrazio per essersi sempre dimostrata disponibile e capace di guidarmi durante lo sviluppo di questa tesi.

Un grazie va anche alla mia correlatrice, la professoressa Gasparini, la quale si è dimostrata sempre gentile e disponibile per ogni chiarimento.

Un grazie va agli amici di una vita: Brove, Ferrario, Silvia e Virgi. Agli amici non si chiede tanto, solamente la possibilità di passare momenti di spensieratezza. Voi ci siete riusciti.

Un grazie va a Matteo, l'unica persona che dal primo giorno della triennale all'ultimo della magistrale mi sia stata vicina, condividendo ogni giorno in università con me. Grazie a te ho capito che in università non si trovano solamente colleghi ma, in rari casi, amici.

Un ultimo grazie va a tutte le persone che ho incontrato e che mi sono state vicine durante questo percorso.

Grazie a tutti!

# Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Stato dell'arte</b>	<b>5</b>
2.1	Introduzione all'affective computing . . . . .	6
2.2	Representation . . . . .	10
2.3	Fusion . . . . .	12
2.4	Translation . . . . .	15
<b>3</b>	<b>Metodo proposto</b>	<b>17</b>
3.1	Meccanismo di attenzione semantico . . . . .	21
3.2	Meccanismo di attenzione visuale . . . . .	25
3.3	Sentiment classification . . . . .	28
3.4	Emotion classification . . . . .	29
<b>4</b>	<b>Esperimenti</b>	<b>30</b>
4.1	Sentiment classification . . . . .	30
4.1.1	Dataset . . . . .	30
4.1.2	Baseline . . . . .	32
4.1.3	Risultati parte testuale . . . . .	33
4.1.4	Risultati parte visuale . . . . .	36
4.1.5	Risultati late fusion . . . . .	39
4.1.6	Confronto risultati con baseline . . . . .	44
4.1.7	Analisi dell'errore . . . . .	45
4.2	Emotion classification . . . . .	48
4.2.1	Dataset . . . . .	48
4.2.2	Baseline . . . . .	50
4.2.3	Risultati parte testuale . . . . .	51
4.2.4	Risultati parte visuale . . . . .	53
4.2.5	Risultati late fusion . . . . .	58
4.2.6	Confronto risultati con baseline . . . . .	66
4.2.7	Analisi dell'errore . . . . .	67
<b>5</b>	<b>Conclusioni</b>	<b>70</b>
<b>6</b>	<b>Sviluppi futuri</b>	<b>71</b>

# 1 Introduzione

Nell'ambito del machine learning, il riconoscimento del testo e delle immagini, occupa una grande parte degli studi presenti in letteratura.

Con la crescita dei social network, come Twitter, Facebook e Instagram, la disponibilità di dati contenenti sia immagine che testo è diventata molto alta. Secondo una ricerca condotta da [Statusbrew](#), nel 2019 ci sono stati mediamente 330 milioni di utenti attivi al mese e quotidianamente 500 milioni di tweets. Questi dati possono far riflettere e poterli analizzare risulterebbe importante per molti scopi.

Effettuare una sentiment analysis su questa grossa quantità di dati potrebbe rendere ancora più comprensibili e rilevanti le opinioni di un utente.

Analizzando le opinioni dei clienti per esempio, un'azienda potrebbe raccogliere dei feedback per capire se un suo prodotto o un suo servizio stiano andando bene o male; viceversa, sarebbe possibile desumere la posizione presa da una certa persona nei confronti di un avvenimento reale riguardo al quale si sta discutendo sulle piattaforme social.

In precedenza molte analisi sono state fatte prendendo in considerazione solo testo o solo immagini. Prendendo in considerazione la coppia si valuta l'influenza dell'immagine sul testo e viceversa.

In questo modo è possibile trovare features correlate o elementi discriminanti tra testo e immagini, aiutando in casi di incertezza in cui sarebbe difficile dare una connotazione piuttosto che un'altra.

In alcuni casi, recuperando delle immagini dal web attraverso l'utilizzo dei tags, è possibile trovare dei dati che sono accomunati da una certa descrizione ma apparentemente sono molto diversi.

In questo esempio sono state raccolte delle immagini sulla base delle emozioni: entrambe sono state recuperate utilizzando il tag *amusement*.



Figura 1: Immagini *amusement*

Entrambe le immagini corrispondono allo stesso tag ma sono chiaramente in contrapposizione: la prima foto contiene colori scuri e poca luce, la seconda invece contiene colori molto accesi e rappresenta una situazione ben più chiara e associabile al tag *amusement* dato che contiene una giostra. Questo è un chiaro esempio in cui l'utilizzo e l'analisi del testo potrebbe risolvere l'incertezza sorta dalla sola analisi delle immagini. Per cercare di aumentare la potenza del modello proposto, è stato introdotto anche un concetto di attenzione. Con quest'ultimo l'analisi non si limita solamente al dato in input, ma si cerca di concentrare l'apprendimento sulle regioni e sulle parole più rilevanti, mettendo in secondo piano quelle meno importanti.

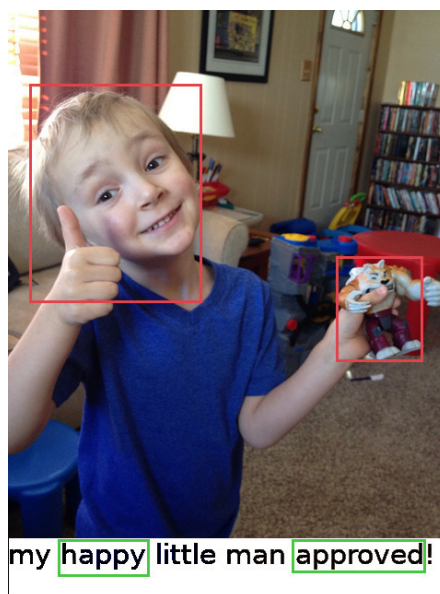


Figura 2: Esempio di attenzione

In questo esempio vengono messe in risalto regioni dell'immagine e parole del testo importanti per l'analisi. Il sorriso del bambino e il giocattolo (box rossi) sono regioni dell'immagine che ritraggono un momento felice, le parole "happy" e "approved" (box verdi) hanno una connotazione positiva. Come si nota dall'esempio, l'attenzione non deve essere incentrata solamente su elementi condivisi da immagine e testo ma può anche fornire informazioni complementari.

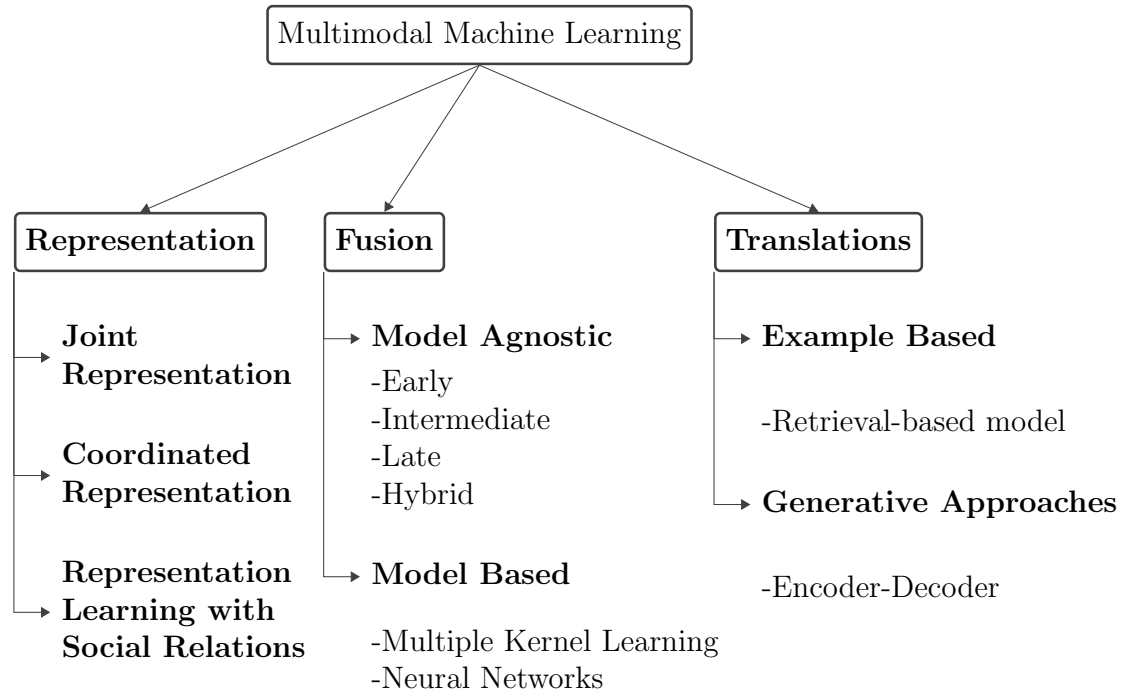
Quindi per entrare più nello specifico, in questa tesi vengono proposti due tipi di studi. Il primo si concentra sulla sentiment analysis, quindi lo studio della polarità dei dati, ai quali può essere associata una label tra *negative* e *positive*.

Il secondo studio verte sull'emotion analysis in cui le label sono otto: *amusement*, *contentment*, *awe*, *excitement*, *sadness*, *disgust*, *anger* e *fear*.

A livello di architettura vengono proposte due pipeline unimodali, una per l'analisi del testo e una per l'analisi dell'immagine.

Una volta ottenuti i risultati provenienti dalle due pipeline, questi vengono fusi tramite late fusion così da restituire un risultato singolo per ogni coppia immagine-testo. Tutte le pipeline saranno corredate di modulo di attenzione. Nel corso dei capitoli vengono mostrati e spiegati i due diversi meccanismi di attention utilizzati durante l'analisi testuale e visuale.

Successivamente alla presentazione dei risultati, viene mostrata anche un'analisi degli errori sui dati testuali, in modo da facilitare sviluppi futuri mettendo in mostra i punti deboli dei modelli e, magari, rilevare correlazioni tra gli errori.



## 2 Stato dell'arte

Prendendo in considerazione i modelli di apprendimento automatico, la rappresentazione con cui i dati devono essere forniti per rendere il modello funzionante ed efficiente risulta essere sempre una sfida non banale.

Con il crescere e l'aumentare di piattaforme social network, il concetto di dati multimodali è divenuto sempre più importante.

La rappresentazione multimodale non è altro che una modalità in cui i dati vengono presentati utilizzando informazioni provenienti da più sorgenti e di diverso tipo.

Infatti, utilizzando come esempio i social network (e.g., Twitter, Instagram, Flickr), i dati che vengono creati e scambiati su queste piattaforme incarnano perfettamente il concetto di dati multimodali: in queste reti sociali è possibile trovare informazioni sotto forma di testo oppure sotto forma di immagini. Queste sono due tra le informazioni più comuni, ma non possono essere escluse da questo discorso informazioni rappresentate sotto forma di audio o video.



Quindi la rappresentazione multimodale risulta essere uno strumento potente, la quale offre la possibilità di valutare diverse sorgenti di informazioni migliorando l'accuratezza di un'eventuale analisi su di esse. Dietro questa possibilità, però si celano alcune difficoltà, tra queste ci sono: la modalità in cui i dati provenienti da diverse sorgenti possono essere combinati, come si può gestire l'eventualità di dati mancanti o, semplicemente, il modo in cui rappresentare le informazioni in maniera sensata e significativa. Basandosi su un metodo d'approccio robusto, il quale non risulti influenzabile dalle difficoltà sopracitate, si possono ottenere ottime performance.

Infatti un lavoro non triviale, che è la base di questa tesi, risulta essere la sentiment analysis multimodale, che negli ultimi anni ha ottenuto molta attenzione.

Nei seguenti sottocapitoli vengono presentati vari approcci presenti in letteratura. In questi elaborati sono contenuti diversi metodi con cui è possibile affrontare problemi di rappresentazione dei dati, problemi di fusione dei dati ed, infine, la traduzione di un'informazione proveniente da una sorgente in una informazione mancante di una sorgente diversa.

## **2.1 Introduzione all'affective computing**

Da un punto di vista preliminare è giusto fare qualche cenno introduttivo all'affective computing.

L'affective computing è un campo molto in voga in termini di ricerca scientifica: il suo obiettivo è quello di sviluppare sistemi intelligenti con i quali risulti possibile capire, riconoscere, prevedere ed interpretare le emozioni e i sentimenti degli umani. Queste componenti sono molto importanti: le persone basano i rapporti umani, le decisioni quotidiane e l'apprendimento su di essi.

Come anticipato precedentemente, con la crescita dei social media, l'analisi delle emozioni e del sentimento è diventata molto utile, permettendo ai ricercatori di capire e comprendere le opinioni degli utenti attivi su diverse piattaforme.

Queste analisi vengono rese fattibili e rilevanti grazie proprio allo sviluppo tecnologico: con la crescita degli smartphone, sono cresciuti i social media ma soprattutto è diventato sempre più facile e veloce creare un qualsiasi tipo di contenuto. In poco tempo si è passati dalla presenza di contenuti prettamente testuali alla presenza di contenuti sotto forma di video, audio e foto.

Il tipo di analisi che può essere fatta sull'enorme mole di dati che viene prodotta quotidianamente, non si limita alle parole presenti in un testo ma può comprendere anche espressioni comportamentali.

Un testo può essere creato avendo prima una buona fase di pensiero e di stesura, diversamente in un video si possono notare alcune particolarità, come un cambiamento della voce lieve durante un discorso oppure un'espressione facciale, rendendo così molto più potente e molto più sofisticato lo studio delle emozioni. In tal modo sarebbe possibile identificare un atteggiamento incondizionato che solamente dall'analisi testuale non sarebbe possibile percepire.

Per questo motivo viene introdotto il concetto di analisi multimodale: le sorgenti da cui arrivano le informazioni non vengono analizzate in maniera separata, bensì combinata. Mantenere l'analisi della parte testuale con l'aggiunta di un'analisi della parte visuale renderebbe uno studio molto più preciso nell'identificare il vero stato emotivo di un utente.

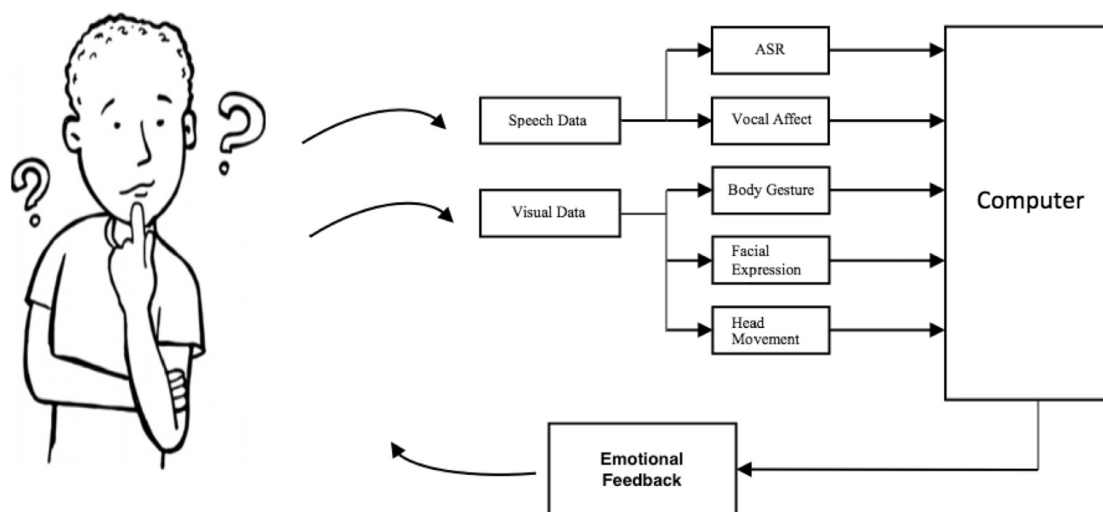


Figura 3: Emotion analysis framework

Infatti la ricerca nel campo dell'affective computing attira sempre più l'attenzione di ricercatori e di aziende. Ciò, legato ai progressi che si stanno ottenendo nel campo dell'elaborazione dei segnali e nel campo dell'intelligenza artificiale, ha contribuito allo sviluppo di sistemi estremamente intelligenti che rilevano ed elaborano le informazioni affettive presenti nelle sorgenti di

dati multimodali.

Molti elaborati presenti in letteratura basano ancora lo studio sull'elaborazione di una modalità singola, ovvero audio, testo o video. Ciò porta questi sistemi a delle limitazioni sotto molti aspetti come, ad esempio, robustezza, accuracy e performance generali, inficiando notevolmente l'applicazione di tali sistemi nel mondo reale.

Infatti l'obiettivo di un utilizzo multimodale dei dati è quello di migliorare l'affidabilità e l'accuracy di questi sistemi durante le predizioni.

Già molti studi e altrettante applicazioni hanno dimostrato il miglioramento e le potenzialità dell'utilizzo di dati provenienti da più sorgenti.

Per questo motivo iniziano ad essere sviluppati framework multimodali, i quali analizzano due o tre sorgenti contemporaneamente: testo, video e/o audio.

Queste sorgenti sono la base della comunicazione tra gli esseri umani, analizzarle contemporaneamente consente di estrarre informazioni semantiche ed affettive, che da un'analisi unimodale non sarebbero minimamente percepibili. Infatti, da alcune ricerche effettuate e presenti in letteratura, l'85% dei sistemi che hanno utilizzato un tipo di sorgente multimodale risultano essere più accurati rispetto ai migliori sistemi unimodali corrispondenti. Il miglioramento medio raggiunto è pari a 9.83%, con una mediana pari a 6.60%.

Avendo introdotto queste nozioni riguardanti i framework multimodali, è giusto spostare l'attenzione e spiegare cos'è l'affective computing.

L'affective computing è un'insieme di tecniche, sviluppate e perfezionate negli anni, che mirano al riconoscimento di espressioni affettive analizzando i dati in diverse condizioni di modalità e di granularità.

Per entrare più nello specifico e dare una visione più chiara di ciò che comprende l'affective computing, bisogna differenziare e presentare alcune analisi che fanno parte di questo tipo di studio.

Come primo esempio può essere presentata la sentiment analysis: in questo tipo di analisi vengono studiati e riconosciuti gli affetti, applicando una differenza netta. Tendenzialmente questo studio viene considerato come task di classificazione binaria, limitando le predizioni sui dati con polarità positiva o polarità negativa.

Un secondo esempio che può essere presentato è l'emotion recognition: in questo tipo di analisi, diversamente dall'analisi mostrata precedentemente, viene effettuato il riconoscimento sulle emozioni in maniera più fine. Non viene più considerato come task di classificazione binaria, bensì la classificazione si basa su un set di label maggiore, il quale rappresenta le emozioni. Un set di label che può essere utilizzato per questo tipo di task è quello creato negli anni '70 dallo psicologo Ekman, contenente le sei emozioni di base che condividono gli umani: felicità, tristezza, paura, rabbia, disgusto e sorpresa.

Con questo capitolo si è voluto quindi introdurre l'affective computing, spiegando e sottolineando l'importanza dell'utilizzo di dati provenienti da fonti multimodali e successivamente presentare un paio di task molto in voga in questi ultimi anni.

Bisogna però sottolineare che gli studi di affect recognition tramite l'utilizzo di dati unimodali rimangono molto importanti: rappresentano un componente fondamentale per il raggiungimento di ottime performance attraverso l'utilizzo di sorgenti multimodali. Senza modelli unimodali efficienti sarebbe impossibile costruire un modello multimodale ben performante.

Inoltre l'analisi della parte testuale risulta essere un buon punto di partenza, affidabile, dimostrandosi molto utile per migliorare nettamente le performance dei modelli di riconoscimento audio e video.

## 2.2 Representation

In questo gruppo vengono elencati dei lavori presenti in letteratura, mostrando il tipo di rappresentazione utilizzata.

Le metodi presentati sono i seguenti: *joint representation*, *coordinated representation* e *representation learning with social relations*.

Come presentato negli approcci di You et al.[29], Baltrusaitis et al.[30], Cao et al.[31], Ngiam et al.[32], Rajagopalan et al.[33], Silberer et al.[34], Srivastava et al.[35], Wang et al.[36], Masci et al.[37], Suk et al.[38], Huang et al.[39], viene utilizzata la *joint representation* effettuando modifiche dettate dallo scenario di utilizzo. Questo tipo di rappresentazione è molto diffusa.

Nei vari modelli proposti i dati vengono raccolti a coppie immagine-testo. Inizialmente vengono gestiti e processati in maniera unimodale e successivamente uniti.

Lo scopo di questa rappresentazione è quello di valutare distintamente i dati raccolti, ma successivamente presentare i risultati in maniera unica, nello stesso spazio di rappresentazione.

Tramite la *coordinated representation*, invece, le pipeline vengono mantenute distinte. Diversamente dalla tecnica presentata precedentemente, le pipeline unimodali rimangono tali e vengono valutate separatamente per tutta l'analisi dei dati. L'unica particolarità consiste nel fatto che vengono imposti alcuni vincoli, ad esempio sulla similarità tra i dati delle diverse pipeline, in modo tale sia possibile trasporarli nello spazio coordinato.

Alcuni esempi di *coordinated representation* sono proposti negli approcci presentati da Frome et al.[40], Kiros et al.[41], Zhang et al.[42], Hardoon et al.[44], Andrew et al.[45], Yan et al.[46], Weston et al.[47], Wang et al.[48]; più particolare è il metodo utilizzato da Peng et al.[43] e da Vendrov et al.[49], nei quali è stato utilizzato un approccio gerarchico.

Una terza modalità di rappresentazione, più marginale, sfrutta la struttura delle *social relations*. Le piattaforme come Twitter, Facebook e simili, si basano sulle *social relations* e potrebbe risultare utile integrare un'ulteriore sorgente di dati oltre alle immagini e al testo.

Alcuni esempi di modelli di questo tipo vengono presentati da Fang et al.[23], Chang et al.[26], Liu et al.[27]; Perozzi et al.[24] presenta il metodo DeepWalk, invece il metodo LINE è presentato da Tang et al.[25]. Zhang et al.[28] incorporano oltre alle *social relations* anche i tags, inoltre sfruttano un'ar-

chitettura gerarchica per apprendere sia features visuali di basso livello, sia features di alto livello per immagini e testo.

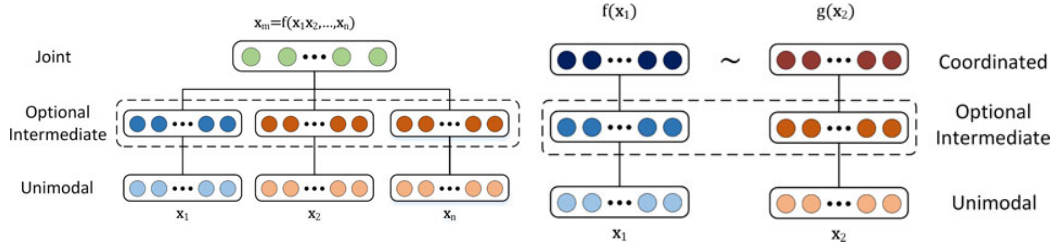


Figura 4: Joint repr.

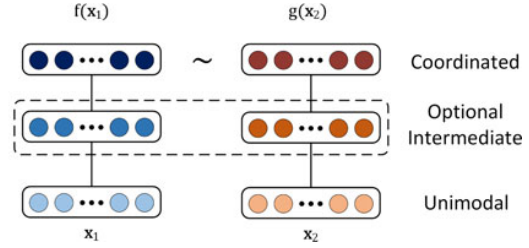


Figura 5: Coordinated repr.



Figura 6: Repr. learning with Social Relations

## 2.3 Fusion

In questo secondo gruppo vengono presentati i diversi tipi di fusion presenti in letteratura.

Essi si dividono in *model agnostic* e *model based*.

Nel primo tipo di fusion, *model agnostic*, sono contenute le tecniche più diffuse: nei lavori presentati da Pérez-Rosas et al.[15], da Poria et al.[16], da Simonyan et al.[17], da Chen et al.[18], da Wöllmer et al.[19], da Poria et al.[20] viene utilizzata la *early fusion*.

Questo tipo di fusione, una volta processati i dati e ottenute le features unimodali, unisce quest'ultime in un'unica rappresentazione multimodale. Successivamente, le features unite, vengono usate come input per il modello di apprendimento scelto.

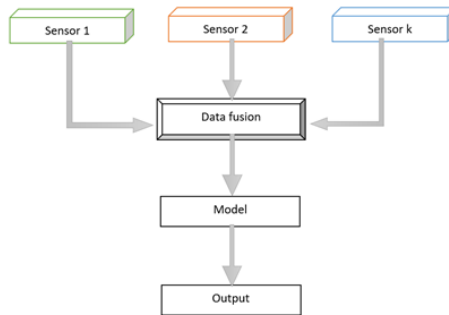


Figura 7: Meccanismo early fusion

Gli elaborati presentati da Wöllmer et al.[12], da Cao et al.[13], da Poria et al.[14], da Fan et al.[11], da Yao et al.[8], da Vielzeuf et al.[9], da Poria et al.[20], da Castellano et al.[7], da Chen et al.[18], da Ramirez et al.[6], invece, utilizzano la *late fusion*.

Come nei meccanismi early fusion, le features vengono estratte dalle pipeline unimodali. Successivamente però non vengono unite in una rappresentazione multimodale e date in input al modello di apprendimento, bensì vengono usate come input per modelli di apprendimento unimodali e, solamente in seguito, vengono unite.

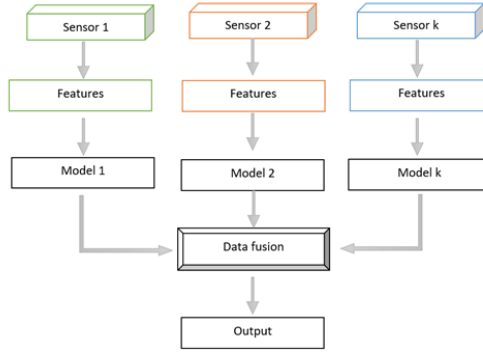


Figura 8: Meccanismo late fusion

Oltre ai due meccanismi più diffusi, vengono presentati altri due tipi di fusion: il primo viene presentato negli elaborati di Chen e al.[21], You e al.[22] e You et al.[29]. Questo tipo viene chiamato *intermediate fusion* e risulta essere un metodo flessibile: permette la fusione delle features a diversi step dell'addestramento del modello. Vengono quindi rese possibili diverse fusioni in diverse profondità della rete.

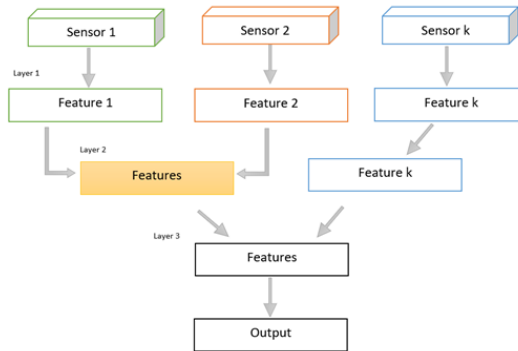


Figura 9: Meccanismo intermediate fusion

La seconda opzione viene rappresentata dai meccanismi di *hybrid fusion* e sono presentati da Yan et al.[10] e da Lan et al.[5]. In questi elaborati vengono uniti diversi tipi di fusione, dando così vita a meccanismi ibridi.



Il secondo gruppo contenente altri metodi di fusion si chiama *model based*. Gli approcci presentati da Poria et al.[16], da Bucak et al.[4], da Jaques et al.[3], da Kahou et al.[2], da Sikka et al.[1] uniscono le features tramite il *multiple kernel learning*. Questo meccanismo permette di non effettuare nessun tipo di trasformazione a livello delle features: riesce a gestire diversi tipi di rappresentazione. Ogni kernel viene calcolato su una feature individuale, ma tramite l'utilizzo di multiple kernel learning risulta possibile combinare diversi tipi di kernel senza effettuare trasformazioni.

Nei lavori presentati da Kahou et al.[2] e da Masci et al.[37] vengono utilizzate le *neural networks* come meccanismi di fusione. Tramite questa architettura risulta possibile processare parallelamente più reti neurali e successivamente unire insieme le metriche di tutte le reti per generare un risultato finale.

Inoltre, applicando tecniche deep alle reti neurali, si ottiene il grosso vantaggio di riuscire a gestire e apprendere da grandi quantità di dati; al contempo, queste tecniche, necessitano di lunghe fasi di addestramento per raggiungere delle buone performance.

## 2.4 Translation

Infine, nell'ultimo gruppo, sono contenuti i metodi con cui si ottengono informazioni mancanti nel dataset o ne viene migliorata la qualità.

Nei modelli *retrieval-based* è presente un dizionario con cui è possibile ottenere informazioni in base all'input: è possibile ricavare la parte testuale dando in input immagini d'esempio e viceversa. In generale fornendo un dato, viene restituito un oggetto il più simile possibile presente nel dizionario, utilizzando quest'ultimo come traduzione del dato in input.

In alcuni approcci viene utilizzato solamente un tipo di retrieval con il quale è possibile ottenere delle immagini fornendo in input del testo; approcci di questo tipo sono contenuti in Chang et al.[26] Altri approcci invece propongono sia il retrieval text-to-image sia image-to-text.

Vari tipi di cross-media retrieval vengono proposti negli approcci di Masci et al.[37], di Wang et al.[48], di Zhang et al.[28], di Huang et al.[39], di Cao et al.[31], di Srivastava et al.[35], di Vondrov et al.[49] e infine nell'approccio di Zhang et al.[42] Peng et al.[43] propone un meccanismo basato su cross-media retrieval tramite l'utilizzo di deep network multiple.

In quest'ultimo tipo di retrieval, fornendo semplicemente un'immagine di un certo oggetto, vengono restituiti diversi tipi di media correlati all'immagine inserita. Per esempio, inserendo un'immagine di una tigre, vengono restituiti l'audio in cui una tigre ruggisce e un video in cui si parla di una tigre.

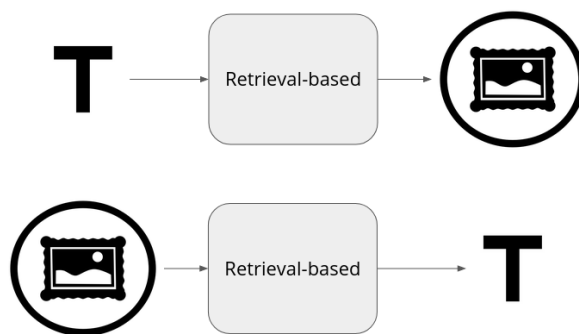


Figura 10: Modelli retrieval-based

Il secondo metodo, contenuto nei modelli denominati *generative approaches*, utilizza gli encoder e i decoder. Il modo in cui lavorano questi modelli è il seguente: inizialmente l'encoder valuta la modalità sorgente e la trasforma sotto forma di rappresentazione vettoriale, successivamente il decoder genera il dato nella modalità desiderata.

In maniera simile al metodo precedente, è presente un dizionario. Come però si può intuire confrontando le Figure 11 e 12, in questo metodo non viene fatta una ricerca nel dizionario per restituire l'informazione mancante, bensì viene creato e addestrato un modello di translation.

Basandosi sugli elementi presenti nel dizionario apprende come tradurre il dato sorgente nel dato che viene richiesto, effettuando il tutto in un singolo passaggio.

Questo tipo di approccio viene presentato da Kiros et al.[41], nel quale viene ricavata l'informazione testuale dando in input delle immagini. Nell'approccio presentato da Ngiam et al.[32] viene utilizzato un deep autoencoder bi-modale, il quale utilizza dati audio e dati video. Un approccio simile viene presentato da Silberer et al.[34], l'unica differenza consiste nei dati utilizzati, ossia vengono utilizzati immagini e testo.

Attraverso questo metodo, dando in input un'immagine, è possibile ottenere la descrizione testuale corrispondente.

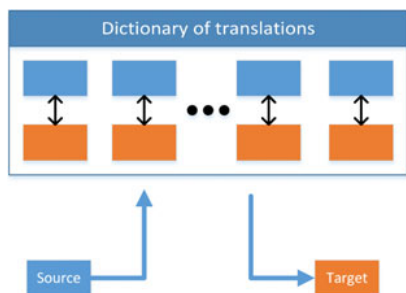


Figura 11: Example based

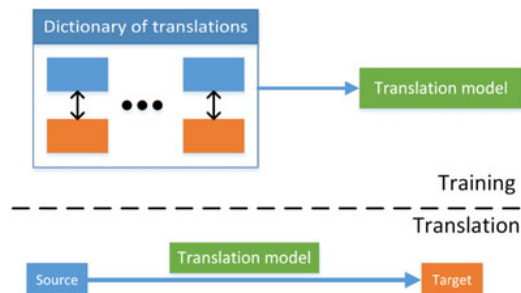


Figura 12: Generative approaches

### 3 Metodo proposto

Per affrontare il problema è stato deciso di proporre una struttura multimodale.

In particolare vengono distinte due pipeline: una per l'analisi dei dati visuali e una per l'analisi dei dati testuali.

Analizzando lo schema presente in Figura 13, è possibile notare la prima pipeline **Visual attention model** composta da un primo componente, chiamato **EfficientNet**, il quale compito risulta essere analizzare l'immagine in input e, attraverso gli strati convoluzionali, riconoscere caratteristiche presenti in quest'ultima. Successivamente, i dati raccolti, vengono passati al componente **visual attention**, il quale ha il compito di concentrare e far risaltare le regioni dell'immagine di maggiore interesse. Una volta applicato il meccanismo di attenzione, la rete convoluzionale prosegue e termina il suo ciclo attraverso gli **FC layers**, attraverso il quale viene effettuata la **predizione** e si ottengono i risultati relativi alle immagini.

In maniera analoga, nella seconda pipeline **Semantic attention model** viene gestita la parte testuale. Nel primo componente, **embedding layer**, il testo viene processato per essere analizzabile dal componente **BERT**. Quest'ultimo è un modello che contiene già un meccanismo di **semantic attention**, il quale mette in risalto le parole più rilevanti a discapito di quelle meno rilevanti. Una volta apprese le nozioni per comprendere il linguaggio, la pipeline prosegue nel componente **FC layers**, attraverso il quale viene effettuata la **predizione**.

Una volta concluse le due pipeline, i dati vengono inseriti nell'ultimo componente. **Late fusion** ha il compito di combinare i risultati calcolati precedentemente e fornire in output un unico risultato, il quale rappresenta la coppia di pipeline.

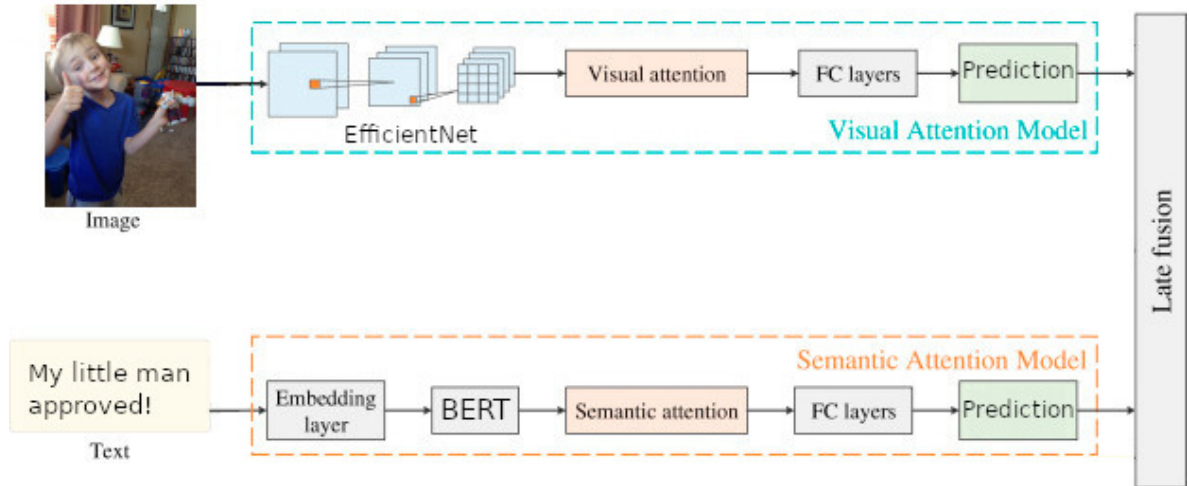


Figura 13: Architettura multimodale

Prende spunto dalla struttura presentata da Huang et al. nel lavoro *Image-text sentiment analysis via deep multimodal attentive fusion*[50].

A differenza del lavoro presentato da Huang et al., i modelli utilizzati sono stati sostituiti da modelli più recenti e più efficienti.

Come introdotto precedentemente, in Figura 13, è possibile notare le due pipeline: la prima dedicata all'apprendimento delle features visuali attraverso l'utilizzo del modello *EfficientNet*[51]; la seconda si occupa dell'apprendimento delle features testuali attraverso il modello *BERT*[52].

Per la parte visuale, la scelta è ricaduta su *EfficientNet* per vari motivi: è stata presentata durante la fine del 2019, è scalabile e necessita di un numero minore di parametri rispetto ai modelli paragonabili.

In particolare viene detta scalabile perchè si definisce sulla base della seguente formula:

$$\begin{aligned}
depth : d &= \alpha^\phi \\
width : w &= \beta^\phi \\
resolution : r &= \gamma^\phi \\
\alpha \cdot \beta^2 \cdot \gamma^2 &\approx 2 \\
\alpha \geq 1, \beta \geq 1, \gamma &\geq 1
\end{aligned} \tag{1}$$

Le tre costanti  $\alpha$ ,  $\beta$ ,  $\gamma$  vengono definite tramite grid search, invece  $\phi$  rappresenta un coefficiente che può essere scelto dall'utente in base a quante risorse si vogliono dedicare alla rete.

Vengono già fornite otto tipologie di rete, da *B0* a *B7*, in cui le costanti sono  $\alpha=1.2$ ,  $\beta=1.1$ ,  $\gamma=1.15$  e il coefficiente  $\phi$  varia.

La baseline, chiamata *EfficientNet-B0*, è rappresentata nell'equazione 1 ed è la rete scelta per il modello proposto. Prendendo in considerazione il numero dei parametri di *EfficientNet-B0*, la rete *ResNet-50* [55] ne utilizza 4.9x e la rete *DenseNet-169* [56] ne utilizza 2.6x.

Come accennato prima, per la parte testuale è stato scelto *BERT*.

La sua architettura è composta da un multi-layer transformer bidirezionale basato su encoding. A differenza di LSTM, modello originariamente scelto, ha il vantaggio di essere bidirezionale e quindi valutare una frase in input in entrambi i versi, aumentandone così la capacità predittiva. Inoltre *BERT* incorpora gli strati di embeddings, riuscendo a gestire in autonomia le varie rappresentazioni che l'input deve avere durante l'elaborazione di quest'ultimo.

*BERT* gestisce in maniera atipica i dati che analizza, applicando due strategie per l'apprendimento: *masked LM* e *next sentence prediction*.

Attraverso la prima strategia, ogni sequenza di parole viene modificata del 15%, sostituendo alle parole scelte un token *[MASK]*: in tal modo, il modello cerca di prevedere il valore originale delle parole oscurate, sfruttando il contesto fornito dalle altre parole non oscurate.

Tramite la seconda strategia, il modello *BERT* riceve in input coppie di frasi e cerca di imparare a capire se la seconda frase fornita risulti effettivamente successiva alla prima frase nel testo originale. A livello di percentuali viene creato un insieme in cui il 50% delle coppie risultano essere successive nel documento di riferimento, l'altro 50% invece contiene due frasi accoppiate in maniera casuale.

Per cercare di rendere ancora più efficiente il modello proposto, nelle pipeline sono stati inseriti meccanismi di attenzione. Attraverso questi ultimi si cerca di aumentare il focus sulle regioni delle immagini e sulle parole maggiormente importanti per la classificazione.

### 3.1 Meccanismo di attenzione semantico

Come anticipato precedentemente, in una frase sono presenti parole più utili alla classificazione rispetto ad altre. Per l'attenzione testuale, è stata sfruttata la struttura di *BERT*. A differenza di altri modelli come *LSTM*, *BERT* è costituito da una struttura chiamata *Transformer* la quale si basa interamente su un meccanismo di attenzione. Attraverso questo meccanismo vengono trovate dipendenze tra dati in input e dati in output. La nuova struttura *Transformer* permette di raggiungere performance migliori pur avendo una sessione di training minore.

L'architettura del *Transformer* si basa su una struttura encoder-decoder, nella quale vengono usati meccanismi di self-attention a più strati e layer fully connected sia per l'encoder che per il decoder.

L'encoder è composto da uno stack di 6 layers identici, in ognuno dei quali sono presenti due componenti: un meccanismo di self-attention multi-head e una rete feed-forward fully connected. Anche il decoder è composto da 6 layers identici ma in ognuno di questi sono presenti tre componenti: un meccanismo di attenzione multi-head masked, un secondo meccanismo di attenzione multi-head che gestisce l'output dell'encoder ed infine una rete feed-forward fully connected. Nel decoder viene aggiunto un meccanismo mascherato perché, avendo gli embeddings sfalsati di una posizione, assicura che la predizione per un elemento  $i$  possa dipendere solamente dagli output noti in posizioni precedenti a  $i$ .



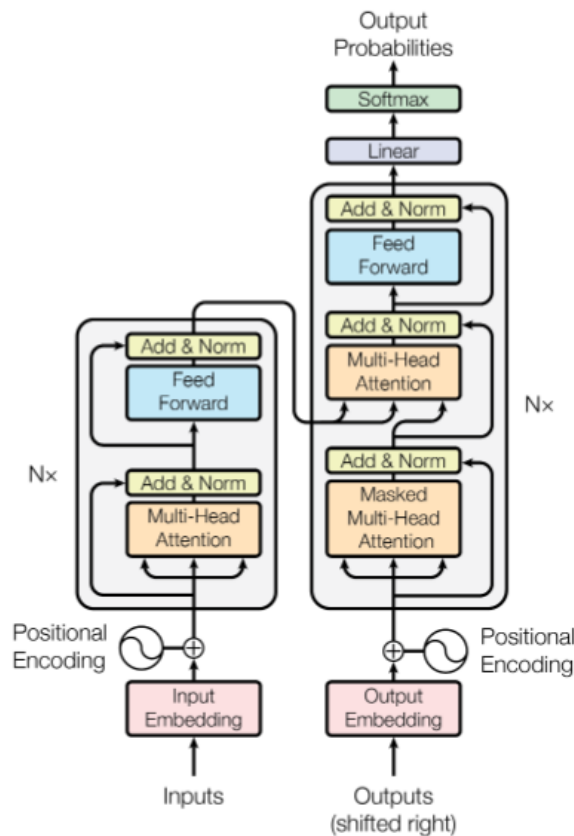


Figura 14: Struttura transformer

Concentrandosi invece sull'attenzione, questa può essere descritta come una funzione definita per confrontare ogni parola presente nella frase con tutte le altre per capire meglio qual è il contributo di ogni singola parola e come codificare l'intera frase.

La frase in input è composta da  $n$  tokens, ogni input viene trasformato in vettori query, key e value  $q_i$ ,  $k_i$  e  $v_i$  attraverso trasformazioni lineari distinte. L'output finale viene calcolato come somma pesata dei vettori value, dove il peso assegnato ad ogni elemento value viene trovato mettendo in relazione il vettore query con l'elemento key corrispondente.

Questo particolare meccanismo viene chiamato *Scaled dot-product attention*: il testo in input è composto da query e key di dimensione  $d_k$  e da valori di

dimensione  $d_v$ . Viene calcolato il prodotto scalare tra il vettore query e tutte le keys, ognuna divisa per  $\sqrt{d_k}$ , e successivamente, per ottenere i pesi relativi agli elementi del vettore value, viene applicata la funzione softmax.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (2)$$

Mettendo insieme più meccanismi *Scaled dot-product attention* viene creato un meccanismo di attenzione multi-head, nel quale i singoli strati di attenzione lavorano parallelamente. In questo modo il modello, che utilizza questo insieme di singole attenzioni, ha la possibilità di gestire informazioni provenienti da diverse rappresentazioni in diverse posizioni.

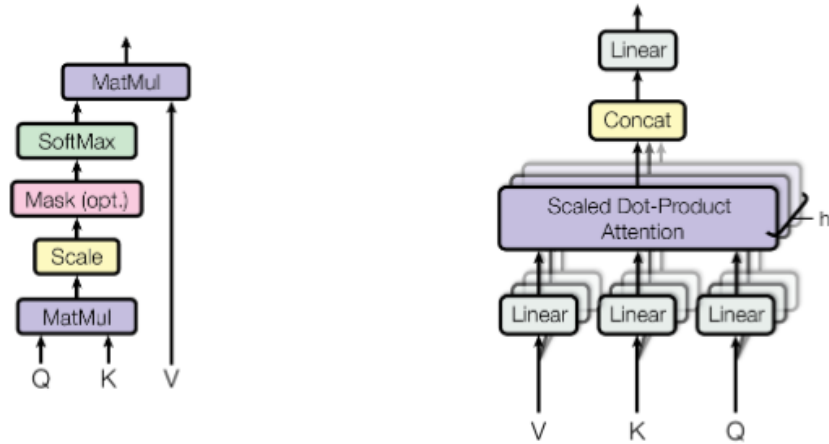


Figura 15: Scaled dot-product attention (sx), multi-head attention (dx)

Per rendere più facile la comprensione di questo meccanismo può essere spiegato nel seguente modo: il vettore query ( $Q$ ) rappresenta quale sia il tipo di informazione che viene cercata, il vettore key ( $K$ ) rappresenta la pertinenza alla query e il vettore value ( $V$ ) rappresenta l'effettivo contenuto dell'input. Una volta presentati questi elementi, l'embedding dell'input viene indicato con una matrice  $X$ , dove il numero delle righe indica il numero di tokens presenti.

Per ottenere i tre vettori  $Q$ ,  $K$ ,  $V$  vengono usati tre pesi  $W^Q$ ,  $W^K$  e  $W^V$ , i quali vengono moltiplicati con l'input  $X$ . Il risultato (attention score) che

si ottiene indica la similarità tra i diversi tokens presenti e quindi premia le parole effettivamente più importanti e penalizza quelle meno rilevanti.

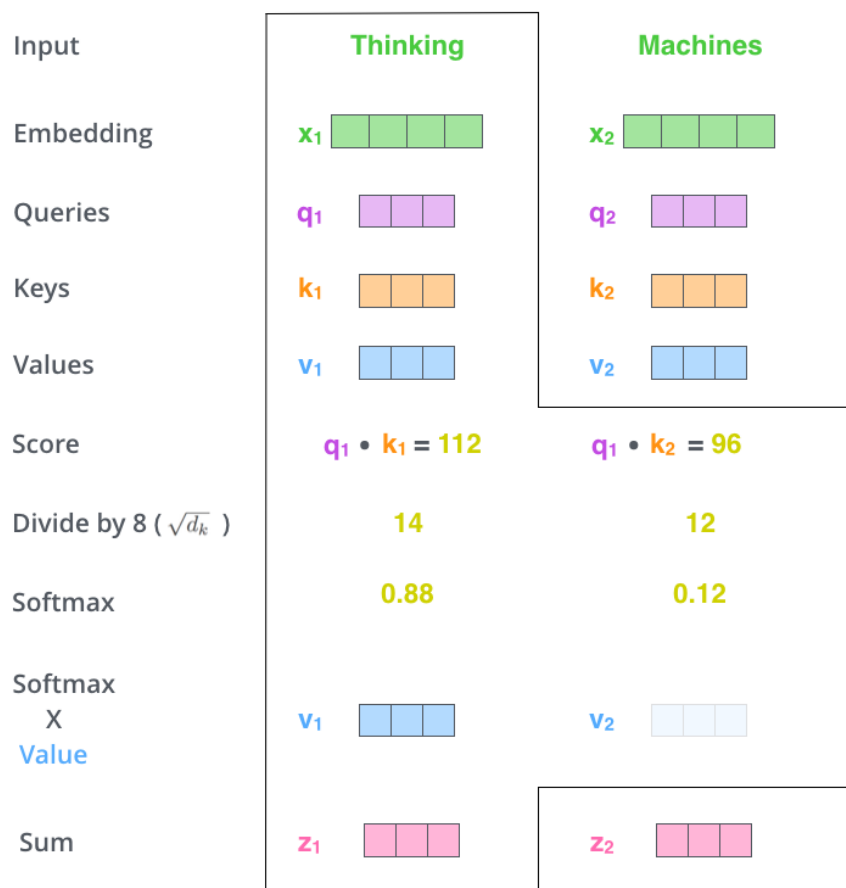


Figura 16: Esempio attention BERT<sup>1</sup>

<sup>1</sup><http://jalammar.github.io/illustrated-transformer/>

### 3.2 Meccanismo di attenzione visuale

Parallelamente alla parte testuale, anche per la parte visuale è stato selezionato un meccanismo di attention.

Come meccanismo è stato scelto quello presente nel lavoro *Image-text sentiment analysis via deep multimodal attentive fusion*[50] e riadattato per funzionare con la rete *EfficientNet*.

Generalmente, come nei testi, anche nelle immagini ci sono regioni più rilevanti e altre meno. Attraverso il meccanismo di attention si cerca di aumentare il divario tra queste zone: alle regioni più importanti viene assegnato un valore più alto, alle regioni meno importanti viene assegnato un valore basso, tendenzialmente vicino allo zero. In questo modo la classificazione dovrebbe essere più efficiente.

Già in altri task, come nella visual sentiment analysis presente nel lavoro di You et al.[29] e nell'image captioning presente nel lavoro di Lu et al.[57], l'applicazione dell'attenzione visuale risulta positiva.

Partendo da questo presupposto è stato scelto di implementare un meccanismo di attenzione di questo tipo: per ogni immagine che viene elaborata dalla rete, vengono prese le features contenute in una matrice e successivamente ne vengono modificati i valori.

Quindi, avendo un insieme di  $n$  immagini, tramite l'applicazione della rete convoluzionale si ottiene una matrice dove ogni elemento rappresenta una singola regione dell'immagine.

$$X = \begin{pmatrix} r_1^1 \dots r_1^j \dots r_1^D \\ \dots & \dots & \dots \\ r_i^1 \dots r_i^j \dots r_i^D \\ \dots & \dots & \dots \\ r_M^1 \dots r_M^j \dots r_M^D \end{pmatrix} \quad (3)$$

dove :  $M = 49, D = 1280, i \leq M, j \leq D$

Per ogni regione dell'immagine viene calcolato un peso  $e$  chiamato *attention score non normalizzato*.

$$H = (W^T X + b)$$

$$H = \begin{pmatrix} e_1^1 \dots e_1^j \dots e_1^D \\ \dots \dots \dots \\ e_i^1 \dots e_i^j \dots e_i^D \\ \dots \dots \dots \\ e_M^1 \dots e_M^j \dots e_M^D \end{pmatrix} \quad (4)$$

$$dove : M = 49, D = 1280, i \leq M, j \leq D$$

$W$  e  $b$  rappresentano dei parametri che vengono imparati durante l'elaborazione di ogni immagine.

Alla matrice  $H$ , composta dagli score precedentemente calcolati, viene applicata una funzione softmax la quale normalizza i valori in un range tra 0 e 1. Quest'ultimi rappresentano la rilevanza al sentimento presente nell'immagine.

$$H = softmax(H)$$

$$H = \begin{pmatrix} \alpha_1^1 \dots \alpha_1^j \dots \alpha_1^D \\ \dots \dots \dots \\ \alpha_i^1 \dots \alpha_i^j \dots \alpha_i^D \\ \dots \dots \dots \\ \alpha_M^1 \dots \alpha_M^j \dots \alpha_M^D \end{pmatrix} \quad (5)$$

$$dove : M = 49, D = 1280, i \leq M, j \leq D, \alpha \in [0; 1]$$

Come ultimo passaggio, i valori della matrice  $X$  vengono moltiplicati con gli score presenti nella matrice  $H$ :

$$X_{i,j} = X_{i,j} * H_{i,j} \quad (6)$$

$$dove : i < 49, j < 1280$$

In maniera pratica ci si posiziona nell'ultimo strato di convoluzione, chiamato *top\_conv* nella rete *EfficientNet*; l'output, che ha dimensione [7, 7, 1280], viene ridimensionato in una matrice bidimensionale da [49, 1280].

Una volta ottenuta questa matrice, chiamata matrice delle features, vengono

applicati i passi precedentemente descritti.

Terminato il processo di attention, la matrice finale viene ridimensionata in modo da renderla adatta agli ultimi layers fully connected, effettuando così la classificazione.

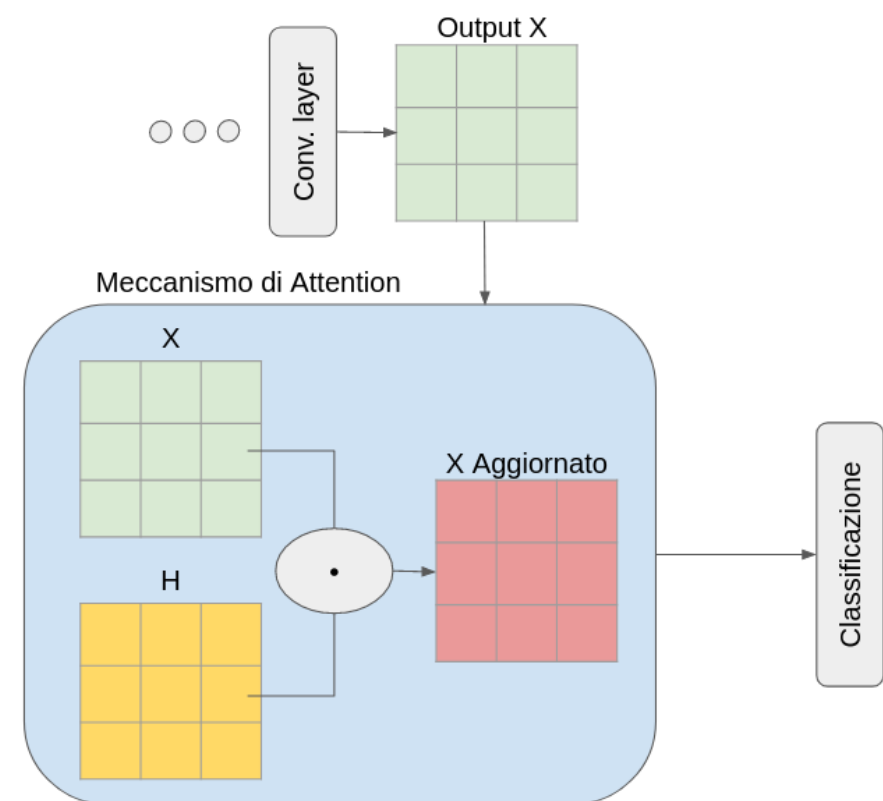


Figura 17: Schema meccanismo di attention

### 3.3 Sentiment classification

In questa tesi sono stati affrontati due task diversi, con dati diversi.

Il primo task affrontato è stata la classificazione del sentimento.

Il dataset è composto da testo e immagini, ad ogni coppia corrisponde una label che può essere *positive* o *negative*. Quindi è un tipo di classificazione binaria.

Di seguito un esempio di due coppie presenti nel dataset:



Immagine	Testo	Label
	Good morning sunshine. Beautiful sea.	Positive
	On this day 14 years ago, Hurricane Katrina changed our lives forever.	Negative

Tabella 1: Esempio dataset

Grazie al lavoro proposto da Vadicamo et al.[53], il dataset utilizzato è disponibile sul sito [Twitter for sentiment analysis](#).

Come detto precedentemente, gli elementi della coppia vengono elaborati singolarmente nelle due pipeline unimodali e congiuntamente nella pipeline multimodale. Successivamente i risultati vengono uniti e presentati.

### 3.4 Emotion classification

Successiva alla classificazione del sentimento, si è provveduto alla classificazione delle emozioni.

Alle coppie immagine-testo viene assegnato una label tra le seguenti:

- *contentment*
- *amusement*
- *awe*
- *excitement*
- *sadness*
- *disgust*
- *anger*
- *fear*



Immagine	Testo	Label
	Firewoks festival at seibuenn park 2012...	Amusement
	cry light(ly) these self-portrait images follow a progression of emotions...	Sadness

Tabella 2: Esempio dataset

Il dataset utilizzato è stato fornito dal Dipartimento di Informatica, Sistemistica e Comunicazione dell'Università degli Studi di Milano-Bicocca ed è presente nell'elaborato di Corchs et al. [54] Come illustrato nel capitolo riguardante la classificazione del sentimento, anche in questo task viene mantenuta la medesima elaborazione delle coppie in input.



## 4 Esperimenti

In questo capitolo vengono mostrati gli esperimenti condotti durante lo svolgimento dei due task precedentemente presentati.

Verranno presentati in maniera approfondita i dataset utilizzati, le misure di performance scelte ed infine verrà mostrata l'analisi dell'errore.

### 4.1 Sentiment classification

#### 4.1.1 Dataset

Il dataset utilizzato per la sentiment classification è disponibile sul sito [Twitter for sentiment analysis](#).

I dati presenti sono stati raccolti tra Luglio e Dicembre 2016 ed equivalgono a circa l'1% del flusso di tweets prodotti a livello globale.

Dai tweet raccolti sono stati scartati quelli che:

- contenevano video, GIF o altri tipi di media diversi da immagini statiche
- non erano scritti in inglese
- avevano il testo composto da meno di 5 parole
- erano retweets

Il dataset viene fornito sottoforma di file *.csv*, nel quale sono presenti un numero identificativo e il testo del tweet. L'elenco delle immagini viene fornito attraverso un secondo file *.csv*, nel quale sono presenti il percorso relativo all'immagine del tweet e la label corrispondente. Per mantenere la corrispondenza tra il tweet e l'immagine, una parte del nome di quest'ultima è uguale all'identificativo del tweet.

Image.csv	
IMG	Label
data/76878/768781748033335296-1.jpg	1

Tweet.csv	
ID	Tweet
768781748033335296	Thank you Jack for bringing...

Figura 18: Esempio coppia immagine-testo

Nel dataset il rapporto testo-immagini non è 1:1 ma ad un testo possono corrispondere più immagini. Originariamente alle coppie immagini-testo poteva essere assegnata una fra queste label: negative, neutral e positive (0, 1, 2). Per scelte di progettazione è stato deciso di considerare solamente le label negative e positive (0, 1), escludendo quindi tutti i dati contrassegnati con la label neutral.

Il dataset finale è stato quindi diviso in 78% train, 11% validation e 11% test. Di seguito una tabella riassuntiva sulla composizione del dataset:

	Negative (0)	Positive (1)
Train	122862	122862
Validation	17000	17000
Test	17000	17000

Tabella 3: Struttura dataset

Per addestrare la parte testuale sono stati utilizzati tutti gli elementi presenti nel dataset, nella Tabella 4 e nella Tabella 5 vengono mostrati i risultati.

Per mancanza di potenza computazionale lato immagini, è stato deciso di utilizzare solamente 40000 dati di train rispetto ai 245724 dati iniziali; unendo

i dati di validation e di test per effettuare gli esperimenti. Infatti anche il modello testuale è stato trainato nuovamente e i dataset di validation e di test sono stati uniti, così da poter fondere i risultati (Tabella 6) con quelli ottenuti nella parte testuale.

#### 4.1.2 Baseline

Per la sentiment classification è stato utilizzato come baseline il lavoro *Image-text sentiment analysis via deep multimodal attentive fusion* presentato da Huang et al.[50]

Il dataset utilizzato in questa tesi non è lo stesso utilizzato nella baseline dato che quest'ultima è risultato irreperibile.

Prendendo come riferimento i test effettuati su un dataset con dati provenienti da Twitter, i risultati ottenuti sono i seguenti: nei test per la parte testuale viene raggiunta un'accuracy del 73.4% tramite l'utilizzo del modello *LSTM*, per la parte visuale viene raggiunta un'accuracy del 66.4% attraverso l'utilizzo della rete *VGG-19*.

L'unione di questi due rami più la pipeline multimodale, ottenuta applicando una early fusion alle due pipeline unimodali, ha restituito un'accuracy pari a 76.3%.

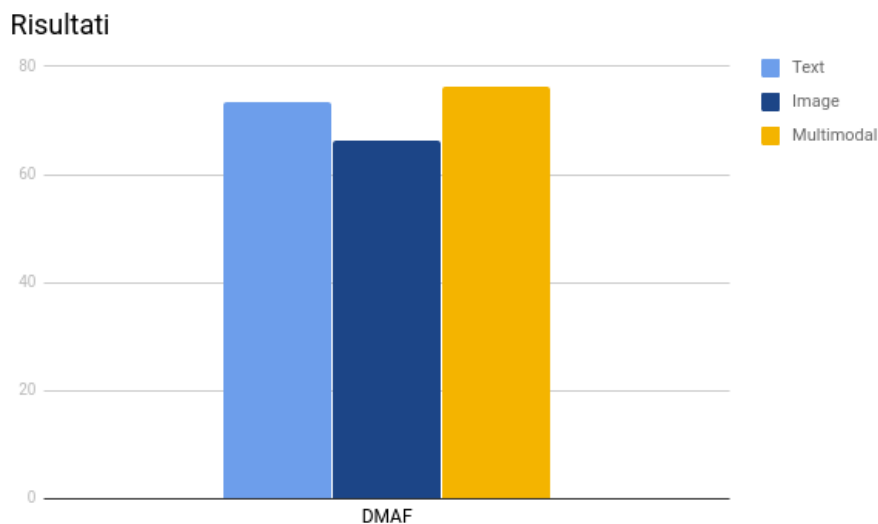


Figura 19: Risultati baseline

### 4.1.3 Risultati parte testuale

Gli esperimenti sulla parte testuale sono stati condotti impostando i seguenti settaggi:

- batch size = 32
- learning rate =  $2e^{-5}$
- steps = 23036
- max sequence length = 128

In fase di validation i risultati ottenuti sono i seguenti:

Precision	Recall	F1-score	Accuracy
0.9911	0.9901	0.9906	0.9906

Tabella 4: Risultati validation

In fase di test i risultati ottenuti sono i seguenti:

Label	Precision	Recall	F1-score	Accuracy
Negative	0.99	0.99	0.99	0.9914
Positive	0.99	0.99	0.99	

Tabella 5: Risultati test

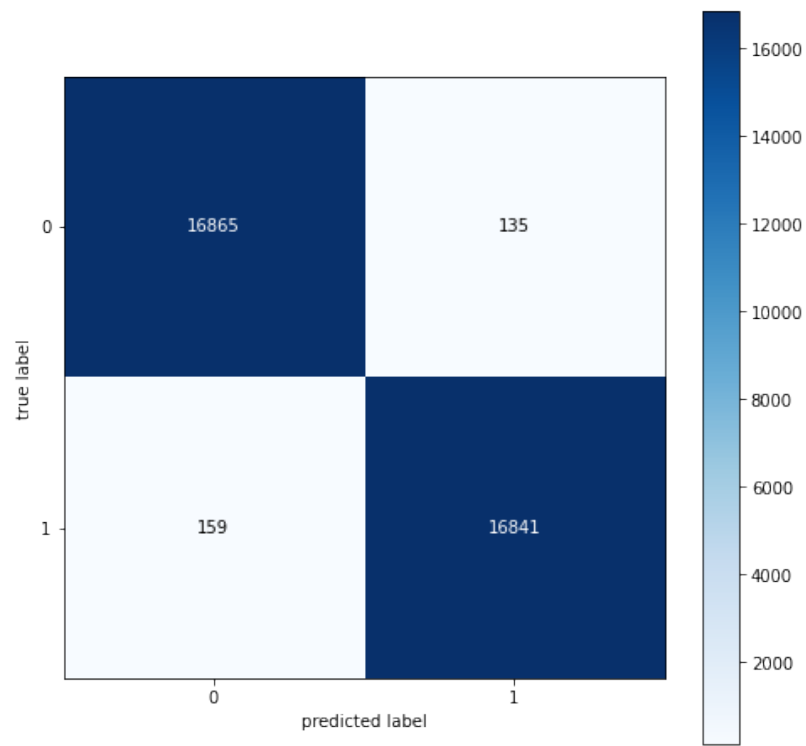


Figura 20: Matrice di confusione

Per correttezza vengono mostrati anche i risultati ottenuti effettuando il test sul dataset ottenuto dall'unione dei dataset di validation e di test. Questi risultati saranno poi fusi con i risultati ottenuti nella parte visuale.

Label	Precision	Recall	F1-score	Accuracy
Negative	0.99	0.99	0.99	0.9909
Positive	0.99	0.99	0.99	

Tabella 6: Risultati test

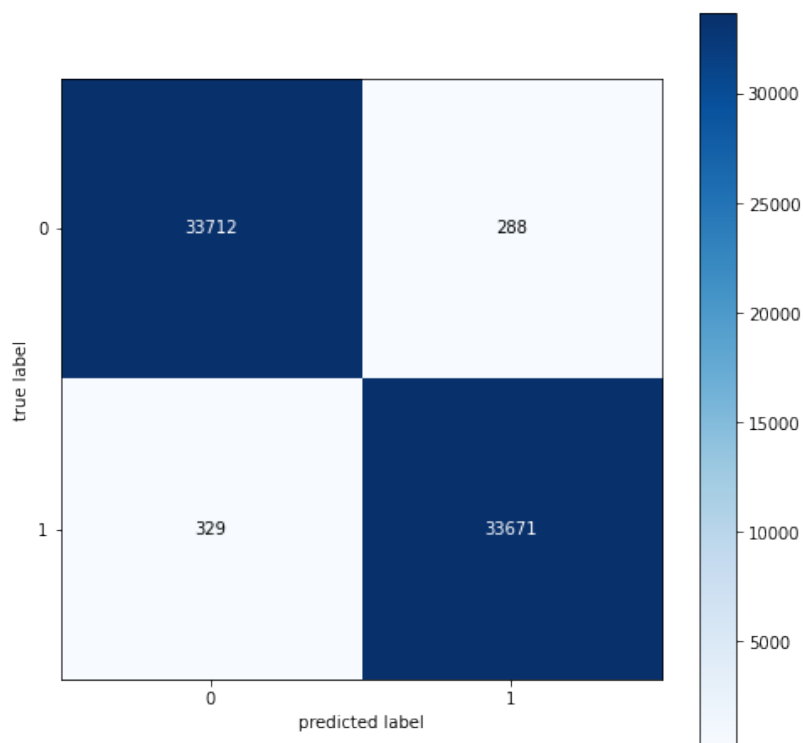


Figura 21: Matrice di confusione

Come si può vedere dalle tabelle e dalle matrici di confusione riportate qui sopra, il modello si comporta in maniera impeccabile sia attraverso la divisione in fase di train, validation, test, sia nella divisione utilizzata per sopperire alle difficoltà riscontrate nel modello visuale.

#### 4.1.4 Risultati parte visuale

Come detto nel capitolo 4.1.1, per questa fase, il dataset è stato gestito in maniera diversa. Per la parte di training è stato utilizzato un sottoinsieme bilanciato da 40000 immagini del dataset di train iniziale, successivamente i dataset di validation e test sono stati uniti per avere così un dataset da 68000 immagini da poter testare.

Questa scelta ha portato un miglioramento rispetto alle prove effettuate in precedenza, a seguire vengono mostrate le impostazioni scelte per il modello e i risultati.

- batch size = 64
- epoche = 10

#### Test senza attention

Label	Precision	Recall	F1-score	Accuracy
Negative	0.56	0.56	0.56	0.5613
Positive	0.56	0.56	0.56	

Tabella 7: Risultati test

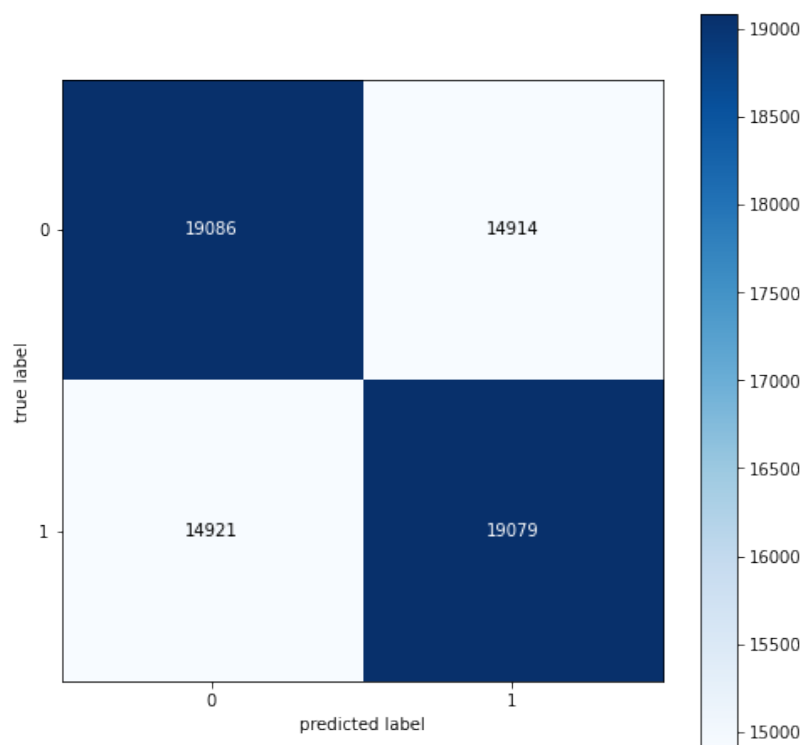


Figura 22: Matrice di confusione

Vengono riportate qui sopra la tabella con i risultati ottenuti in fase di test e la matrice di confusione corrispondente. Il modello riesce a comportarsi abbastanza bene, tenendo conto delle difficoltà riscontrate durante la fase di training. Per riuscire a terminare quest'ultima fase è stato necessario utilizzare impostazioni più grezze che hanno causato una perdita di accuratezza.



### Test con attention

Label	Precision	Recall	F1-score	Accuracy
Negative	0.59	0.64	0.62	0.6019
Positive	0.61	0.56	0.59	

Tabella 8: Risultati test

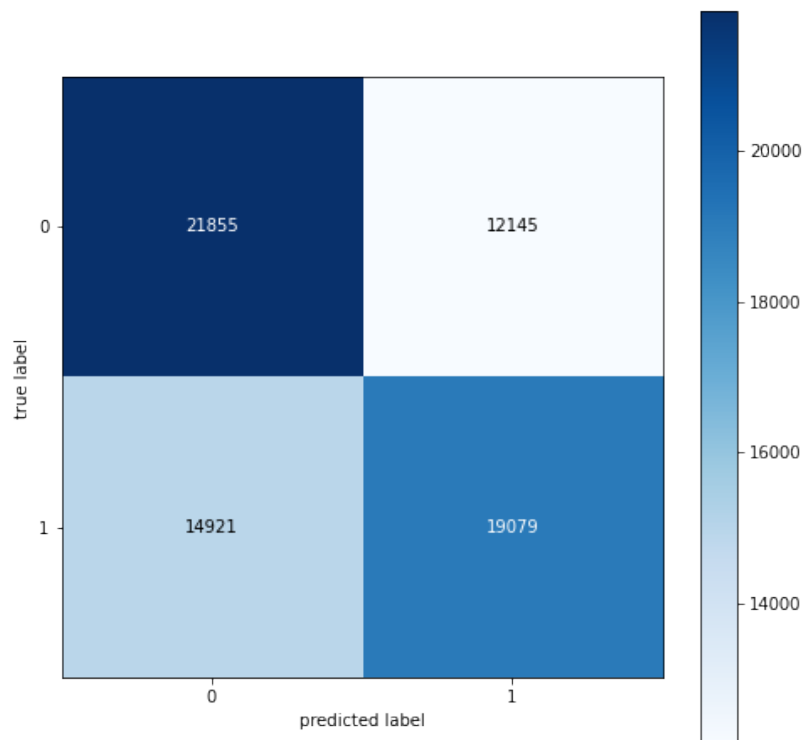


Figura 23: Matrice di confusione

Come si può notare, confrontando la Tabella 8 con la Tabella 7, l'applicazione del meccanismo di attenzione porta un leggero miglioramento. Anche in questo caso, come spiegato nel commento ai risultati ottenuti senza l'utilizzo del meccanismo di attention, sono state riscontrate difficoltà.

#### 4.1.5 Risultati late fusion

In questa sezione viene presentato l'esito finale ottenuto applicando una late fusion ai risultati provenienti dai rami visuale e testuale.

Come detto in precedenza, per necessità, sono stati uniti i dataset di validation e di test. I risultati vengono divisi per applicazione o meno del meccanismo di attenzione visuale.

I risultati sono stati combinati secondo due tecniche: *massimo* e *media*.

Fornendo in input un dato, ogni modello predice due probabilità: una probabilità che il dato sia positivo e una probabilità che sia negativo. Di conseguenza per ogni coppia immagine-testo sono presenti due coppie di probabilità.

Utilizzando il criterio del massimo, viene scelto il valore maggiore tra le due probabilità per la label negativa e il valore maggiore tra le due probabilità per la label positiva. Ottenute queste due probabilità viene scelto il massimo assegnando la label corrispondente alla coppia immagine-testo.

Utilizzando il criterio della media, viene trovato il valore medio nella coppia di probabilità per la label negativa e il valore medio nella coppia di probabilità per la label positiva. Ottenute queste due probabilità viene scelto il massimo assegnando la label corrispondente alla coppia immagine-testo.

#### Test senza attention

##### Criterio decisionale - Massimo

Label	Precision	Recall	F1-score	Accuracy
Negative	0.56	0.56	0.56	0.5613
Positive	0.56	0.56	0.56	

Tabella 9: Risultati late fusion

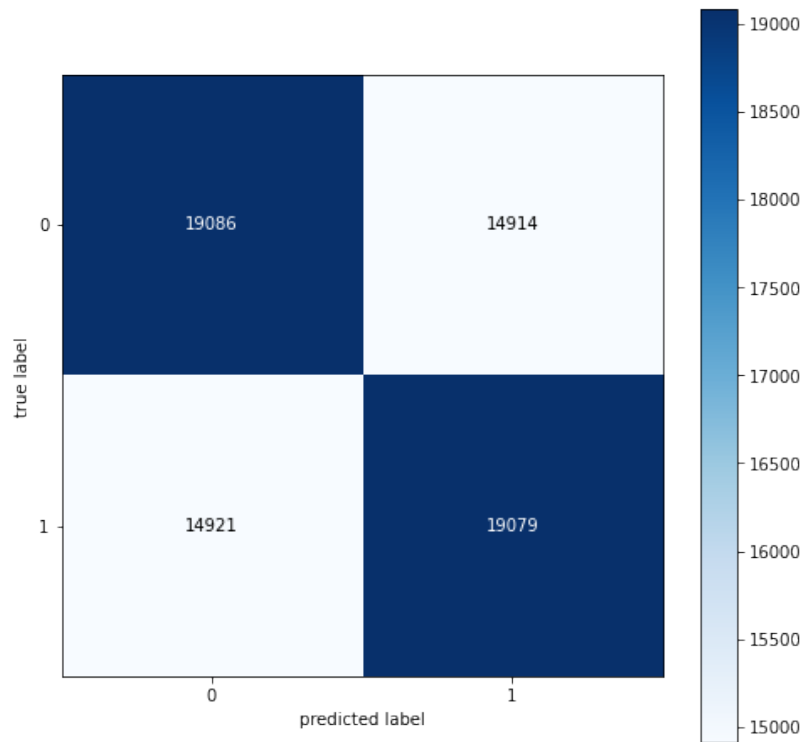


Figura 24: Matrice di confusione

A causa delle difficoltà riscontrate dal modello visuale, le probabilità predette da quest'ultimo non risultano essere equamente distribuite e di conseguenza risultare prossime al valore 1 per la label predetta. Ciò non succede nel modello testuale che, pur funzionando meglio, ha una distribuzione di probabilità più equa.

Scegliendo il massimo come metodo di fusione, i risultati tendono a somigliare a quelli ottenuti nel ramo visuale: le probabilità presenti nei risultati ottenuti nella parte testuale, pur essendo migliori, non riescono a superare le probabilità prossime a 1 della parte visuale.

### Criterio decisionale - Media

Label	Precision	Recall	F1-score	Accuracy
Negative	0.99	0.99	0.99	0.9909
Positive	0.99	0.99	0.99	

Tabella 10: Risultati late fusion

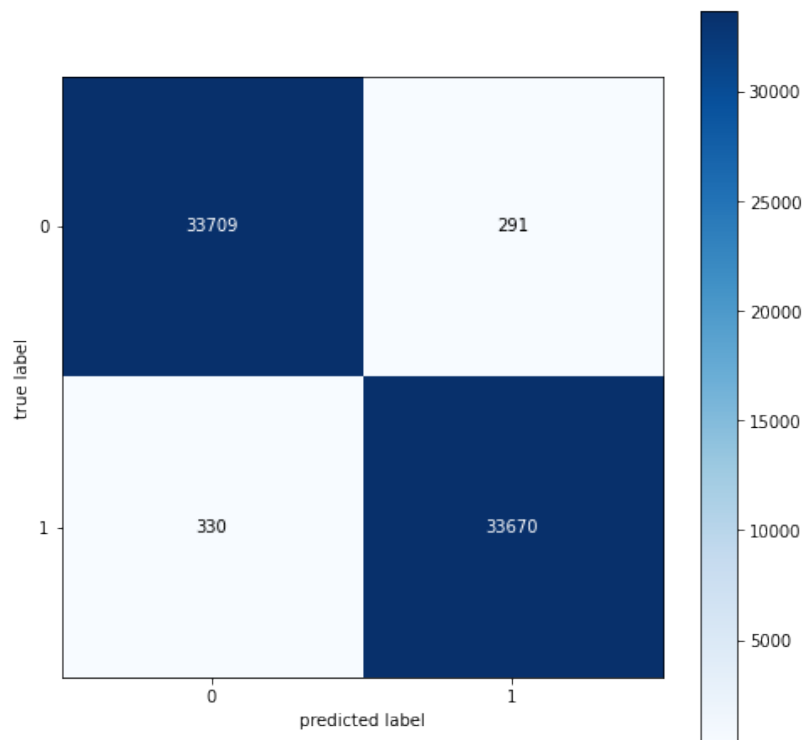


Figura 25: Matrice di confusione

In questo caso, diversamente dalla fusione tramite massimo, i risultati ottenuti nella parte testuale riescono a migliorare quelli ottenuti nella parte visuale. Risulta così essere un metodo di fusione più adatto.

### Test con attention

#### Criterio decisionale - Massimo

Label	Precision	Recall	F1-score	Accuracy
Negative	0.59	0.64	0.62	0.6019
Positive	0.61	0.56	0.59	

Tabella 11: Risultati test

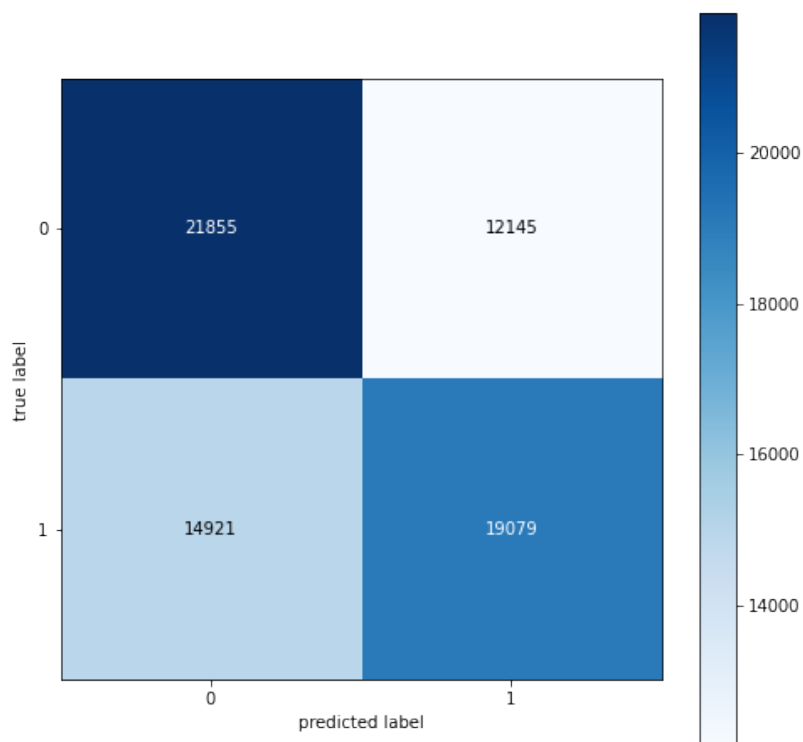


Figura 26: Matrice di confusione

Valgono le medesime motivazioni utilizzate nel commento alla Tabella 9.

### Criterio decisionale - Media

Label	Precision	Recall	F1-score	Accuracy
Negative	0.99	0.99	0.99	0.9909
Positive	0.99	0.99	0.99	

Tabella 12: Risultati late fusion

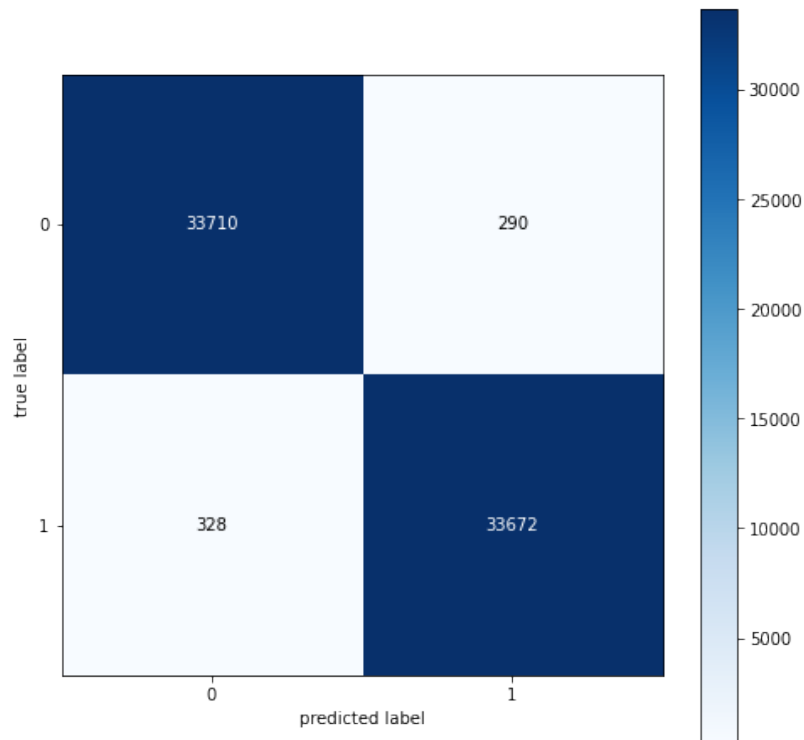


Figura 27: Matrice di confusione

Come si può notare dalla Tabella 12, anche in questo caso i risultati vengono equilibrati tramite la fusione utilizzando la media. Prendendo in considerazione la Figura 27 e confrontandola con la Figura 25, è presente un leggerissimo miglioramento. Seppur minimo, il miglioramento ottenuto attraverso l'applicazione del meccanismo di attention si può notare.

#### 4.1.6 Confronto risultati con baseline

In questa sezione vengono confrontati i risultati ottenuti negli esperimenti precedenti con quelli contenuti nella baseline. I risultati vengono divisi per uso del meccanismo di attention; per quanto riguarda la fusione vengono presi quelli ottenuti fondendo i risultati tramite la media, tramite la quale sono stati raccolti i risultati migliori.

Per la parte testuale vengono riportati i risultati ottenuti attraverso l'utilizzo del dataset composto sia dai dati di validation sia dai dati di test, presenti nella Tabella 6.

	Text	Image	Multimodal
DMAF[50]	73.4%	66.2%	76.3%
Attention	99.09%	60.19%	99.09%
No Attention	99.09%	56.13%	99.09%

Tabella 13: Confronto risultati

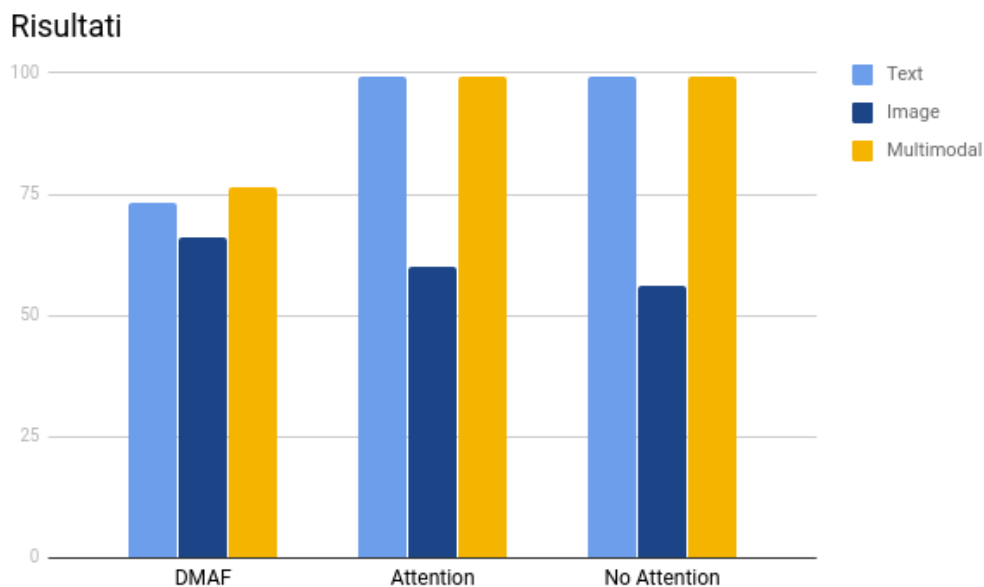


Figura 28: Confronto risultati

Confrontando i risultati ottenuti con quelli della baseline, è possibile notare un netto miglioramento lato testo. L'utilizzo del modello *BERT* porta molti vantaggi, migliorando del 26% circa le performance in confronto a quelle ottenute tramite l'utilizzo del modello *LSTM* utilizzato da Huang et al.[50]. Considerazione opposta vale per i risultati ottenuti lato visuale: tenendo sempre in considerazione le difficoltà riscontrate, il meccanismo di attention avvicina le performance, diminuendo la differenza di accuracy dal 10% al 6%. Grazie alle ottime performance ottenute nel lato testuale, il risultato della valutazione multimodale supera quello ottenuto nella baseline di circa il 23%.

#### 4.1.7 Analisi dell'errore

In questo capitolo viene fatta un'analisi sulle predizioni sbagliate restituite dal modello testuale. Il dataset di test utilizzato per effettuare l'analisi rispetta la divisione presentata nella Tabella 3.

Inizialmente si può affermare che, vendendo anche i risultati presentati nella Tabella 5, il modello si comporta in maniera quasi perfetta. Su 34000 dati in input, solamente 294 risultano essere errati, pari allo 0.86%.

In particolare 159 frasi positive vengono predette come negative e 135 frasi negative vengono predette come positive.

Come prima analisi è stata controllata la composizione delle frasi: ossia da quante parole e da quanti caratteri sono composte.

Label	N. Medio Parole	N. Medio Caratteri
Negativo	11.30	66.90
Positivo	12.10	76.55
Totale	11.70	71.72

Tabella 14: Predizioni corrette

Label	N. Medio Parole	N. Medio Caratteri
Negativo	13.34	79.56
Positivo	13.65	80.90
Totale	13.51	80.29

Tabella 15: Predizioni errate



Successivamente sono state controllate le 10 parole più presenti, divise per label, nelle frasi predette correttamente e in quelle predette in maniera errata.

Negativo	Positivo
like	happy
get	love
can't	birthday
fuck	day
u	great
look	good
bad	best
people	beautiful
shit	thank
hate	much

Tabella 16: Predizioni corrette

Neg → Pos	Pos → Neg
love	love
good	good
like	look
best	really
can't	great
people	miss
thanks	can't
get	get
look	got
wait	lol

Tabella 17: Predizioni errate

In questo confronto è possibile notare che, nelle frasi predette in maniera corretta, tra le 10 parole più usate, nemmeno una viene condivisa tra le due label. Il contrario invece succede nelle predizioni errate dove ben la metà delle parole più usate vengono condivise: questa informazione, unita al fatto che tendenzialmente le frasi in cui è stata fatta una predizione errata sono più lunghe, porta a pensare che il non utilizzo di termini facilmente connotabili e la maggior lunghezza del testo rendano difficile la predizione da parte del modello.

Quindi l'utilizzo di termini più rappresentativi per il sentimento presente nella frase e una maggiore attenzione sul non inserimento di termini che non danno informazioni utili, porta a delle migliori performance da parte del modello.

Analizzando ulteriormente le frasi predette in maniera errata, spiccano tre motivazioni plausibili per il quale è stato commesso l'errore: la prima è la presenza di ironia nel testo, la quale porta il modello a comprenderne in maniera opposta il significato. Questa motivazione è adatta anche al secondo motivo di errore, ovvero la litote: una figura retorica con il quale si cerca di attenuare o enfatizzare un concetto.

La terza motivazione riguarda la presenza nel testo di informazioni che facciano riferimento alla foto in sé, magari con uno scopo pubblicitario, oppure

una descrizione di ciò che è stato utilizzato per effettuare quest'ultima.  
Alcuni esempi di frasi predette in maniera errata che rientrano nelle tre motivazioni mostrate precedentemente:

Tweet	Label	Predizione
I will never forget this part omg #SMOSHLIVE	Pos	Neg
The Holocaust was never this much fun	Neg	Pos
I'm not happy	Neg	Pos
Every time I try to have an intelligent conversation with a #trump supporter this is the response I get	Neg	Pos
#ThursdayThoughts Spicy, funny and just a little bit naughty,go on treat yourself, only 99p	Neg	Pos
I'm posting the images here too since more people gets to see it this way	Neg	Pos
SA Barrel Horse Association's program for #burrapicnicraces. Dont miss out! #barrelracing	Pos	Neg
Oculus Wants Us to Pay How Much for Its Overdue To...via @gizmodo #technology #innovation	Pos	Neg

Tabella 18: Esempi predizioni

## 4.2 Emotion classification

### 4.2.1 Dataset

Il dataset utilizzato per l'emotion classification è stato fornito dal Dipartimento di Informatica, Sistemistica e Comunicazione dell'Università degli Studi di Milano-Bicocca.

Fino al lavoro presentato da Corchs et al.[54], in letteratura non erano presenti dataset contenenti coppie immagini-testo validate sulle emozioni elencate precedentemente. Data questa mancanza, sono state raccolte più di 3 milioni di immagini, utilizzando come keywords di ricerca le 8 emozioni.

Successivamente le immagini e le label associate sono state verificate da cinque addetti, i quali assegnavano *Yes* o *No* per confermare o meno la label associata ad un'immagine.

Quelle con tre o più *Yes* sono state raccolte e utilizzate anche in questa tesi. Prima di iniziare l'elaborazione, i dati sono stati controllati: alle iniziali 20588 coppie immagini-testo ne sono state tolte 541. Molte di queste avevano la parte testuale mancante, altre avevano il testo rappresentato non in inglese e, infine, alcune erano dei dopppioni.

Il risultato finale è un file *.csv*, il quale contiene un identificativo, il link Flickr dell'immagine, il nome assegnato all'immagine, il testo del tweet e la label corrispondente.

Il link rimanda direttamente all'immagine originale (es:[http://farm1.staticflickr.com/53/4086686053\\_85920608c0\\_z.jpg](http://farm1.staticflickr.com/53/4086686053_85920608c0_z.jpg)), il nome dell'immagine contiene la label assegnata.



Figura 29: contentment\_42343.jpg

Il dataset finale non risulta essere bilanciato, in particolare è popolato nella maniera seguente:

Label	Label numerica	Numero
Amusement	0	4287
Anger	1	1025
Awe	2	2883
Contentment	3	4491
Disgust	4	1477
Excitement	5	2610
Fear	6	844
Sadness	7	2430
TOTALE		20047

Tabella 19: Dataset emotion

Prima di effettuare i test sulla parte testuale, sono state effettuate delle operazioni sui tweet.

In particolare sono stati eliminati tutti i link e le emoticon presenti nel testo. Inoltre sono state eliminate tutte le occorrenze delle label nel testo, così da rendere più veritiero l'apprendimento.

#### 4.2.2 Baseline

Come detto precedentemente, il lavoro *Ensemble learning on visual and textual data for social image* proposto da Corchs et al.[54] è stato utilizzato come baseline per la classificazione delle emozioni.

In particolare, è possibile presentare i risultati provenienti dagli esperimenti effettuati da Corchs et al. Per la parte testuale raggiunge un'accuracy del 71% attraverso l'utilizzo del modello *SVM*. Per la parte visuale sono stati condotti due tipi diversi di test: il primo in cui sono state utilizzate *features visuali hand-crafted*, nel secondo sono state utilizzate *features visuali deep*. Nel primo test è stata raggiunta un'accuracy pari a 51.9% tramite l'utilizzo dell'albero decisionale *RandomSubSpace*, nel secondo test è stata raggiunta un'accuracy pari a 57.3% attraverso l'utilizzo del modello *SVM*. Di conseguenza sono state effettuate due fusioni tramite *Late-BMA*: la prima in cui sono state usate le *features visuali hand-crafted* in cui è stata raggiunta un'accuracy del 74%, nella seconda sono state utilizzate le *features visuali deep* raggiungendo un'accuracy pari 76%.

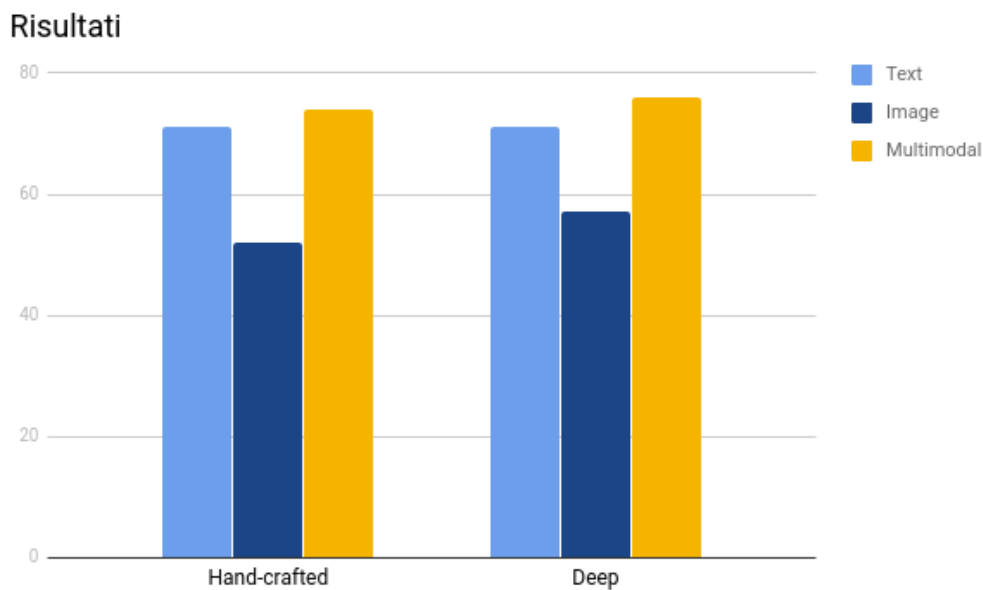


Figura 30: Risultati baseline

Dalla Figura 30 è possibile notare delle performance migliori tramite l'uti-

lizzo di tecniche *deep*. In ogni caso vengono utilizzate entrambe nel confronto successivo presente nel capitolo [Confronto risultati con baseline](#).

### 4.2.3 Risultati parte testuale

I test per la parte testuale dell’emotion classification sono stati condotti utilizzando la *10-Fold Cross Validation*. I settaggi sono stati scelti dopo varie prove e sono i seguenti:

- batch size = 32
- learning rate =  $2e^{-5}$
- steps = 1690
- warmup proportion = 0.1
- max sequence length = 128

In questo task è stato introdotto il warmup.

Nei dataset abbastanza differenziati potrebbe presentarsi un problema di *early over-fitting*; per limitarne l’influenza e diminuire il numero di epoche può essere applicata questo tipo di funzione.

Nel caso in cui non venisse implementata, durante il training sarebbero necessarie più epoche per far convergere il modello in maniera desiderata.

In maniera pratica, durante l’inizio della fase di training, il learning rate assume un valore minore del valore settato inizialmente e cresce con il passare delle iterazioni fino a raggiungere il valore impostato.

Label	Precision	Recall	F1-score	Accuracy
Amusement	0.98	0.99	0.98	0.9617
Anger	0.94	0.86	0.90	
Awe	0.94	0.98	0.96	
Contentment	0.97	0.96	0.96	
Disgust	1.00	0.96	0.98	
Excitement	0.93	0.98	0.95	
Fear	0.84	0.93	0.88	
Sadness	0.96	0.95	0.95	

Tabella 20: Risultati test

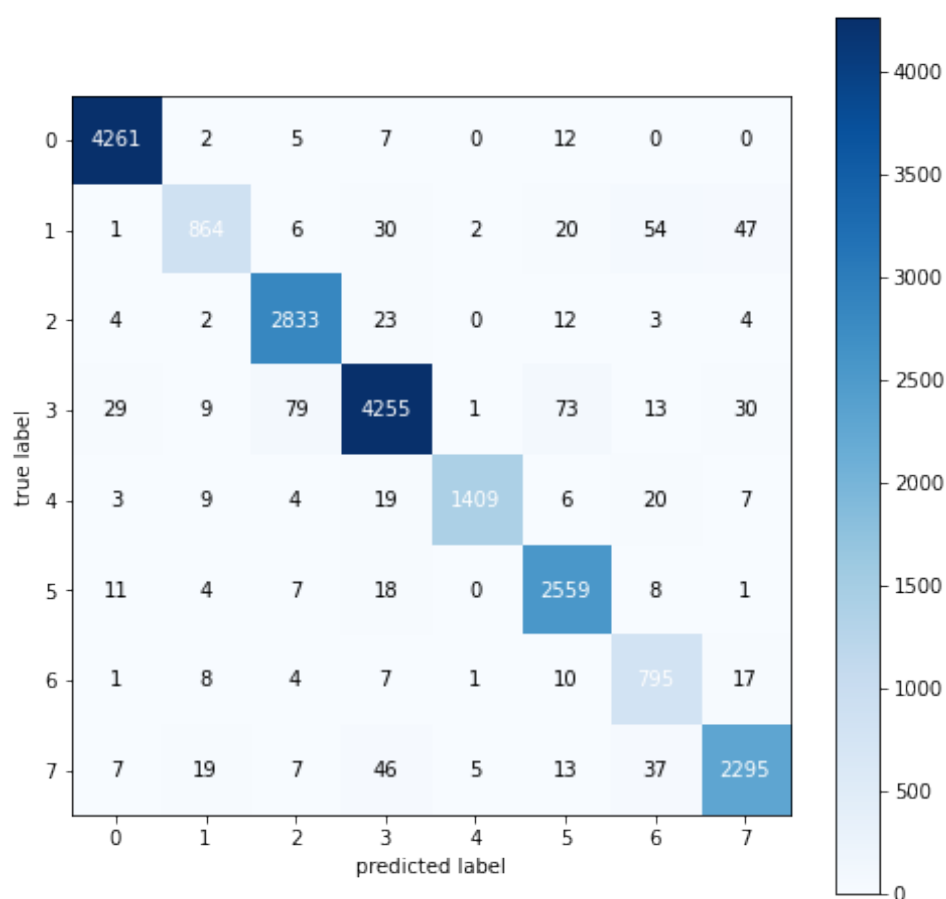


Figura 31: Matrice di confusione

Come è possibile notare dalla Tabella 20 e dalla Figura 31, il modello *BERT* si comporta in maniera più che soddisfacente. Il modello non accusa la minor quantità di dati rispetto al task precedente e nemmeno la presenza di otto classi non bilanciate.

#### 4.2.4 Risultati parte visuale

Per la parte visuale sono stati condotti due diversi test: in uno è stato applicato il meccanismo di attention, nell'altro non è stato applicato. Per entrambi, come nella parte testuale, è stata utilizzata la *10-Fold Cross Validation*. I settaggi scelti sono i seguenti:

- batch size = 32
- epoche = 10

#### Test senza attention

Label	Precision	Recall	F1-score	Accuracy
Amusement	0.25	0.24	0.25	0.1761
Anger	0.05	0.04	0.05	
Awe	0.15	0.21	0.18	
Contentment	0.24	0.24	0.24	
Disgust	0.10	0.04	0.06	
Excitement	0.14	0.16	0.15	
Fear	0.03	0.02	0.03	
Sadness	0.12	0.12	0.12	

Tabella 21: Risultati test no attention



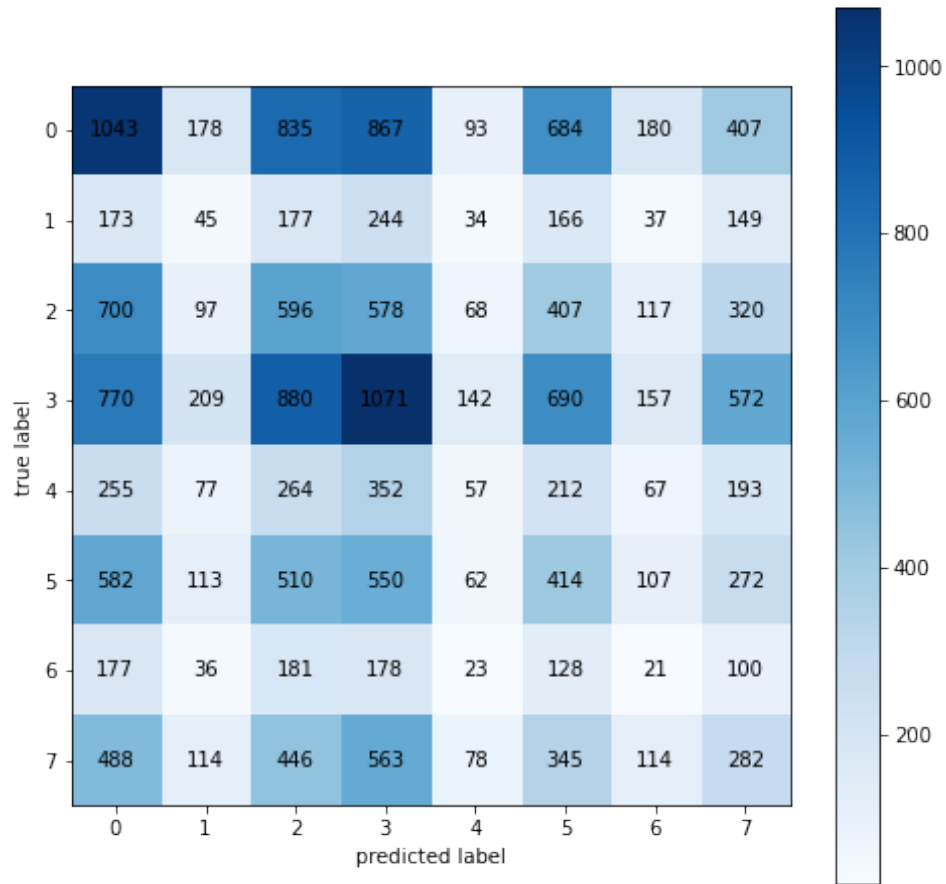


Figura 32: Matrice di confusione

Contrariamente dalla parte testuale, il modello *EfficientNet* accusa la poca quantità di dati e lo sbilanciamento delle classi.

Pur cambiando i settaggi, le performance non migliorano: aumentando le epoche il modello tende all'*over-fitting*.

Dalla Figura 32 si può notare che confonde molto le classificazioni dei dati, avendo comunque una concentrazione più alta nelle classi più popolate: la classe contentment (3) e la classe amusement (0).

**Test con attention**

Label	Precision	Recall	F1-score	Accuracy
Amusement	0.23	0.25	0.24	0.1751
Anger	0.05	0.03	0.04	
Awe	0.14	0.19	0.16	
Contentment	0.23	0.24	0.24	
Disgust	0.10	0.04	0.06	
Excitement	0.14	0.16	0.15	
Fear	0.05	0.03	0.03	
Sadness	0.13	0.12	0.13	

Tabella 22: Risultati test attention

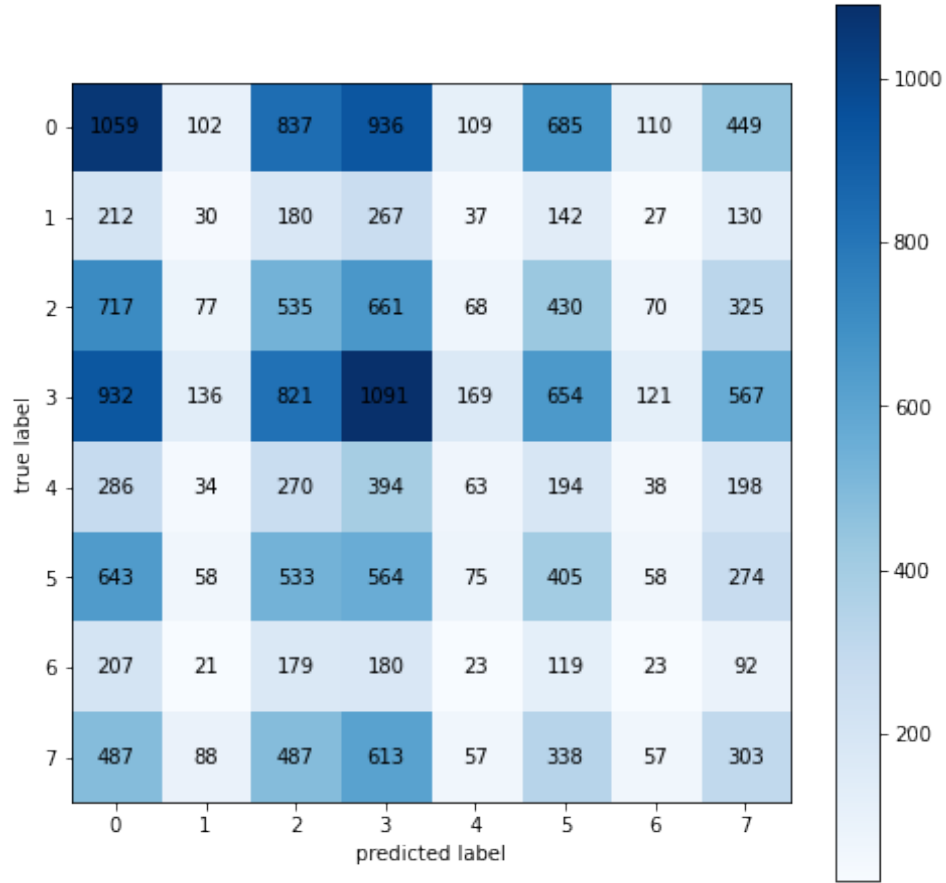


Figura 33: Matrice di confusione

Durante le prove sono stati raccolti questi risultati che non sembrano soddisfacenti. La motivazione per il quale sono stati ottenuti questi risultati è la seguente: la rete scelta, *EfficientNet*, risulta essere più complessa a livello strutturale e, a livello dei parametri, più numerosa rispetto alla rete *AlexNet* utilizzata nel lavoro di Corchs et al.[\[54\]](#) Queste informazioni, unite alla piccola numerosità del dataset, portano la rete a non apprendere bene.

La rete, fino all'ultimo strato convoluzionale, è pretrainata sul dataset *ImageNet* e i pesi imparati vengono mantenuti; ma negli strati successivi i pesi devono essere appresi e risultano essere troppi rispetto alla piccola quantità

di dati utilizzati per il training.

Inserire altri layer nascosti peggiorerebbe ulteriormente le performance. Infatti come si può notare dalla Tabella 22, applicando il meccanismo di attention il quale aumenta il numero dei parametri, è presente un peggioramento rispetto ai risultati presenti nella Tabella 21.

#### 4.2.5 Risultati late fusion

In questa sezione viene presentato l'esito finale ottenuto applicando una late fusion ai risultati provenienti dai rami visuale e testuale.

I risultati vengono divisi per applicazione o meno del meccanismo di attenzione visuale e sono stati combinati secondo due tecniche: *massimo* e *media*. Inserendo in input un dato, ogni modello predice otto probabilità: una per ogni label in cui il dato è classificabile. Di conseguenza per ogni coppia immagine-testo in input sono presenti otto coppie di probabilità.

Utilizzando il criterio del massimo, per ogni coppia di probabilità viene scelto il valore maggiore. Ottenute queste otto probabilità, viene scelto il massimo assegnando la label corrispondente alla coppia immagine-testo.

Utilizzando il criterio della media, viene trovato il valore medio per ogni coppia di probabilità. Successivamente, ottenute queste otto probabilità, viene scelto il massimo assegnando la label corrispondente alla coppia immagine-testo.

##### Test senza attention

##### Criterio decisionale - Massimo

Label	Precision	Recall	F1-score	Accuracy
Amusement	0.25	0.24	0.25	0.1760
Anger	0.05	0.04	0.05	
Awe	0.15	0.21	0.18	
Contentment	0.24	0.24	0.24	
Disgust	0.10	0.04	0.06	
Excitement	0.14	0.16	0.15	
Fear	0.03	0.02	0.03	
Sadness	0.12	0.12	0.12	

Tabella 23: Risultati late fusion no attention

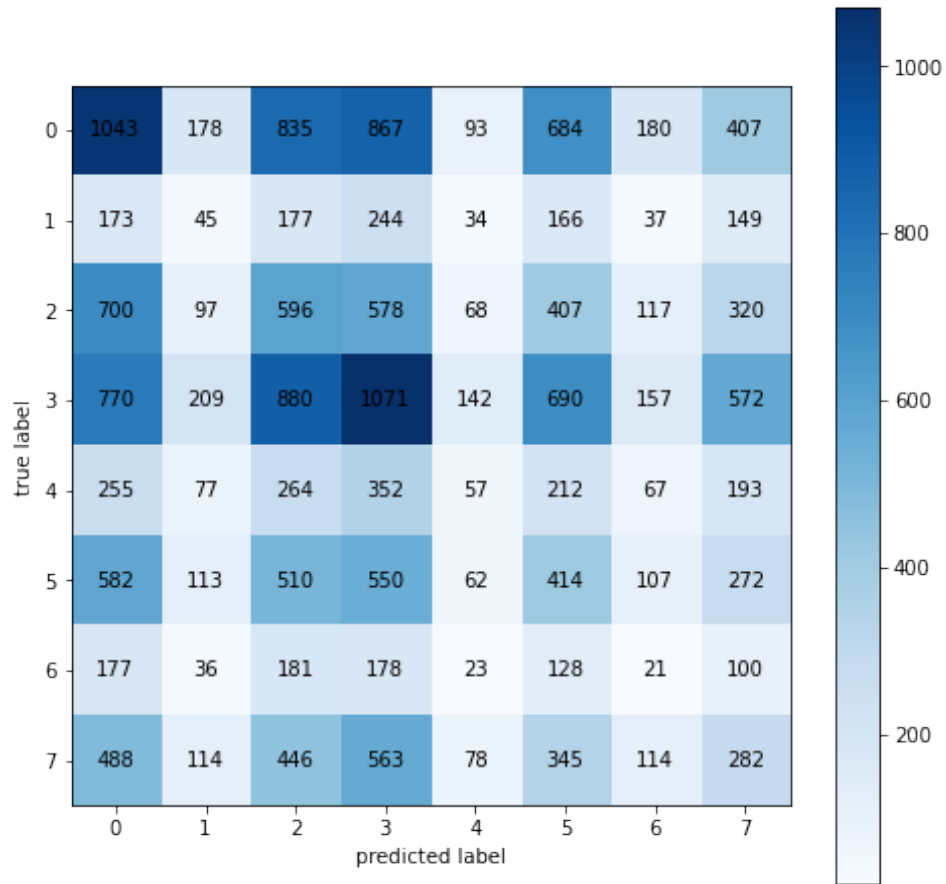


Figura 34: Matrice di confusione

Nella Tabella 23 è possibile notare che, come nel task precedente, la fusione tramite massimo viene completamente influenzata dalle performance ottenute nel modello visuale.

I risultati in Figura 34, essendo molto influenzati dal modello visuale, si concentrano nelle classi più popolose, rappresentate da un blu scuro.

**Criterio decisionale - Media**

Label	Precision	Recall	F1-score	Accuracy
Amusement	0.97	0.97	0.97	0.8864
Anger	0.83	0.72	0.77	
Awe	0.86	0.94	0.90	
Contentment	0.90	0.89	0.89	
Disgust	0.97	0.59	0.73	
Excitement	0.86	0.92	0.89	
Fear	0.69	0.85	0.76	
Sadness	0.87	0.91	0.89	

Tabella 24: Risultati late fusion no attention

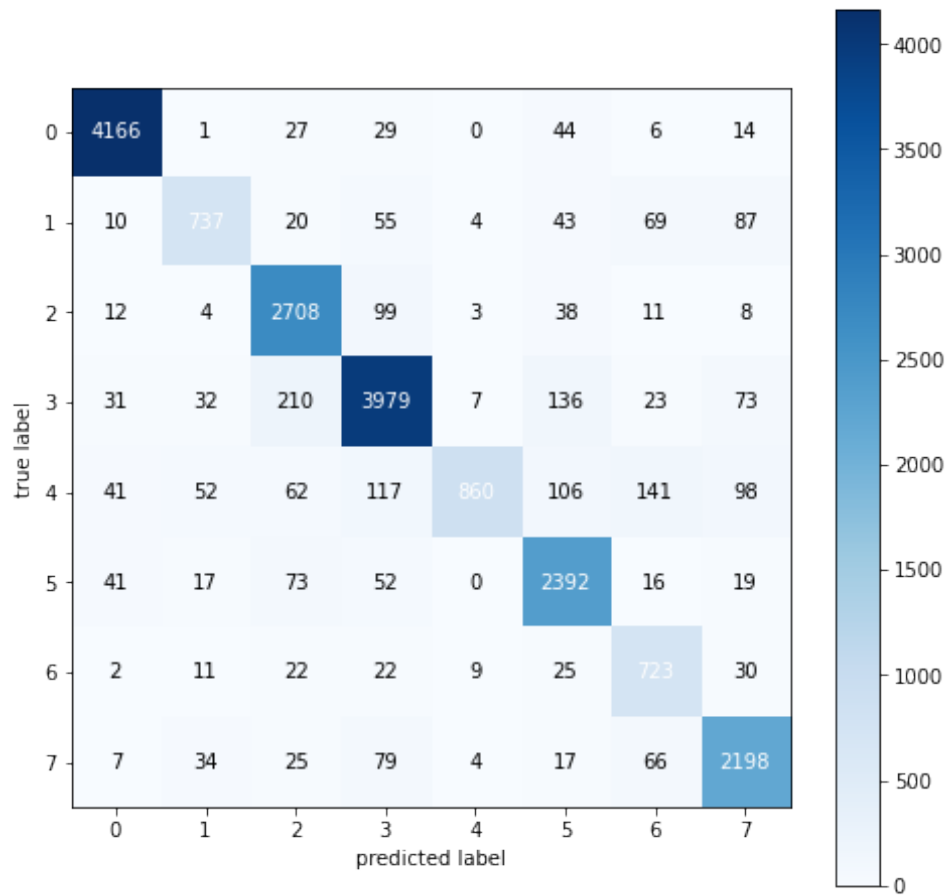


Figura 35: Matrice di confusione

La fusione tramite media risulta essere migliore: le buone performance ottenute nella parte testuale riescono a migliorare e rendere meno impattanti le performance ottenute nella parte visuale.



**Test con attention**

**Criterio decisionale - Massimo**

Label	Precision	Recall	F1-score	Accuracy
Amusement	0.23	0.25	0.24	0.1751
Anger	0.05	0.03	0.04	
Awe	0.14	0.19	0.16	
Contentment	0.29	0.24	0.24	
Disgust	0.10	0.04	0.06	
Excitement	0.14	0.16	0.15	
Fear	0.05	0.03	0.03	
Sadness	0.13	0.12	0.13	

Tabella 25: Risultati late fusion attention

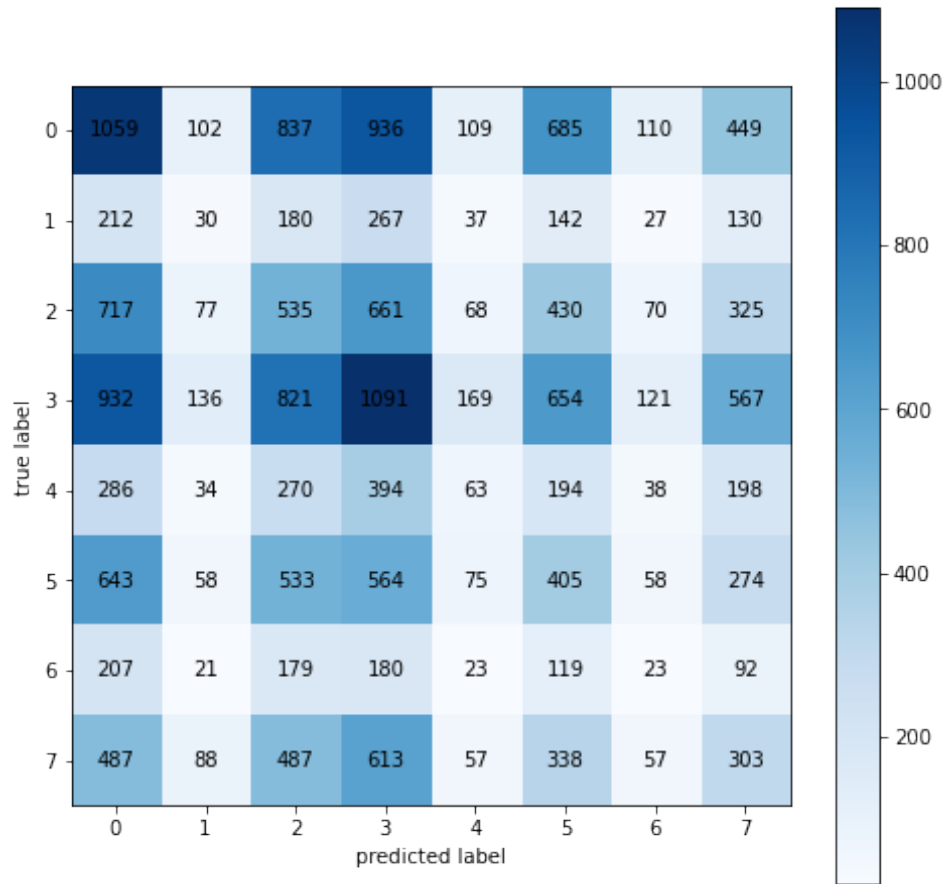


Figura 36: Matrice di confusione

Come nella presentazione dei risultati ottenuti nella parte visuale, applicando il meccanismo di attention le prestazioni peggiorano. Il metodo di fusione tramite massimo si dimostra non affidabile e troppo influenzabile dalle cattive performance del modello visuale.

Come nella Figura 34, anche nella Figura 36 è intuibile che il modello non riesca a classificare in maniera discreta.

**Criterio decisionale - Media**

Label	Precision	Recall	F1-score	Accuracy
Amusement	0.97	0.97	0.97	0.8888
Anger	0.84	0.72	0.77	
Awe	0.87	0.94	0.90	
Contentment	0.90	0.89	0.90	
Disgust	0.97	0.59	0.73	
Excitement	0.86	0.91	0.89	
Fear	0.69	0.86	0.77	
Sadness	0.86	0.91	0.89	

Tabella 26: Risultati late fusion attention

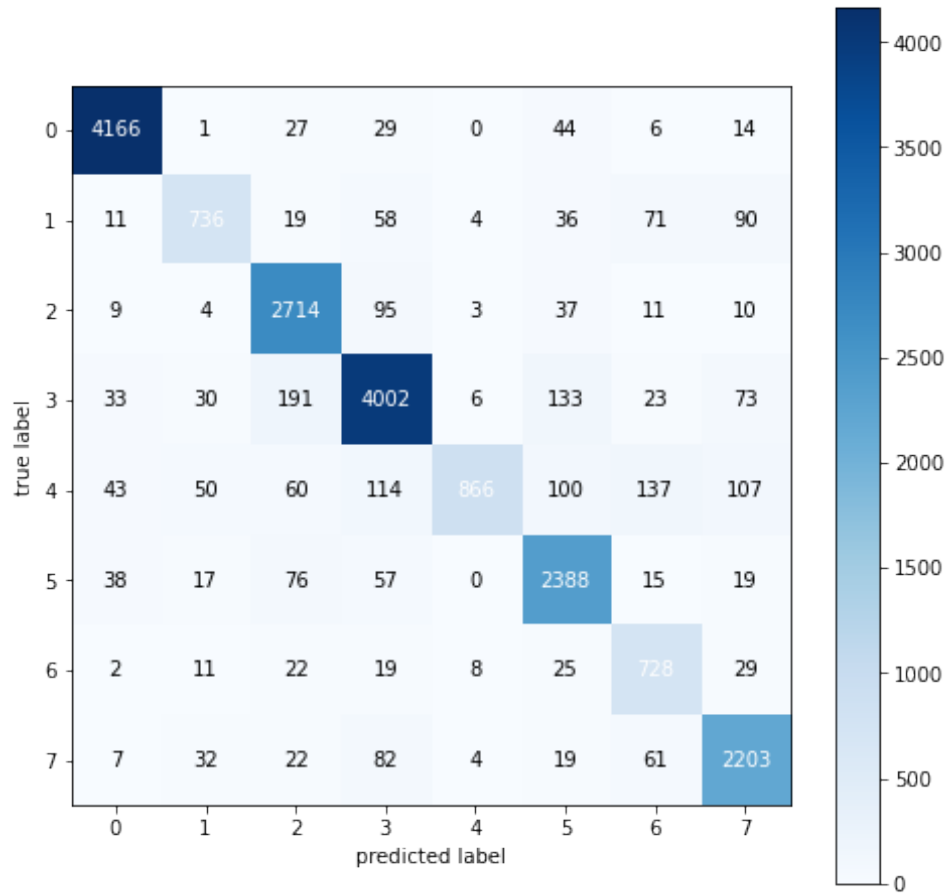


Figura 37: Matrice di confusione

In questo caso, confrontando le Tabelle 26 e 24, i risultati sono quasi simili. Il metodo di fusione tramite media risulta essere il migliore, il quale coglie l'applicazione del meccanismo di attention e bilancia le performance dei due modelli.

#### 4.2.6 Confronto risultati con baseline

In questa sezione vengono confrontati i risultati ottenuti negli esperimenti precedenti con quelli contenuti nella baseline. I risultati vengono divisi per uso del meccanismo di attention; per quanto riguarda la fusione vengono presi quelli ottenuti fondendo i risultati tramite la media, tramite la quale sono stati raccolti i risultati migliori.

	Text	Image	Multimodal
Hand-crafted	71%	51.9%	74%
Deep	71%	57.3%	76%
Attention	96.17%	17.51%	88.88%
No Attention	96.17%	17.60%	88.64%

Tabella 27: Confronto risultati

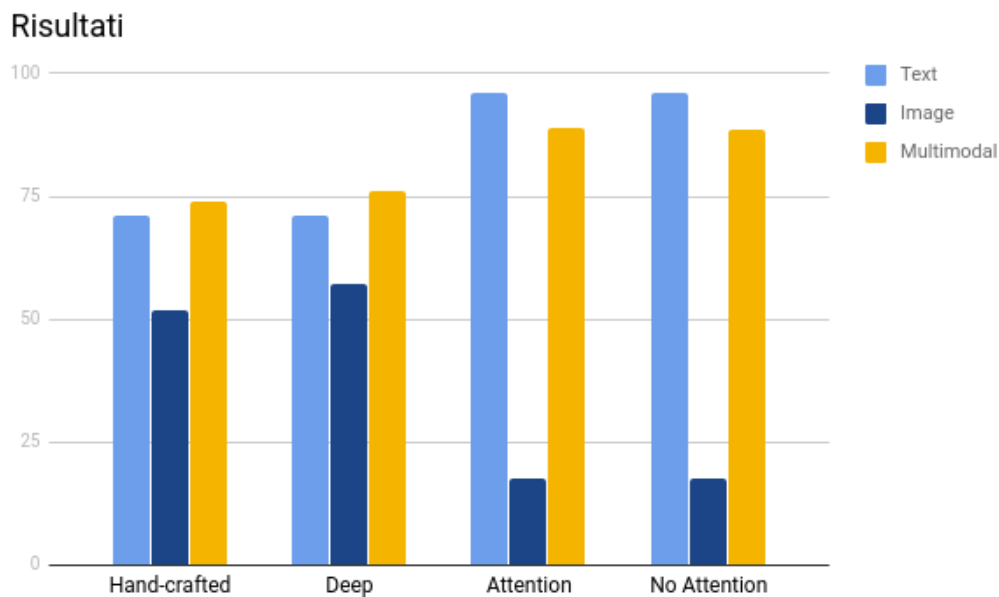


Figura 38: Confronto risultati

Confrontando i risultati ottenuti in questa tesi con i risultati ottenuti nel lavoro di Corchs et al.[54] è possibile notare che il modello testuale *BERT*

supera del 25% i migliori risultati ottenuti nella baseline.

Nella parte testuale invece il modello proposto crolla, pagando una differenza del 40% circa rispetto al risultato migliore ottenuto nella baseline tramite l'utilizzo di tecniche *deep*.

Come nel task precedente, anche in questo caso le ottime performance della parte testuale risolleivano le performance ottenute nella parte visuale, restituendo due risultati dal punto di vista multimodale che superano del 12% il miglior risultato ottenuto nella baseline.

#### 4.2.7 Analisi dell'errore

In maniera analoga all'analisi dell'errore per la sentiment classification, anche in questa sezione viene effettuata un'analisi sulle predizioni errate effettuate dal modello testuale BERT sull'intero dataset presentato nel capitolo 4.2.1. Utilizzando una *10-fold Cross Validation*, il modello raggiunge un'accuracy del 96.17%. Sul totale di 20047 frasi date in input, vengono effettuate 831 predizioni errate.

Label	Predizioni errate	% errore
Amusement	17	0.4
Anger	172	16.8
Awe	52	1.8
Contentment	300	6.7
Disgust	60	4.1
Excitement	51	2.0
Fear	49	5.8
Sadness	130	5.4

Tabella 28: Predizioni errate

Ricordando che il dataset non è bilanciato, le label in cui vengono commessi più errori risultano essere: *anger*, *contentment*, *fear*, *disgust* e *sadness*. Escludendo la label *contentment*, le altre sono le quattro label con il minor numero di dati: questo porta il modello a non apprendere bene e, di conseguenza, non riuscire a distinguere correttamente i dati.

Infatti, concentrandosi sugli errori delle quattro label meno popolose, si può notare una cosa interessante: nel 52% degli errori il modello riesce a connotare la frase come negativa, ma non riesce a classificarla in maniera corretta.

Invece nel 33% degli errori, le frasi in input vengono classificate come *contentment* o *amusement*: comportamento ipotizzabile dato che il dataset non risulta essere bilanciato e queste due label rappresentano quasi il 45% della totalità dei dati.

Come presentato nell'analisi precedente, anche in questo caso vengono analizzate le frasi e la loro composizione:

Label	N. Medio Parole	N. Medio Caratteri
Amusement	42.51	282.03
Anger	67.26	423.37
Awe	96.02	614.74
Contentment	52.29	325.14
Disgust	52.02	310.87
Excitement	108.35	671.85
Fear	87.93	573.59
Sadness	49.10	302.20
Totale	65.76	415.24

Tabella 29: Predizioni corrette

Label	N. Medio Parole	N. Medio Caratteri
Amusement	118.64	745.24
Anger	88.77	548.33
Awe	110.08	791.31
Contentment	84.88	486.76
Disgust	224.73	1338.66
Excitement	134.78	838.10
Fear	155.59	1009.89
Sadness	59.05	358.13
Totale	101.24	617.65

Tabella 30: Predizioni errate

Dalle tabelle qui sopra è facilmente intuibile che, come nell'analisi precedente, la brevità e l'inserimento di informazioni più adatte al task che si sta compiendo, portano a delle performance migliori.

Ciò che incuriosisce però è il numero medio dei caratteri nelle predizioni errate delle label *disgust* e *fear*.

Analizzando i testi classificati in maniera errata con queste due label, colpisce la presenza nelle parole con più occorrenze dei termini *mrd*, *vid* e *delany*. Esiste un motivo ed è il seguente: dalla piattaforma *Flickr* sono state raccolte immagini, con relativi testi, provenienti dall'album *Act in the Living Present - The Life of Martin Robinson Delany*. Questa storia parla del dottore Delany, il quale fu un'attivista per la lotta contro la tratta degli schiavi. Sicuramente un argomento molto negativo, ma dato che le immagini sono state fornite con lunghe descrizioni, questo ha portato troppe informazioni che il modello non è riuscito a imparare e classificare in maniera corretta, limitandolo alla comprensione della polarità negativa.

Analizzando invece la totalità dei testi classificati in maniera errata è risultato che nel 55% dei casi la descrizione contiene più informazioni sull'attrezzatura utilizzata per effettuare la foto o sulle caratteristiche tecniche delle foto stessa, rispetto ad una descrizione dell'emozione provata dall'utente.



## 5 Conclusioni

In questo capitolo viene fatto il punto della situazione, valutando e prendendo coscienza di tutto il lavoro presentato nei capitoli precedenti.

Il lavoro affrontato non è stato banale, è stata affrontata una tematica molto importante per il momento storico che si sta vivendo.

Quindi è stato deciso di partire da un lavoro che sembrava più congeniale possibile e riadattarlo. Il modello proposto di Huang et al.[50], sviluppato solamente per la sentiment classification, è risultato il più adatto per affrontare gli studi di questa tesi. Aggiornare i modelli presenti con modelli più recenti come *BERT* e *EfficientNet* è un compito riuscito a metà: per la parte testuale si è ottenuto un netto miglioramento, raggiungendo performance quasi perfette, ponendo così un punto di partenza affidabile per eventuali sviluppi futuri. Questo modello è riuscito a comportarsi egregiamente in entrambi i task, non accusando la differenza di numerosità dei dataset.

Diverso è il discorso per il modello visuale, ossia *EfficientNet*: purtroppo sono stati riscontrati diversi problemi. Nel primo task si è presentato un problema dal punto di vista dei terminali: avendo tanti dati da valutare sarebbero state necessarie macchine più potenti e con meno vincoli, quest'ultimi dettati dalla piattaforma [Colab](#) fornita da Google, la quale non permette sessioni di training superiori alle 12 ore. Per questo motivo il numero di dati di training è stato diminuito e i dataset di validation e test sono stati uniti in un unico dataset.

Nel secondo task invece si è riscontrato un problema con la poca numerosità di dati: il modello utilizzato, pur essendo pretrainato, contiene la parte finale che è composta da layers con pesi da apprendere. I dati presenti nel dataset per l'emotion classification non sono risultati sufficienti per addestrare correttamente la parte non pretrainata. Per questo motivo non si nota un miglioramento nemmeno tramite l'aggiunta del meccanismo di attenzione perché, inserendo quest'ultimo, il numero dei parametri che il modello deve imparare aumenta ancora di più.

Contrariamente, nel task di sentiment classification si è notata una differenza con l'applicazione del meccanismo di attenzione. Pur incontrando i problemi citati precedentemente, il meccanismo ha portato un miglioramento nelle performance.

Spostando invece l'attenzione sulla late fusion, si può dire che porta un miglioramento generale, rendendo meno impattanti gli errori o i problemi

presenti in una delle due pipeline. Come visto precedentemente, anche nel task di emotion classification, riesce a sopperire alle difficoltà del modello visuale, restituendo comunque delle ottime performance.

Infine, tramite le analisi degli errori, sono emerse delle somiglianze tra i due task: seppur con qualche differenza, in entrambi i casi è stata riscontrata una correlazione tra errore e presenza di contenuto impersonale nel testo. Il testo rappresentante di caratteristiche fotografiche, con un'impronta pubblicitaria o contenente ironia, porta il modello a classificare in maniera errata con un peggioramento delle performance.

## 6 Sviluppi futuri

Possibili sviluppi futuri consistono nell'implementazione di una pipeline multimodale, nella quale la pipeline testuale e la pipeline visuale vengono unite tramite early fusion prima di effettuare la classificazione.

In questo modo le pipeline diventerebbero tre, in cui una di queste sfrutterebbe le features più o meno discriminanti prima di effettuare la classificazione. In questo modo casi di incertezza verrebbero valutati prima di assegnare una label e successivamente, a livello di late fusion, valutare tre fonti: una visuale, una testuale e una testuale-visuale.

Un ulteriore step per ottenere un miglioramento delle performance sarebbe quello di arricchire il dataset utilizzato per l'emotion classification, aumentando il numero di dati e successivamente bilanciare le 8 classi presenti.

Inoltre, utilizzando il modello BERT come punto di riferimento, potrebbe essere utilizzata una diversa rete convoluzionale per la visuale. Potrebbe essere scelta una rete con una struttura diversa e meno complessa, la quale riesca ad ottenere buona performance ma con un minor utilizzo di risorse computazionali.

## Riferimenti bibliografici

- [1] K. Sikka, K. Dykstra, S. Sathyanarayana, G. Littlewort, and M. Bartlett, *Multiple kernel learning for emotion recognition in the wild*, in Proc. 16th Int. Conf. Multimodal Interaction, 2013, pp. 517–524.
- [2] S. E. Kahou, et al., *EmoNets: Multimodal deep learning approaches for emotion recognition in video*, J. Multimodal User Interfaces, vol. 10, pp. 99–111, 2015
- [3] N. Jaques, S. Taylor, A. Sano, and R. Picard, *Multi-task, multi-kernel learning for estimating individual wellbeing*, in Proc. Multimodal Mach. Learn. Workshop Conjunction NIPS, 2015, pp. 1–7.
- [4] S. S. Bucak, R. Jin, and A. K. Jain, *Multiple kernel learning for visual object recognition: A review*, IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 7, pp. 1354–1369, Jul. 2014.
- [5] Z. Z. Lan, L. Bao, S. I. Yu, W. Liu, and A. G. Hauptmann, *Multimedia classification and event detection using double fusion*, Multimedia Tools Appl., vol. 71, pp. 333–347, 2014.
- [6] G. A. Ramirez, T. Baltrusaitis, and L.-P. Morency, *Modeling latent discriminative dynamic of multi-dimensional affective signals*, in Proc. Int. Conf. Affective Comput. Intell. Interaction Workshops, 2011, pp. 396–406.
- [7] G. Castellano, L. Kessous, and G. Caridakis, *Emotion recognition through multiple modalities: Face, body gesture, speech*, Affect and Emotion Human-Computer Interaction: LNCS. Berlin, Germany: Springer Verlag, 2008.
- [8] A. Yao, D. Cai, P. Hu, S. Wang, L. Sha, and Y. Chen, *HoloNet: towards robust emotion recognition in the wild*, in ACM International Conference on Multimodal Interaction, 2016, pp. 472–478.
- [9] V. Vielzeuf, S. Pateux, and F. Jurie, *Temporal multimodal fusion for video emotion classification in the wild*, pp. 569–576, 2017.
- [10] J. Yan et al., *Multi-clue fusion for emotion recognition in the wild*, in ACM International Conference on Multimodal Interaction, 2016, pp. 458–463.

- [11] Y. Fan, X. Lu, D. Li, and Y. Liu, *Video-based emotion recognition using CNN-RNN and C3D hybrid networks*, in Proceedings of the 18th ACM International Conference on Multimodal Interaction, 2016, pp. 445-450: ACM.
- [12] M. Wöllmer, F. Weninger, T. Knaup, B. W. Schuller, C. Sun, K. Sagae, L. Morency, *Youtube movie reviews: Sentiment analysis in an audio-visual context*, IEEE Intelligent Systems 28 (3) (2013).
- [13] D. Cao, R. Ji, D. Lin, S. Li, *A cross-media public sentiment analysis system for microblog*, Multimedia Syst. 22 (4) (2016) 479-486.
- [14] S. Poria, E. Cambria, D. Hazarika, N. Majumder, A. Zadeh, L. Morency, *Context-dependent sentiment analysis in user-generated videos*, in: R. Barzilay, M. Kan (Eds.), Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers, Association for Computational Linguistics, 2017, pp.873-883.
- [15] V. Pérez-Rosas, R. Mihalcea, L. Morency, *Utterance-level multimodal sentiment analysis*, in: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 500 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers, The Association for Computer Linguistics, 2013, pp. 973-982.
- [16] S. Poria, I. Chaturvedi, E. Cambria, A. Hussain, *Convolutional MKL based multimodal emotion recognition and sentiment analysis*, in: F. Bonchi, J. Domingo-Ferrer, R. A. Baeza-Yates, Z. Zhou, 505 X. Wu (Eds.), IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain, IEEE, 2016, pp. 439-448.
- [17] K. Simonyan, A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, CoRR abs/1409.1556.
- [18] S. Chen and Q. Jin, *Multi-modal dimensional emotion recognition using recurrent neural networks*, in Proc. 5th Int. Workshop Audio/Visual Emotion Challenge, 2015, pp. 49-56.
- [19] M. Wöllmer, A. Metallinou, F. Eyben, B. Schuller, and S. Narayanan, *Context-sensitive multimodal emotion recognition from speech and facial*

- expression using bidirectional LSTM modeling*, in Proc. 15th Annu. Conf. Int. Speech Commun. Assoc., 2010, pp. 2362–2365.
- [20] S. Poria, E. Cambria, and A. F. Gelbukh, *Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis*, in Proc. Conf. Empirical Methods Nat.Lang. Process. (EMNLP), 2015, pp. 2539–2544.
- [21] M. Chen, S. Wang, P. P. Liang, T. Baltrusaitis, A. Zadeh, L. Morency, *Multimodal sentiment analysis with word-level fusion and reinforcement learning*, in: E. Lank, A. Vinciarelli, E. E. Hoggan, S. Subramanian, S. A. Brewster (Eds.), Proceedings of the 19th ACM International Conference on 510 Multimodal Interaction, ICMI 2017, Glasgow, United Kingdom, November 13 - 17, 2017, ACM, 2017, pp. 163–171.
- [22] Q. You, J. Luo, H. Jin, J. Yang, *Cross-modality consistent regression for joint visual-textual sentiment analysis of social multimedia*, in: P. N. Bennett, V. Josifovski, J. Neville, F. Radlinski (Eds.), Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, San Francisco, CA, USA, February 22-25, 2016, ACM, 2016, pp. 13–22.
- [23] C. Fang, H. Jin, J. Yang, and Z. Lin, *Collaborative feature learning from social media*, in Proc. IEEE Conf. Comput. Vis. Pattern Recognit.(CVPR), Boston, MA, USA, Jun. 2015, pp. 577–585.
- [24] B. Perozzi, R. Al-Rfou, and S. Skiena, *DeepWalk: Online learning of social representations*, in Proc. ACM SIGKDD, 2014, pp. 701–710.
- [25] J. Tang et al., *LINE: Large-scale information network embedding*, in Proc. 24th Int. Conf. World Wide Web (WWW), 2015, pp. 1067–1077.
- [26] S. Chang et al., *Heterogeneous network embedding via deep architectures*, in Proc. 21st ACM SIGKDD Int. Conf. Knowl. Disc. Data Min., 2015, pp. 119–128.
- [27] S. Liu, P. Cui, W. Zhu, S. Yang, and Q. Tian, *Social embedding image distance learning*, in Proc. ACM Int. Conf. Multimedia, MM, Orlando, FL, USA, Nov. 2014, pp. 617–626.

- [28] H. Zhang, X. Shang, H. Luan, M. Wang, and T. Chua, *Learning from collective intelligence: Feature learning using social images and tags*, ACM Trans. Multimedia Comput. Commun. Appl., vol. 13, no. 1, p. 1, 2016.
- [29] Q. You, L. Cao, H. Jin, J. Luo, *Robust visual-textual sentiment analysis: When attention meets tree-structured recursive neural networks*, in: A. Hanjalic, C. Snoek, M. Worring, D. C. A. Bulterman, B. Huet, A. Kelliher, Y. Kompatsiaris, J. Li (Eds.), Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016, ACM, 2016, 520, pp. 1008–1017.
- [30] T. Baltrusaitis, N. Banda, and P. Robinson, *Dimensional Affect recognition using continuous conditional random fields*, in Proc. 10th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit., 2013, pp. 1–8.
- [31] Y. Cao, M. Long, J. Wang, Q. Yang, and P. S. Yu, *Deep visual-semantic hashing for cross-modal retrieval*, in Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2016, pp. 1445–1454.
- [32] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, *Multimodal deep learning*, in Proc. 29th Int. Conf. Mach. Learn., 2011, pp. 689–696.
- [33] S. S. Rajagopalan, L.-P. Morency, T. Baltrusaitis, and R. Goecke, *Extending long short-term memory for multi-view structured learning*, in Proc. Eur. Conf. Comput. Vis., 2016, pp. 338–353.
- [34] C. Silberer and M. Lapata, *Learning grounded meaning representations with autoencoders*, in Proc. Annu. Meet. Assoc. Comput. Linguistics, 2014, pp. 721–732.
- [35] N. Srivastava and R. R. Salakhutdinov, *Multimodal learning with deep Boltzmann machines*, in Proc. 28th Int. Conf. Neural Inf. Process. Syst., 2012, pp. 2949–2980.
- [36] D. Wang, P. Cui, M. Ou, and W. Zhu, *Learning compact hash codes for multimodal representations using orthogonal deep structure*, IEEE Trans. Multimedia, vol. 17, no. 9, pp. 1404–1416, Sep. 2015.
- [37] J. Masci, M. M. Bronstein, A. M. Bronstein, and J. Schmidhuber, *Multimodal similarity-preserving hashing*, IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 4, pp. 824–830, Apr. 2014.

- [38] H.-L. Suk, S.-W. Lee, and D. Shen, *Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis*, NeuroImage, vol. 101, pp. 569–582, Nov. 2014.
- [39] F. Huang et al., *Learning social image embedding with deep multimodal attention networks*, in Proc. Thematic Workshops ACM Multimedia, Mountain View, CA, USA, Oct. 2017, pp. 460–468.
- [40] A. Frome, G. Corrado, and J. Shlens, *DeViSE: A deep visual-semantic embedding model*, in Proc. 28th Int. Conf. Neural Inf. Process. Syst., 2013, pp. 2121–2129.
- [41] R. Kiros, R. Salakhutdinov, and R. S. Zemel, *Unifying visual-semantic embeddings with multimodal neural language models*, Trans. Assoc. Comput. Linguistics, pp. 1–13, 2015.
- [42] D. Zhang and W.-J. Li, *Large-scale supervised multimodal hashing with semantic correlation maximization*, in Proc. 26th AAAI Conf. Artif. Intell., 2014, pp. 2177–2183.
- [43] Y. Peng, X. Huang, and J. Qi, *Cross-media shared representation by hierarchical learning with multiple deep networks*, in Proc. IJCAI, 2016, pp. 3846–3853.
- [44] D. R. Hardoon, S. Szedmák, and J. Shawe-Taylor, *Canonical correlation analysis: An overview with application to learning methods*, Neural Comput., vol. 16, no. 12, pp. 2639–2664, 2004.
- [45] G. Andrew, R. Arora, J. A. Bilmes, and K. Livescu, *Deep canonical correlation analysis*, in Proc. ICML JMLR Workshop Conf., vol. 28, 2013, pp. 1247–1255.
- [46] F. Yan and K. Mikolajczyk, *Deep correlation for matching images and text*, in Proc. IEEE CVPR, 2015, pp. 3441–3450.
- [47] J. Weston, S. Bengio, and N. Usunier, *IE: scaling up to large vocabulary image annotation*, in Proc. IJCAI, 2011, pp. 2764–2770.
- [48] L. Wang, Y. Li, and S. Lazebnik, *Learning deep structure-preserving image-text embeddings*, in Proc. CVPR, 2016, pp. 5005–5013.

- [49] I. Vendrov, R. Kiros, S. Fidler, and R. Urtasun, *Order-embeddings of images and language*, in Proc. Int. Conf. Learn. Representations, 2016.
- [50] F. Huang, X. Zhang, Z. Zhao, J. Xu and Z. Li, *Image-text sentiment analysis via deep multimodal attentive fusion*, in Knowl.Based Syst., vol. 167, 2019, pp. 26–37.
- [51] M. Tan, Quoc V. Le, *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*, arXiv:1905.11946, 2019.
- [52] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, arXiv:1810.04805, 2019.
- [53] L. Vadicamo, F. Carrara, A. Cimino, S. Cresci, F. Dell’Orletta, F. Falchi and M. Tesconi, *Cross-Media Learning for Image Sentiment Analysis in the Wild*, in IEEE International Conference on Computer Vision Workshops (ICCVW), 2017, pp. 308-317.
- [54] S. Corchs, E. Fersini, and F. Gasparini, *Ensemble learning on visual and textual data for social image emotion classification*, 2017.
- [55] K. He, X. Zhang, S. Ren and J. Sun, *Deep residual learning for image recognition*, CVPR, 2016, pp. 770-317.
- [56] G. Huang, Z. Liu, L. Van Der Maaten and K.Q. Weinberger, *Densely connected convolutional networks*, CVPR, 2017.
- [57] J. Lu, C. Xiong, D. Parikh and R. Socher, *Knowing When to Look: Adaptive attention via a visual sentinel for image captioning*, IEEE Computer Society, 2017, pp. 3242–3250.