UNIVERSITÀ
DEGLI STUDI
FIRENZE

PhD Program in Smart Computing
Dipartimento di Ingegneria dell'Informazione (DINFO)

Constrained Affective Computing

Lisa Graziani

Dissertation presented in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Smart Computing

PhD Program in Smart Computing
University of Florence, University of Pisa, University of Siena

Constrained Affective Computing

Lisa Graziani

Advisor:

Prof. Marco Gori

Head of the PhD Program:

Prof. Paolo Frasconi

Evaluation Committee:
Prof. Bart De Moor, *Katholieke Universiteit Leuven*
Prof. Walter Kropatsch, *Technology University Wien*

ciclo XXXIII - December 2020

To all those who make me feel good

Acknowledgments

Before I started my PhD I did not know anything about Artificial Intelligence and Machine Learning, and I have to say that it has been a nice discovery. In these three years I have learnt a lot from this experience, but I am only at the tip of the iceberg.

First of all, I would like to thank my advisor Marco Gori for introducing me to the world of artificial intelligence, for giving me confidence, and for the insightful comments. I would like to express my sincere thanks to Stefano Melacci, to whom I owe a large part of my work. He helped me and he taught me a lot about machine learning and the research world. I would like to thank also the other professors of the SAILab, Michelangelo Diligenti, Monica Bianchini, Marco Maggini, and Franco Scarselli, for being always kind and helpful.

I would like to thank the members of the supervisory committee, Oswald Lanz and Roberto Tagliaferri, for their time and for all the useful suggestions. I would like to thank also the members of the evaluation committee, Bart De Moor and Walter Kropatsch, for accepting to be my reviewers, for their time and for their precious comments.

Thanks to Francesco for his help, for always explaining me ideas clearly, and for dispensing me trash music, so making me feel happy. Although we have not worked together, I want to thank Giorgia for the good times we had. She is not only a "trash companion", but I think she is also a good friend. Thanks to my lab partners Andrea and Matteo, with whom I have been on the same page many times, for these three years spent together and for always helping me. Despite we know each other for a short time, I would like to thank Giovanna for showing immediately her kindness and help. I want to thank Enrico for helping me when I needed. Even though I had not time to get to know him, I think he is a very good person. Thanks to the other colleagues, who I cannot acknowledge one by one because they are too many. From each of them I learnt something, for better or worse.

I would like to thank Claudio Saccà, Maurizio Masini, and Oronzo Parlangeli, for giving me the opportunity to participate in the project about speech.

A big thank you goes to my family for always supporting me and especially for bearing me, and in particular to my little brother, who is a big man for me.

Thanks to my train travelling companions, Antonella, Maria Isabel, Fabrizio, and Lara, for the good times together, for the takeaway aperitifs, and for the precious advises. Thanks to my ex colleagues in Mathematics, Anna and Chiara, who are still in department to keep me company sometimes.

Thanks to my best friends, Fabiola, Roberta, Deborah, Alessia, Agnese, Samuele, and Manuela, for being part of my life for many years. Thanks to my friends in Torrita, and in particular to Erica for the nice trips that we did together. Thanks to my soccer team, with which I share the moments of fun. I want to thank also my ex

work colleagues, Silvia, Enrico, and Leonardo, for teaching me a lot about the world of work.

Finally, for the first time, I want to thank myself, for getting involved in a new field, for never surrendering even in difficult moments, and for still being myself (even if it is not always the right thing).

Abstract

Emotions have an important role in daily life, influence decision-making, human interaction, perception, attention, self-regulation. They have been studied since ancient times, philosophers have been always interested in analyzing human nature and bodily sensations, psychologists in studying the physical and psychological changes that influence thought and behavior. In the early 1970s, the psychologist Paul Ekman defined six universal emotions, namely anger, disgust, fear, happiness, sadness, and surprise. This categorization has been taken into account for several studies. In the late 1990s, Affective Computing was born, a new discipline spanning between computer science, psychology, and cognitive science. Affective Computing aims at developing intelligent systems able to recognize, interpret, process, and simulate human emotions. It has a wide range of applications, as healthcare, education, games, entertainment, marketing, automated driver assistance, robotics, and many others. Emotions can be detected from different channels, such as facial expressions, body gestures, speech, text, physiological signals. In order to enrich human-machine interaction, the machine should be able to perform tasks similar to humans, such as recognizing facial expressions, detecting emotions from what it is said (text) and from how it said (audio), and it should be able also to express its own emotions.

With the great success of deep learning, deep architectures have been employed also for many Affective Computing tasks. In this thesis, thinking about an emotional and intelligent agent, a detailed study of emotions has been carried out using deep learning techniques for various tasks, such as facial expression recognition, text and speech emotion recognition, and facial expression generation. Nevertheless, deep learning methods to properly perform in general require a great computing power and large collections of labeled data. To overcome these limitations we exploit the framework of Learning from Constraints, which needs few supervised data and enables to exploit a great quantity of unsupervised data, which are easier to collect. Furthermore, such approach integrates low-level tasks processing sensorial data and reasoning using higher-level semantic knowledge, so allowing machines to behave in an intelligent way in real complex environments. These conditions are reached requiring the satisfaction of a set of constraints during the learning process. In this way a task is translated into a constrained satisfaction problem. In our case, considering that knowledge could not be always perfect, the constraints are softly injected into the learning problem, so allowing some slight violations for some inputs.

In this work different constraints have been employed in order to exploit knowledge that we have on the problem. In facial expression recognition, a predictor that detects emotions from the full face is enforced by three coherence constraints. One exploits the temporal sequence of the expression, another relates different face sub-parts (eyes, nose, mouth, eyebrows, jaw), and

the last relates two feature representations. In text emotion recognition First Order Logic (FOL)-based constraints are used to exploit a great quantity of unlabeled data and data labeled with Facebook reactions. In facial expression generation cyclic-consistency FOL constraints are employed to translate a neutral face into a specific expression, and other logical rules are used to decide what emotion to generate putting together inputs coming from different channels. Finally, some logical constraints are proposed to develop a system that recognizes emotion from speech, and we built an Italian dataset that might be helpful to implement such model.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivations

Emotions play a key role in our daily life, as matter of fact, affect decision-making, learning, communication, self-regularization. We interact with other people communicating our psychological and physiological reactions and we try to comprehend our psychophysiological changes to adapt ourselves to the external environment. Each emotion has a crucial role in everyday life, for instance, fear can help to avoid a dangerous situation, in the long term anger could bring to stress conditions, or sadness could lead to depression. Emotions are studied across several fields including computer science. Over the past two decades, researchers have been attempting to build machines able to recognize, interpret, process, and simulate human affects. All these studies are enclosed in a specific discipline, called Affective Computing (see Section 2.5). Affective Computing has a wide range of applications, such as healthcare, education, marketing, entertainment, automated driver assistance, and many others. Systems handling emotions can be used to help people with Autism Spectrum Disorder, to identify causes of stress or depression, to enhance students' interest in distant learning, or to suggest new products in e-commerce, to name a few. Emotions can be detected from different channels, such as facial expressions, body movements, text, speech, and physiological signals. In order to enrich interaction between human and machine, we need to build an agent who has ability similar to humans. For this reason the machine should be able to recognize facial expressions from the interlocutor, to detect emotions from what it said (text) and from how it is said (speech), and to express its emotions (generating facial expressions for instance). Machine Learning and Deep Learning (Section 2.1) are suitable to address Affective Computing problems, since they allow machines to handle something not specifically programmed, as emotions.

Deep learning methods, that exploit deep architectures and large datasets, are yielding state-of-the-art results in several Affective Computing tasks [116]. In emo-

tion recognition from text, deep learning techniques can go over the single word that expresses an emotion and understand the context, facing the language ambiguity. In facial expression recognition deep learning methods exploit large quantity of data to try to learn in different conditions, such as pose or illumination variations and in presence of occlusions. The other advantage of deep learning approaches is that are more straightforward, since feature representations and classification models are developed at the same time. For instance, in speech emotion recognition, we do not need to extract handcrafted acoustic features, but we can directly provide to deep networks the input signal, or in facial expression recognition we can directly provide to the network the face image. Deep networks are not only used for classification tasks, but also for generation tasks (Section 2.1), such as facial expression generation.

In general, deep learning approaches for Affective Computing tasks are fully supervised. However, a supervised deep learning algorithm generally achieves acceptable performance with thousands of labeled examples per category. For real world applications, many times finding a large collection of labeled data is expensive or even unachievable. For this reason, we need approaches that use few supervised data and that can exploit also unsupervised data. Furthermore, emotions are strictly connected with the external environment, therefore we need to construct heavily structured learning environments, a feature that has been mostly neglected in deep learning approaches. The latter are still mostly seen as black-boxes, whereas we want to build machines that behave in an intelligent way in real complex environments. As a result, we need an approach that integrates logical reasoning and deep learning, in order to inject prior knowledge into the problem.

1.2 Contributions

In this thesis, deep learning techniques are employed to study several tasks about emotions, considering the six universal emotions defined by Ekman, namely anger, disgust, fear, happiness, sadness and surprise. We regard categorical approaches, namely representing emotions as discrete categories (see Appendix A.1 for more details). In particular, we work on text predicting emotions and Facebook reactions, on images and videos recognizing facial expressions. Moreover with images we generate facial expressions. On speech a preliminary study for emotion recognition is performed, constructing a dataset containing audio labeled with emotions.

In order to inject into the learning prior knowledge that we have on the problem, we follow the framework of Learning from Constraints (Section 2.4) in solving the aforementioned tasks. Such approach integrates low-level tasks processing sensorial data and reasoning using higher-level semantic knowledge, so allowing deep neural networks to learn in a way more similar to humans. Knowledge is expressed

as a collection of constraints, that are exploited during the training minimizing a loss function containing them. In this way a learning process can be thought as a *constrained* satisfaction problem.

Moreover Learning from Constraints does not require to use a great quantity of supervised data, that is a deep learning limitation (as seen in Section 1.1), and exploits also unlabeled data that are used to enforce the constraints.

The constraints that we employ might not model a perfect knowledge or might not be valid for all the cases. This is the reason why we prefer soft constraints, namely that should not be perfectly satisfied, so allowing some slight violations of them for some inputs. The soft optimization is typically obtained with penalty functions, that are minimized participating to the overall cost function of the problem. The penalty functions are weighted by a scalar greater than zero, that gives less or more importance to each constraint.

We mainly consider constraints based on First Order Logic (Section 2.2) in order to define a more formal representation of knowledge. These logic formulas to be inserted in the learning problem are converted into real-valued functions using fuzzy logic (Section 2.3).

The contributions of this thesis may be summarized as follows.

1. The task of facial expression recognition is addressed with a Convolutional Neural Network (CNN)-based model, analyzing and exploiting face sub-parts (mouth, nose, eyebrows, eyes, jaw). Three constraints, which describe three different types of coherence, are inserted in the learning problem. The first is a temporal coherence that, in presence of video sequence, enforces the predictions to smoothly change over time. The second is a constraint among face parts, that enforces the prediction on full face to be coherent with the predictions on the other sub-parts, so that we can grasp more fine-grained information not easily caught from the entire face. The latter is a coherence between two categories of representations. This approach allows us to exploit unsupervised data in addition to few labeled data, to study face sub-parts and to detect emotions in presence of occlusions.

2. On text we present a model that jointly learns to predict Facebook reactions and to detect emotions. Logical constraints are used to express how reactions are connected to emotions and vice-versa. This approach enables to train an emotion classifier even with few textual data labeled with emotions. Generating artificial labels, i.e., defining a fixed mapping between emotions and reactions and augmenting the training data with these new labels, is a rigid and sometimes ambiguous conversion. For this reason, it seems reasonable to use soft constraints to define relationships between reactions and emotions.

3. Facial expression generation is addressed in a more articulate way, building an application able to generate the expression that a given person would do reading a text. The way the system alters the input face is the outcome of a decision process that involves the information extracted from the provided text, from the input face itself, or from other sources of knowledge. Such information is merged using logical rules, which establish what expression to generate. These rules are not integrated into the learning, because we have no supervised data from which we can learn how to mix the information. The model employed to translate the neutral face into the final expression, based on Generative Adversarial Networks (GANs), instead follows the theory of Learning from Constraints. First Order Logic is exploited to describe cyclic consistency and the classic conditions to train GANs (see Section 2.1).

4. We introduce the problem of speech emotion recognition, and we present an Italian emotional speech dataset that we built extracting clips from movies. This corpus has been constructed with the aim of developing in the future a model of speech emotion recognition. Some ideas to face the problem exploiting constraints will be suggested.

1.3 Structure of the Thesis

The thesis is organized as follows. Chapter 2 presents the theories and the models that will be employed in the following chapters. In particular, Section 2.1 introduces Machine Learning and Deep Learning and gives an overview of the main neural network architectures. Furthermore it presents the generative models that will be employed to generate facial expressions. Section 2.2 introduces Classical Logic, and in particular First Order Logic, while Section 2.3 presents Fuzzy Logic, that is exploited to integrate constraints based on First Order Logic formulas into the learning process. Section 2.4 explains the theory of Learning from Constraints, that allow us to integrate prior knowledge into the learning. Section 2.5 presents the relatively new discipline Affective Computing, discussing the several applications and the different channels of emotion detection and expression.

Chapter 3 addresses the problem of facial expression recognition, presenting the model developed during my PhD. In particular, we describe the feature representations, the model structure and the coherence constraints employed to enforce the predictors. We report several results, considering predictions both on single frames and on video sequences, on full face and on face sub-parts, and with and without constraints. We study also the expression detection in presence of occlusions.

Chapter 4 deals with the task of text emotion recognition, proposing a model that jointly learns to detect emotions and to predict Facebook reactions. We describe the

model structure, the data organization, and the logic rules used during the training to enforce the predictors of reactions and emotions. We report the results on different datasets for the tasks of emotion detection and of Facebook reaction prediction, with several configurations, and with and without constraints.

Chapter 5 proposes a model which generates the facial expression that a certain person would make reading a text. We describe the process of information extraction from the inputs, the logic rules employed to decide what expression to generate, and the generative model used to transform the neutral input face. We report quantitative results for the sub-models used to extract information from the inputs and qualitative results for the generative model, and we explain how the web application based on the full model works.

Chapter 6 introduces the problem of speech emotion recognition, and presents an Italian Speech Emotional dataset built to develop, in the future, a system that detects emotions in real applications. For the moment, we have exploited an existing deep approach to evaluate the quality of such dataset.

At the beginning of Chapters 3-6, we report the related works of the tasks presented in such chapters. In Chapter 3, 4 and 6 we also provide an overview of the existing datasets that are about the specific tasks that we have addressed.

Chapter 7 outlines a summary of the given contributions and proposes some possible directions for future work.

Finally, Appendix A gives an overview on the main theories about emotions, such as the Ekman's study about the six universal categories, and on the different approaches of emotion classification.

Chapter 2

Background

2.1 Machine Learning & Deep Learning

Artificial Intelligence (AI) is a successful and very timely field with several applications and research topics. *Artificial Intelligence* refers to the ability of machines to perform tasks similar to what a human brain would do, such as learning, reasoning, understanding and elaborating language (Natural Language Processing), and processing visual data (Computer Vision). Machine Learning is a branch of Artificial Intelligence where a computer learns something from given examples in an autonomous way, without receiving precise instructions as it happens in classical programming, and then it uses what has learned to make predictions on new examples. Machine learning has numerous applications, as object recognition, face recognition, video surveillance, fraud detection, recommendation for products and services, sentiment analysis, language translation, search engine, virtual personal assistant, robot control, self driving vehicles, email spam and malware filtering, speech recognition, medical diagnosis, and many others.

In the following, we only introduce some basic techniques of machine learning. For a more comprehensive coverage of the fundamentals we suggest [11] and [49], and [46] for Deep Learning. Deep learning [46] is a subfield of machine learning that uses deep architectures to learn complex functions. Over the past few years, it has been tremendously growing, due in part to more powerful computers and larger datasets.

Machine learning tasks are usually described in terms of how the machine learning system should process an example [46]. Many kinds of tasks can be solved with machine learning, and in the following some of the most common are reported:

- **Classification:** In a classification task it is asked to specify which of k categories some input belongs to, namely it is asked to learn a function $f : \mathbb{R}^n \to \{1, ..., k\}$. A simple example is the system of anti-spam filtering, which classifies the

email in spam or not-spam. Another example is object recognition, where the input is an image and the output is an index identifying the object in the image.

- **Regression**: In this type of task, the learning algorithm is asked to predict a numerical value given some input, namely it is asked to output a function $f : \mathbb{R}^n \to \mathbb{R}$. An example of a regression task is the prediction of the expected claim amount that an insured person will make, or the prediction of future prices of houses.

- **Clustering:** In clustering, data are divided into groups, but unlike what happens in classification, classes are not known. Similar objects are grouped in the same group according to a certain measure distance or a statistical distribution. Clustering techniques may apply in several fields, such as biology, marketing, social science, social network analysis, and so on.

- **Anomaly detection:** In this type of task, the learning algorithm analyzes a set of events or objects, and flags some of them as being unusual or atypical. A wide range of techniques are used to discover outliers from a certain probability distribution. An example of an anomaly detection task is credit card fraud detection.

- **Synthesis and sampling:** In this type of task, the machine learning algorithm is asked to generate new examples that are similar to those in the training dataset (see the subsection "Generative Models" below).

- **Missing features:** In this type of task, the learning algorithm is given a new example $x \in \mathbb{R}^n$, but with some components x_i of x missing. The goal is providing a prediction of the values of the missing entries.

- **Density estimation:** In the density estimation task, the learning algorithm is asked to learn a function $p_{model} : \mathbb{R}^n \to \mathbb{R}$, where $p_{model}(x)$ is a probability density function (if x is continuous) or a probability mass function (if x is discrete) on the space that the examples were drawn from.

Learning can be divided into supervised, unsupervised, reinforced and semi-supervised learning. In *supervised learning* the data x and the label y are provided, and the goal is to learn a function to map x into y. Examples of supervised tasks are classification, regression, object detection, semantic segmentation, image captioning. Some methods widely used in supervised learning are Support Vector Machines (SVMs) [12, 24], k-nearest neighbors, decision tree [14], Naive Bayes, Artificial Neural Networks (ANNs). In *unsupervised learning* all the examples are not labeled and the aim is to identify some hidden common structures of the data. Examples of unsupervised tasks are clustering, dimensionality reduction, feature

learning, density estimation. Some common machine learning methods used in unsupervised learning are Principal Component Analysis (PCA), k-means clustering, autoencoders. Halfway between supervised and unsupervised learning there is *semi-supervised learning*, where only a subset of training data is labeled while the remaining are unsupervised. This approach enables to reduce the cost associated with the labeling process. In *Reinforcement Learning* [131] the system (agent) interacts with a dynamic environment in which it tries to reach an objective, having a reward for each correct action that it performs. Reinforcement learning can be used, for instance, to teach a vehicle to drive by itself or an agent to play some games against an adversary.

One of the main machine learning models are Artificial Neural Networks (ANNs), which take inspiration from the biological neural networks in the human brains. An ANN consists of some layers composed by a groups of nodes (neurons) connected by weighted edges which constitute the knowledge of the network. The first and simplest type of artificial neural network devised was the Feedforward Neural Network (FNNs), so called because data are forward processed, from the input layer to the output layer eventually passing through several hidden layers. In a FNN connections between the nodes do not form a cycle. It has been demonstrated that FNNs can approximate every continuous function [26].

One of the main deep architectures are the Convolutional Neural Networks (CNNs) [72, 73], that are inspired by the mammalian visual cortex, so they are mainly applied to image data. A typical block of a CNN is composed by three stages [46]: in the first stage, a convolutional layer performs several convolutions in parallel to produce linear activations. In the second stage, each linear activation is provided to a non-linear activation function, and in the third stage a pooling function is applied. Pooling function replaces the output of the network at a certain location with a summary statistic of the nearby outputs. For instance, the max pooling operation reports the maximum output within a rectangular neighbourhood. Pooling allows the representation to be invariant to small translations of the input, and this is very useful in object recognition, for instance.

Another among the most important deep architectures are the Recurrent Neural Networks (RNNs) [117], that are ideal for capturing temporal dynamics and processing sequential data of arbitrary length. For this reason, they are applicable to several tasks, such as language modeling, speech recognition, time series prediction, etc. Bidirectional Recurrent Neural Networks [125] (BRNNs) combine an RNN that moves forward through time, beginning from the start of the sequence, with another RNN that moves backward through time beginning from the end of the sequence. Hochreiter and Schmidhuber [59] introduced the Long Short-Term Memory (LSTM), that is able to learn long-term dependencies, by providing gate mechanisms to add and forget information selectively. They are widely used in

sequence-to-sequence learning, as in machine translation [8, 130].

Generative Models

Generative models generate new data instances, that can be images, videos, text, audio, while discriminative models discriminate between different kinds of data instances. More formally, given a set of data instances X and a set of labels Y, generative models try to capture the joint probability $p(X, Y)$ (or just $p(X)$ if there are no labels), whereas discriminative models capture the conditional probability $p(Y|X)$.

Two of the most common generative models are Variational Autoencoders (VAEs) [67] and Generative Adversarial Networks (GANs) [47]. A VAE is a particular autoencoder, i.e., an encoder-decoder architecture that learns to copy its input to its output. A simple autoencoder considers deterministic encoder and decoder, while a VAE considers a probabilistic version of them. As matter of fact, a VAE encodes inputs as distributions over the latent space rather than as points. During the VAE training, the input is encoded as distribution over the latent space, then a point from the latent space is sampled from that distribution, the sampled point is decoded and the reconstruction error can be computed, and finally, the reconstruction error is backpropagated through the network. The loss function, that is minimized during the training, is composed of a reconstruction term, that tends to make the encoding-decoding scheme as performant as possible, and of a regularization term, that tends to regularize the organisation of the latent space, by making the distribution returned by the encoder close to a Gaussian distribution.

Differently, GANs do not deal with any explicit probability density estimation. They are composed by two networks, a generator which learns to capture the data distribution, and a discriminator which estimates the probability that a sample comes from the data distribution rather than from the model distribution. In other words, the discriminator tries to distinguish between real and fake images, and the generator tries to fool the discriminator by generating real-looking images. Discriminator and generator are jointly trained in a minimax game, in which the objective function is

$$\min_G \max_D \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))], \tag{2.1}$$

where $\mathbf{x}$ denotes the real data distribution from $p_{data}(\mathbf{x})$, $\mathbb{E}$ denotes the expectation, and $\mathbf{z}$ is the vector from the random noise distribution $p_{\mathbf{z}}(\mathbf{z})$. Discriminator D wants to maximize the objective function 2.1, such that $D(\mathbf{x})$ is close to 1 (real) and $D(G(\mathbf{z}))$ is close to 0 (fake), whereas generator G wants to minimize it, such that $D(G(\mathbf{z}))$ is close to 1 (i.e., discriminator is fooled into thinking generated $G(\mathbf{z})$ is real). In practice, at the beginning of the training, when G is yet poor, D can reject samples with high confidence because they are clearly different from the training

data. In this case, $\log(1 - D(G(\mathbf{z})))$ satures. For this reason, it is better to train G to maximize $\log(D(G(\mathbf{z})))$ than to minimize $\log(1 - D(G(\mathbf{z})))$.

In general GANs tend to produce sharper images which look more realistic than VAEs [39]. For this reason GANs are widely used for several tasks, such as image and video generation, text to image translation, super resolution, style transfer, artifact removal, music generation, data augmentation, etc. GANs, on the other hand, are very difficult to optimize as they do not converge easily. Moreover there are not general criteria for the quantitative evaluation of the results. Some measures to evaluate GANs are proposed in [86]. Usually, humans check whether the generated images are perceptually realistic or not.

One of the most widely used variation of GANs are the Conditional GANs [93], in which a conditional vector is added along with the noise vector. In this way the generated result is not a generic example from a not known noise distribution, but follows a specific condition. The conditional vector is added both to the generator and the discriminator.

2.2 Classical Logic

In Classical Logic any proposition (i.e., a declarative sentence) has associated either the truth value 1 (true) or 0 (false) [35]. Depending on the granularity in the theory, a logic can be defined by a propositional or predicate language.

The syntax of *propositional logic* is built from the following symbols:

- a set of propositional variables $p_1, p_2, \ldots$;
- logical connectives: $\wedge$ conjunction, $\vee$ disjunction, $\neg$ negation, $\rightarrow$ implication, $\leftrightarrow$ equivalence;
- two constants $\bot$ and $\top$ denoting the *False* and *True* proposition, respectively.

These elements are combined according to the following inductive definition to build the set of formulas:

- propositional variables and constants are formulas;
- if ϕ, ψ are formulas, then $\phi \wedge \psi$, $\phi \vee \psi$, $\neg\phi$, $\phi \rightarrow \psi$, $\phi \leftrightarrow \psi$ are formulas.

Each formula represents a proposition whose truth value has to be evaluated. The truth evaluation of a formula is a mapping from the set of propositional variables to $\{0, 1\}$. Therefore the truth value of a compound proposition is a function of the truth values of its components and the logical connectives occurring in it. Table 2.1 shows the truth tables of the logical connectives.

Table 2.1: Truth tables of classical logic connectives.

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

In propositional logic there are two important logically equivalent statements, known as De Morgan's laws. They state that the negation of a disjunction is the conjunction of the negations, and the negation of a conjunction is the disjunction of the negations. In formal language such rules are written as in the following.

Definition 2.2.1 (De Morgan's laws)**.** *Given ϕ and ψ propositions*

$$\neg(\phi \vee \psi) \leftrightarrow \neg\phi \wedge \neg\psi$$

$$\neg(\phi \wedge \psi) \leftrightarrow \neg\phi \vee \neg\psi$$

First Order Logic

Propositional logic deals with simple declarative propositions, while First Order Logic (FOL), also known as predicate logic, covers predicates, functions and quantification. For this reason FOL is more suitable for contexts where some relational knowledge among the objects of a certain domain can be expressed. With FOL we can explicitly represent the elements of a domain (with terms), we can express property between individuals and relationships between two or more individuals (with predicates), we can quantify a property, asserting that is valid for at least an individual or for all the individuals. For instance, consider the two sentences "Elizabeth is a queen" and "Victoria is a queen". In a propositional language, these sentences are unrelated and have to be denoted by two different variables. On the other hand, the predicate "is a queen" occurs in both sentences, which have a common structure that could be denoted, for instance, by $queen(Elizabeth)$ and $queen(Victoria)$.

The alphabet of a first-order language contains the following symbols:

- set of predicates $\mathcal{P}$, each one with its arity;
- set of functions $\mathcal{F}$, each one with its arity;
- set of constants $\mathcal{C}$;
- set of variables $\mathcal{V}$;
- logical connectives $\vee, \wedge, \neg, \rightarrow, \leftrightarrow$;

- two constants $\bot$ and $\top$ denoting the *False* and *True* proposition, respectively;
- $\exists, \forall$ universal and existential quantifiers, respectively;
- parenthesis $(,)$.

Note that functions are different from predicates. A function takes one or more arguments, and returns a value, while a predicate takes one or more arguments, and is either true or false.

The set of terms will refer to objects in a certain domain and it is inductively defined as follows:

- variables and constants are terms;
- if $t_1, \ldots, t_n$ are terms and $f^{(n)}$ is an n-ary function, then $f^{(n)}(t_1, \ldots, t_n)$ is a term.

In first order logic, the set of formulas is defined upon predicates as:

- if $t_1, \ldots, t_n$ are terms and $p^{(n)}$ is an n-ary predicate, then $p^{(n)}(t_1, \ldots, t_n)$ is a formula, said atomic formula;
- if ϕ, ψ are formulas, then $\phi \wedge \psi$, $\phi \vee \psi$, $\neg\phi$, $\phi \rightarrow \psi$, $\phi \leftrightarrow \psi$ are formulas;
- if x is a variable and ϕ is a formula, then $(\forall x)\phi$, $(\exists x)\phi$ are formulas.

The semantic of a FOL formula is a boolean value, i.e., $\{0, 1\}$ as in propositional logic.

Example 2.2.1 (FOL formula)**.** *The sentence "Each number prime and greater than 2 is odd" can be converted into the following FOL-based formula:*

$$\forall x(prime(x) \wedge x > 2 \rightarrow odd(x))$$

2.3 Fuzzy Logic

The term *fuzzy logic* was firstly used by Zadeh [148] in 1965. Fuzzy logic aims at modeling the imprecise ways of reasoning, that play an essential role in the remarkable human ability to make rational decisions in an environment of uncertainty and imprecision [149]. In fuzzy logic the truth-value of a formula, instead of assuming two values $\{0, 1\}$ as in Classical Logic, can assume any value in the interval $[0, 1]$, where 0 denotes absolute false and 1 absolute true. This kind of logic allows us to deal with continuous values and is used to indicate the degree of truth represented by a formula. Connectives and quantifiers are converted using the fuzzy generalization of First Order Logic [101]. The three main fuzzy logics are Gödel, Lukasiewicz and Product. They can be defined by *t-norms* [55] that model the logical AND.

Table 2.2: The truth functions of fundamental t-norms, residuum ($\Rightarrow$), negation, material implication ($\rightarrow$).

	Product	Lukasiewicz	Gödel
$x \wedge y$	$x \cdot y$	$\max\{0, x+y-1\}$	$\min\{x, y\}$
$x \vee y$	$x+y-x \cdot y$	$\min\{1, x+y\}$	$\max\{x, y\}$
$\neg x$	$1-x$	$1-x$	$1-x$
$x \Rightarrow y$	$x \leq y ? 1 : \frac{y}{x}$	$\min\{1, 1-x+y\}$	$x \leq y ? 1 : y$
$x \rightarrow y$	$1-x+x \cdot y$	$\min\{1, 1-x+y\}$	$\max\{1-x, y\}$

Definition 2.3.1 (t-norm)**.** *A t-norm is a function* $T : [0,1] \times [0,1] \rightarrow [0,1]$ *which satisfies the following properties for all* $x, y, z \in [0,1]$*:*

$$T(x,y) = T(y,x) \tag{2.2}$$
$$T(x, T(y,z)) = T(T(x,y), z) \tag{2.3}$$
$$T(x,z) \leq T(y,z) \quad \textit{if} \quad x \leq y \tag{2.4}$$
$$T(x,1) = x \tag{2.5}$$
$$T(x,0) = 0. \tag{2.6}$$

T is a continuous t-norm if is a continuous function.

The main t-norms are:

- **Product.** $T(x,y) = x \cdot y$
- **Gödel** (or minimum). $T(x,y) = \min\{x,y\}$
- **Lukasiewicz.** $T(x,y) = \max\{0, x+y-1\}$

Table 2.2 shows the algebraic operations corresponding to the three fundamental continuous t-norms. Through the De Morgan's laws (def. 2.2.1) and the double negation law, i.e., $\neg\neg x \leftrightarrow x$, the disjunction $x \vee y$ is written as the negation of conjunction of negations, namely $\neg(\neg x \wedge \neg y)$.

Given a continuous t-norm we can define the corresponding residuum $\Rightarrow$ that generalizes the notion of implication and it is determined by

$$x \Rightarrow y = \max\{z : T(x,z) \leq y\}.$$

We can also obtain the material implication $x \rightarrow y$ by the negation ($\neg$) and by the AND operator, i.e., $\neg x \vee y$. For instance, if we take the product t-norm T,

$$T(x \rightarrow y) = T(\neg x \vee y) = 1 - x + y - y + x \cdot y = 1 - x + x \cdot y.$$

Quantifiers. Let's consider the case in which a formula contains quantifiers. With no loss of generality, we restrict our attention to FOL formulas in the Prenex Normal Form (PNF) form, where all the quantifiers $(\forall, \exists)$ and their associated quantified variables are placed at the beginning of the expression. Let's consider a FOL formula with variables $x_1, x_2 \dots$ assuming values in the finite sets $X_1, X_2, \dots$, and $P = \{p_1, p_2, \dots\}$ the vector of predicates [44]. We indicate with $p_j(X_j)$ the set of possible groundings [1] for the j-th predicate, and with $P(X)$ all possible grounded predicates, such that $P(X) = p_1(X_1) \cup p_2(X_2) \cup \dots$. Assuming all the predicates are evaluated in $[0, 1]$, then the truth degree of a formula containing an expression E can be computed by fuzzy logic operators according to Table 2.2.

The universal quantifier over a variable x_i is defined as the minimum of the t-norm generalization $t_E(\cdot)$:

$$\forall\, x_i E(P(X)) \;\rightarrow\; \Phi_\forall(P(X)) = \min_{x_i \in X_i} t_E(P(X)).$$

For the existential quantifier, the truth degree is instead defined as the maximum of the t-norm expression over x_i:

$$\exists\, x_i E(P(X)) \;\rightarrow\; \Phi_\exists(P(X)) = \max_{x_i \in X_i} t_E(P(X)).$$

However, a small modification to the universal quantifier is made to enable faster convergence during the training. The min operator over the t-norm values is replaced by the average over the set:

$$\exists x_i\, E(P(X)) \rightarrow \Phi_\exists(P(X)) = \frac{1}{|X_i|} \sum_{x_i \in X_i} t_E(P(X)).$$

Example 2.3.1. *The formula $\exists x_1 \forall x_2 A(x_1) \wedge B(x_2)$, using the product t-norm, is translated into the following continuous function:*

$$\Phi(P(X)) = \frac{1}{|X_1|} \sum_{x_1 \in X_1} \max_{x_2 \in X_2} A(x_1) \cdot B(x_2).$$

2.4 Learning from Constraints

Most real-world problems correspond with learning environments that are heavily structured, so it is important to construct appropriate representations of the environmental information. The framework of Learning from Constraints [49] integrates the ability of classical machine learning techniques to learn from continuous feature-based representations with the ability of reasoning using higher-level semantic knowledge. This approach bridges the symbolic and sub-symbolic worlds

[1] A grounding is a predicate evaluation on a certain element of a domain.

[44]. A set of constraints, that include information on labeled data and prior knowledge on the learning environment, have to be satisfied during the learning process. The combination of machine learning and constraints has been considered by several authors in the literature [9, 28, 90].

In Learning from Constraints both *hard* and *soft* constraints have been studied [45]. The first ones are those that have to be perfectly satisfied, while the latter can be violated, at the cost of some penalization. Soft constraints are preferred in the cases in which knowledge that they model is not perfect, or in which they cannot be valid for all the examples. For instance, soft constraints are typically used to model supervisions, since the target annotation might be subject to human error or might be incoherent due to the participation of various annotators.

Semi-supervised learning is a special case of Learning from Constraints in which supervised pairs are enforced together with constraints that describe the data distribution [49].

Let's formalize the approach of Learning from Constraints [32]. Consider a multi-task learning problem, where a set of T functions $\boldsymbol{f} = \{f_1, \ldots, f_T\}$ must be learned. We assume that a set of H constraints $\phi_h(\boldsymbol{f}), 0 \leq \phi_h(\boldsymbol{f}) \leq 1, (h = 1, \ldots, H)$ that describe the prior knowledge about the learning problem are provided. Let X_j be the sample of patterns of the function f_j. Multiple functions can share the same sample of patterns, i.e., $X_i = X_j$ for $i \neq j$. To keep the notation simple we limited the description to unary predicates, but it can be extended to predicates of any arity. Let $\boldsymbol{f}(X) = f_1(X_1) \cup f_2(X_2) \cup \ldots$ collects the groundings for all functions.

Following the classical penalty approach for constrained optimization, constraints can be enforced by penalizing their violation on the sample of data together with another term which forces the fitting of the supervised data for each function:

$$C[\boldsymbol{f}(X)] = \sum_{k=1}^{T} \Big(\overbrace{\|f_k\|^2}^{reg.} + \overbrace{\lambda_l \sum_{x \in \epsilon_k} L(f_k(x), y_k(x))}^{labeled} \Big) + \overbrace{\sum_{h=1}^{H} \lambda_h \mathcal{L}(\phi_h(\boldsymbol{f}(X)))}^{constraints} \tag{2.7}$$

where the first term is a regularization term penalizing non smooth solutions, $\epsilon_k \subset X_k$ is a set of labelled examples for the function f_k, $L(\cdot, \cdot)$ is a loss function, $y_k(x)$ is the target for the example x for the k-th task, λ_l is the weight for the labeled portion of the cost function, $\mathcal{L}(\cdot)$ is the loss function for the part about constraints, and λ_h is the weight for the h-th constraint. Higher is λ_h and not respecting the h-th constraint is costlier. $\mathcal{L}$ is a monotonically decreasing function, that should be equal to zero when the formula ϕ_h is true (i.e. equal to 1). In general the following mappings are exploited:

$$\mathcal{L}(\phi_h(\boldsymbol{f})) = 1 - \phi_h(\boldsymbol{f}), \tag{2.8}$$

$$\mathcal{L}(\phi_h(\boldsymbol{f})) = -\log(\phi_h(\boldsymbol{f})). \tag{2.9}$$

Constraints can be represented as First Order Logic clauses, which provide a formally well-defined representation for abstract knowledge. In this case, fuzzy logic (Section 2.3) is used to bring these formulas into the learning problem. As matter of fact, t-norms are employed to convert FOL-formulas into differentiable functions.

The integration of FOL logic rules into the learning has been considered in several works [60, 91, 126]. Nevertheless, the proposed approaches are still limited in the kind of knowledge that can be integrated. In [60] the definition of the model is limited to universally quantified formulas and to a small set of logic operators, while in [91] it is limited to Horn clauses [2].

In several applications, it might happens that the number of constraints is large and adding a new one could bring to an inconsistency of the overall system. In this case, these (partially) inconsistent constraints will not be hardly satisfied. With soft constraints instead, we can give more or less importance to a rule, by finding an appropriate compromise. In this way the system will try to satisfy the conditions that are more weighted for most of the examples, and the ones that are less weighted for the remaining ones. However, FOL rules provide a well-defined representation of knowledge, so we can more easily check the consistency of the constrained system or if some constraints are unnecessary.

2.5 Affective Computing

Affective Computing is a relatively new interdisciplinary field of research spanning the areas of computer science, psychology, and cognitive science. In 1997, Rosalind Picard defined "Affective Computing" as the computing that relates to, arises from, or influences emotions [108]. Affective computing aims to enable intelligent systems to recognize, feel, express and interpret human emotions, so that can enrich the interactivity between human and machine. Emotions are detected in different ways, such as in speech, in facial expressions, in body movements, in text, or in physiological signals. A machine detects emotional information capturing data with sensors. A webcam can capture facial expressions or body movements, a microphone can gather speech, other sensors can detect emotional signals by directly measuring physiological data [13], such as skin temperature, galvanic resistance, Electrocardiography (ECG), Blood Volume Pulse (BVP), Electromyography (EMG), Electroencephalography (EEG) [150].

Body gestures provide strong and reliable signals to detect an individual's emotional state [27], especially to help people with Autism Spectrum Disorder (ASD) [107]. Simple gestures can communicate clear messages, as lifting the shoulders when we do not know the answer or clapping the hands when we appreciate some-

[2] Horn clause is a disjunction of literals (a propositional variable or its negation) with at most one positive, i.e. not-negated literal.

thing. There are different approaches to detect body gestures. Appearance-based models extract information directly from images or videos, 3D model-based algorithms employ volumetric models, and skeletal-based algorithms use a virtual skeleton to focus on essential parts of the body.

There are two main approaches to classify emotions, i.e., continuous and categorical. The first uses dimensions such as negative vs. positive, calm vs. aroused, while the second uses discrete classes. In general emotions are categorized by the six universal emotions defined by Ekman, namely anger, disgust, fear, happiness, sadness, surprise. Some of the leading theories about emotions are presented in Appendix A.

Humans usually communicate and express their emotions in multimodal way, combining facial expressions, gestures, voice, etc. In recent years, methods that exploit multimodal data have been developed, so improving the accuracy of the overall result [111]. In general three channels are fused, namely text, audio and video. The challenge of multimodal analysis is how and at what stage to join data from the various modalities. There are two types of fusions, i.e., *feature-level* (or *early fusion*) and *decision-level* (or *late fusion*). In early fusion the features extracted from various channels are combined in a unique feature vector, which is used for the classification. The advantage is that the model can capture correlation between the modalities. The disadvantage is the time synchronization, as the features obtained from diverse channels might widely differ. In late fusion each modality is processed independently and the results are fused to obtain the final decision. The advantage is that the fusion process is easier, because the decisions coming from the various modalities in general are similar. On the other hand, training each classifier independently is time consuming. Some works employed a hybrid approach, exploiting the advantages of feature-level and decision-level fusion.

Affective Computing has a wide range of applications, as healthcare, education, games, marketing, automated driver assistance, entertainment, and so on. In the field of healthcare, through wellness monitoring, affective computing can be useful to create an individual profile identifying causes of stress, anxiety, depression [154]. Technologies handling emotions are employed especially to help people with ASD [144]. Several studies demonstrate that emotional responses of individuals with ASD are less differentiate and more negative, and that they have difficulties in evaluating their own emotions. Methods based on facial expressions are used both to detect ASD and to help children with ASD to produce expressions [29].

Affective computing in education is helpful to understand students' learning and interest in order to formulate appropriate teaching plans [146]. This is important especially in e-learning applications, where the emotional incentive between teachers and students is very poor. In this scenario, the teacher can adapt the pedagogical situation when a learner is bored, interested or frustrated.

Affective computing can be useful also to make the driving experience more secure, reading user's physiological signals through components with which he/she naturally interacts, such as the steering wheel [127]. Recently, technologies have been developed for warning the driver if he/she is sleepy, angry, unconscious or unhealthy to drive, lowering the speed or stopping the vehicle if necessary.

Systems detecting emotions are employed for commercial uses, in order to understand whether some products are appreciated and what elements are of greatest interest. Recommendation systems suggest movies, TV series, musics or products to the users according to their previous responses, learning their preferences, in a way. Affective computing has other various applications in human-computer interaction, such as affective mirrors allowing the individual to see how he/she performs, or emotion monitoring agents alerting before sending an angry email, or music players selecting tracks based on mood, and so on.

Chapter 3

Facial Expression Recognition

In this chapter the problem of facial expression recognition is addressed, that consists in detecting emotions in facial images or videos. We investigate the application of a pool of *Convolutional Neural Networks* (*CNNs*) with the aim of building recognizers of expressions in static images, that can be further applied to video sequences. Both (*i.*) *appearance* and (*ii.*) *shape features* are considered, but, differently from most of the existing works, we do not hand-engineer shape features, and we let the CNNs learn the right representations from special shape-only images. We propose a model that considers (*iii.*) *face sub-parts* in addition to the entire face, motivated by the need of gaining deeper insights in the role of each component. Then, we move to the Semi-Supervised setting, exploiting (*iv.*) video data. The unsupervised portion of the training data is used to enforce three kinds of coherence:

- *temporal coherence* among consecutive frames;
- *part-based coherence* in each frame, i.e., a coherent prediction among the CNNs that operate on the different face parts;
- *coherence between appearance and shape*-based representation for each face part.

Such constraints bridge the functions computed by the CNNs at training time while, at test time, we study the output of each learned predictor independently. This allows us to concretely grasp the importance of each face part, especially in presence of occlusions, and to evaluate how each single predictor benefits from the information transfer activated by the proposed constraints.

Facial expressions are one of the most powerful and universal signals of emotion manifestation, and have been studied across several disciplines, such as psychology, neuroscience, sociology and computer science. Facial features of expressions are mostly located around mouth, nose, and eyes, and their locations are essential

in explaining and categorizing expressions [34]. Despite the large number of advanced psychological experiments about the human perception and recognition of emotions, we can trivially figure out that different face parts have a different impact in the way humans recognize emotions: the role of eyebrows when we are angry or the way we treat our mouth when we are happy or surprised, for example.

Although the task of facial expression recognition is widely studied and much progress has been made, it still remains a challenging problem, due to the variability and complexity of facial expressions. Moreover different personal attributes, such as age, gender, ethnic backgrounds make this task more trivial. Facial expression detection has several applications, such as in healthcare (pain detection, monitoring of depression, helping individuals with the autism spectrum disorder or with facial paralysis), in e-commerce, in sociable robots, for driver assistance, in cartoons to create characters' expressions, and in many other human-computer interaction systems.

This chapter is organized as follows. In the next two sections the related works and the main datasets of facial expressions are mentioned. Section 3.3 formalizes the problem of facial expression recognition. In Section 3.4 we describe the different representations of the input data that are provided to our model and the basic structure of the predictors. Then we introduce a set of constraints that are enforced in the learning stage of the classifiers (Section 3.5), and we present our experimental analysis (Section 3.6). Finally, some examples are presented to illustrate cases in which the shape-based representation allows the system to detect the right expression while the appearance-based representation fails, such as in presence of occlusions over some face parts as mouth or nose (Section 3.7).

3.1 Related works

We can find several approaches that exploit Machine Learning with the aim of learning to categorize emotions from examples [76, 121]. In general the classification is made into the six universal emotions (see Section A.1), whereas few works tried to detect non-basic affective states from facial expressions, such as fatigue, pain [79], and mental states like agreeing/disagreeing, concentrating, interest [147], frustration [64] and insecurity. Emotions are also represented in the continuous space considering affect dimensions [54] (see Section A.2). Two facial features can be considered, namely *appearance-based*, which use textural information by considering the intensity values of the pixels, and *shape-based*, which ignore texture and describe shape explicitly, generally extracting landmarks points.

Most of approaches of facial expression recognition are about using still images [84, 96], while several more recent works also consider video sequences where actors start with a neutral expression and generate a non-neutral one [83, 151]. The

learning framework is usually fully supervised, and supervision is either about each training image or about each video sequence. Works that exploit video data focus on the importance of the temporal evolution of the input face. The system proposed by Fan and Tjahjadi [41] processes four sub-regions of the face: forehead, eyes/eyebrows, nose and mouth. They used an extension of the spatial pyramid histogram of gradients and dense optical flow to extract spatial and dynamic features from video sequences, and adopted a multi-class SVM-based classifier with one-to-one strategy to recognize facial expressions. With the coming of deep learning, many deep networks-based approaches are employed to perform the task of facial expression recognition in end-to-end way. The architecture more suited to learn spatial features directly from images are the CNNs [15]. Jung et al. [63] propose a neural-network-based method where two different networks are exploited: the first one extracts appearance features from image sequences, learning temporal correlations, while the other network extracts shape features from a set of facial landmarks. The two nets are combined to yield the final decision on the emotion class. Happy and Routray [56] identify salient areas with generalized discriminative features for expression classification. They only use appearance-based features, and do not consider the time domain. The framework from Jain et al. [62] recognizes facial expressions from video sequences by modeling temporal variations within shapes. They show that shape provides important information that is sometimes hard to grasp from appearance only. Zhang and Huang [151] propose a mixed model which include a "temporal" and a "spatial" network. The former captures dynamic features from consecutive frames, while the latter is about extracting static features from still frames.

Considering real scenarios, the main challenges in facial expression recognition are head-pose variations, illumination variations, and occlusions. To address these problems, Wang [142] developed a region based deep attention architecture, which adaptively integrates visual clues from regions and whole faces. Levi [75] proposed a more robust approach to illumination changes, mapping images to Local Binary Pattern (i.e., a texture descriptor which constructs local representations by comparing each pixel with its surrounding neighborhood of pixels [103]).

3.2 Datasets

There are two types of facial expression datasets, i.e. posed and spontaneous [138]. In the first, the participants are asked to act different expressions, so making easier the recognition but less applicable in real world scenarios. In the second, expressions are natural, so more realistic, but more difficult to detect. Moreover, in this case to obtain a reliable dataset some experts have to annotate the images and some labels may be ambiguous. There are many corpora containing images or videos

Figure 3.1: Example of a sequence of CK+. It starts with a neutral face and ends with surprised expression.

of facial expressions, some of them collected for emotion recognition competitions, such as FER2013 [48] and Emotion Recognition in the Wild (EmotiW) [31]. In what follows, we report the most famous or the easily accessible ones.

- **CK+** [85]: the Extended Cohn-Kanade (CK+) database contains 593 sequences, recorded in laboratory by 123 subjects (different age and gender), starting from neutral expression and ending with the peak of the expression (Fig. 3.1). The video sequences, composed of 10-60 frames, are labeled with an emotion among the six basic plus contempt.

- **MMI** [137]: the MMI database is laboratory controlled and contains 740 images and 2,900 videos of 25 people. The sequences begin with a neutral expression and reach a peak near the middle before returning to the neutral expression. The six universal emotions are collected.

- **MUG** [2]: MUG is a collection of posed and induced facial expression image sequences, captured in a controlled laboratory environment with high resolution and no occlusions. The sequences begin and end at neutral state and follow the onset-apex-offset temporal pattern [1].

- **ADFES** [139]: the Amsterdam Dynamic Facial Expression Set contains nine dynamic filmed expressions, that are joy, anger, sadness, fear, disgust, surprise, contempt, pride, and embarrassment. The sequence starts with neutral face and ends with the peak of the expression (Fig. 3.2). The subjects are divided into male/female and North-European/Mediterranean, and are recorded in frontal view and in two different head-turning versions (faces turning toward and away from viewers).

- **FER2013** [48]: FER2013 is collected automatically from web images, that are adjusted, cropped and resized to 48×48 pixels. It contains 35,887 images with the six basic emotions plus neutral.

[1] The temporal evolution of an expression is usually described by four phases, that are *neutral* in which there are not signs of muscular activity, *onset* where the muscular contraction begins, *apex* where the peak of the expression is reached and *offset* in which there is muscular relaxation.

Figure 3.2: Example of a sequence of ADFES. It starts with a neutral face and ends with angry expression.

Figure 3.3: WSEFEP examples. A same subject performs 7 expressions (anger, disgust, fear, happiness, sadness, surprise, neutral).

- **AFEW** [31] and **SFEW** [30]: The Acted Facial Expressions in the Wild (AFEW) database contains 1809 video clips collected from movies with spontaneous expressions, different head poses, occlusions and illuminations. The Static Facial Expressions in the Wild (SFEW) was created by selecting 1766 static frames from the AFEW database. The emotions collected are the six universal plus neutral.

- **WSEFEP** [104]: Warsaw Set of Emotional Facial Expression Pictures contains 210 high quality photographs of the six universal emotions plus neutral, performed by 30 subjects (Fig. 3.3).

- **Multi-PIE** [53]: The Multi-PIE database contains 755,370 images from 337 subjects under 15 viewpoints and 19 illumination conditions in up to four recording sessions. The expressions performed are slightly different from the six basics: smile, surprised, squint, disgust, scream and neutral.

- **EmotioNet** [40]: EmotioNet is a large-scale database with one million facial expression images collected from Internet. The images are labeled with the six basic emotions and with ten compound expressions [33].

- **RAF-DB** [77]: The Real-world Affective Face Database (RAF-DB) contains 29,672 different facial images downloaded from Internet. With manually crowd-sourced annotation and reliable estimation, each example is labeled with the six basic emotions plus neutral and eleven compound emotions.

- **AffectNet** [97]: AffectNet contains more than one million images collected from Internet, of which 450,000 were manually annotated with the six basic emotions plus neutral and contempt, and with the intensity of valence and arousal (see Appendix A.2 for the definition of valence and arousal).

- **ExpW** [152]: The Expression in-the-Wild Database (ExpW) contains 91,793 faces downloaded using Google image search. Each image was manually annotated with one of the six universal emotions plus neutral.

3.3 Problem Definition

The task of facial expression recognition that we consider consists in building a classifier that predicts one of the six universal emotions, plus the *neutral* case, and that we collect into the set Y, codified with indices from 1 to 7. The most popular inputs of the recognizer are images of faces, represented in foreground, usually with frontal orientation. When video data are considered, the recognition problem focusses on short video clips where a transition from the *neutral* state toward one the six emotions is recorded. Processing videos instead of still images can improve the recognition performance because facial expressions involve variations of the facial muscles along the temporal dimension. However, classifiers that are specifically trained to build a latent representation from a video clip $\mathcal{V}$ before taking a decision, cannot be immediately applied to classify images. Differently, image-based classifiers can process single frames $\{\mathcal{I}_t\}$ of a video (being t the time index) to produce a final decision over a time window, so they are more versatile from the point of view of easiness of deployment in different real-world applications. The facial expression recognition problem is usually faced in the "Fully-Supervised" setting, and, in the case of videos, the available datasets are composed of labeled video clips where we do not have access to the labelings of the single frames. Nonetheless, obtaining supervised data is costly, while nowadays is pretty easy to have access to collections of unsupervised frontal view faces (web, social networks, smartphones, ...) or unsupervised video recordings (video conference/call applications). This suggests that studying the "Semi-Supervised" setting, where a portion of the training data is labeled and a larger portion is unsupervised, can be a promising way to approach the recognition task.

Motivated by the need of building a versatile emotion recognition system, we focus on a predictor that operates on still images and that we can use to make predictions on video data. The system can be trained exploiting both video and image data in a Semi-Supervised setting, taking advantage of the temporal evolution described by the video format. In detail, we consider a classifier $f(\cdot)$ that produces a decision $y \in Y$ for each input image $\mathcal{I}$, or for a set of consecutive frames belonging to a time window W (that covers a video clip, for example),

$$y = f(\mathcal{I}) \tag{3.1}$$

$$y = \texttt{majority}_{t \in W} \{f(\mathcal{I}_t)\} \ , \tag{3.2}$$

where `majority` is the majority-voting function, that returns the most frequent prediction in the time window W. Differently from the existing approaches, our system can be trained using labeled and unlabeled image datasets, collected in $\mathcal{D}_{\mathcal{I}}$, or labeled and unlabeled frames extracted from the previously described labeled video sequences, collected in $\mathcal{D}_{\mathcal{V}}$. Due to the aforementioned properties of the existing video datasets (containing transitions from *neutral* to a certain emotion), we can artificially generate $\mathcal{D}_{\mathcal{V}}$ by labeling as *neutral* the very first frames of each video clip, and by assigning the provided video label to the last frames of the sequence. The frames in the internal portion of the sequence are not labeled. Formally, we have

$$\mathcal{D}_{\mathcal{I}} = \{(\mathcal{I}_i, y_i),\ i = 1, \ldots, l\} \cup \{(\mathcal{I}_i, \texttt{none}),\ i = l+1, \ldots, l+u\}\ ,$$

where $y_i \in Y$ is the image label, and the rightmost set is fully unlabeled. Then,

$$\mathcal{D}_{\mathcal{V}} = \{\mathcal{D}_{\mathcal{V}_z},\ z = 1, \ldots, v\}\ ,$$

where v is the number of available video clips and $\mathcal{D}_{\mathcal{V}_z}$ is a sequence extracted from the z-th clip,

$$\begin{aligned} \mathcal{D}_{\mathcal{V}_z} \quad = \quad & ((\mathcal{I}_{z,t}, \texttt{neutral}),\ t = 1, \ldots, \alpha|\mathcal{V}_z|) \oplus \\ & ((\mathcal{I}_{z,t}, \texttt{none}),\ t = \alpha|\mathcal{V}_z| + 1, \ldots, \beta|\mathcal{V}_z|) \oplus \\ & ((\mathcal{I}_{z,t}, y_z),\ t = \beta|\mathcal{V}_z| + 1, \ldots, |\mathcal{V}_z|)\ , \end{aligned}$$

being $\oplus$ the sequence concatenation operator, $\mathcal{I}_{z,t} \in \mathcal{V}_z$ the t-th frame of the z-th video, and $0 < \alpha < \beta < 1$, arbitrarily chosen. In this case $y_z \in Y \setminus \{\texttt{neutral}\}$ is the label provided with the video clip $\mathcal{V}_z$ (`neutral` is the identifier of the *neutral* class). We notice that $\mathcal{D}_{\mathcal{V}}$ is more informed than $\mathcal{D}_{\mathcal{I}}$, since it also stores the image/frame order and the frame grouping with respect to the videos. For this reason, we can consider $\mathcal{D}_{\mathcal{I}}$ to be an instance of the more general representation $\mathcal{D}_{\mathcal{V}}$, and from here and out, we will focus on data represented as in $\mathcal{D}_{\mathcal{V}}$ without reducing the generality of what we described so far, and we will compactly indicate it with $\mathcal{D}$.

3.4 Feature Representation and Model Structure

Our model is based on CNNs that process two categories of representations of the input image/frame $\mathcal{I}$. Such categories consist in *appearance*-based (i.e, visual) representations and a *shape*-based representations. In both the cases, we do not consider the whole $\mathcal{I}$, but only the rectangular area that is covered by the target face. We localize the face first, and then we crop the image accordingly. This choice is crucial when processing inputs with multiple faces or when the face is not well positioned at the center of the image (or more generally, at a position incoherent with the training data). The *appearance*-based representation of the face is simply a grayscale instance

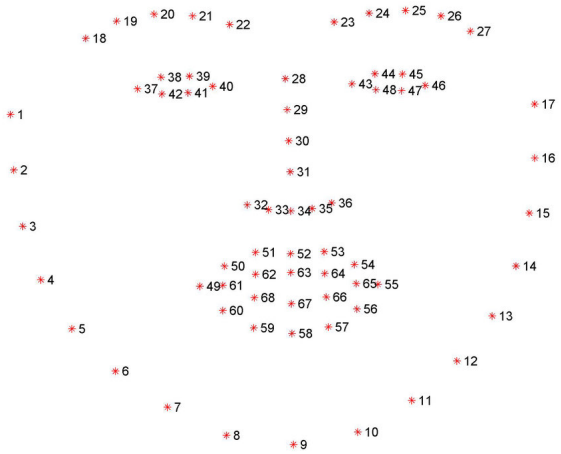


Figure 3.4: The indexes of the 68 coordinates extracted from a human face with the *dlib* facial landmark predictor. Facial landmarks are used to localize and represent salient regions of the face, that are eyes, eyebrows, nose, mouth, and jaw.

of the cropped face. In the case of the *shape*-based representation, we still focus on the same cropped region, but we extract a set of shape features that essentially describe the contours of the face parts, and that, in this work, consist of a set of facial landmark points. However, instead of stacking their 2D coordinates into a vector (that is only possible if the set of points is consistent among different faces), we consider a more generic approach in which the shape is simply represented by an artificial image with uniform background and in which the landmarks points are depicted at their coordinates. This allows us to treat the shape in a way that is similar to what we do with the appearance, and it opens the possibility of providing different shape "sketches" that are not only based on landmark points (but also on contour lines, for example).

In order to study the effects of the different face parts in the recognition process, we computed the appearance and shape representations for the face and for all the face parts: mouth, nose, eyes, eyebrows. First we detected face area exploiting the localizer of Viola and Jones [141], which uses the classic Histogram of Oriented Gradients (HOG) features combined with a linear classifier, an image pyramid, and a sliding window detection scheme. Then we extracted the 68 landmark points [65] shown in Fig. 3.4 [2]. Cropping around each set of part-related landmarks (adding a small padding), we obtained 7 instances of appearance-based representations of the input $\mathcal{I}$ and 8 shape-based ones, since in the case of shape we also included the landmarks associated to the jaw contour. Figure 3.5 shows the overall 15 representations that we generate. We resized these representations to the following sizes: face area 200×200, mouth area 80×50, eye area 60×30, eyebrow area 100×30, nose area 60×100 pixels, jaw area 200×170.

[2] We used OpenCV `https://opencv.org/` and the "dlib" library `http://dlib.net/`

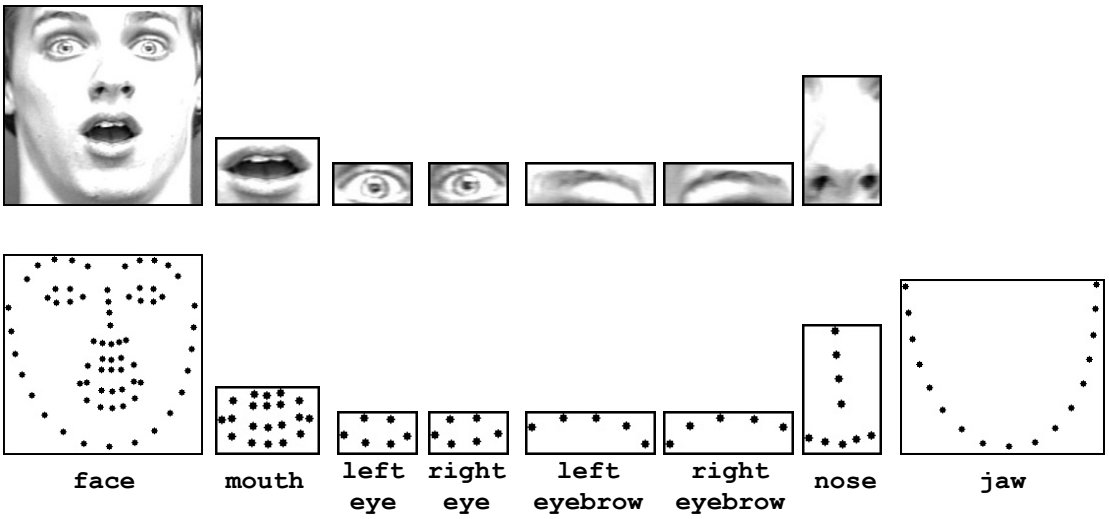


Figure 3.5: Representations extracted from an input image: 7 appearance-based representations (top) and 8 shape-based representations (bottom), that we implement by sketching landmark points in artificial images.

We implemented a pool of 15 CNNs, each of them processing one of the aforementioned representations (Figure 3.6). The generic CNN_h associated to the h-th representation has two convolutional layers followed by max pooling, and some fully connected layers terminated with a softmax activation that outputs a probability distribution over the emotions in Y. We indicate with $p_h(\cdot)$ the function computed by such CNN_h. All the hidden neural units have ReLu activation functions. The face-related CNNs have 32 and 64 filters on the two convolutional layers, respectively, and two fully connected layers (64 and $|Y| = 7$ neurons). The other CNNs, that are based on inputs with smaller sizes, exploit 16 and 32 filters, and a single fully connected layer ($|Y| = 7$ neurons). The output of each of the 15 CNNs, when followed by an arg max operation (assuming 1-based indexing), is a possible instance of the function f in Eq. (3.1) and Eq. (3.2). Formally, for a given h,

$$\begin{aligned} x_h &= \texttt{representation}_h(\mathcal{I}) \\ p_h(x_h) &= \texttt{CNN}_h\,(x_h) \\ f(\mathcal{I}) &= \arg\max p_h(x_h)\,, \end{aligned}$$

where x_h is the h-representation of the input, and $p_h(x_h)$ outputs a vector of size $|Y|$ that sums to 1. Even if our final goal is to focus on the case in which h is the index of the full-face-based classifier, in Section 3.6 we will evaluate the quality of multiple instances of f, considering the predictors on the face parts too.

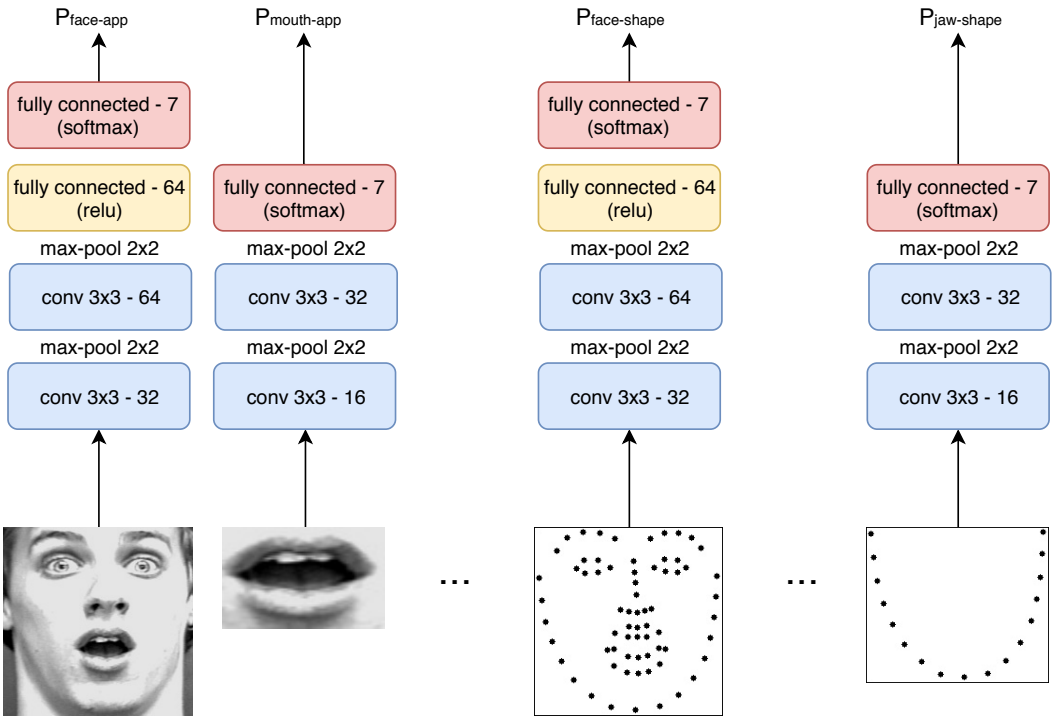


Figure 3.6: Structure of the 15 CNNs employed. Each CNN processes one of the input representations. Each network ends with a softmax activation that outputs a probability distribution over the six universal emotions plus neutral class.

3.5 Coherence Constraints

We trained the pool of CNNs by minimizing an objective function involving the cross-entropy $L(p_h(x_h), y)$ between the outputs of the networks and the available labels (one-hot encoding), considering the training data $\mathcal{T} \subset \mathcal{D}$. The cross-entropy only exploits the labeled pairs in $\mathcal{T}$. However, our objective function is also composed by the penalties associated to the fulfilment of "coherence constraints" that we enforce on all the samples of $\mathcal{T}$, being them labeled or not. We have considered three types of coherence, namely "temporal coherence", "coherence among the predictions on the face parts" and "coherence between appearance and shape". The former enforces the CNNs to be coherent over time for each video sequence, i.e., it enforces the predictions to smoothly change along the time axis. This constraint introduces a regularizing effect, since it prevents the system from developing unstable models that abruptly change their decisions among consecutive frames, as shown in Figure 3.7.

The part-based coherence enforces each full-face-representation-based classifiers to take decisions that are coherent with the ones taken (*on average*) by the other part-based classifiers (and vice-versa), as represented in Figure 3.8. The idea behind this constraint is that the committee of the local (i.e. part-based) predictors could

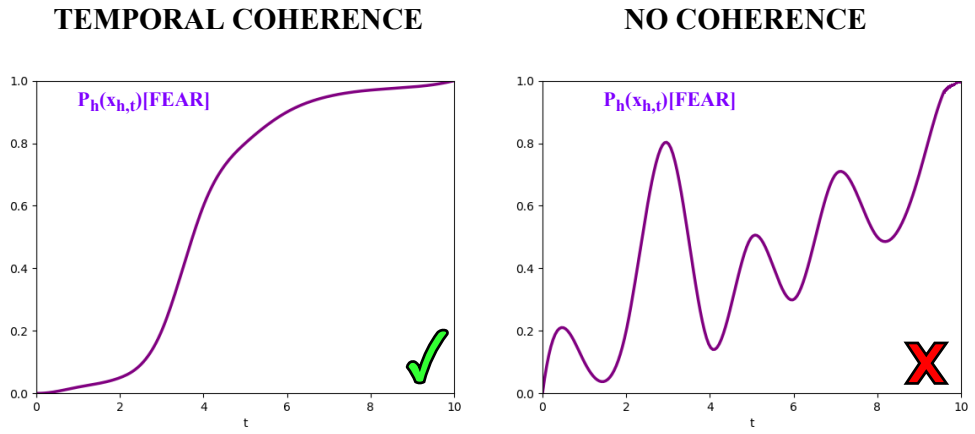


Figure 3.7: The trend of the predictions along the time axis of a specific emotion classifier (in this case "fear"). We consider a transition from a neutral state to a specific emotion, and only the first and the last time instant are paired with supervisions ($t = 0$ not-fear, $t = 10$ fear). In the left picture, the temporal coherence enforces the predictions to smoothly change along time axis. When such coherence is not enforced, the trend of the predictions in the inner portions of the time time window might oscillate (right picture).

provide important fine-grained information that the global (face-based) predictor might not have been able to capture. Our main expectation from this condition is to improve the classifier that processes the full-face input, but this might also have opposite effects, as we will show in Section 3.6.

The coherence between appearance and shape enforces the prediction of the appearance-based classifier to be coherent with the prediction of the shape-based classifier for each part (excluding the jaw) – we remark that the enforcement of each of the coherence constraints only happens at training time.

In detail, given three scalars $\lambda_t, \lambda_c, \lambda_r \geq 0$ that weigh the importance of the coherence (soft) constraints, we define our objective function as the sum of three contributions (cross-entropy on the supervised examples, temporal coherence, part-based coherence) both for the appearance and shape-based representation, and of the coherence between appearance and shape. In what follows, for the sake of simplicity, we report each contribution considering the appearance-based representation (the shape-based case is equivalent). In detail, we have:

i. Loss on the supervised examples:

$$\sigma_{app}(\mathcal{T}) = \sum_{h} \sum_{\substack{i=1 \\ y_i \neq \texttt{none}}} w_i \cdot L(p_{h,app}(x_{h,app,i}), y_i), \tag{3.3}$$

where L is the cross-entropy loss function and the index h spans over the 7 appearance-based classifiers (or the 8 shape-based classifiers). The index i spans over all the pairs in $\mathcal{T}$, and, for the sake of simplicity, we used the no-

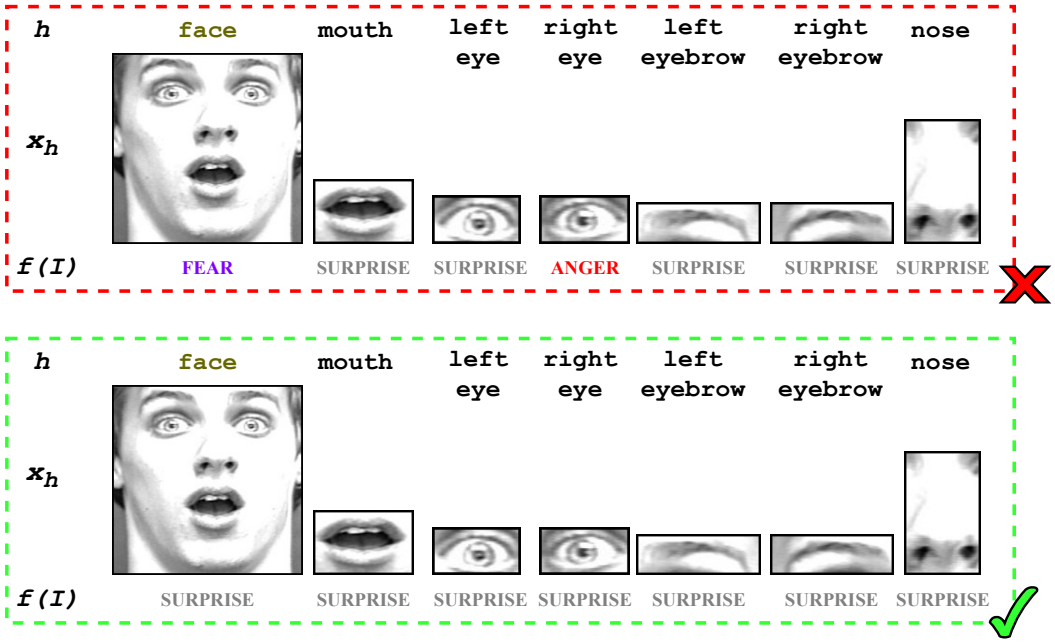


Figure 3.8: Part-based coherence ensures that the prediction of the full-face classifier is coherent with the (average) prediction of the other parts classifiers. Above, an example of a constraint which is not satisfied is reported, while below the predictions between the full face and the other parts are coherent.

tation $y_i \neq \texttt{none}$ to indicate that we consider only the labeled examples. The scalar weights w_i are used to give custom weights to the examples, and we used them to give more importance to the classes that are less represented in $\mathcal{T}$.

ii. Temporal coherence:

$$\tau_{app}(\mathcal{T}) = \sum_{h} \sum_{z=1}^{v} \sum_{t=2}^{|\mathcal{V}_z|} (1 - p_{h,app}(x_{h,app,(z,t-1)})' \cdot p_{h,app}(x_{h,app,(z,t)})), \tag{3.4}$$

where the notation (z,t) is the index of the t-th frame in the z-th video sequence belonging to $\mathcal{T}$, while $'$ is the transpose operator.

iii. Part-based coherence:

$$\rho_{app}(\mathcal{T}) = \sum_{h \neq \texttt{face}} \sum_{i} (1 - p_{\texttt{face},app}(x_{\texttt{face},app,i})' \cdot p_{h,app}(x_{h,app,i})), \tag{3.5}$$

where the notation `face` is used to indicate the index associated with the full-face input.

We notice that since $p_{\cdot}(\cdot)$ is a probability distribution, so the dot products involving two instances of $p_{\cdot}(\cdot)$ are 1 when such instances are equivalent (and the coherence constraints are fulfilled). The temporal constraint involves dot products between the predictions on pairs of consecutive frames in the same video clip. We kept the same structure to build the part-based constraint, where the averaging operation on the part-based classifiers is evident when $\sum_{h \neq \texttt{face}}$ is moved right before the second term of the dot product $p_{\texttt{face}}(x_{\texttt{face},\cdot,i})' \cdot p_h(x_{h,\cdot,i})$.

Still focussing on the case of appearance-based input representation, the objective function that we aim at minimizing (with respect to the weights of the CNNs) is the sum of the three contributes defined above,

$$\gamma_{app}(\mathcal{T}) = \sigma_{app}(\mathcal{T}) + \lambda_t \, \tau_{app}(\mathcal{T}) + \lambda_c \, \rho_{app}(\mathcal{T}). \tag{3.6}$$

We notice that the aforementioned constraints are only focussed on a given input representation (either shape or appearance). However, similarly to what happens in the coherence between the full face and its parts, we can defined a coherence constraint between appearance and shape-based representations that is

$$\phi(\mathcal{T}) = \sum_{h \neq \texttt{jaw}} \sum_{i} (1 - p_{h,app}(x_{h,app,i})' \cdot p_{h,shape}(x_{h,shape,i})). \tag{3.7}$$

The index h does not consider the jaw since it is only available in the shape-based inputs, and each dot product involves an appearance and a shape-based representation of h. Finally, the final objective function of our model is the sum of all contributes introduced so far,

$$\gamma(\mathcal{T}) = \gamma_{app}(\mathcal{T}) + \gamma_{shape}(\mathcal{T}) + \lambda_r \, \phi(\mathcal{T}). \tag{3.8}$$

3.6 Experimental Results

In order to validate our model, we used the Extended Cohn-Kanade dataset (CK+) described in Section 3.2. We excluded the sequences associated to "contempt", which is not included into the six universal emotions.

In order to build the Semi-Supervised set $\mathcal{D}$ described in Section 3.3, we selected $\alpha = 0.1$ and $\beta = 0.7$. So, as represented in Figure 3.9, for each sequence of the dataset the first 10% of the frames were labeled with neutral expression, the last 30% with the specific emotion and the remaining frames were unsupervised. We generated 5 randomizations of the whole dataset, and divided each of them into training (70%), validation (15%), and test sets (15%), keeping the original distribution of the classes in each set. The validation data was used to validate the model parameters, while the

Figure 3.9: A sequence of the CK+ dataset, where we assign *neutral* label to the initial frames and *surprise* label to final frames, while the internal frames are left unlabeled.

test partition was used to measure the quality of the model. The results presented in this section are averaged over the 5 test partitions. Each collection of training data consists of about $\approx 4,000$ frames, out of which $\approx 1,500$ are labeled, and they are organized into ≈ 200 sequences, while the validation data is composed of ≈ 600 frames, out of which ≈ 200 are labeled, and organized into ≈ 30 sequences. Since examples from the "neutral" class are much more represented with respect to other examples, we set $w_i = 0.1$ in Eq. (3.3) if i is an example from the neutral class, $w_i = 1$ otherwise.

Initially we excluded coherence between appearance and shape, setting $\lambda_r = 0$. We selected the optimal λ_c, λ_t by a grid-search in $\{10^{-10}, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-4}, 10^{-2}\}$, measuring frame-level accuracy (i.e., only the labeled validation frames are considered). We implemented our model using TensorFlow, and we minimized Eq. (3.8) by the Adam-based optimizer (starting learning rate 0.001), mini-batches of size 96, and we trained the model for multiple epochs, stopping the procedure when the validation error started increasing.

We performed experiments comparing a system with no-coherence-constraints ($\lambda_c = \lambda_t = 0$) with other models that include either temporal or part-based coherence. We compared the cases of single-frame-level predictions (where only the labeled portion of the test set is considered) and the case of video-sequence-level predictions, following the decision rules of Eq. (3.1) and Eq. (3.2), respectively (where W covers the full video sequence). An example of sequence-level decision is represented in Figure 3.10, where the prediction is the most predicted emotion on the frames of the video-sequence, excluding the class "neutral", since the CK+ dataset does not contain sequences labeled with this class. Since examples of the different classes are not balanced in the given dataset, and in order to provide a more informative set of results, we measured two types of accuracies, namely `Micro` and `Macro` accuracies. The former is simply the percentage of correctly labeled frames/sequences, while the latter is the average of the percentages of correctly labeled frames/sequences in each emotion class.

Table 3.1 shows the results we obtain when testing the classifiers that operate on the full-face inputs, considering both appearance and shape representations. We also report results of an additional classifier obtained by averaging the outputs of

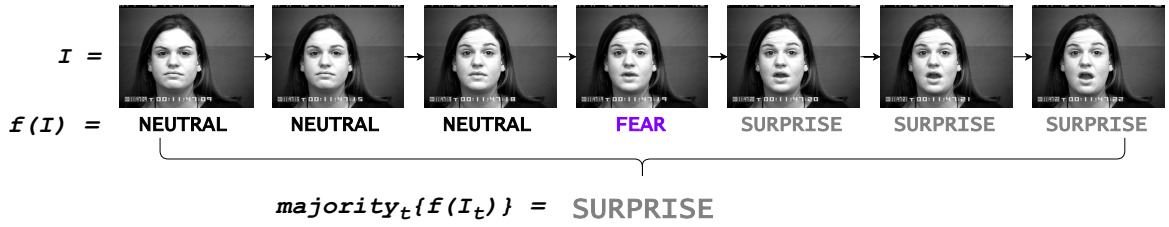


Figure 3.10: Example of prediction on a video sequence. The prediction consists in the most predicted class on the frames of the sequence (excluding "neutral").

Table 3.1: Micro and macro accuracies (std dev. in brackets) at image and video (sequence) level of the full-face-based classifiers (appearance and shape representations) and of an ensemble of the 15 classifiers (average of 15 outputs, both shape and appearance). Results without coherence constraints (None), with Part-based coherence and Temp-oral coherence (results where coherence improves the accuracy are in bold).

	Images						Videos					
	% Micro Acc			% Macro Acc			% Micro Acc			% Macro Acc		
	None	Part	Temp	None	Part	Temp	None	Part	Temp	None	Part	Temp
$\texttt{Face}_{app.}$	78.9 (3.6)	78.0 (2.0)	**81.1** (3.0)	71.2 (2.8)	**72.8** (2.2)	**72.2** (7.4)	75.3 (5.1)	**77.0** (3.4)	**80.0** (2.9)	64.0 (3.2)	**66.8**(3.1)	**64.4** (10.3)
$\texttt{Face}_{shape}$	71.8 (3.0)	**71.9** (3.1)	**72.5** (2.9)	61.1 (2.9)	**61.3** (3.0)	**62.1** (2.7)	68.5 (3.0)	68.1 (3.1)	**69.4**(2.9)	54.0 (2.9)	53.5 (3.0)	**55.5** (2.7)
$\texttt{Avg}_{all}$	73.7 (4.1)	71.4 (3.1)	72.1 (4.8)	71.9 (3.9)	70.2 (3.3)	69.7 (3.7)	78.3 (4.9)	77.9 (2.5)	**80.4** (5.5)	65.6 (6.5)	**65.9** (3.9)	64.8 (7.4)

the full set of 15 classifiers (thus mixing appearance and shape data).

Temporal coherence always improves the quality of the face-based classifiers, up to 5% in the case of sequences (micro). In the case of macro-accuracy we observe larger standard deviations, that are due to the effects of the predictions on the classes with a smaller number of examples. [3] Such classes are less-frequently predicted, and asking for a strong temporal regularization sometimes further reduces such frequency. Coherence among parts helps in a less evident manner, especially when using shapes. Shape is less informative than appearance, resulting in a performance drop of $\approx$ 10%. The average-based classifier is only in some cases better than the face-based ones. Constraints are less effective in this case (even if we get a strong micro accuracy in videos + temporal coherence). This suggests that, in this case, mixing the 15 classifiers together is not a promising direction, mostly because some of them have low performances that can degrade the average quality of the system. This model can be compared with other approaches taking the full-face classifier based on appearance, enforced during the training by the other classifiers through the coherence constraints.

To gain better insights about the last comment, Table 3.2 reports the accuracies for all the part-based classifiers. The mouth area is a very effective input for facial expression recognition, that can sometimes compete with the full-face. This is more

[3] In the case of full-face-based classifier (appearance), we selected the optimal λ_t using image-level predictions on the validation data, leading to $\lambda_t = 10^{-8}$ and $\lambda_t = 10^{-2}$ in the case of micro and macro accuracy, respectively.

Table 3.2: Micro and macro accuracies (std dev. in brackets) at image and video level of all the part-based classifiers (appearance and shape representation). Results without coherence constraints (None), with Part-based coherence and Temp-oral coherence (results where coherence improves the accuracy are in bold).

	Images						Videos					
	% Micro Acc			% Macro Acc			% Micro Acc			% Macro Acc		
	None	Part	Temp	None	Part	Temp	None	Part	Temp	None	Part	Temp
Mouth$_{app.}$	70.5 (3.5)	68.6 (3.0)	**72.8** (2.6)	71.5 (6.7)	70.8 (5.8)	**73.3** (4.4)	77.5 (7.7)	72.3 (9.0)	75.7 (6.4)	73.0 (9.5)	66.4 (8.4)	69.9 (8.7)
Left-eye$_{app.}$	42.3 (6.0)	41.4 (6.0)	40.0 (4.2)	41.3 (6.5)	39.1 (4.9)	38.5 (3.9)	49.4 (8.4)	**50.6** (4.1)	47.2 (5.9)	42.7 (5.8)	41.3 (2.7)	40.2 (6.3)
Right-eye$_{app.}$	42.0 (5.6)	42.0 (7.3)	40.6 (5.2)	40.8 (5.7)	40.5 (5.7)	38.8 (5.6)	46.8 (2.3)	**47.2** (4.9)	**47.7** (2.9)	39.8 (1.7)	39.2 (3.0)	38.9 (3.7)
Left-eyebrow$_{app.}$	40.5 (6.8)	37.7 (7.3)	38.4 (9.1)	40.1 (6.1)	37.4 (7.5)	37.6 (8.4)	43.0 (9.7)	41.7 (9.2)	42.1 (11.1)	35.2 (7.7)	34.3 (9.1)	34.3 (9.6)
Right-eyebrow$_{app.}$	40.1 (2.5)	39.7 (2.4)	**40.4** (2.9)	40.1 (3.5)	39.5 (2.8)	**40.3** (3.1)	43.4 (4.6)	42.5 (5.5)	**43.8** (3.2)	36.5 (6.6)	35.6 (6.8)	35.9 (4.0)
Nose$_{app.}$	43.6 (2.9)	**44.1** (5.5)	43.4 (4.0)	41.6 (3.4)	**42.4** (4.8)	**42.0** (3.7)	44.3 (4.9)	**47.7** (5.1)	**47.2** (2.8)	35.4 (4.3)	**38.8** (4.3)	**38.9** (3.1)
Mouth$_{shape}$	64.3 (2.3)	63.8 (3.5)	63.4 (3.2)	64.4 (4.7)	63.4 (4.8)	**66.2** (4.9)	71.9 (2.5)	**74.0** (3.7)	70.6 (2.8)	64.3 (4.2)	**66.1** (6.0)	**67.3** (5.0)
Left-eye$_{shape}$	35.8 (3.4)	34.5 (3.7)	35.2 (2.6)	33.2 (3.9)	33.0 (3.4)	32.5 (2.3)	45.1 (5.8)	44.7 (8.5)	45.1 (4.5)	36.6 (7.1)	**37.2** (6.1)	**38.3** (4.1)
Right-eye$_{shape}$	40.7 (3.2)	40.6 (2.7)	**41.5** (3.0)	36.9 (2.4)	**37.2** (2.1)	**37.9** (2.0)	51.9 (2.2)	**52.8** (3.7)	**56.2** (3.7)	39.4 (3.1)	**41.5** (3.3)	**44.9** (3.9)
Left-eyebrow$_{shape}$	31.2 (4.4)	31.0 (3.8)	30.1 (3.5)	31.8 (1.8)	**31.9** (2.0)	31.7 (3.7)	36.2 (6.7)	34.5 (3.4)	34.9 (3.5)	28.7 (5.1)	28.7 (3.0)	**29.3** (4.1)
Right-eyebrow$_{shape}$	34.3 (4.2)	33.9 (3.7)	34.1 (3.5)	34.3 (5.2)	33.4 (4.5)	33.6 (4.9)	40.4 (5.0)	40.0 (5.9)	**41.3** (6.7)	33.9 (5.6)	33.1 (5.0)	33.8 (7.0)
Nose$_{shape}$	30.8 (3.7)	30.4 (3.2)	**30.9** (4.2)	30.6 (5.6)	**31.0** (5.0)	**31.6** (5.2)	37.5 (5.0)	35.7 (3.7)	34.0 (1.4)	31.4 (5.4)	28.5 (5.6)	**31.8**(4.4)
Jaw$_{shape}$	37.4 (3.7)	37.2 (3.7)	37.0 (3.5)	34.1 (4.6)	**34.9** (4.3)	33.8 (4.0)	40.9 (2.5)	40.9 (2.1)	40.0 (3.7)	30.5 (2.5)	**31.3** (2.7)	29.8 (2.7)

evident in the case of videos, when comparing shape-based representations of face and mouth. As expected, the other parts are worse than the full-face, since they are just local views. The addition of both coherences sparsely helps in improving the local classifiers, with a preference toward temporal coherence. The worst results are obtained by eyebrows and nose in shape-based classification. Interestingly, the eye-based predictors score the most effective results after face and mouth in video sequences. While their appearance representation is altered when eyes get closed, their shape representation is more stable. The results on left eye and right eye are a bit different and this is due to the fact that wrinkles can be asymmetric, or that an eye can be closed, or to the variation of lighting and pose. This analysis suggests that an accurate choice of a sub-portion of the face parts could significantly help the part-based coherence constraint (since some of the parts are not very informative).

As already mentioned in Section 3.5, temporal coherence yields homogeneous predictions over the sequences, without oscillations along the temporal axis. In Figure 3.11 we report an example showing that temporal coherence produces an uniform trend in the predictions on the sequence ("surprise" emotion is sketched). The model without temporal coherence produces an oscillating trend on the sequence, predicting also wrong emotions as "disgust" and "anger" as long as time passes. Differently, temporal coherence leads to a smooth variation from the neutral state to the emotional one.

In Table 3.3 we show the results on single emotion classes for face and mouth appearance-based classification, focussing on the case where no coherence is introduced and the ones with a selection of the best $\lambda_c > 0$ and $\lambda_t > 0$ from the previously described experiments. "Fear" and "sadness" classes are difficult to classify because they do not involve strong facial movements, while "happiness" and "surprise" are easy to recognize. The mouth-based model has difficulties with the

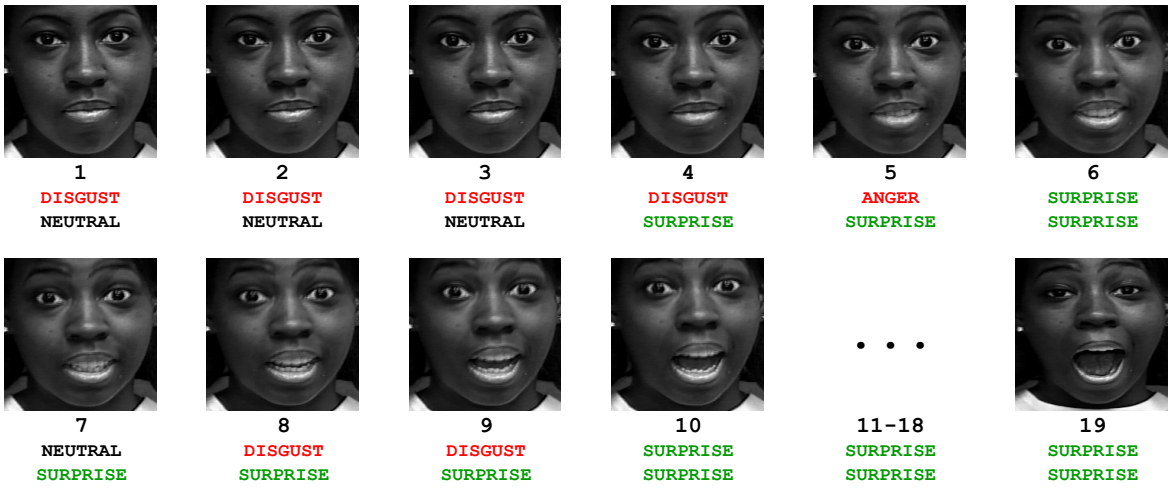


Figure 3.11: Predictions in a video sequence that starts with a neutral expression and ends with surprise. For each frame we report the prediction of the model without coherences (top) and the prediction with temporal coherence (bottom). The wrong predictions are in red.

Table 3.3: Accuracies on each class of full-face and mouth classifiers (appearance). Results without coherence constraints, with PART-based coherence and TEMP-oral coherence (results where coherence improves the accuracy are in bold).

	IMAGES							VIDEOS					
	Anger	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*	*Neutral*	*Anger*	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*
face$_{app.}$ NONE	73.7	69.2	56.1	92.5	29.5	96.1	81.1	77.1	62.2	33.3	90.9	25.0	95.4
face$_{app.}$ PART	68.0	**78.2**	**75.2**	**98.2**	24.3	**97.4**	68.6	68.6	**71.1**	**53.3**	90.9	20.0	**96.9**
face$_{app.}$ TEMP	**77.1**	**81.8**	50.0	**97.5**	26.2	95.5	**81.8**	77.1	**73.3**	**40.0**	**98.2**	25.0	**96.9**
mouth$_{app.}$ NONE	66.4	69.6	59.4	92.7	75.6	96.6	40.2	77.1	73.3	46.7	78.2	75.0	87.7
mouth$_{app.}$ PART	66.4	**81.8**	**65.1**	**95.0**	59.6	95.5	32.2	62.9	**77.8**	46.7	74.6	55.0	81.5
mouth$_{app.}$ TEMP	**67.8**	**80.4**	58.8	**94.8**	72.0	95.2	**44.1**	74.3	**77.8**	40.0	74.6	65.0	87.7

"neutral" class, since some emotions do not evidently alter the mouth area (the face model does not show this issue). In the "sadness" class, where the face-based model scores low accuracies, the mouth-based classifier is much more performant. This suggests that the face-related network has difficulties in developing a generalizable representation for the whole face to identify "sadness". Larger training data could help in this case. Temporal coherence shows better performance on "neutral", "anger" (image-level only), and "disgust" emotions. It is also helpful in the "happiness" class, where the face model performs a close-to-flawless classification. Introducing coherence among parts improves the recognition of "disgust", "fear" (face only), "happiness" (image-level only), and it slightly improves the accuracy of "surprise" for the face-based predictor.

In addiction to these results, we report that eye-based recognition reaches very good results on the "surprise" class; the accuracy of right-eye classifier with temporal coherence is 88.2%. This happens because the eyes are wide open in surprise expressions, and thus easily recognizable. Differently, the "neutral" class is not rec-

Table 3.4: Confusion matrix for the seven classes. Rows: actual emotions; columns: predicted emotions.

	Anger	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*	*Neutral*
Anger	165	24	0	0	0	0	22
Disgust	9	159	0	0	2	4	20
Fear	6	0	56	22	0	9	19
Happiness	0	0	8	302	0	0	0
Sadness	50	13	0	0	32	0	20
Surprise	1	0	0	0	0	292	12
Neutral	48	7	0	0	13	11	351

ognizable at all from the eyebrows. Nose-based classification (appearance) reaches an accuracy of 79.4% with temporal coherence in the "disgust" class, where the nose is wrinkled.

In Table 3.4 we report the confusion matrix for the seven classes for one of the best models described so far (similar considerations hold for almost all the models). As already mentioned, we generated 5 randomizations of the dataset, so we calculated 5 confusion matrices, one for each split $i \in \{1, ..., 5\}$ of the test set, where the predictions are made by the model trained on the training set i, and then we summed the values of each matrix, thus obtaining a single matrix. "Anger" is confused with "disgust" because in both classes the eyebrows are very similar. "Fear" is confused with "happiness" since both emotions are characterised by open mouth and, in particular, clenched teeth. "Sadness" is difficult to recognize both because it is not a very enhanced expression and because the class is poorly represented in the dataset. As a matter of fact, this emotion is confused with "anger", due to the similar features on the mouth area. All classes, except "happiness", are sometimes misclassified with "neutral" in the examples where the expressions are just slightly evident.

If we do not consider a few exceptions (that are not worth being reported), we did not obtain results with evident improvements when exploiting both the temporal and part-based coherence. On the one hand this is due to the larger difficulty in finding the optimal parameters that weigh the two constraints, on the other hand it might also be due to the relatively small size of the dataset.

We performed other experiments including the coherence constraint between appearance and shape (Eq. 3.7), and selecting the best value of $\lambda_r > 0$ by cross-validation. We observed that this type of coherence helps to further improve the accuracies of models that also exploit the temporal coherence, that is the case that we consider in Table 3.5. In particular, we get improvements of the overall micro and macro accuracies of the face-based model (appearance), mostly due to improvements of the quality of the classification of some specific emotions, such as "anger",

Table 3.5: Micro and macro overall accuracies (All) and plain accuracies on each class of full-face classifier (appearance). Results without coherence constraints, with TEMP-oral coherence only and with TEMP-oral and coherence between appearance and shape (results where coherence between appearance and shape improves the accuracy respect to temporal coherence only are in bold).

	IMAGES								
	Micro Acc	*Macro Acc*	*Anger*	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*	*Neutral*
face$_{app.}$ NONE	78.9	71.2	73.7	69.2	56.1	92.5	29.5	96.1	81.1
face$_{app.}$ TEMP	81.1	72.9	77.1	81.8	50.0	97.5	26.2	95.5	81.9
face$_{app.}$ TEMP+APPSHAPE	80.7	**72.9**	73.7	81.2	49.0	94.8	**31.3**	94.6	**85.8**
	VIDEOS								
	Micro Acc	*Macro Acc*	*Anger*	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*	*Neutral*
face$_{app.}$ NONE	75.3	64.0	77.1	62.2	33.3	90.9	25.0	95.4	–
face$_{app.}$ TEMP	80.0	68.4	77.1	73.3	40.0	98.2	25.0	96.9	–
face$_{app.}$ TEMP+APPSHAPE	**80.4**	**69.9**	**80.0**	**80.0**	**46.7**	89.1	25.0	**98.5**	–

"disgust" (even by 6.7%), "fear" and "surprise", at sequence (video) level. Differently, improvements are less evident at frame level, with the exception of the "neutral" class, where the decision of the classifier is more stable than when using appearance only.

A demo with the model described in this chapter has been developed. [4] It has been trained in a larger quantity of data, taking more datasets and augmenting the images. Rotations, translations and illumination changes are performed. In real-time, the system detects the face from webcam and classifies the expression in the six universal emotions plus neutral, and it outputs the probability distribution on the seven classes.

3.7 Occlusions

Recognizing facial expressions in presence of occlusions is challenging and is a real scenario, especially in the current situation, where mouth and nose are covered by masks. Even if appearance-based representations generally lead to better results than shape-based ones, we experienced that they are less effective when there are occlusions covering portions of the face area. In order to investigate the robustness of our model, we manually created some test images that include occlusions: we took the last frame (the more expressive) of each sequence of the CK+ dataset (309 images), and we covered some face parts, such as mouth or nose. We selected one of the best models described so far, and in Table 3.6 we report the accuracies associated to the cases in which the shape-based representation performs better than appearance (full-face classifier). In the case of "anger", when the mouth is covered, the accuracy of the appearance-based classifier is only 35.6% while the shape-based one

[4] Demo is freely accessible at `https://sailab.diism.unisi.it/facial-emotions/`

Table 3.6: Accuracies (Acc) of full-face classifiers (appearance and shape) on images with occlusions.

Emotion	Covered Part	% Acc_{app}	% Acc_{shape}
anger	`mouth`	35.6	75.6
disgust	`mouth`	61.0	78.0
disgust	`nose`	81.4	83.1
happiness	`nose`	66.7	98.6
sadness	`mouth`	67.9	71.4
sadness	`nose`	53.6	75.0
surprise	`nose`	95.2	98.8

yields 75.6%. As we have discussed in Section 3.6, this emotion is not easy to recognize, and covering an important part such as the mouth makes the task even more difficult. Differently, the shape-based classifier can capture more robust features that go beyond the appearance even when the mouth is occluded. Another considerable improvement of the shape with respect to appearance can be observed when the nose is occluded in images with happy expressions. In fact, the appearance-based classifier sometimes confuses them with "fear", where the mouth is in general as open as in "happiness". The good performance of the shape-based representation is due to the fact that it is a more compact and less variable representation than the appearance-based one. As a result, the CNNs are less influenced by the occlusion phenomenon that we simulated.

In Figure 3.12 we report some examples with occlusions in which the shape-based classifier predicts the right emotion, while the appearance-based gets it wrong. The first example from the left represents "anger", but when the mouth is covered, the appearance-based classifier predicts "fear". The mouth is clearly different in these two emotions, so occluding it, the appearance-based classifier cannot grasp the right expression from the other face parts. The second and third examples represent "disgust", and when the wrinkled mouth is covered the appearance classifier predicts "fear", while if the wrinkled nose is occluded the expression is confused with "anger". In the following case, covering the nose, the appearance-based classifier predicts "fear" instead of "happiness", because it focuses on the open mouth, not considering the more relaxed nose. In the last example, depicting "sadness", the appearance classifier predicts "surprise", not seeing if the mouth is wide open or down.

3.8 Discussion

Facial expressions are a powerful tool to detect and communicate emotions. Recognizing facial expressions from static images or video sequences is a widely studied

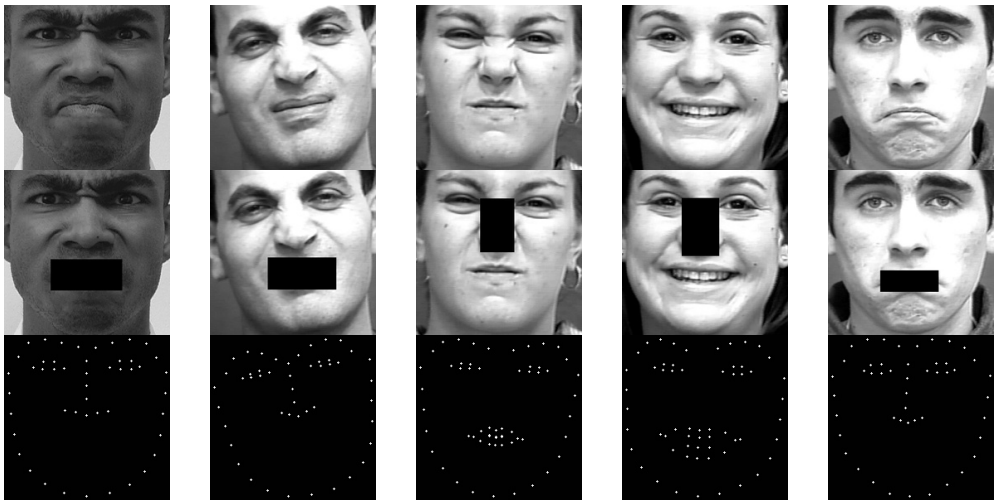


Figure 3.12: Examples of images with occlusions where the shape-based classifier predicts the right emotion (green) whereas the appearance-based classifier gets wrong (red). From top to bottom: the original image (appearance), the images with occlusion (appearance, shape), the wrong prediction of the appearance-based classifier (red) and the right prediction of the shape-based classifier (green).

but still challenging problem. The recent progresses obtained by deep neural architectures, or by ensembles of heterogeneous models, have shown that integrating multiple input representations leads to state-of-the-art results. In particular, the appearance and the shape of the input face, or the representations of some face parts, are commonly used to boost the quality of the recognizer.

We presented a Convolutional Neural Networks-based approach to facial expression recognition, that processes distinct face parts, represented using visual (appearance) or shape-only features. In the latter case, we treated shape as a generic input of the learnable model, without manually engineering its representation. We studied the importance of the different representations on the task at hand, showing an analysis that involved all the considered face parts, and reporting results of experiments on a popular dataset composed of six basic emotions, plus the neutral case. We found that the mouth area is a very effective input for recognizing emotions, that can sometimes compete with the full-face.

We proposed the introduction of coherence constraints among the face-part predictors, between predictions on consecutive time instants, and between appearance and shape representations, casting the learning problem in the Semi-Supervised setting and using video data. These types of coherence are not limited to CNNs, but can be combined with other deep architectures. To exploit the coherence constraints so defined it is required that the model outputs probability distributions.

Our experimental results have shown that coherence constraints improve the quality of the recognizer, especially in the case of temporal coherence combined with coherence between appearance and shape, thus offering a suitable basis to profitably exploit unsupervised video sequences. We found out that, in this case, averaging the outputs of the full set of 15 classifiers (mixing appearance and shape data) is not a promising direction, mostly because some of them have low performances. The best solution seems to be considering only the full-face appearance-based classifier, that has received improvements by the other face parts through the part-based coherence, and by the shape-based representation through the coherence between appearance and shape. To improve the performance of the full-face predictor more datasets can be used for training.

The potentiality of this approach, in addition to the possibility to exploit unsupervised data, is that it can be used to study single face parts and to recognize expressions in less straightforward conditions, as in presence of occlusions or illumination changes. As matter of fact, we have seen that, when some face parts are covered, the shape-based classifier detects the expressions better than the appearance-based classifier. This is important especially in the current situation where mouth and nose are covered by masks, so it is very difficult to detect the facial expression with a simple full-face classifier based only on appearance.

Chapter 4

Text Emotion Recognition

In this chapter we deal with the task of text emotion recognition, which classifies textual data according to a large set of classes. A similar task concerning the topic of emotions in text is sentiment analysis, that instead classifies a text in positive or negative (sometimes neutral).

We propose a neural network-based model to jointly learn the task of *emotion detection* and the task of *predicting Facebook reactions* [50] from text. It consists of a *Bidirectional Recurrent Neural Network* (*BRNN*) to encode the input sentence, and two predictors associated with the considered tasks. Predictors are not independent, but are linked by prior knowledge on the relationships between emotion detection and reaction prediction. Such knowledge is represented by *First Order Logic* (*FOL*) *formulas* (Section 2.2), which allow us to naturally express how reactions are connected to emotion classes and vice-versa. Following the framework of Learning from Constraints, FOL formulas are converted into polynomial constraints, through *t-norms*, and softly enforced into the learning problem, thus tolerating some violations.

Text emotion recognition is a challenging task due to language complexity and ambiguity. Moreover irony and sarcasm can be used to express negative sentiments with positive words, therefore for a machine it is difficult to detect the emotion without understanding the context. This task, as well as sentiment analysis, has several application, such as business, psychology, education. For example, conversational systems can adapt their language in function of the perceived user emotions, digital marketing platforms can customize recommendations, social media marketing strategies can be changed in function of the estimated emotions triggered when posting contents, comments on social networks can help to predict the results of some elections, systems can warn users before sending an inappropriate message.

News, blogs, reviews, posts or comments on social networks are a precious source of information for building large datasets of annotated multimedia contents, or for mining users' behaviours and other user-related information. We focus on the case of Facebook, where users can express their feeling on a post through the so called

Figure 4.1: Facebook reactions: `LIKE`, `LOVE`, `HAHA`, `WOW`, `SAD`, `ANGRY`.

"reactions", that are `LOVE`, `HAHA`, `WOW`, `SAD`, `ANGRY`, together with the widely known `LIKE` (Figure 4.1). While `LIKE` represents a universal and generic expression of a positive feedback, the other reactions are more fine-grained, and somewhat related to the categories of emotions. However, this relationship is weak and distant, since some reactions can be loosely associated to emotional categories, sometimes with large ambiguity. For example, `WOW` expresses "surprise" but it can be also used to describe contents where the astonishment is accompanied by "fear". Moreover, Facebook reactions are the outcome of a tagging process where users might follow superficial and strongly subjective criteria to react. For instance, taggers might attach `SAD` or `ANGRY` reactions to comments from users that are associated to a rival sport team or to a political party that is in contrast with the one of the tagger, independently on the precise content of the text of the comment.

This chapter is organized as follows. In the next two sections the related works and the main datasets containing texts labeled with emotions are mentioned. Section 4.3 describes the data organization and the proposed model, while Section 4.4 focusses on the logic constraints. Experimental results are provided in Section 4.5.

4.1 Related Works

In the case of categorical emotion detection, sentences are usually classified into the six universal emotions defined by Ekman (see Appendix A.1). The task of emotion detection from text has been the subject of a large number of studies, mostly distinguished into lexicon-based and machine learning-based approaches (or hybrid solutions). Lexicon-based approaches employ linguistic models or prior knowledge for the classification task, and they essentially give a score to a sentence using a predefined sentiment lexicon [66, 129]. The advantage is that they do not need labelled data, while a disadvantage is that sentences describing emotions with words that do not appear in the vocabulary, would be not well classified. In [1] the authors propose an unsupervised context-based emotion detection method that does not rely on any affect dictionaries or annotated training data. A constraint optimization framework based on lexicon is presented in [143]. Machine learning-based methods usually exploit supervised learning algorithms trained on annotated corpora. The approach of [112] focusses on Twitter data, while [18] uses a heterogeneous emotion-annotated dataset to recognize the six basic emotions. Herzig [58] focuses on an ensemble model, strongly exploiting pre-trained, dense word-embedding rep-

resentations.

In the last years, deep learning has become popular also in the task of text emotion recognition. In [69] recurrent neural networks and transfer learning are combined, where the network is pre-trained for sentiment analysis task, while the output layer is subsequently tuned to the task of emotion recognition. In [19] a LSTM-based approach is proposed to detect only three emotions (anger, happiness and sadness) in textual dialogues, combining both semantic and sentiment based representations. Attention mechanisms have been also employed for emotion recognition from text [10, 89]. In these two specific works, attention is combined with deep architectures to detect affect in English tweets.

Finally, we observed that recently some works about Facebook reactions have been developed. Some authors trained emotion classification models using Facebook reactions [110, 113], while others tried to learn to predict Facebook reactions in a given domain, bootstrapping the system with the outcome of emotion mining [70]. Reactions are usually manually mapped to (a subset of) the aforementioned universal emotions, providing a form of distant supervision.

4.2 Datasets

In this section an overview of textual datasets used for affective computing is provided. There are few textual corpora labelled with emotions, and most of them contain few sentences.

- **AffectiveText**. AffectiveText (or SemEval-2007) [128] contains 1,250 short newspaper headlines, taken from major newspapers as New York Times, CNN, BBC News. This dataset was created for an unsupervised competition and was split into 250 sentences of trial data and 1000 sentences of test data. Sentences are labeled with the six basic emotions, and each of them is scored in a range from 0 to 100.

- **ISEAR**. International Survey on Emotion Antecedents and Reactions [122] contains 7,666 sentences from questionnaires about emotional experiences covering anger, disgust, fear, joy, sadness, shame and guilt.

- **Fairy Tales**. Fairy Tales [4] contains 176 short stories by three authors: B.Potter, H.C.Anderson and Grimm's. There are 1,207 sentences, each of them is annotated with four labels, related to the primary emotion and mood of the annotators. The emotions considered are anger, disgust, fear, happiness, sadness, positive surprise, negative surprise and neutral.

- **Aman**. This dataset [5] contains 4,090 blog posts, which are retrieved using seed words that represent the six universal emotions. Four people manu-

ally annotated the sentences with the basic emotions plus two categories, i.e., mixed emotion and no emotion.

- **SemEval-2018**. This corpus [95] contains tweets in English, Arabic, and Spanish. It has been created for a competition which task was to classify tweets with moderate and high emotions. Each tweet is annotated with neutral or with one or more of eleven given emotions, which are anger, disgust, fear, joy, love, optimism, pessimism, sadness, surprise and trust. Also the intensity of the emotion label is provided as annotation.

- **SemEval-2019**. This corpus [20] consists of textual dialogues between two individuals. It has been created for a competition which task was to detect contextual emotions in text. Each conversation is either labeled as joy, anger, sadness, or others.

4.3 Model and Data Organization

We consider a multi-task setting where two predictors $p_r(x)$ and $p_e(x)$ operate on the same data x, that is a short input text. Such predictors are associated to the task of reaction classification (p_r) and emotion classification (p_e), respectively. In this work, both the tasks consist in predicting the most dominant reaction/emotion when processing a text x. In detail, $p_r(x) \in [0,1]^R$ outputs a probability distribution over R reactions, and, analogously, $p_e(x) \in [0,1]^E$ outputs a probability distribution over E classes of emotions. We select the emotion-reaction pair associated to the largest probabilities.

Following the classical pipeline of several machine learning-based approaches in Natural Language Processing (NLP), the input text x is tokenized into words $x_0, \ldots, x_t$ belonging to a fixed-size vocabulary. Each word is embedded into a learnable latent dense representation, also known as "word embedding", and an LSTM recurrent neural network [59] processes the sequence of word embeddings in both directions (BRNN [125]). The forward and backward states are then concatenated, producing an embedded latent representation of x, that is provided as input to Feed-Forward Networks (FNNs) with softmax activation functions in the output layers, thus implementing p_r and p_e, respectively. The choice of sharing the same latent representation of x with both predictors is due to the fact that the two prediction tasks are certainly correlated. Finally, during the training stage, the FNNs are connected by constraints that are devised from FOL rules, and that will be described in Section 4.4. The whole architecture is reported in Fig. 4.2.

Our model is trained using a heterogeneous collection of text $\mathcal{T}$ of partially labeled and unlabeled data, composed by the union of three disjoint sets, $\mathcal{T}_r$, $\mathcal{T}_e$, $\mathcal{T}_u$, that, in turn, consist of pairs (x, y), where y is either a reaction label, an emotion

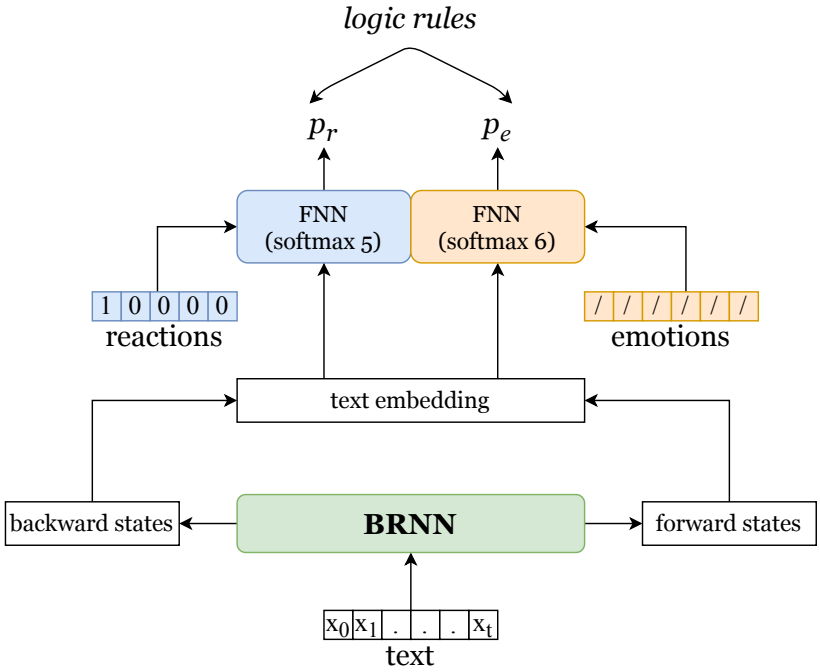


Figure 4.2: The proposed model. The input text is tokenized into words $x_0, \ldots, x_t$ belonging to a vocabulary. Each word is embedded into a learnable dense representation (word embedding). A BRNN processes the sequence of word embeddings in both directions. The forward and backward states are concatenated, producing an embedded latent representation of the text, which is provided as input to 2 FNNs, one for reactions and one for emotions. The two FNNs end with softmax activation functions, that output the probability distribution on reactions p_r and on emotions p_e. When training the network, we feed it with text either labeled with emotions or reactions, and logic constraints bridge the two predictors p_r and p_e.

label, or a dummy placeholder (i.e., unlabeled data), respectively. (i) The set $\mathcal{T}_r$ is a collection of Facebook posts, each of them labeled with one out of $R = 5$ reaction classes, encoded with a one-hot vector y_r of size R. We did not consider the class `LIKE`, since it is too generic, and we selected the most frequent reaction class in each post. Moreover, $\mathcal{T}_r$ is composed only by those posts with at least τ reaction hits in total ($\tau = 20$ in our experience), and where the most frequent reaction has a number of hits that is greater than the number of hits of all the other reactions scaled by a factor γ (we set $\gamma = 0.4$). (ii) The set $\mathcal{T}_e$ is a collection of sentences, each of them labeled with one of the $E = 6$ universal emotions, encoded with a one-hot vector y_e of size E.[1] We exploited existing datasets to build $\mathcal{T}_e$, keeping only the most dominant emotion in the case of multi-labeled data. (iii) Finally, the set $\mathcal{T}_u$ is a collection of unlabeled text, that in our experience, consists of a large collection of Facebook posts without reactions. Each sample is paired with a dummy label vector $y_{\texttt{none}}$. This set

[1] In our experience we did not consider the *neutral* class, that, however, could be easily introduced in the proposed model.

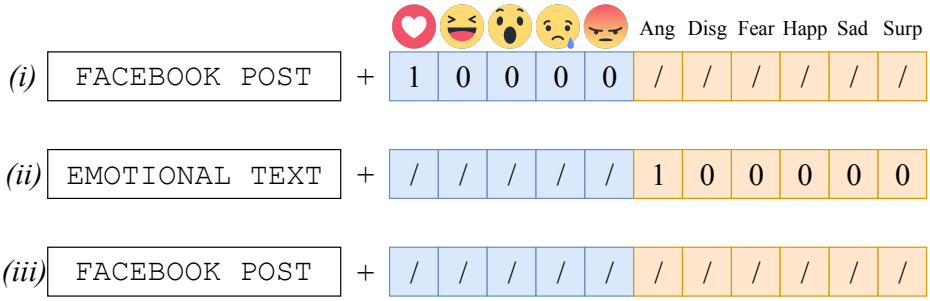


Figure 4.3: Sample representatives of the types of data included in our heterogeneous training set. (i) A Facebook post paired with the reaction label LOVE (encoded with the blue 1-hot vector) and no emotion labels. (ii) Text paired with the emotion class *anger* (orange 1-hot vector) and no reaction labels. (iii) An unlabeled Facebook post.

is exploited to enforce the logic constraints in space regions that are not covered by the labeled portion(s) of the training set. This allows the model to learn predictors that better generalize the information associated to the logic formulas. A sketch that summarizes the types of training data used in this work is reported in Fig. 4.3.

4.4 Jointly Learning Reactions and Emotions with Constraints

Before introducing our approach, we mention that the simplest way to bridge the tasks of emotion and reaction classification is to generate artificial labels, i.e., to define a fixed mapping between emotions and reactions and augment the training data with these new labels (see, for example [110], Table 1). Considering the emotion/reaction classes of Section 4.3, a reasonable mapping from reactions to emotions, represented with the notation "ground truth" $\rightarrow$ "new label", is the following one: LOVE $\rightarrow$ *happiness*, WOW $\rightarrow$ *surprise*, HAHA $\rightarrow$ *happiness*, SAD $\rightarrow$ *sadness*, ANGRY $\rightarrow$ *anger*. Similarly, we can map emotions to reactions: *anger* $\rightarrow$ ANGRY, *disgust* $\rightarrow$ ANGRY, *fear* $\rightarrow$ WOW, *happiness* $\rightarrow$ HAHA, *sadness* $\rightarrow$ SAD, *surprise* $\rightarrow$ WOW. However, this manual conversion is rigid and sometimes ambiguous. For example, no reactions are converted into labels of classes *fear* and *disgust*, and no emotions are mapped into the reaction LOVE.

We propose to describe the mappings between emotion and reaction classes using FOL formulas and to develop a multi-task system that learns from them, following the framework of Learning from Constraints. Each class is associated to a predicate, whose truth degree is computed using a function that, for simplicity, we indicate with the name of the class itself. These predicates can be seen as the components of the vectorial functions $p_r(x)$ and $p_e(x)$, i.e., $p_r(x) = [\texttt{HAHA}(x), \texttt{SAD}(x), \texttt{ANGRY}(x), \texttt{LOVE}(x), \texttt{WOW}(x)]$, and $p_e(x) = [anger(x), disgust(x), fear(x), happiness(x), sadness(x),$

surprise(x)]. We define the following rules,

$$\forall x\, \texttt{HAHA}(x) \Rightarrow \textit{happiness}(x) \quad (4.1)$$
$$\forall x\, \texttt{SAD}(x) \Rightarrow \textit{sadness}(x) \quad (4.2)$$
$$\forall x\, \texttt{ANGRY}(x) \Rightarrow \textit{anger}(x) \vee \textit{disgust}(x) \quad (4.3)$$
$$\forall x\, \texttt{LOVE}(x) \Rightarrow \textit{happiness}(x) \quad (4.4)$$
$$\forall x\, \texttt{WOW}(x) \Rightarrow \textit{surprise}(x) \vee \textit{fear}(x) \quad (4.5)$$
$$\forall x\, \textit{anger}(x) \Rightarrow \texttt{ANGRY}(x) \quad (4.6)$$
$$\forall x\, \textit{disgust}(x) \Rightarrow \texttt{ANGRY}(x) \quad (4.7)$$
$$\forall x\, \textit{fear}(x) \Rightarrow \texttt{WOW}(x) \quad (4.8)$$
$$\forall x\, \textit{happiness}(x) \Rightarrow \texttt{HAHA}(x) \vee \texttt{LOVE}(x) \quad (4.9)$$
$$\forall x\, \textit{sadness}(x) \Rightarrow \texttt{SAD}(x) \quad (4.10)$$
$$\forall x\, \textit{surprise}(x) \Rightarrow \texttt{WOW}(x)\,. \quad (4.11)$$

Notice that these rules do not include negations, that is due to the probabilistic relationship (softmax) that we introduced in the output of the predictors (if a function goes toward 1, all the others will automatically go toward 0).[2]

We defined our FOL formulas after having analyzed the content of various Facebook posts and the associated reactions. Implications 4.3-4.5-4.9 include an ambiguous mapping, modeled with the $\vee$ operator (disjunction). The second predicate that we reported in each disjunction corresponds to a less trivial mapping that, at a first glance, might not always seem obvious. However, in our experience, we found these cases to be more frequent than expected. We report an example for each of them: `WOW` could be *fear* instead of *surprise* (Eq. 4.5),

> *Snake on a plane: Frightening moment on an Aeromexico flight when a large snake fell from overhead mid-flight. The flight made a quick landing and animal control took the stowaway into custody.*

Emotion *happiness* could be converted into `LOVE` instead of `HAHA` (Eq. 4.9),

> *When I got a wedding ring of diamond from the boy I loved.*

The reaction `ANGRY` could be eventually mapped into *disgust* (Eq. 4.3),

> *The San Antonio police chief said that former officer Matthew Luckhurst committed a vile and disgusting act that violates our guiding principles.*

Our rules are converted into real-valued polynomials by means of t-norms, that are functions modeling the logical AND whose output is in $[0, 1]$ (as seen in Section 2.3). We used the Product t-norm, where the logical AND is simply the product of the involved arguments. In turn, this choice transforms $a \Rightarrow b$ into the polynomial

[2]We did not write the rules in a more compact form using the double implication $\Leftrightarrow$, since we will differently weigh the impact of some of them, as it will be clear shortly.

$1 - a + a \cdot b$. Constraining the FOL formula to hold true leads to enforcing the t-norm-based polynomials to be 1, so we get equality constraints, e.g., $1 - a + a \cdot b = 1$ in the previous example. We introduce these constraints into the learning problem in a soft manner using penalty functions, so that the system might decide to violate some of them for some input x (in our implementation, we used the penalty $-\log(\cdot)$).

Formally, the multi-task function that we minimize to learn the model is

$$\sum_{(x,y_r)\in\mathcal{T}_r} L\left(p_r(x), y_r\right) + \sum_{(x,y_e)\in\mathcal{T}_e} L\left(p_e(x), y_e\right) + \sum_{j=1}^{J} \sum_{(x,\cdot)\in\mathcal{T}} w_j \phi_j\left(p_r(x), p_e(x)\right), \tag{4.12}$$

where we avoided reporting the scaling factors in front of each term of the summation, to keep the notation simpler. The function L is the cross-entropy loss. The first term takes only the data labeled with reactions and it is the supervised loss on reactions. The second term is the supervised loss on emotions, and the last term contains the logic rules. With ϕ_j we indicate the penalty term associated to the j-th FOL formula, weighed by the scalar $w_j > 0$. With J we indicate the number of logic rules (in this case $J = 11$). Each ϕ_j might only consider some of the output components of $p_r(x)$ and $p_e(x)$, depending on the FOL formula that it implements. For example the penalty term for the first rule is

$$\phi_1(p_r(x), p_e(x)) = -\log(1 - \texttt{HAHA}(x) + \texttt{HAHA}(x) \cdot \mathit{happiness}(x)),$$

where the FOL formula is converted into a real-valued function by means of the product t-norm (Section 2.3).

Notice that FOL formulas are constrained to hold true on all the available training data, including the large collection of unlabeled text $\mathcal{T}_u$. This allows the system to learn predictors that fulfil the FOL rules in regions of the input space that might not be covered by the labeled data, thus increasing the information transfer between the two tasks. Thanks to this formulation, we can differently weigh the impact of each constraint in function of the confidence we have on it, tuning the parameters w_j. For example, constraints associated to formulas 4.4-4.7-4.8 are weaker than the other ones, and we decided to keep their weight small. The implication 4.7 is not always suitable, because `LOVE` reaction is sometimes used for compassion or affect, while the formulas 4.7 and 4.8 are weaker because the opposite mappings imply a disjunction.

4.5 Experimental Results

In order to evaluate the proposed model, we created a heterogeneous data collection that follows the organization described in Section 4.3. In particular, we considered a large public dataset of *Facebook posts* that are scraped from Facebook pages of

Table 4.1: Number of Facebook posts for each reaction, and number of *unlabeled* posts (top). Number of texts for each emotion class, covering three public datasets (bottom).

	LOVE	WOW	HAHA	SAD	ANGRY	*Unlabeled*	TOTAL
Facebook Posts	31801	13807	17552	16689	15775	100000	195624

	Anger	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*	TOTAL
Affective Text	91	42	194	441	265	217	1250
ISEAR	1087	1082	1089	1090	1083	0	5431
Fairy Tales	146	64	166	445	264	100	1185

newspapers, such as ABC News, BBC, CNN, The New York Times, The Wall Street Journal, The Washington Post, Time, Usa Today. [3] Data was filtered accordingly to what we described in Section 4.3, ending up with $\approx 200,000$ posts, out of which $100,000$ are left unlabeled. Then, we collected the most popular datasets, described in Section 4.2, containing text labeled with emotions, namely *AffectiveText*, *ISEAR*, and *Fairy Tales*. For the purpose of this experimentation, we took from AffectiveText the emotion with the highest score. From ISEAR we discarded the classes shame and guilt since they are not part of the universal emotions, and mapped "joy" to "happiness" (the class "surprise" is missing). From Fairy Tales we kept only sentences with four identical labels (three for the class "disgust", due to the small number of samples). In Table 4.1 we report the details of the data exploited in this experimentation.

We evenly divided our heterogeneous datasets into 3 splits, keeping the original data distribution among classes. Each split is further divided into training, validation and test sets, with special attention in preparing the test data. In particular, the test set is composed of 15% of the labeled Facebook posts, merged with one of ISEAR, Fairy Tales, Affective Text. As a matter of fact, each of such emotional datasets is small sized (considering the number of classes and the intrinsic difficulty of the learning task), and it has different properties w.r.t. the other two ones. We experienced that training and testing on subportions of the same emotional dataset leads to performances that do not reflect the concrete quality of the system when it is deployed and tested in a generic context. Differently, training and testing on different emotional datasets offers a more realistic perspective of the generalization quality of the resulting system. The training set includes 70% of the labeled Facebook posts and 80% of the two emotional datasets which are not present in the test set, plus the unlabeled Facebook posts. The validation set is composed of the remaining data, that is, 15% of labeled posts and 20% of the two emotional datasets which are not used as test set. We preprocessed all the data converting text to lowercase, removing URLs, standardizing numbers with a special token, removing brackets, separating

[3] `https://data.world/martinchek/2012-2016-facebook-posts`

punctuation and hashtags. Then, we created a vocabulary composed of the most frequent $10,000$ words and we truncated sentences longer than 30 words, to make them more easily manageable by the BRNN.

We evaluated architectures with differently sized word embeddings (from 50 to 300 units each), states of the BRNN (in the range $[50, 200]$), hidden layers (and number of units) of the final FNNs (up to 2 hidden layers). After a first exploratory experimentation, we focussed on models with word embeddings of size 100, BRNN with a hidden state composed of 100 units and final predictors with no hidden layers, that were providing the best results in the validation data. Then, we kept validating in more detail all the other model parameters (learning rate, the possibility of introducing drop-out right after the BRNN, weight of the logic constraints w_j). We considered the (macro) F1 scores on each task to evaluate the quality of our models, and we early stopped the training procedure whenever the average F1 score on the validation data was not increased after 20 epochs (keeping the model associated to the best F1 score found so far).

We compared the following models:

- PLAIN. The model of Fig. 4.2, without logic constraints ($w_j = 0, \forall j$).
- CONSTR. The same as PLAIN, but including logic constraints ($w_j > 0, \forall j$).
- ARTIFICIAL. The same as PLAIN, where the training data are augmented with artificially mapped classes as described at the beginning of Section 4.4.
- *+Emb*. Variant of the models above, based on pre-trained word embeddings of size 300 (the popular Google word2vec model).[4]

We first evaluate the quality of the system in the task of reaction prediction. In Table 4.2, we can appreciate how introducing logic constraints constantly improves the quality of the predictor in all the reaction classes. Using artificial labels from emotional data is far from giving the same benefits of logic constraints, and we did not experience advantages in using pre-trained word embeddings, that might be due to the inherent noise in the reaction prediction task.

Moving to the task of emotion classification, we report the results we obtained in the previously described test sets, that correspond to three different emotional datasets. In Table 4.3 we focus on testing in the ISEAR data. Logical rules always allow the model to improve the macro-averaged F1 scores. We notice that the F1 score on "disgust" and "fear" classes is largely better than when not using constraints. In fact, without exploiting the logical rules of Eq. 4.3 and 4.5 there is no transfer of information from reaction data, and the supervised portion of the training set is not enough to learn good predictors. Interestingly, this consideration does

[4]In this case, after our initial exploratory experimentation, we selected a BRNN with state size 200, and reaction predictor with a hidden layer of size 25.

Table 4.2: F1 scores on Facebook reactions (test data, averaged over the 3 data splits - std dev. in bracket). Three models are compared: without constraints (PLAIN), with constraints (CONSTR), with artificial labels (ARTIFICIAL). +*Emb*: variant with pre-trained word embeddings. Bold: cases in which constraints introduce improvements.

	LOVE	WOW	HAHA	SAD	ANGRY	Macro Avg
PLAIN	0.630 (0.009)	0.354 (0.008)	0.440 (0.009)	0.532 (0.014)	0.329 (0.012)	0.457 (0.007)
CONSTR	**0.639** (0.162)	**0.371** (0.013)	**0.443** (0.003)	**0.535** (0.005)	**0.347** (0.007)	**0.467** (0.007)
ARTIFICIAL	0.596 (0.051)	0.324 (0.015)	0.393 (0.028)	0.451 (0.077)	0.303 (0.030)	0.413 (0.038)
PLAIN+*Emb*	0.614 (0.019)	0.343 (0.014)	0.425 (0.012)	0.531 (0.007)	0.345 (0.013)	0.452 (0.006)
CONSTR+*Emb*	**0.638** (0.007)	**0.347** (0.003)	**0.437** (0.005)	**0.538** (0.012)	**0.356** (0.009)	**0.463** (0.003)
ARTIF.+*Emb*	0.608 (0.031)	0.323 (0.006)	0.375 (0.031)	0.446 (0.070)	0.311 (0.002)	0.412 (0.030)

Table 4.3: F1 scores on emotion classification (ISEAR). Bold: cases in which constraints introduce improvements.

	Anger	*Disgust*	*Fear*	*Happiness*	*Sadness*	Macro Avg
PLAIN	0.313	0.009	0.170	0.452	0.420	0.227
CONSTR	0.200	**0.185**	**0.272**	0.395	0.419	**0.245**
ARTIFICIAL	0.186	0.025	0.039	0.126	0.246	0.104
PLAIN+*Emb*	0.366	0.149	0.383	0.522	0.466	0.314
CONSTR+*Emb*	**0.383**	0.146	0.381	**0.551**	**0.486**	**0.324**
ARTIF.+*Emb*	0.160	0.002	0.039	0.128	0.262	0.098

not hold when using pre-trained embeddings, where the performances of the not-constrained model are already close to the constrained one. In this case, all the other classes are improved instead. Finally, artificial labels do not seem a promising solution.

The results on the Fairy Tales test data are shown in Table 4.4, still confirming the improvements introduced by constraints in the average case. Since "surprise" is poorly represented in the labeled portion of the training set (being it not included in ISEAR data), results in this class are pretty low. While artificial labels help in "surprise", they sometimes lead to very bad results. This is even more evident when using pre-trained embeddings, where the system constantly overfits the training data. Notice that the F1 scores on the validation splits were very promising when using such embeddings, but, as we mentioned when describing the experimental setting, the system badly generalizes to out-of-sample data that is related-but-not-fully-coherent with the training (validation) sets.

In the case of Affect Text test data (Table 4.5) constraints still increase the macro F1, but not when using pre-trained embeddings. We observe a less coherent behaviour with respect to the previous test sets, and this is due to the fact that Affective Text is composed of sentences that are significantly shorter than the ones of the other datasets, and they are evocative of multiple emotions in which it is harder to distinguish the most-dominant one.

Table 4.4: F1 scores on emotion classification (Fairy Tales). Bold: cases in which constraints introduce improvements.

	Anger	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*	Macro Avg
PLAIN	0.238	0.151	0.397	0.533	0.410	0.018	0.291
CONSTR	**0.288**	**0.184**	0.362	0.533	0.400	0.038	**0.301**
ARTIFICIAL	0.261	0.029	0.079	0.598	0.471	0.101	0.256
PLAIN+*Emb*	0.365	0.137	0.451	0.546	0.365	0.037	0.317
CONSTR+*Emb*	**0.367**	0.127	0.424	0.521	**0.476**	**0.068**	**0.331**
ARTIF.+*Emb*	0.156	0.035	0.078	0.064	0.109	0.009	0.075

Table 4.5: F1 scores on emotion classification (Affective Text). Bold: cases in which constraints introduce improvements.

	Anger	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*	Macro Avg
PLAIN	0.162	0.100	0.282	0.514	0.289	0	0.224
CONSTR	**0.187**	**0.111**	**0.282**	0.493	0.295	0	**0.228**
ARTIFICIAL	0.182	0	0.010	0.586	0.383	0.198	0.227
PLAIN+*Emb*	0.153	0.113	0.369	0.571	0.396	0.054	0.276
CONSTR+*Emb*	0.022	**0.117**	0.324	**0.577**	**0.447**	0	0.248
ARTIF.+*Emb*	0.126	0.047	0	0.059	0.093	0.045	0.062

In Fig. 4.4 we report precision and recall (averaged on the test splits, when needed) associated to the results of Table 4.2, 4.3, 4.4, 4.5. When predicting reactions and using constraints, we observe improvements in *both* precision and recall in the case of 3 out of 5 classes. When predicting emotions, improvements are usually either in terms of precisions *or* in terms of recall (we count a similar number of cases in which precision is improved and cases in which recall is improved).

Comparing our experimental analysis with existing literature that is about emotion detection is not straightforward. Existing approaches make use of lexical resources or focus on settings that are pretty different from the one we selected (they test on splits that are taken from the same emotional dataset, thus providing better results [94, 143]). However, we found that, in some cases, our model is competitive with popular algorithms. Table 4.6 reports the F1 scores of existing models, emphasizing the cases in which our CONSTR+*Emb* outperforms them. In Affective Text, we compared with the WN-AFFECT system (based on WordNet Affect), and a model based on LSA to compute representations of emotion words [129] (even if they considered a multi-label learning problem). On the same data, as well as in ISEAR, we also considered the CNMF model from [66], based on non-negative matrix factorization, that was evaluated on a subset of the emotions we considered in this work. Finally, we compared with (what we refer to as) the Wikipedia model from [1], that was trained on texts taken from Wikipedia and tested on the ISEAR data (and other

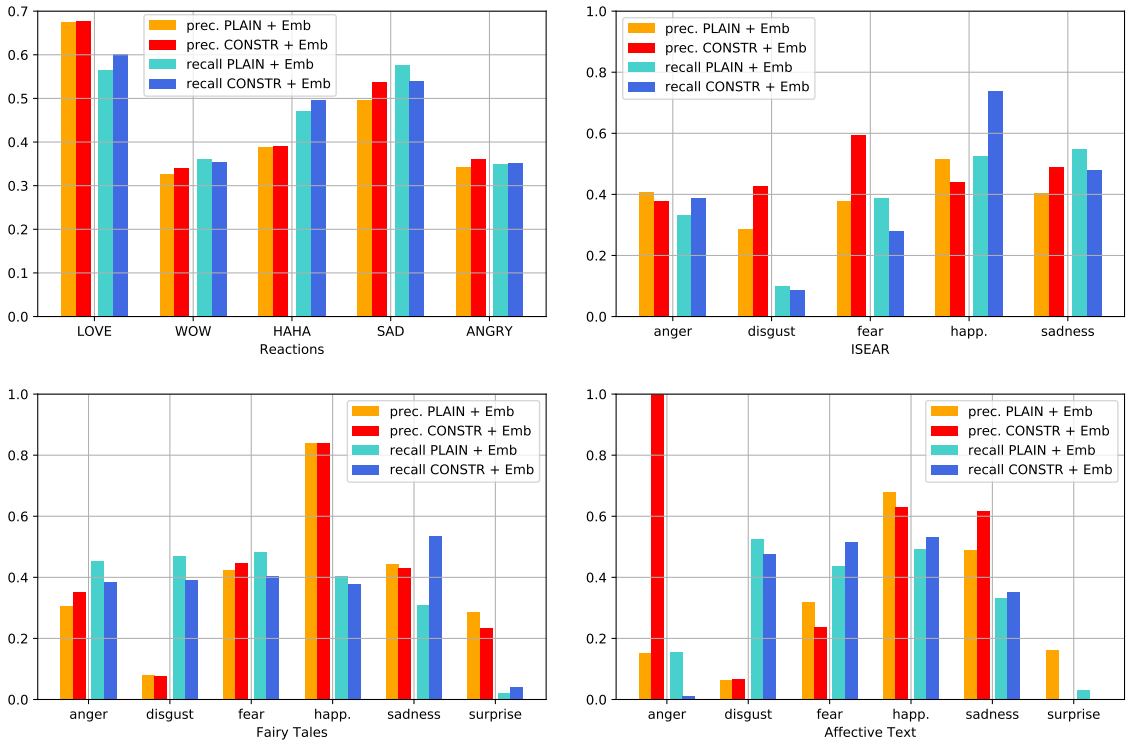


Figure 4.4: Precision and recall associated to the results of Table 4.2, 4.3, 4.4, 4.5 (left-to-right, top-to-bottom), comparing PLAIN+*Emb* with CONSTR+*Emb*.

Table 4.6: Results of existing approaches. We indicate with $*$ those cases in which our model (CONSTR+*Emb*) outperforms the result reported in this table.

	Anger	*Disgust*	*Fear*	*Happiness*	*Sadness*	*Surprise*	Macro Avg
	ISEAR						
CNMF [66]	0.579	-	0.056*	0.010*	0.017*	-	-
WIKIPEDIA [1]	0.413	0.430	0.517	0.514*	0.396*	-	0.454
	Affective Text						
WN-AFFECT [129]	0.061	-	0.033*	0.011*	0.066*	0.069	0.040*
LSA [129]	0.112	0.039*	0.219*	0.308*	0.206*	0.141	0.176*
CNMF [66]	0.278	-	0.618	0.648	0.475	-	-

datasets).[5]

4.6 Discussion

Recognizing emotions from text is a challenging task, due to the ambiguity and the complexity of the language. Humans sometimes use sarcasm and irony to express negative sentiments with positive words, so a machine should be able to under-

[5]We did not consider Fairy Tales since existing approaches usually merge "anger" and "disgust", and also because the sentence truncation strongly affected this dataset.

stand the context to detect the proper emotion. Moreover few and small datasets containing texts labeled with emotions are available. To overcome this problem, we proposed to jointly learn the tasks of emotion detection and Facebook reaction prediction, when processing raw text. While such tasks share several analogies, mapping emotion classes to Facebook reactions (and vice-versa) can easily become ambiguous. Our system exploits First Order Logic formulas to model the task relationships, and it learns from such formulas, also exploiting large collections of unlabeled training data. The logic rules are converted with the product t-norm into real-valued functions, in order to be integrated into the learning problem.

The provided experimental analysis has shown that bridging these two tasks, by means of FOL-based constraints, leads to improvements in the prediction quality that clearly goes beyond more naive approaches in which artificial labels are generated in the data preprocessing stage. The results are not yet straightforward, because we used test data that are far different from training data, and however this is a difficult task in itself. In the future, the model could be improved introducing lexical resources or using Transformers [140] instead of RNNs, even if with transformers, computational costs, in terms of memory and time complexity, would increase significantly.

Chapter 5

Facial Expression Generation

In this chapter we address the task of facial expression generation, that in our case, is handled in a more articulate way, presenting an application that generates a facial expression in relation to the emotion detected from a text [51]. We developed a system that answers the question *how will a certain person react when reading a given post?* For example, we might want to see what would be the face of a famous Hollywood star or of a popular politician when reading a newspaper title or a Facebook post that is about something that he/she cares of. Not all the people would react at the same way reading the same content. For example, given a newspaper title that is about politics, a person who dislikes such topic will react differently compared to a person who likes it, while, on average, the female audience is less interested into posts that are about motors compared to males. We do clearly underline that in this work we report only general preferences, we do not want to create stereotypes (I am, myself, the exception that proves the rule: I am female, I play football and I like Formula 1).

Differently from most existing generation-based applications, the way the system alters the input face is the outcome of a decision process that is not user-selected and that involves the information extracted from the provided post (newspaper title, short-text, etc.), from the input face itself, or from other sources of knowledge. In other words, we bridge two processes, namely (*i.*) *information extraction* and (*ii.*) *image generation*, with the purpose of creating a system that generates new data in function of the information that is either given or extracted from the input signals.

(*i.*) Our system automatically extracts information about the sex and the expected age of the input person using CNNs, while those details that cannot be grasped from the face picture are explicitly provided to the system. In particular, we consider the case of the topics of interest (or not-interest) of the considered person, that can be compared with the topics of the input post, automatically extracted by means of a RNN-based pipeline. As matter of fact, on the input text we perform the tasks of *topic prediction* and *emotion detection*. The most

dominant emotion is detected using the model described in Chapter 4, that employs constraints based on *First Order Logic* (*FOL*) formulas.

(*ii.*) In order to generate the final expression we exploited the widely used *Generative Adversarial Networks* (*GANs*) (Section 2.1), following the approach in [80] and its FOL-based instance [87]. In particular, our model includes a generator and a discriminator for each class of emotions, and it learns to translate pictures of "neutral" faces into pictures that are associated with target emotional states. Cyclic and identity constraints are described as logic rules and are inserted in the learning problem through t-norms. Also the conditions on the generator and the discriminator (see Eq. 2.1) are represented as logic formulas. In this way the generation problem is translated into a constrained satisfaction problem. This approach presents a learning scheme easier to understand and is flexible to develop new generation tasks by simple logic descriptions.

FOL is also used to mix the information extracted from the inputs, where a *t-norm* based implementation allows the system to handle the distribution of probabilities yielded by the four networks and the topics of interest, and to decide which emotion to generate. In this specific case the logic rules are not inserted into the learning stage.

This chapter is organized as follows. Section 5.1 presents some works of image-to-image translation. Section 5.2 reports the proposed system introducing all its sub-models, in particular Section 5.2.4 describes the constrained generative model. Section 5.3 focuses on describing the decision process based on FOL rules. Section 5.4 provides numerical and qualitative results and shows the system interface.

5.1 Related works

The huge popularity of GANs [47] has led to the development of a large number of approaches that are aimed at extending and refining the generative process, showing impressive results in the context of image-to-image translation [22, 61, 80, 153]. In [153] the so-called cycleGANs are presented, which enable to translate an image from a domain X to a domain Y in absence of paired examples. Given a function $G : X \rightarrow Y$ and an inverse mapping $F : Y \rightarrow X$, it is required that $F(G(x)) = x$ and $G(F(y)) = y$. Another generative model that uses unpaired data is the UNsupervised Image-to-image Translation (UNIT) approach [80], of which we exploit a variant in Section 5.2.4. UNIT combines Variational AutoEncoders (VAEs) (see Section 2.1) with Coupled GANs [81], which are GANs where two generators share weights to learn the joint distribution of images in multiple domains.

Real-world applications based on generative models have spread out, especially in the case of smartphone applications, since image-to-image translation becomes

strongly appealing when it is about human faces. Once we are given a input face, several approaches show how to change hair color [22], add glasses [71, 87], change sex [22, 71, 87], get older or younger [22, 71].

Our generative case, namely translation from neutral face into a specific expression, is proposed in [22], where a generative adversarial network learns the mappings among multiple domains using a single generator and discriminator. The approach in [105] generates videos of the six basic facial expressions given a neutral face image. It processes separately facial expression dynamics and face appearance using two different GAN architectures. The first is a conditional version of the Wasserstein GAN [7] that is used to generate new facial expression motions, and then the second conditional GAN transforms the generated facial landmark sequences into video frames by adding the texture information.

5.2 Generating Facial Expressions Associated with Text

Our model processes an input image of a person and it focusses on his/her face. Additional knowledge about the person depicted in the image is provided, together with a short text (post, newspaper title, etc.). The system generates an artificial image with the facial expression that the input subject is expected to have after having read the short text. The entire model is composed of several modules, compactly shown in Fig. 5.1. In the rest of this section we describe the details of each module that is involved in the information extraction/organization process and of the final image generation procedure.

5.2.1 Gathering Information from the Input Image

The system processes a visual representation of the person that we want to consider, in a neutral expression, frontal view. Such representation might show the target person in different poses, wearing different clothes, with different backgrounds, etc., and the system is able to exploit and alter only the area around the face, keeping intact all the rest of the image. Several types of information could be extracted from the original visual input that, due to the variability of such input, might only be available in some input data and not visible in others (types of background, types of clothes, objects that the person is holding, etc.). In the context of this work we focus on two features that are common to all the expected inputs since they are estimated by only looking at the face region, and that we can extract in a pretty accurate way: gender and age.

In detail, the system processes an input image I that depicts the face of the person that we want to alter. The face is first localized and the face region is cropped and

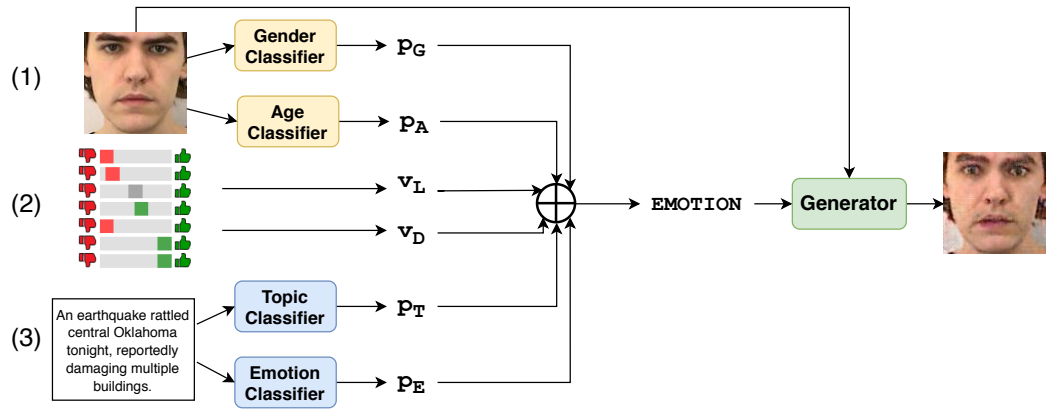


Figure 5.1: The main computational blocks of the proposed system. Three different inputs are provided to the system: (1) a picture of person, from which the face is localized and extracted; (2) information about the topics that the person likes or dislikes, paired with a degree that indicates how strongly each topic is either liked or not (symbolically represented by the 6 sliders); (3) a short-text. The input data undergo an information extraction stage, whose outcome is a set of distribution of probabilities or set of scores over different attributes (p_G, p_A, v_L, v_D, p_E, p_T). A logic-based decisional process (represented by the circled cross) yields the target emotion that the input person is expected to have when reading the short text. Finally, a Neural Network-based generator automatically generates the target facial expression.

rescaled to 100×100 pixels (RGB). Then, it is fed to two CNNs that act as classifiers of the *gender* and the *age* of the face in I. Both the classifiers yield precisions over two classes, distinguishing either between the *male* and *female* classes and between the *young* and *old* classes. The last layer of each network is based on the softmax activation function, so the output of each network is composed of the probabilities of the target classes, that are $p_G \in [0,1]^2$ in the case of the gender classifier and $p_A \in [0,1]^2$ in the case of the age classifier.

5.2.2 Gathering Information from the Input Text

The system also processes a short text typically composed by a single sentence or just a few sentences, that might be, for example, a newspaper title or a post in a social network. We focus on the emotion-related information that is carried by such textual data and on its topic. From a very abstract perspective, the former is information that will have an important role in determining *how* the input subject is expected to react when reading the input text, while the latter is what allows the system to decide *if* the input subject is somewhat interested in the provided text. Of course, one might think of more fine grained or more structured information that can be extracted from text. However, we followed the same simplicity principle that was also followed when deciding which information to extract from the input image.

As made in Section 4.3, an input text t is tokenized into words $t_0, \ldots, t_n$ that are mapped into the symbols of a fixed-size vocabulary V. Then, each word is embedded into a learnable latent dense representation, also known as "word embedding", and an LSTM processes the sequence of word embeddings in both directions. The forward and backward states of the LSTM are then concatenated, producing an embedded latent representation of the whole text t. In particular, our system computes two independent latent representations of t, namely $\hat{t}_T \in \mathbb{R}^{d_T}$ and $\hat{t}_E \in \mathbb{R}^{d_E}$, that are provided as input to two feed-forward networks with softmax activation function in the output layer, where d_T and d_E are the sizes of the embeddings. The first network is about a *topic classifier*, that yields the probability distribution $p_T \in [0,1]^7$ over 7 topics, that are *fashion, motors, politics, religion, science, sport, technology*. The second network is about an *emotion classifier*, that outputs the probability distribution $p_E \in [0,1]^6$ over the six universal emotions. This is the model presented in Section 4.3, discarding the reaction predictor. However the emotion classifier has been enforced during the training by the reaction labels through the logic constraints 4.1-4.11.

5.2.3 Organizing the Input Knowledge

Further knowledge about the input subject is provided to the system, in order to allow it to better characterize the details of the considered person and to estimate a more appropriate facial expression with respect to the considered input text. Our system can be fed with information related to how strongly the input subject likes or dislikes one of the aforementioned topics (*fashion, motors, politics, religion, science, sport, technology*), modeling the strength of each liking/disliking attribute with a real value in $[0,1]$. In particular, 1 indicates a strong liking/disliking of a topic and 0 means that the person does not care at all. We end up with two vectors of scores, each of them in $[0,1]^7$ (since we have 7 topics). The first vector, v_L, collects the scores about how strongly the input subject likes the 7 topics, while the second vector v_D, collects the scores about how strongly he/she dislikes the 7 topics. Of course, we assume that if $v_L(k) > 0$ then $v_D(k) = 0$, and vice-versa, being $v_L(k)$ ($v_D(k)$) the k-th component of v_L (v_D). In other words, if the target person likes a topic, the disliking score is zero. In our GUI, this information is inserted by means of 7 sliders. For each topic k the user can define a value $v(k)$ between -5 and 5. If such value is positive, then $v_L(k)$ is set to $\frac{v(k)}{5}$ and $v_D(k) = 0$. Otherwise, $v_D(k)$ is set to $\frac{|v(k)|}{5}$ and $v_L(k) = 0$. Notice that if no knowledge is provided, the two vectors are set to zeros (i.e., do not care).

5.2.4 Generating Facial Expressions

Once the input information has been processed to formulate a decision on the target emotion to generate (whose details are postponed to Section 5.3), a Neural Network-based generative model handles the face region from the input image I, and it generates an altered instance of the input face with an expression that communicates the target emotion. Our model generates RGB images at the resolution of 100×100 pixels that is rescaled and superimposed in the face area of I.[1]

We notice that we do not have the use of training data in which each face in a neutral expression is paired with the same face in a non-neutral expression. For this reason, we exploit a variant of the UNIT approach [80], introduced in Section 5.1. In particular, we consider a FOL-based implementation of UNIT [87], that follows the same t-norm-based implementation that we used in the emotion classifier.

We have seven domains, one for each emotion class plus the neutral class, and the notation x_i indicates an image associated with the i-th universal emotion ($i = 0, \ldots, 5$) or a neutral face ($i = N$). During the training stage, to the model is given a tuple of 7 training images, one for each class (not necessarily belonging to the same subject). If $\bar{x}_{i \to j}$ is the fake image with emotion j generated from the input x_i, then the output data of the model consist of 19 generated pictures, that are $\bar{x}_{h \to h}$, $\bar{x}_{h \to N}$, for $h = 0, \ldots, 5$, and $\bar{x}_{N \to i}$, for $i = 0, \ldots, 5, N$. The reason why we need to generate 19 pictures is due to the constraints that we have to impose while training the model, and it will become clear shortly. Internally, the model is composed by 7 encoders and 7 decoders, as shown in Figure 5.2. The encoders $e_i(x_i)$, $i = 0, \ldots, 5, N$ project their inputs onto a shared latent space Z. Notice that each encoder e_i only processes inputs of class i, i.e., x_i. Then, decoders $g_i(z)$, $i = 0, \ldots, 5, N$ are able to decode data from the shared space Z, generating fake pictures. The encoder e_i and decoder g_i are feed-forward Neural Networks, and they act as VAE for the domain of class i. Following [80], the weights of the last few layers of the encoders (that extract high-level representation of the input images) are shared, and the same sharing principle is also applied to the weights of the first few layers of the decoders (that decode high-level representations for reconstructing the input images). In particular, each encoder is composed by 4 convolutional layers (followed by Leaky ReLU) and 3 residual blocks. A residual block is shared with the 7 encoders and another residual block is shared with the 7 decoders. Each decoder consists of 3 residual blocks, 3 deconvolutional layers (2 with Leaky ReLU and the last with tanh).

In order to learn the weights of the networks, we have 7 discriminators d_i, $i =$

[1] In order to avoid discontinuities in those areas that are close to the borders of the generated picture, the system computes a weighted average of such generated image and the original face. In particular, we smoothly weigh in a decreasing way the contributes of the generated pixels while moving closer to the borders. The oval area at the middle of the generated image is left untouched (it is not averaged).

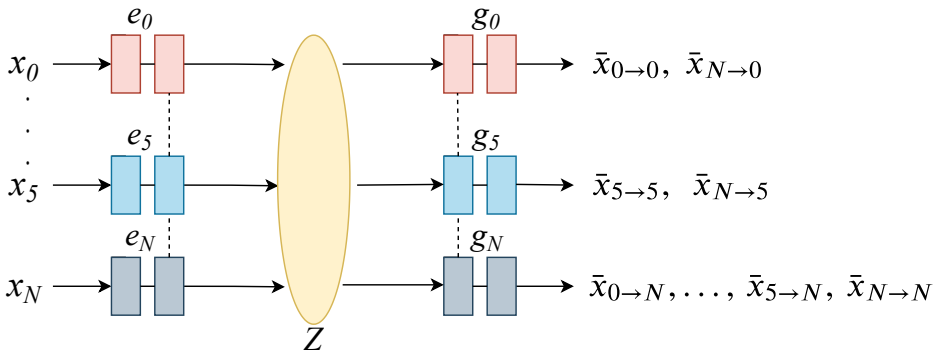


Figure 5.2: An example of the structure of our generative model during the training stage, showing the input/output of the model. The input data consists of a tuple of 7 training images, one for each emotion class plus neutral (N). The model is then composed of 7 encoders e_i, $i = 0, \ldots, 5, N$ that project their inputs onto a shared latent space Z. Each small vertical rectangle indicates some layers of neurons. Then we have 7 decoders that decode data from the shared space Z. The connection weights of the last layers of the encoders are tied (dashed lines), as well as the connection weights of the first layers of the decoders. Each decoder g_h, $h = 0, \ldots, 5$ outputs a pair of images, $\bar{x}_{j\to h}$ with $j \in \{h, N\}$, that are the re-generated image x_h and the image with emotion h generated from input neutral face x_N, respectively. Decoder g_N outputs 7 images, that are the images with neutral expressions generated from the input images with the 6 universal emotions and from the neutral one.

$0, \ldots, 5, N$ that must distinguish the real images of class i from the ones that are artificially generated by the model. In particular, each discriminator is composed by 5 convolutional layers (4 with Leaky ReLU and the last with sigmoid). The learning criterion consists in trying to fool the discriminators, generating images that are not easily distinguishable from the fake ones, while improving the discriminators themselves, as in classic GANs. However, the model is also subject to a set of FOL-based constraints that ensure cycle-consistency. In this work, we adapted the FOL constraints to the case of emotion generation, taking care of handling the neutral class in a special way, since our final goal is to learn how to generate an emotion-related facial expression from a neutral face, and not to learn to convert each emotion into any other emotion. First of all we constrain the system to be able to re-generate the input, so that for each class i we have

$$\forall x_i \quad g_i(e_i(x_i)) = x_i, \tag{5.1}$$

where the equality operator expresses a pixel by pixel similarity between the images. We impose cycle consistency to constraint the generation procedure to translate an image x_i from domain i to domain j and to translate it back to domain i getting the exact same image x_i (and not another image of class i). This constraint is needed to enforce the system to generate output images that are alterations of the input one, and not new faces that have nothing to do with the input. In detail, for each h, the

system gets an image with neutral expression x_N and it generates a new image with emotion h, that is $\bar{x}_{N\to h}$. Then, the system must be able to reconstruct the initial x_N when $\bar{x}_{N\to h}$ is provided as input, i.e.,

$$\forall x_N \quad g_N(e_h(\bar{x}_{N\to h})) = x_N. \tag{5.2}$$

The same cycle consistency is enforced in the case of an input image x_h that is translated into a neutral one $\bar{x}_{h\to N}$, and then translated back into x_h, i.e.

$$\forall x_h \quad g_h(e_N(\bar{x}_{h\to N})) = x_h. \tag{5.3}$$

The generated images must fool the discriminators, i.e., they have to be detected as real ones:

$$\forall x_N \quad d_h(g_h(\bar{x}_{N\to h})) \tag{5.4}$$

$$\forall x_h \quad d_N(g_N(\bar{x}_{h\to N})). \tag{5.5}$$

At the same time the discriminators must keep their capability of recognizing the real images from the generated ones. If $d_i(x)$ is a function in $[0,1]$ that is 1 when x is real and 0 when it is fake, for each $h = 0,\ldots,5$ we have

$$\forall x_h \quad d_h(x_h) \wedge \neg d_N(\bar{x}_{h\to N}), \tag{5.6}$$

$$\forall x_N \quad d_N(x_N) \wedge \neg d_h(\bar{x}_{N\to h}). \tag{5.7}$$

The product t-norm (see Section 2.3) was selected to convert the aforementioned logic formulas into real valued functions. The encoders e_i and the decoders g_i ($i = 0,\ldots,5,N$) are trained to the satisfaction of the constraints (5.1)-(5.5), while the discriminators d_i are trained to satisfy the equations (5.6) and (5.7). Once training has ended, we only need to keep e_N and g_h, $h = 0,\ldots,5$, that is what the system needs to generate a facial expression from a neutral input.

5.3 Fuzzy Logic-based Decision Process

Once the system is given an input image I, a short text t and target person-related knowledge k, the information that is either automatically extracted using neural models or simply reorganized from k, is encoded into the vectors p_G, p_A, p_T, p_E, v_L and v_D, as described in Section 5.2. Such vectors are probability distributions over certain classes (p_G, p_A, p_T, p_E) or vector of independent probabilities (v_L, v_D), in both the case each element is in $[0,1]$. The system needs to take a decision on which emotion to generate, and we propose to integrate the outcome of the information extraction stage using FOL formulas and cast them into a fuzzy logic setting

by converting them using t-norms. This choice allows us to exploit the uncertainty in the output of the neural networks or in the provided knowledge and to naturally embed such uncertainty in the decision process. The reason why we rely on these formulas is that we have no data from which we could learn how to mix the probabilities. However, the proposed rules are differentiable, so they could allow us to integrate learning also at this stage, if supervised data will become available for this task.

We create FOL predicates with the same names of the classes of information that we extracted/organized from the input image and text, temporarily discarding emotion-related predictions. In detail, we get the predicates $\texttt{female}(I)$, $\texttt{male}(I)$, $\texttt{old}(I)$, $\texttt{young}(I)$, $\texttt{fashion}(t)$, $\texttt{motors}(t)$, $\texttt{politics}(t)$, $\texttt{religion}(t)$, $\texttt{science}(t)$, $\texttt{sport}(t)$, $\texttt{tech}(t)$. Each element of the aforementioned vectors p_G, p_A, p_T is the truth degree of one of such predicates. For example, the first element of p_G is the truth degree of predicate $\texttt{female}(I)$, while its second element is the truth degree of $\texttt{male}(I)$, and so on. In the case of the target person-related knowledge, we use the following predicate names associated with the truth degrees in v_L: $\texttt{likes\_fashion}(k)$, $\texttt{likes\_motors}(k)$, $\texttt{likes\_politics}(k)$, $\texttt{likes\_religion}(k)$, $\texttt{likes\_science}(k)$, $\texttt{likes\_sport}(k)$, $\texttt{likes\_tech}(k)$. Similarly, the predicates $\texttt{dislikes\_fashion}(k)$, $\texttt{dislikes\_motors}(k)$, $\texttt{dislikes\_politics}(k)$, $\texttt{dislikes\_religion}(k)$, $\texttt{dislikes\_ science}(k)$, $\texttt{dislikes\_sport}(k)$, $\texttt{dislikes\_tech}(k)$, are associated with the truth degrees in v_D.

We also introduce three special predicates that are related to the generation of a neutral face (i.e., the system does not alter the input face), $\texttt{neutral}(I)$, to the generation of the most-dominant emotion in the input text, $\texttt{gen\_emo}(I)$ (that can be retrieved by $\arg\max p_E$), and the generation of a disgusted expression, $\texttt{gen\_disgust}(I)$. These predicates are not associated to pre-computed truth scores, and we devised a list of FOL implications that we can use to determine if these predicates are true (i.e., 1) or false (i.e., 0), so that the face expression generator will react accordingly. We considered two categories of rules. The first category is inspired by the principle that if a person is interested in the topic of the input text, then it should react by showing the facial expression associated to the emotion that is detected in such text. If there is not a specific interest on such topic, then we can follow some commonly assumed statistical relations between being male/female/young/old and certain topics (for example, it is pretty frequent – not always the case – for male audience to have interest in motors, sports and not being interested in fashion; similarly, younger people might be less interested into politics than adults). Please consider that these rules are just simple examples, with no attempts to model realistic stereotypes, and our system is general with respect to them. The first category of rules includes (all the rules in this section are intended to hold for $\forall I, \forall t, \forall k$),

1) $\texttt{fashion}(t) \wedge (\texttt{female}(I) \vee \texttt{like\_fashion}(k)) \Rightarrow \texttt{gen\_emo}(I)$

2) $\texttt{fashion}(t) \wedge \texttt{male}(I) \wedge \neg\texttt{like\_fashion}(k) \Rightarrow \texttt{neutral}(I)$

3) $\texttt{motors}(t) \wedge (\texttt{male}(I) \vee \texttt{like\_motors}(k)) \Rightarrow \texttt{gen\_emo}(I)$

4) $\texttt{motors}(t) \wedge \texttt{female}(I) \wedge \neg\texttt{like\_motor}(k) \Rightarrow \texttt{neutral}(I)$

5) $\texttt{politics}(t) \wedge [(\texttt{male}(I) \wedge \texttt{old}(I)) \vee \texttt{like\_politics}(k)] \Rightarrow \texttt{gen\_emo}(I)$

6) $\texttt{politics}(t) \wedge [(\texttt{female}(I) \vee \texttt{young}(I)) \wedge \neg\texttt{like\_politic}(k)] \Rightarrow \texttt{neutral}(I)$

7) $\texttt{religion}(t) \wedge [(\texttt{female}(I) \wedge \texttt{old}(I)) \vee \texttt{like\_religion}(k)] \Rightarrow \texttt{gen\_emo}(I)$

8) $\texttt{religion}(t) \wedge [(\texttt{male}(I) \vee \texttt{young}(I)) \wedge \neg\texttt{like\_religion}(k)] \Rightarrow \texttt{neutral}(I)$

9) $\texttt{science}(t) \wedge \texttt{like\_science}(k) \Rightarrow \texttt{gen\_emo}(I)$

10) $\texttt{science}(t) \wedge \neg\texttt{like\_science}(k) \Rightarrow \texttt{neutral}(I)$

11) $\texttt{sport}(t) \wedge [(\texttt{male}(I) \wedge \texttt{young}(I)) \vee \texttt{like\_sport}(k)] \Rightarrow \texttt{gen\_emo}(I)$

12) $\texttt{sport}(t) \wedge [(\texttt{female}(I) \vee \texttt{old}(I)) \wedge \neg\texttt{like\_sport}(k)] \Rightarrow \texttt{neutral}(I)$

13) $\texttt{tech}(t) \wedge (\texttt{young}(I) \vee \texttt{like\_tech}(k)) \Rightarrow \texttt{gen\_emo}(I)$

14) $\texttt{tech}(t) \wedge \texttt{old}(I) \wedge \neg\texttt{like\_tech}(k) \Rightarrow \texttt{neutral}(I)$.

Notice that $\neg\texttt{like\_*}$ has nothing to do with disliking something, and it only means do-not-care about such topic. The second category of rules deals with cases in which a person does not like the topic of the input text. In this case, the system will generate a disgusted facial expression. Formally, we have,

15) $\texttt{fashion}(t) \wedge \texttt{dislike\_fashion}(k) \Rightarrow \texttt{gen\_disgust}(I)$

...

21) $\texttt{tech}(t) \wedge \texttt{dislike\_tech}(k) \Rightarrow \texttt{gen\_disgust}(I)$.

The premise of each rule is transformed into a polynomial using the product t-norm (see Section 2.3). The system evaluates all the premises, finding the one that holds with a larger truth degree. The associated conclusion is set to true (i.e., 1) and the corresponding action is taken on the facial expression generation. Notice that if there is a strong uncertainty in the topic of the input text (max p_T is small, < 0.55 in our experiments) then the rules are discarded and $\texttt{gen\_emo}(I)$ is directly set to true.

Table 5.1: Accuracies for topic classifier (top) and emotion classifier (bottom).

fashion	*motors*	*politics*	*religion*	*science*	*sport*	*tech*	micro acc.	macro acc.
0.782	0.684	0.741	0.645	0.775	0.879	0.704	0.779	0.774
anger	*disgust*	*fear*	*happiness*	*sadness*	*surprise*		micro acc.	macro acc.
0.394	0.608	0.662	0.732	0.607	0.238		0.598	0.540

5.4 Experimental Results

We experimentally evaluated the quality of each of the modules that compose our system. In order to train each of them we collected different datasets, either popular datasets or data that we crawled from the web and semi-automatically labeled (data from Wikipedia and images from the web). Then we will describe our implementation of the whole system showing qualitative results (Section 5.4.1).

CelebA [82] is a face attributes dataset with several celebrity images. This collection was used to train the gender and age classifiers and it was also used in the generation module. In the former case, we considered the images annotated with both gender and age information, that were preprocessed by localizing the face areas, cropping them and resizing such area to 100×100. We obtained almost 100,000 images of females, almost 70,000 of males, 40,000 of adults and 130,000 images of young people. In each task we divided the data into training, validation and test sets (70%,15%,15% of the data, respectively – this is what we did in all the experiments), and selected the optimal architecture by focussing on two-layer CNNs followed by max pooling layers and evaluating different filter sizes, number of filters, number of fully connected layers. The final models were composed by 32 and 64 filters of size 3×3, two fully connected layers with 516 hidden and 2 output neurons (number of classes). We obtained satisfying (micro) accuracies on the test splits, i.e., 97.3% for gender classification and 87.5% for age classification.

The topic classifier was trained in supervised manner on texts collected from Wikipedia. We automatically crawled Wikipedia pages from categories and subcategories that matched (or were similar) to the selected 7 topics. We collected $\approx$ 200,000 short texts (each of them composed of a sentence or up to two shorter sentences). We evaluated architectures with differently sized word embeddings and states of the BRNN, multiple fully connected layers and number of hidden units. After the validation stage, we selected word embeddings of size 200, BRNNs with a hidden state composed of 200 units and a single fully connected layer.

In the Table 5.1 (top) we report the model accuracies, that are slightly below 80%, both in the micro and macro cases. The error analysis suggested that is mostly due to the fact that there is clearly some ambiguity in determining the topic of those texts that are about, for example, biographical information, that are indeed present in Wikipedia pages.

Figure 5.3: Automatically generated faces (test data). On the first column the input neutral face is shown. The following columns are outputs of our model.

The model we used for emotion classification from text shares the same overall structure of the just described topic classifier and, as already mentioned, is the model presented in Chapter 4. From the entire structure, we took only the emotion predictor, that however has been enforced during the training with a large collection of unsupervised text and with Facebook posts that include reactions, exploiting logic constraints. In this case we used pre-trained word embeddings of size 300 (from the popular Google word2vec model), and 2 fully connected layers with 25 and 6 neurons respectively (we have 6 classes). In Table 5.1 (bottom) we report the results we obtained.

The face expression generator was trained using several images of facial expressions, subdivided into the six universal emotions and the neutral class. Even if we used several public datasets (ADFES [139], CK+ [85] (only the colour ones), WSEFEP [104], MUG [2]), we also downloaded images from the web and collected faces from non-emotion-related datasets (CelebA). Then, we classified these images using the emotion classifier we presented in Chapter 3 (we took only the full face predictor) and we manually filtered out the cases in which the classifier was not confident and the most evident cases in which the emotion class was inappropriate. All the images were cut around the face area and resized to 100×100 pixels (RGB). In this case, we do not have a clear measure of performance, so we visualized the images generated from a held-out set after 6 consecutive epochs, and we stopped the training procedure when results were satisfying. In Fig. 5.3 we show some samples generated by the final model.

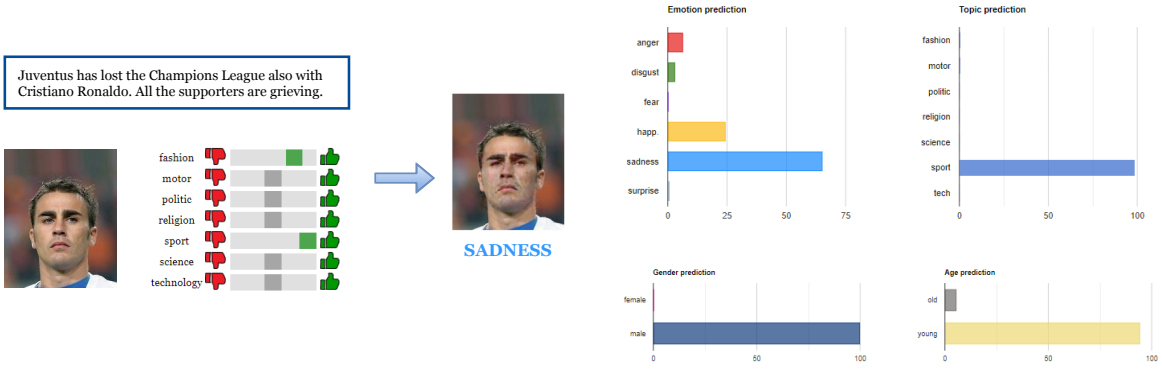


Figure 5.4: Example showing some portions of our web interface. Left: input data, i.e., face image, text and additional knowledge. Middle: the generated image (*sadness*). Right: the histograms of the information extraction stage. The target person is a football player, he likes sport and somewhat fashion (see the sliders). The system is predicting that the topic of the post is sport, and the dominant emotion is sadness. For this reason, the final facial expression is sadness.

5.4.1 System Interface and Generations

We implemented a web interface (HTML5, Javascript) from which the user can upload a picture or select preexisting templates. [2] The user can use 7 sliders to indicate how strongly the target person is interested/not-interested in the 7 topics. Finally, the post from which we want to estimate a facial expression is provided in a text area. The backend of the system consists of a TensorFlow implementation of the previously described neural models, with a Python server that is queried by the web interface.

In Figure 5.4 we report an example of our system, showing portions of the web interface. The input text is about sport, the subject is both young and male and he also likes sport, so the rule with highest premise value is the number 11 of Section 5.3. The emotion predictor classifies the text with *sadness* ($\arg\max p_E$), so a sad expression is generated. In Figure 5.5 we show another example with the same person of Figure 5.4, but with a different input text. The topic predictor classifies the text as fashion and the emotion predictor classifies the text as *disgust*. Since the subject is also a bit interested in fashion (fuzzy score, smaller than 1), the rule with high premise is the number 1 of Section 5.3, so a disgusted expression is generated.

Fig. 5.6 shows an example in which different people react in different ways to the same post, talking about technology. The first subject likes technology and the rule activated is rule number 13, which indicates to generate the emotion coming from the text, that is *fear*. The second person dislikes technology and the automatically selected rule is number 21 (that indicates to generate a disgusted expression). The third subject does not care about this topic and it is an old woman, so rule 14

[2] `https://sailab.diism.unisi.it/emoreact`

Figure 5.5: Another example of our system with the same subject of Fig. 5.4 (we do not report the histograms of gender and age and the sliders, since they are the same as in Fig. 5.4). The subject likes fashion a bit, the text is classified as *disgust*, and that is the generated facial expression.

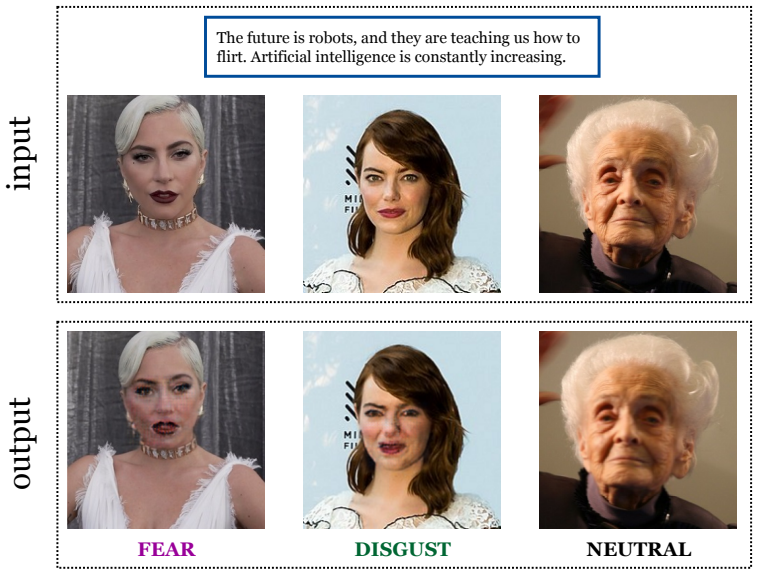


Figure 5.6: An example of different reactions to the same post. On top the original faces (input), on bottom the corresponding generated expressions (output). (*Image copyrights belong to their respective owners.*)

indicates to keep a neutral expression. Similarly, in Fig. 5.7 we have an input text that talks about fashion. The first person is female and she likes this topic, therefore the rule selected by the system is rule 1, which indicates to generate the emotion detected in the input text, i.e., *happiness*. The second subject dislikes fashion and the system selects rule 15 that indicates *disgust*. The last person is male and not interested in this topic, so we get a neutral expression due to rule 2.

5.5 Discussion

Generative models are widely employed for image generation, especially with the coming of GANs. Differently from existing approaches, our system bridges infor-

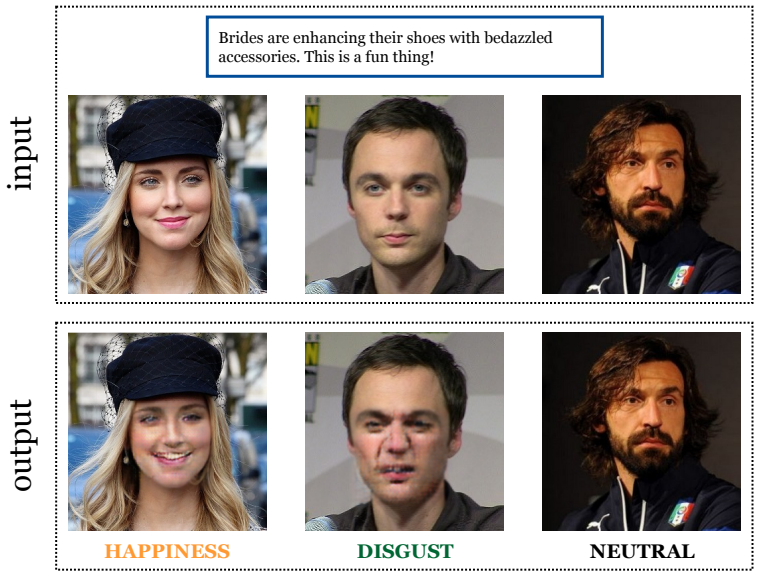


Figure 5.7: Another example of different reactions to the same post. (*Image copyrights belong to their respective owners.*)

mation extraction and image generation. As a matter of fact, we presented a Neural Network-based system that is capable of extracting information from a face picture and a short-text and of automatically generating the facial expression that is associated with the expected reaction of the input person when reading such short-text. While the information extraction process is completely based on CNNs for the image data and LSTMs for the text data, the decision process is based on First Order Logic. Thanks to t-norms, the logic formulas are bridged with the outputs of the neural models, so that the system can also exploit the uncertainty associated to the predictions in order to decide the target reaction. These specific logic rules do not follow the framework of Learning from Constraints, because we have no labeled data from which we could learn what expression to generate. However, if supervised data will become available, these rules could be integrated into the learning.

A Generative Adversarial Network-based model is used to transform a neutral expression into a target emotion, exploiting multiple generators and discriminators, whose constraints are described by FOL formulas. This generative approach allows us to describe the learning scheme in a more understandable way and to easily re-implement it for new image-to-image translation problems, without looking for specific hand-crafted cost functions.

Our experimental analysis reports the outcome of a complete experience composed of stacked neural models, assessing the experimental quality of each of them. The qualitative results of the generated images are still a little poor. We need more powerful machines and more data to improve the quality of the generated examples.

Chapter 6

Speech Emotion Recognition

In this chapter we address the task of *Speech Emotion Recognition* (*SER*), which consist in detecting emotions from speech signals. An *emotional speech dataset*, called *Opera*, has been built extracting clips from acted or dubbed Italian movies, thus obtaining examples of spoken sentences close to real life. This work is part of a project of the Department of Social, Political and Cognitive Sciences of the University of Siena [120]. Known participants labeled the clips extracted from movies, and only those with concordant evaluations were selected. We thus structured a corpus of spoken sentences labelled with the six universal emotions plus the neutral category. In order to evaluate the quality of our dataset we exploited an existing deep model, composed by a *Convolutional Recurrent Neural Network* (*CRNN*) with an attention model. The aim is to use this corpus to train a speech emotion recognition system for real applications.

Speech emotion recognition has numerous applications, such as business, robots, call centre environments [74], games, education, healthcare, crime investigations, and many others. It is a challenging task for various reasons. Voice features keep on changing by age and gender, so a machine should be trained to differentiate these classes. Intonation of emotions shows dissimilarities among different languages, so a model trained on specific data could not be performant for other languages. Moreover real life conversations are noisy and extremely variable, so it is difficult to learn common patterns. Recognizing emotions from speech is difficult also for human ear. Humans help themselves in the detection trying to understand the context. Sometimes to grasp the context, SER is accompanied with other tasks, such as face recognition, topic detection, speech gender and age recognition.

We can consider three kinds of features in speech, namely *prosodic*, *spectral* and *linguistic*. [1]

- Prosodic features are those that can be more easily perceived by humans and are extracted from long segments of speech such as sentences, words and syl-

[1] Other features subdivisions are made as well [3].

Table 6.1: Relationship between prosodic features and emotions.

Emotion	Pitch (average)	Energy	Speech rate
anger	very much higher	much higher	slightly faster
disgust	very much slower	lower	very much slower
fear	very much higher	normal or higher	much faster
happiness	much higher	higher	faster or slower
sadness	slightly lower	lower	slightly slower
surprise	much higher	higher	much faster
neutral	much lower	normal	slower

lables [43]. Among these there are *pitch*, which represents the vibration frequency of the vocal cords during the sound production and describes how high/low the subject speaks, *energy* that is the distribution of the signal amplitude values in time, and *speech rate* which describes the rate of words or syllables uttered over a unit of time. There are correlations between prosodic features and emotional states, as shown in Table 6.1. For example, anger can be described by a faster speech rate, high energy and pitch frequency, while sadness is represented by slower speech rate, lower energy and pitch.

- Spectral features are obtained by converting the time domain signal into the frequency domain signal using Fast Fourier Transform (FFT). They are usually extracted from speech segments of length 20-30 milliseconds. The most widely used spectral feature is Mel Frequency Cepstral Coefficient (MFCC), that is based on a linear cosine transform of a log power spectrum on a non-linear Mel scale of frequency [98]. Other spectral features are Linear Prediction Cepstral Coefficient (LPCC), Perceptual Linear Prediction Coefficient (PLPC), Log Frequency Power Coefficient (LFPC). Spectral features are visually represented with spectrograms. The horizontal axis represents time, while the vertical axis represents frequency. The amplitude (or loudness) of a particular frequency at a particular time is represented by the third dimension, i.e., color. The color scale is red-green-blue, where blue corresponds to low amplitudes and red to high amplitudes.

- Linguistic features are based on semantic information contained in speech. Considering linguistic features is more demanding, since a language model is required to recognize the word sequence of the utterance. For this reason, there are few works which combine acoustic and linguistic features to improve recognition performance [124]. Non linguistic vocalizations can be helpful to detected emotion from speech as well as linguistic features. As matter of fact, studies in cognitive sciences showed that listeners seem to be rather accurate in decoding some non basic affective states such as distress, anxiety, boredom

from non linguistic vocalizations such as laughs [135], coughs, cries [106], sighs, and yawns.

This chapter is organized as follows. Section 6.1 presents the existing works on speech emotion recognition, while Section 6.2 mentions the main datasets containing audio data labeled with emotions. Section 6.3 explains how the Opera corpus has been constructed and the labeling process. In Section 6.4 we provide experimental results that evaluate the quality of the data. Finally, in Section 6.5 some constraints are suggested to build a model able to recognize emotions from speech.

6.1 Related works

For the task of speech emotion recognition several features and classifiers have been considered, such as Hidden Markov Models (HMMs) [102, 123], Gaussian Mixture Models (GMMs) [68], Support Vector Machines (SVMs) [145]. In [100] the authors combine facial expressions and speech with late fusion, adopting Random Forest for the decision task. Traditional speech emotion recognition approaches use handcrafted acoustic features, such as pitch, energy, Mel-Frequency Cepstral Coefficients, Perceptual Linear Prediction coefficient [6, 100].

More recent approaches employ Deep Neural Networks, that enable to automatically detect the features without a manual extraction. Usually spectrograms obtained from the speech signal are directly processed by the deep networks. In general the architecture utilized is a combination of CNNs and RNNs [42, 78, 134, 136]. In [136] speech and visual information are combined in an end-to-end manner, where the outputs coming from the two channels (a CNN processes the speech signal) are fused together and provided to an LSTM. Later, approaches which use attention mechanisms have been developed to focus on emotionally salient parts of the utterances [21, 92, 99, 114]. The first to combine RNNs with attention models was Mirsamadi [92], followed by Ramet [114]. Neumann [99] combined attention mechanisms with CNNs, while Chen [21] used CRNNs. Just lately, Transformers [140], a self-attention model able to get state of the art results in translation and in other NLP tasks, have been employed also for SER. In [133] only the encoder of the Transformer is used to learn in a more robust way to localize the relevant salient parts of speech.

6.2 Datasets

Several speech corpora have been created in a wide variety of languages for developing emotional systems that work on audio [132]. They can be divided into three categories:

- *Acted* (or *simulated*), namely recorded by actors or trained volunteers. These are the most straightforward datasets to train a model, but the further from real world scenarios. As matter of fact, here the acoustic features of the utterances may be exaggerated, while more subtle features may be ignored.
- *Invoked*, i.e., obtained provoking an emotional reaction using staged situations. As compared to acted datasets, these are more naturalistic.
- *Spontaneous*, namely obtained by recording speakers in natural situations or extracting speech from movies, TV programs, call center conversations, radio talks. These datasets are hard to obtain due to the difficulty to classify emotions in "wild" situations, but are the most realistic.

Below we report some of the most famous corpora for speech emotion detection.

- **IEMOCAP**. Interactive Emotional Dyadic Motion Capture [17] is the most widely used dataset in English. It consists of 12h of audio-visual recordings divided into five sessions. In each session a pair of speakers (a female and a male) performed selected emotional scripts and improvised scenarios designed to elicit specific emotions. It contains 10039 utterances with an average duration of 4.5s. The data were labeled with the classes anger, disgust, fear, happiness, sadness, surprise, neutral, excitement and frustration.
- **SAVEE**. Surrey Audio-Visual Expressed Emotion [57] is an English corpus, containing 480 utterances recorded from four male actors. The six basic emotions plus the neutral category are considered.
- **EmoDB**. Berlin Emotional Database [16] is a German acted dataset consisting in about 800 utterances displayed by five male and five female actors. The emotions performed are neutral, fear, happiness, anger, sadness, disgust, and boredom.
- **VAM**. Vera am Mittag database [52] is a German speech corpus of spontaneous emotions, which was recorded from a German TV talk-show. It contains 1018 emotional utterances by 47 speakers (11 males and 36 females). The emotion labels were given from human evaluators on a continuous valued scale for three emotion primitives: valence, activation and dominance (see Section A.2 for explanation of valence, activation, dominance).
- **RECOLA**. *RECOLA* [115] (REmote COLlaborative and Affective interactions) is a multimodal corpus of spontaneous interactions in French. 46 participants were recorded in dyads during a video conference while were completing a task requiring collaboration. Different multimodal data, i.e., audio, video, ECG and physiological (electrocardiogram, and electrodermal activity), were

recorded continuously and synchronously. Six annotators measured emotions continuously on two dimensions, i.e., arousal and valence, as well as social behavior labels on five dimensions.

- **Emovo**. Emovo [25] is the first dataset of emotional speech for the Italian language. It contains 588 utterances simulated by six actors (three males and three females), subdivided into the six universal emotions plus neutral.

6.3 Opera: an Emotional Speech Dataset

6.3.1 Corpus Construction

Opera corpus is part of a wider project aiming at building end-to-end applications for real world scenarios. As we have seen in Section 6.2, the existing emotional speech datasets are mainly simulated by actors or invoked. The limitation of both these types of corpora is that the number of different voices is small, and besides the acted ones are not spontaneous. To create a robust application capable to identify emotions in real world scenarios, we preferred to rely on a dataset containing a relevant number of different voices and in which emotions arise naturally during conversation. For these reasons, we decided to exploit clips extracted from a collection of movies. To our best knowledge, a dataset containing spontaneous speech in Italian labeled with emotions does not exist.

Clips were extracted from a collection of 141 both Italian or Italian dubbed movies. Different genres of movies were selected to guarantee that all desired emotions would be available. All movies have an high audio quality with sampling frequency of 44.1 or 48 KHz and a stereo channel. First subtitles were exploited to select timestamp intervals of a candidate clip. In this process, only clips involving a single actor and with a time-length at least of 2.5 seconds were selected. Then, a group of people discarded the ones were actor's voice was covered by other sounds.

Finally, remaining clips were available for evaluation. We considered the six universal emotions plus neutral. About 50 people were asked to evaluate clips selecting between one or two of the seven emotions or skipping it whenever an emotion not belonging to the universal ones was recognized. We ensured that a single clip was evaluated only one time by the same person and at least by three people. The participants could specify up to two emotions for each clip and for each emotion they could specify the degree of that emotion, as low, medium or high. In natural conversations multiple emotions can be expressed at the same time, and as a consequence, the possibility to report a degree of intensity for each emotion helps in expressing their inherent fuzzy nature.

	anger	*disgust*	*fear*	*happiness*	*sadness*	*surprise*	*neutral*
anger	**537**	-	-	-	-	-	-
disgust	155	**232**	-	-	-	-	-
fear	15	1	**142**	-	-	-	-
happiness	0	2	1	**318**	-	-	-
sadness	32	16	34	0	**410**	-	-
surprise	43	26	41	76	28	**351**	-
neutral	20	13	12	42	59	92	**537**

Table 6.2: Emotion distribution for 3235 clips selected with an average approach. Diagonal: number of clips with a single emotion. Under the diagonal: number of clips with two emotions.

6.3.2 Corpus Characteristic

From the manual evaluation we obtained 5078 wav audio clips sampled at 44.1 or 48 KHz. For each clip up to 3 votes have been collected. Considering that each rater could assign up to two labels for an example and that a degree could be assigned to each emotion, we decided to use an average approach to automatically select the dominant emotion for each clip. We assigned to each level a correspondent numeric approximation: 0 for not recognized, 0.33 for low, 0.66 for medium and 1 for high degree. Then we summed all the votes obtained by the clip and computed the average level for each possible emotion. Let's suppose that we have the three following votes for a clip:

#1	*happiness* - 1 , *surprise* - 0.33	
#2	*happiness* - 0.66	
#3	*surprise* - 0.66	*happiness* - 0.553
	threshold = 0.33	

The resulting emotion distribution for this clip is $happiness = 0.553$ and $surprise = 0.33$. Defining a threshold, this criterion permits to identify, in most cases, the dominant emotion. When there is not a dominant one, we considered the clip as multi-labeled or even to be discarded if all emotions are below the chosen threshold. For our purposes we are interested in identifying those clips where there are up to two dominant emotions. Using a threshold of 0.33, we obtained a corpus consisting on a total of 3235 examples whose emotion distribution is reported in Table 6.2. On the diagonal of the symmetric matrix is reported the number of clips with a single dominant emotion. Out of the diagonal there is the number of clips with two dominant emotions. The evaluations seem to be consistent, as matter of fact "happiness" is never with "anger" or "sadness", and "anger" is accompanied by "disgust" frequently. Sometimes "happiness" and "surprise" are together, and maybe this happens when euphoria is expressed.

6.4 Experimental Results

We have not precise methodologies to evaluate the effectiveness and quality of Opera. As first test, we exploited the model presented in [21], that achieves good results in terms of unweighted average recall on IEMOCAP and EmoDB (see Section 6.2 for the description of these datasets). We suppose that if our data are reasonable, such consolidated existing model should perform rather well also on them. Certainly it is not an accurate approach, but is a first evaluation before building a specific speech emotion recognition system for real applications.

The model proposed in [21] is a three-dimensional attention-based CRNN. Firstly, log Mel-spectrogram (static, deltas, and delta-deltas) are extracted from speech signals as the 3-D CNN input, and then 3-D CNN is combined with LSTM. An attention layer is used to focus on the emotionally relevant frames of the speech. Finally, a fully connected layer with softmax as activation function gives the probability distribution on emotions. The CNN input $X \in \mathbb{R}^{t \times f \times c}$ is essentially a 3-D feature representation of the speech, where t is the frame length, f is the number of Mel-filter bank and $c = 3$ is the number of feature channels, i.e., static, deltas, and delta-deltas. In [21] experiments were performed on IEMOCAP and EmoDB, where in the first dataset only four classes were considered, i.e., *happiness*, *anger*, *sadness* and *neutral*. At the beginning of our experimental analysis on Opera corpus we considered only the same four emotions, and then we took all the seven classes of the dataset. First of all, we selected from the entire corpus, a subset of clips with a single dominant emotion. The resulting dataset consists of 315 clips labeled as "happiness", 410 for "sadness", "anger" and "neutral", 350 for "surprise", 140 for "fear" and 230 for "disgust".

We split the clips into sub-parts with maximum duration of 4 seconds. Considering that most of the examples have length between 3 and 6 seconds, we adopted a criterion which guarantees that only few data contain padding or are divided in no more than two sub-parts. We considered also overlapping between sub-parts. The clips are split as follows. (i) Padding is applied at the end in those clips shorter than $4s$. (ii) If the length is between 4 and $5.5s$ we take the last $4s$ of the clip, since emotions usually emerge after few seconds from the beginning of the speech. (iii) If the length is between 5.5 and $6s$ we obtain two clips, one of $4s$ starting from the beginning and the other of $4s$ from the end. (iv) If the length is greater than $6s$ we generate two clips, one of $4s$ starting from $1s$ and the other of $4s$ from the end. In the training phase each sub-segment is considered as one example, while in the validation and test phases more sub-segments of a same clip are evaluated as a single example.

In our configurations the frame length t is 400 and the number of filters f is 40, so the CNN input is $X \in \mathbb{R}^{400 \times 40 \times 3}$. We used Adam optimizer with learning rate

	recall %	precision %	F1 %
4 classes	60.2 (3.2)	61.1 (3.1)	60.6 (3.1)
7 classes	48.7 (1.5)	53.5 (6.0)	50.8 (3.1)

Table 6.3: Macro average recall, precision, F1 score, on test set with four and seven classes. In brackets standard deviation.

	anger	*disgust*	*fear*	*happiness*	*sadness*	*surprise*	*neutral*
anger	**58.54**	7.32	7.32	4.88	3.66	15.71	12.2
disgust	8.7	**39.13**	0	8.7	15.22	5.71	28.26
fear	10.71	3.57	**67.86**	3.57	7.14	4.29	7.14
happiness	14.29	12.7	3.17	**34.92**	12.7	15.71	12.7
sadness	3.66	2.44	1.22	6.1	**67.07**	14.29	14.63
surprise	15.71	5.71	4.29	15.71	14.29	**27.14**	17.14
neutral	6.1	7.32	0	4.88	21.95	17.14	**54.88**

Table 6.4: Confusion matrix on seven classes (percentages). On the rows the targets and on the columns the predictions.

10^{-4} and batch size 40. For both the experiments with four and seven classes, we evaluated architectures with different numbers of convolutional layers, and we performed 5-fold cross validation, taking three folds for training set, one for validation and the remaining one for the test. We considered CNNs with number of layers from 4 to 6, where all the convolutional layers are followed by Leaky ReLU. Table 6.3 shows the results calculated on the test set on four and seven classes. We reported the macro average recall, precision and F1 score on the five folders, for the model which provided best macro average recall in validation set, that was with 5 convolutional layers. With four classes, the results are good, compared with the results in [21] on IEMOCAP. [2] Increasing the number of classes the scores are lower as expected but still acceptable considering that Opera is a dataset of spontaneous speech from hundreds of different speakers and that contains emotions difficult to recognize from audio, as shown in Table 6.4. The poorer results come from "happiness", as it happens in [21] and from "surprise" and "disgust" that are not easy to classify from speech also for humans. Nevertheless no emotions are absolutely misclassified. The results are impressive on the class "fear", where we have few data, and other good scores come from "anger", "sadness" and "neutral", as in [21].

6.5 Discussion

Speech emotion recognition has several applications and is a challenging problem, due to voice changes in gender and age, and expression variation in different lan-

[2] In [21] the recall on four classes is 64.74% and standard deviation 5.44.

guages. Moreover the existing datasets used to train SER models present limitations. Acted corpora are furthest from real contexts and in general contain voices of few people. Invoked dataset are more naturalistic, but also in this case usually the number of examples and of different subjects is low. Motived by these drawbacks, we constructed a speech emotional dataset extracting clips from movies. In this way we collected a larger number of utterances from different genders, ages and situations. This type of data seems to better represent real scenarios. To our best knowledge, we built the first spontaneous dataset containing speech in Italian labeled with emotions.

We performed a preliminary experimentation using an existing deep model to evaluate the quality of our dataset. The results obtained on this corpus are at least comparable with the results of other widely used datasets, thus offering the opportunity to exploit it in real contexts. The next step will be exactly to develop a new speech emotion recognition system that can be employed in real applications. In order to improve the model performance, some constraints could be injected into the learning problem. In this case, where the targets annotation could be subjected to human error, soft constraints seem to be a good solution.

Logic formulas can be defined in a similar way as made for text in Section 4.4. Supposing that the model ends with a fully connected layer with softmax activation function, for each example x it outputs the probability distribution on emotions $p(x)$. Each class is associated to a predicate, that can be seen as the component of the vectorial function $p(x)$, i.e.,

$$p(x) = [anger(x), disgust(x), fear(x), happiness(x), sadness(x), surprise(x), neutral(x)] .$$

We can define the following rules which contrast opposite emotions:

(1) $\forall x\ anger(x) \vee sadness(x) \Rightarrow \neg happiness(x)$

(2) $\forall x\ happiness(x) \Rightarrow \neg(anger(x) \vee sadness(x))$

(3) $\forall x\ disgust(x) \Rightarrow \neg happiness(x)$

(4) $\forall x\ happiness(x) \Rightarrow \neg disgust(x)$

(5) $\forall x\ fear(x) \Rightarrow \neg happiness(x)$

(6) $\forall x\ happiness(x) \Rightarrow \neg fear(x)$

(7) $\forall x\ anger(x) \vee disgust(x) \vee fear(x) \vee happiness(x) \vee sadness(x) \vee surprise(x) \Rightarrow \neg neutral(x)$

(8) $\forall x\ neutral(x) \Rightarrow \neg(anger(x) \vee disgust(x) \vee fear(x) \vee happiness(x) \vee sadness(x) \vee surprise(x))$.

Formulas (1) and (2) simply say that if an utterance expresses *anger* or *sadness* cannot express *happiness* at the same time, and vice-versa. Implications (3)-(6) should be less weighted, and state that *happiness* cannot be in presence with *disgust* and *fear*. Formulas (7) and (8) assert that if an example is classified as *neutral*, cannot contain

emotions, and vice-versa. As made in Section 4.4, these logic rules, to be exploited during the learning, can be converted into real-valued functions through t-norms.

We could also think to use audio data with labels different from the basic emotions (if available), connecting these labels and emotions through logic rules, as made for text with Facebook reactions. We could try to impose a temporal coherence similarly as made for facial expression recognition (Eq. 3.4). The audio clip can be divided in sub-parts, and each of them will be provided to the network. Temporal coherence will ask the prediction in the sub-part t to be coherent with the prediction of the sub-part $t + 1$.

Chapter 7

Conclusions

This chapter summarizes the contribution of the thesis and discusses avenues for future research.

In this thesis we addressed several Affective Computing tasks with deep learning-based approaches, that allow machines to handle something not specifically programmed, as emotions. We have seen that emotions can be integrated into machines in different ways and they are important for several applications, such as healthcare, education, automatic driver assistance, entertainment, and so on. Thinking to a machine that behaves in a similar way to humans, in this work we dealt with several problems, such as facial expression recognition, text emotion recognition, facial expression generation, and speech emotion recognition. We handled categorical approaches, considering the six universal emotions defined by Ekman (anger, disgust, fear, happiness, sadness, and surprise), and the neutral class, as well.

We addressed these tasks in a novel way following the framework of Learning from Constraints, which integrates low-level tasks processing sensorial data and reasoning using higher-level semantic knowledge. Prior knowledge is formally expressed as constraints that should be satisfied during the training. In this way, we developed heavily structured learning environments, where machines could act more cleverly. We used also constraints based on First Order Logic (FOL) which provided an expressive and formally well-defined representation for knowledge. FOL-based formulation enables to better deal with cases in which the number of constraints is large, and so it can be more easy to avoid inconsistency in the problem definition.

Moreover, with Learning from Constraints we avoided to use a great quantity of labeled data, a deep learning limitation, and we exploited also unsupervised data, that are easier to collect. This approach based on constraints can be used also for tasks different from the ones we addressed, where the number of supervised data is small and where you want to formally integrate additional knowledge into the learning problem.

For the problem of facial expression recognition, we developed a model based on CNNs that detects expressions in static images and that can be further applied to video sequences. We also considered the effects of classifying representations of different face parts. We exploited three coherence constraints to improve the performance of the full face classifier. A temporal coherence enforced the predictions to be coherent over time, a part-based coherence enforced the prediction of the full-face classifier to be coherent with the (average) prediction of the other face part classifiers, and a coherence between feature representations enforced the prediction of the appearance-based classifier to be coherent with the prediction of the shape-based classifier for each part. We also experimented that this model is able to recognize expressions in presence of occlusions. This is an extremely up to date topic, as nowadays most of us have part of our faces covered, due to the need of wearing a mask in day to day life.

To address the task of text emotion recognition, a few datasets containing textual data labeled with emotions exist. To overcome this problem we proposed a neural network-based model that jointly learns the tasks of emotion detection and Facebook reaction prediction, when processing raw text. FOL-based formulas were exploited to easily express relations between reactions and emotions. In this way, we used few data labeled with emotion classes, data labeled with reactions and a large collection of unsupervised data. We found out that the introduction of logic rules improves both the tasks, and this model performs better than the model with artificial labels, where the training data are augmented with fixed mappings between reactions and emotions.

We addressed the problem of facial expression generation in a more extensive way, considering the reaction that a person would make reading a text. We bridged two processes, i.e., information extraction from inputs and image generation. FOL-based formulas were used to mix the information extracted from the inputs and and to decide which emotion to generate. These logic rules were not employed during the training, because we had no labeled data from which we could learn what emotion to generate. If supervised data become available, the logic formulas could be clearly injected into the learning problem. To translate the neutral input face into the specific expression, instead we followed the same t-norm-based implementation we used to train the emotion classifier on text. This generative approach based on FOL formulas allowed us to describe the learning scheme in a easier and clearer way, and can be effortlessly re-implemented for other image translation tasks.

For speech emotion recognition, we built a dataset extracting clips from acted or dubbed Italian movies. Existing emotional speech datasets are essentially acted or invoked. The first ones are not naturalistic and in general both contain few different voices. On our experience instead, we collected utterances from different genders, age, situations, so closer to real life scenarios. To the best of our knowledge, this is

the first emotional speech dataset in Italian language.

Clearly, all the presented models can be improved. If more powerful machines become available, we will try to use Transformers for the emotion classifiers on text and on speech, and we could obtain better qualitative results for the expression generation. The emotion predictor on text can be improved also considering lexical resources. The quality of the facial expression recognizer can be enhanced adding more data to the training set.

As future work and perspective, the starting point for us, will be to try to train a speech emotion recognizer with the dataset presented in Chapter 6, with the aim of applying it to real world scenarios. To improve the model some constraints can be inserted in the learning problem, as suggested in Section 6.5. We could also try to combine speech and language, defining some constraints that allow us it.

Looking at possible applications, the facial expression classifier, the text emotion predictor and the speech emotion recognizer can be joined to improve the overall performance. Since such models coming from different modalities are already trained, the best solution would probably be to apply a late fusion (see Section 2.5). Moreover, imagining an agent that behaves as close as possible to a human, we could also think of integrating the facial expression generator with the other models proposed.

Appendix A

Theories of Emotions

Emotions have been investigated for many years and several theories about that have been developed. Some researchers consider emotions as discrete while others prefer to categorized them in multiple dimensions.

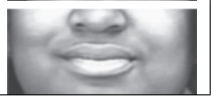
A.1 Categorical approaches

Some theories state that humans have a set of basic emotions recognizable across different cultures, which are considered as discrete categories. These categorical approaches assume that different emotions arise from separate neural systems.

A.1.1 Paul Ekman

Paul Ekman, American psychologist, is one of the greatest researchers about emotions. In the late 1960s, he studied non-verbal behaviours in an isolated tribe in Papua Nuova Guinea, and demonstrated that facial expressions and emotions are not determined by culture, but are universal. He defined six basic emotions, namely *anger, disgust, fear, happiness, sadness, surprise* [37]. In 1992 Ekman extended the list, adding emotions defined secondary: amusement, contempt, contentment, embarrassment, excitement, guilt, pride, relief, satisfaction, sensory pleasure, and shame [36, 38]. By 1978, Ekman and Friesen developed the *Facial Action Coding System* (*FACS*), a system for describing facial expressions into individual components of muscle movements, called Action Units (AUs). Initially 30 AUs based on facial muscle movements were defined, later details on head movements and eye positions were added (now there are 44 AUs in total) [23]. An AU consists of three basic parts: AUnumber, FACS name, and muscular basis. Table A.1 shows some examples of AUs. FACS is helpful in the diagnosis of mental disorders, lie detection, psychology, animation, interview, and other environments involving communications.

Table A.1: Examples of Action Units (AUs). An AU is composed by an identifier number, the name, and the muscles involved in movement.

AUNumber	FACS name	Muscolar basis	Example
1	Inner Brow Raiser	Frontalis, Pars Medialis	
2	Outer Brow Raiser	Frontalis, pars lateralis	
4	Brow Lowerer	Corrugator supercilii, Depressor supercilii	
5	Upper Lid Raiser	Levator palpebrae superioris	
6	Check Raiser	Orbicularis oculi, pars orbitalis	
9	Nose Wrinkler	Levator labii superioris alaquae nasi	
10	Upper Lip Raiser	Levator labii superioris	
11	Nasolabial Deepener	Zygomaticus minor	
12	Lip Corner Puller	Zygomaticus major	

Ekman's research about facial micro-expressions inspired the award-winning tv series *Lie to Me*, of which he analyzed each episode. Moreover, he made her own scientific contribution to the creation of the successful movie *Inside Out*, which focuses on emotions and family dynamics.

A.1.2 Plutchik's wheel

In 1980, Robert Plutchik created a wheel of emotions, where there are four couples of primary emotions, i.e., *joy* versus *sadness*, *anger* versus *fear*, *trust* versus *disgust*, and *surprise* versus *anticipation* [109]. In Figure A.1 the model is shown. The 8 primary emotions are located in the second circle and the opposite ones are across from each other. Moving to the center of the circle, emotions and colors are more intense, while moving to the outer layers, colors become less saturated, and the intensity of the emotions drops. Other affective states are represented as combinations of the primary emotions. For instance love is the combination of joy and trust, guilt of joy

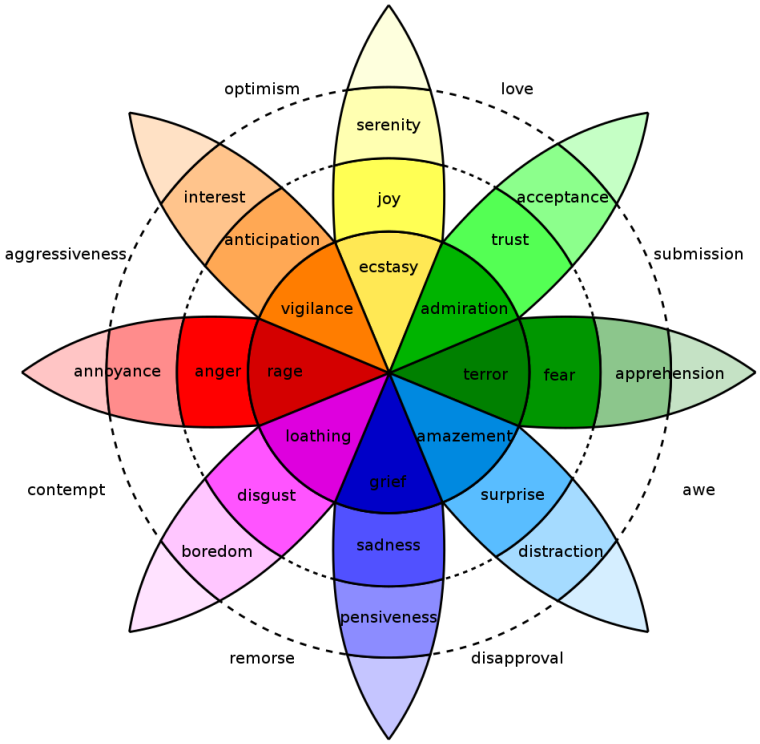


Figure A.1: Plutchik's wheel, composed by 8 primary bipolar emotions: joy - sadness, anger - fear, trust - disgust, and surprise - anticipation. Emotions intensity is represented by the color and is more intense when they move from the outside to the center of the wheel. Emotions with no color are a mix of two primary emotions.

and fear, delight of joy and surprise.

A.2 Dimensional approaches

Dimensional approaches represent emotions as coordinates in a multi-dimensional space. These models are contrary to the ones based on categorical emotions, as matter of fact, they suggest that a common and interconnected neurophysiological system is responsible for all affective states.

A.2.1 Russell's circumplex

In 1980, Russell devised the circumplex model, which represents an affective state as a point in a two dimensional space [118]. In Figure A.2 the circumplex is shown, where the axis x describes the *valence* (unpleasant-pleasant continuum) and the axis y the *arousal* (deactivation-activation continuum). Valence represents the intrinsic attractiveness or averseness of an emotion, while arousal represents the physiolog-

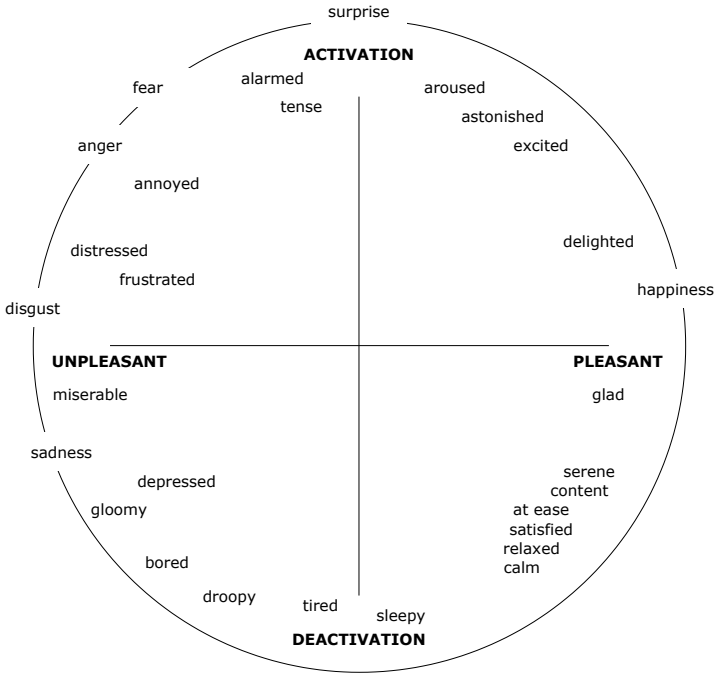


Figure A.2: Russell's circumplex. Emotions are distributed in a two-dimensional circular space. The x-axis is the continuum between unpleasant and pleasant emotions. The y-axis is the continuum between low and high arousal. The six universal emotions are located on the border for greater visual impact.

ical and psychological state of being reactive to stimuli. Happiness, for instance, is conceptualized as an emotional state with strong valence and moderate arousal. Surprise has a very strong activation, but is neutral in the unpleasant-pleasant continuum.

A.2.2 PAD model

Pleasure-Arousal-Dominance (PAD) model has been developed by Russell and Mehrabian from 1974 [88, 119]. It uses three orthogonal dimensions to represent all emotions, namely pleasure-displeasure, arousal-nonarousal and dominance-submissiveness. The pleasure-displeasure dimension measures how pleasant is an emotion, and indicates the valence. For this reason the PAD model is also called VAD (Valence-Arousal-Dominance) model. The arousal-nonarousal dimension measures the intensity of the emotion, while dominance-submissiveness represents the dominant and controlled nature of the emotion. Dominance is related to freedom or limitation regarding an individual's behaviour. It makes possible to distinguish anger from anxiety, attention from surprise, contempt from impotence. Figure A.3 shows how the six universal emotions are placed in the three dimensional space. For instance, both anger and fear have low valence and high arousal, the difference is that

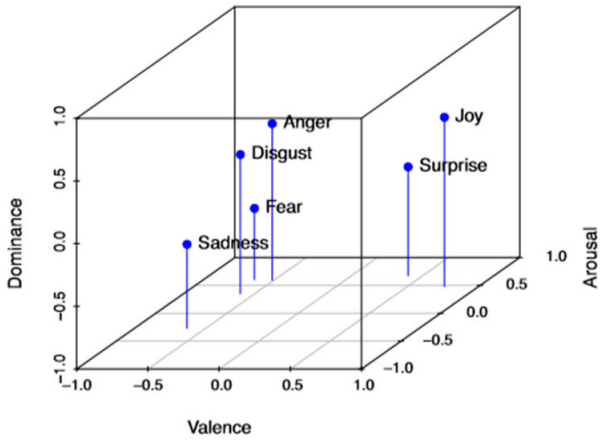


Figure A.3: Valence-Arousal-Dominance model. Emotions are represented in a three dimensional space.

anger is a dominant emotion while fear is submissive.

Appendix B

Publications

Journal papers

1. **Lisa Graziani**, S. Melacci, M. Gori, "Coherence Constraints in Facial Expression Recognition", *Intelligenza Artificiale*, pages:79–92, 2019. **Candidate's contributions**: designed algorithms, carried out theoretical analyses, experimental setup.

Peer reviewed conference papers

1. **Lisa Graziani**, S. Melacci, M. Gori, "The Role of Coherence in Facial Expression Recognition", *International Conference of the Italian Association for Artificial Intelligence*, pages:320–333, 2018. **Candidate's contributions**: designed algorithms, carried out theoretical analyses, experimental setup. (**Best student paper award**)

2. **Lisa Graziani**, S. Melacci, M. Gori, "Jointly Learning to Detect Emotions and Predict Facebook Reactions", *International Conference on Artificial Neural Networks* , pages:185–197, 2019. **Candidate's contributions**: designed algorithms, carried out theoretical analyses, experimental setup.

3. **Lisa Graziani**, S. Melacci, M. Gori, "Generating Facial Expressions Associated with Text", *International Conference on Artificial Neural Networks*, pages:621-632, 2020. **Candidate's contributions**: designed algorithms, carried out theoretical analyses, experimental setup.

Papers under review

1. **Lisa Graziani**, M. Gori, S. Melacci, "A Language Modeling-Like Approach to Sketching", Submitted to: *Neural Networks*. **Candidate's contributions**: designed algorithms, carried out theoretical analyses, experimental setup.

2. C. Saccà, **Lisa Graziani**, O. Parlangeli, M. Masini "Opera: an Italian Dataset for Speech Emotion Recognition.", Submitted to: *International Journal of Speech Technology*. **Candidate's contributions**: carried out theoretical analyses, dataset evaluation, experimental setup.

Bibliography

[1] Agrawal, A. and An, A. (2012). Unsupervised emotion detection from text using semantic and syntactic relations. In *Proceedings of the The International Joint Conferences on Web Intelligence and Intelligent Agent Technology*, pages 346–353. IEEE.

[2] Aifanti, N., Papachristou, C., and Delopoulos, A. (2010). The mug facial expression database. In *11th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS 10*, pages 1–4. IEEE.

[3] Akçay, M. B. and Oğuz, K. (2020). Speech emotion recognition: Emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers. *Speech Communication*, 116:56–76.

[4] Alm, C. (2008). *Affect in Text and Speech*. PhD thesis, University of Illinois.

[5] Aman, S. and Szpakowicz, S. (2007). Identifying expressions of emotion in text. In *International Conference on Text, Speech and Dialogue*, pages 196–205. Springer.

[6] Anagnostopoulos, C.-N., Iliou, T., and Giannoukos, I. (2015). Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011. *Artificial Intelligence Review*, 43(2):155–177.

[7] Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR.

[8] Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

[9] Bauml, M., Tapaswi, M., and Stiefelhagen, R. (2013). Semi-supervised learning with constraints for person identification in multimedia data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3602–3609.

[10] Baziotis, C., Athanasiou, N., Chronopoulou, A., Kolovou, A., Paraskevopoulos, G., Ellinas, N., Narayanan, S., and Potamianos, A. (2018). Ntua-slp at semeval-2018 task 1: Predicting affective content in tweets with deep attentive rnns and transfer learning. *arXiv preprint arXiv:1804.06658*.

[11] Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer.

[12] Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152.

[13] Bota, P. J., Wang, C., Fred, A. L., and Da Silva, H. P. (2019). A review, current challenges, and future possibilities on emotion recognition using machine learning and physiological signals. *IEEE Access*, 7:140990–141020.

[14] Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

[15] Burkert, P., Trier, F., Afzal, M. Z., Dengel, A., and Liwicki, M. (2015). Dexpression: Deep convolutional neural network for expression recognition. *arXiv preprint arXiv:1509.05371*.

[16] Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W. F., and Weiss, B. (2005). A database of german emotional speech. In *Ninth European Conference on Speech Communication and Technology*.

[17] Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., and Narayanan, S. S. (2008). Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42(4):335.

[18] Chaffar, S. and Inkpen, D. (2011). Using a heterogeneous dataset for emotion analysis in text. In *Canadian Conference on Artificial Intelligence*, pages 62–67. Springer.

[19] Chatterjee, A., Gupta, U., Chinnakotla, M. K., Srikanth, R., Galley, M., and Agrawal, P. (2019a). Understanding emotions in text using deep learning and big data. *Computers in Human Behavior*, 93:309–317.

[20] Chatterjee, A., Narahari, K. N., Joshi, M., and Agrawal, P. (2019b). Semeval-2019 task 3: Emocontext contextual emotion detection in text. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 39–48.

[21] Chen, M., He, X., Yang, J., and Zhang, H. (2018). 3-d convolutional recurrent neural networks with attention model for speech emotion recognition. *IEEE Signal Processing Letters*, 25(10):1440–1444.

[22] Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., and Choo, J. (2018). Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8789–8797.

[23] Cohn, J. F., Ambadar, Z., and Ekman, P. (2007). Observer-based measurement of facial expression with the facial action coding system. *The handbook of emotion elicitation and assessment*, 1(3):203–221.

[24] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

[25] Costantini, G., Iaderola, I., Paoloni, A., and Todisco, M. (2014). Emovo corpus: an italian emotional speech database. In *International Conference on Language Resources and Evaluation (LREC 2014)*, pages 3501–3504. European Language Resources Association (ELRA).

[26] Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314.

[27] de Gelder, B. and Hortensius, R. (2014). The many faces of the emotional body. In *New frontiers in social neuroscience*, pages 153–164. Springer.

[28] De Una, D., Rümmele, N., Gange, G., Schachte, P., and Stuckey, P. J. (2018). Machine learning and constraint programming for relational-to-ontology schema mapping. In *IJCAI*, volume 2018, page 27th.

[29] Deriso, D., Susskind, J., Krieger, L., and Bartlett, M. (2012). Emotion mirror: a novel intervention for autism based on real-time expression recognition. In *European Conference on Computer Vision*, pages 671–674. Springer.

[30] Dhall, A., Goecke, R., Lucey, S., and Gedeon, T. (2012). Collecting large, richly annotated facial-expression databases from movies. *IEEE Annals of the History of Computing*, 19(03):34–41.

[31] Dhall, A., Ramana Murthy, O., Goecke, R., Joshi, J., and Gedeon, T. (2015). Video and image based emotion recognition challenges in the wild: Emotiw 2015. In *Proceedings of the 2015 ACM on international conference on multimodal interaction*, pages 423–426.

[32] Diligenti, M., Roychowdhury, S., and Gori, M. (2017). Integrating prior knowledge into deep learning. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 920–923. IEEE.

[33] Du, S., Tao, Y., and Martinez, A. M. (2014). Compound facial expressions of emotion. *Proceedings of the National Academy of Sciences*, 111(15):E1454–E1462.

[34] Duchenne, G.-B. and de Boulogne, G.-B. D. (1990). *The mechanism of human facial expression*. Cambridge university press.

[35] Ebbinghaus, H.-D., Flum, J., and Thomas, W. (2013). *Mathematical logic*. Springer Science & Business Media.

[36] Ekman, P. (1992). An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200.

[37] Ekman, P. and Friesen, W. V. (1971). Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2):124.

[38] Ekman, P. E. and Davidson, R. J. (1994). *The nature of emotion: Fundamental questions.* Oxford University Press.

[39] El-Kaddoury, M., Mahmoudi, A., and Himmi, M. M. (2019). Deep generative models for image generation: A practical comparison between variational autoencoders and generative adversarial networks. In *International Conference on Mobile, Secure, and Programmable Networking*, pages 1–8. Springer.

[40] Fabian Benitez-Quiroz, C., Srinivasan, R., and Martinez, A. M. (2016). Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5562–5570.

[41] Fan, X. and Tjahjadi, T. (2015). A spatial-temporal framework based on histogram of gradients and optical flow for facial expression recognition in video sequences. *Pattern Recognition*, 48(11):3407–3416.

[42] Fayek, H. M., Lech, M., and Cavedon, L. (2017). Evaluating deep learning architectures for speech emotion recognition. *Neural Networks*, 92:60–68.

[43] Fox, A. et al. (2000). *Prosodic features and prosodic structure: The phonology of suprasegmentals*. Oxford University Press.

[44] Gnecco, G., Gori, M., Melacci, S., and Sanguineti, M. (2015). Foundations of support constraint machines. *Neural computation*, 27(2):388–480.

[45] Gnecco, G., Gori, M., Melacci, S., and Sanguineti, M. (2015). Learning with mixed hard/soft pointwise constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 26(9):2019–2032.

[46] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

[47] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

[48] Goodfellow, I. J., Erhan, D., Carrier, P. L., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D.-H., et al. (2013). Challenges in representation learning: A report on three machine learning contests. In *International conference on neural information processing*, pages 117–124. Springer.

[49] Gori, M. (2017). *Machine Learning: A constraint-based approach*. Morgan Kaufmann.

[50] Graziani, L., Melacci, S., and Gori, M. (2019). Jointly learning to detect emotions and predict facebook reactions. In *International Conference on Artificial Neural Networks*, pages 185–197. Springer.

[51] Graziani, L., Melacci, S., and Gori, M. (2020). Generating facial expressions associated with text. In *International Conference on Artificial Neural Networks*, pages 621–632. Springer.

[52] Grimm, M., Kroschel, K., and Narayanan, S. (2008). The vera am mittag german audio-visual emotional speech database. In *2008 IEEE international conference on multimedia and expo*, pages 865–868. IEEE.

[53] Gross, R., Matthews, I., Cohn, J., Kanade, T., and Baker, S. (2010). Multi-pie. *Image and Vision Computing*, 28(5):807–813.

[54] Gunes, H. and Schuller, B. (2013). Categorical and dimensional affect analysis in continuous input: Current trends and future directions. *Image and Vision Computing*, 31(2):120–136.

[55] Hájek, P. (2013). *Metamathematics of fuzzy logic*, volume 4. Springer Science & Business Media.

[56] Happy, S. and Routray, A. (2015). Automatic facial expression recognition using features of salient facial patches. *IEEE Transactions on Affective Computing*, 6(1):1–12.

[57] Haq, S., Jackson, P. J., and Edge, J. (2009). Speaker-dependent audio-visual emotion recognition. In *AVSP*, pages 53–58.

[58] Herzig, J., Shmueli-Scheuer, M., and Konopnicki, D. (2017). Emotion detection from text via ensemble classification using word embeddings. In *Proceedings of the International Conference on Theory of Information Retrieval*, pages 269–272. ACM.

[59] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[60] Hu, Z., Ma, X., Liu, Z., Hovy, E., and Xing, E. (2016). Harnessing deep neural networks with logic rules. *arXiv preprint arXiv:1603.06318*.

[61] Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.

[62] Jain, S., Hu, C., and Aggarwal, J. K. (2011). Facial expression recognition with temporal modeling of shapes. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1642–1649. IEEE.

[63] Jung, H., Lee, S., Yim, J., Park, S., and Kim, J. (2015). Joint fine-tuning in deep neural networks for facial expression recognition. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 2983–2991. IEEE.

[64] Kapoor, A., Burleson, W., and Picard, R. W. (2007). Automatic prediction of frustration. *International journal of human-computer studies*, 65(8):724–736.

[65] Kazemi, V. and Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1867–1874.

[66] Kim, S. M., Valitutti, A., and Calvo, R. A. (2010). Evaluation of unsupervised emotion models to textual affect recognition. In *NAACL HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 62–70. ACL.

[67] Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

[68] Koolagudi, S. G., Reddy, R., and Rao, K. S. (2010). Emotion recognition from speech signal using epoch parameters. In *2010 international conference on signal processing and communications (SPCOM)*, pages 1–5. IEEE.

[69] Kratzwald, B., Ilić, S., Kraus, M., Feuerriegel, S., and Prendinger, H. (2018). Deep learning for affective computing: Text-based emotion recognition in decision support. *Decision Support Systems*, 115:24–35.

[70] Krebs, F., Lubascher, B., Moers, T., Schaap, P., and Spanakis, G. (2018). Social emotion mining techniques for facebook posts reaction prediction. In *Proceedings of the 10th International Conference on Agents and Artificial Intelligence*.

[71] Lample, G., Zeghidour, N., Usunier, N., Bordes, A., Denoyer, L., and Ranzato, M. (2017). Fader networks: Manipulating images by sliding attributes. In *Advances in Neural Information Processing Systems*, pages 5967–5976.

[72] LeCun, Y., Bengio, Y., et al. (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995.

[73] LeCun, Y. et al. (1989). Generalization and network design strategies. *Connectionism in perspective*, 19:143–155.

[74] Lee, C. M. and Narayanan, S. S. (2005). Toward detecting emotions in spoken dialogs. *IEEE transactions on speech and audio processing*, 13(2):293–303.

[75] Levi, G. and Hassner, T. (2015). Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. In *Proceedings of the 2015 ACM on international conference on multimodal interaction*, pages 503–510.

[76] Li, S. and Deng, W. (2020). Deep facial expression recognition: A survey. *IEEE Transactions on Affective Computing*.

[77] Li, S., Deng, W., and Du, J. (2017). Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2852–2861.

[78] Lim, W., Jang, D., and Lee, T. (2016). Speech emotion recognition using convolutional and recurrent neural networks. In *2016 Asia-Pacific signal and information processing association annual summit and conference (APSIPA)*, pages 1–4. IEEE.

[79] Littlewort, G. C., Bartlett, M. S., and Lee, K. (2007). Faces of pain: automated measurement of spontaneousallfacial expressions of genuine and posed pain. In *Proceedings of the 9th international conference on Multimodal interfaces*, pages 15–21.

[80] Liu, M.-Y., Breuel, T., and Kautz, J. (2017). Unsupervised image-to-image translation networks. In *Advances in neural information processing systems*, pages 700–708.

[81] Liu, M.-Y. and Tuzel, O. (2016). Coupled generative adversarial networks. In *Advances in neural information processing systems*, pages 469–477.

[82] Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738.

[83] Long, F. and Bartlett, M. S. (2016). Video-based facial expression recognition using learned spatiotemporal pyramid sparse coding features. *Neurocomputing*, 173:2049–2054.

[84] Lopes, A. T., de Aguiar, E., De Souza, A. F., and Oliveira-Santos, T. (2017). Facial expression recognition with convolutional neural networks: coping with few data and the training sample order. *Pattern Recognition*, 61:610–628.

[85] Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., and Matthews, I. (2010). The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 ieee computer society conference on computer vision and pattern recognition-workshops*, pages 94–101. IEEE.

[86] Lucic, M., Kurach, K., Michalski, M., Gelly, S., and Bousquet, O. (2018). Are gans created equal? a large-scale study. *Advances in neural information processing systems*, 31:700–709.

[87] Marra, G., Giannini, F., Diligenti, M., and Gori, M. (2019). Constraint-based visual generation. In *International Conference on Artificial Neural Networks*, pages 565–577. Springer.

[88] Mehrabian, A. and Russell, J. A. (1974). *An approach to environmental psychology.* the MIT Press.

[89] Meisheri, H. and Dey, L. (2018). Tcs research at semeval-2018 task 1: Learning robust representations using multi-attention architecture. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 291–299.

[90] Melacci, S., Maggini, M., and Gori, M. (2009). Semi–supervised learning with constraints for multi–view object recognition. In *International Conference on Artificial Neural Networks*, pages 653–662. Springer.

[91] Minervini, P., Demeester, T., Rocktäschel, T., and Riedel, S. (2017). Adversarial sets for regularising neural link predictors. In *Uncertainty in Artificial Intelligence-Proceedings of the 33rd Conference, UAI 2017*. Curran Associates Inc.

[92] Mirsamadi, S., Barsoum, E., and Zhang, C. (2017). Automatic speech emotion recognition using recurrent neural networks with local attention. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2227–2231. IEEE.

[93] Mirza, M. and Osindero, S. (2014). Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

[94] Mohammad, S. (2012). Portable features for classifying emotional text. In *Proceedings of the Conference of the NAACL HLT*, pages 587–591. ACL.

[95] Mohammad, S., Bravo-Marquez, F., Salameh, M., and Kiritchenko, S. (2018). Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, pages 1–17.

[96] Mollahosseini, A., Chan, D., and Mahoor, M. H. (2016). Going deeper in facial expression recognition using deep neural networks. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–10. IEEE.

[97] Mollahosseini, A., Hasani, B., and Mahoor, M. H. (2017). Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10(1):18–31.

[98] Muda, L., Begam, M., and Elamvazuthi, I. (2010). Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques. *arXiv preprint arXiv:1003.4083*.

[99] Neumann, M. and Vu, N. T. (2017). Attentive convolutional neural network based speech emotion recognition: A study on the impact of input features, signal length, and acted speech. *Proc. Interspeech 2017*, pages 1263–1267.

[100] Noroozi, F., Marjanovic, M., Njegus, A., Escalera, S., and Anbarjafari, G. (2017). Audio-visual emotion recognition in video clips. *IEEE Transactions on Affective Computing*, 10(1):60–75.

[101] Novák, V., Perfilieva, I., and Mockor, J. (2012). *Mathematical principles of fuzzy logic*, volume 517. Springer Science & Business Media.

[102] Nwe, T. L., Foo, S. W., and De Silva, L. C. (2003). Speech emotion recognition using hidden markov models. *Speech communication*, 41(4):603–623.

[103] Ojala, T., Pietikainen, M., and Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987.

[104] Olszanowski, M., Pochwatko, G., Kuklinski, K., Scibor-Rylski, M., Lewinski, P., and Ohme, R. K. (2015). Warsaw set of emotional facial expression pictures: a validation study of facial display photographs. *Frontiers in psychology*, 5:1516.

[105] Otberdout, N., Daoudi, M., Kacem, A., Ballihi, L., and Berretti, S. (2020). Dynamic facial expression generation on hilbert hypersphere with conditional wasserstein generative adversarial nets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[106] Pal, P., Iyer, A. N., and Yantorno, R. E. (2006). Emotion detection from infant facial expressions and cries. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 2, pages II–II. IEEE.

[107] Piana, S., Stagliano, A., Camurri, A., and Odone, F. (2013). A set of full-body movement features for emotion recognition to help children affected by autism spectrum condition. In *IDGEI International Workshop*.

[108] Picard, R. (1997). Affective computing. cambridge, massachustes institure of technology.

[109] Plutchik, R. (1980). A general psychoevolutionary theory of emotion. In *Theories of emotion*, pages 3–33. Elsevier.

[110] Pool, C. and Nissim, M. (2016). Distant supervision for emotion detection using facebook reactions. In *Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 30–39.

[111] Poria, S., Cambria, E., Bajpai, R., and Hussain, A. (2017). A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, 37:98–125.

[112] Qadir, A. and Riloff, E. (2014). Learning emotion indicators from tweets: Hashtags, hashtag patterns, and phrases. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1203–1209.

[113] Raad, B. T., Philipp, B., Patrick, H., and Christoph, M. (2018). Aseds: Towards automatic social emotion detection system using facebook reactions. In *International Conference on High Performance Computing and Communications; on Smart City; on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 860–866. IEEE.

[114] Ramet, G., Garner, P. N., Baeriswyl, M., and Lazaridis, A. (2018). Context-aware attention mechanism for speech emotion recognition. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 126–131. IEEE.

[115] Ringeval, F., Sonderegger, A., Sauer, J., and Lalanne, D. (2013). Introducing the recola multimodal corpus of remote collaborative and affective interactions. In *2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG)*, pages 1–8. IEEE.

[116] Rouast, P. V., Adam, M., and Chiong, R. (2019). Deep learning for human affect recognition: insights and new developments. *IEEE Transactions on Affective Computing*.

[117] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

[118] Russell, J. A. (1980). A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161.

[119] Russell, J. A. and Mehrabian, A. (1977). Evidence for a three-factor theory of emotions. *Journal of research in Personality*, 11(3):273–294.

[120] Saccà, C., Graziani, L., Parlangeli, O., and Masini, M. (2020). Opera: an italian dataset for speech emotion recognition. Submitted to: International Journal of Speech Technology.

[121] Sariyanidi, E., Gunes, H., and Cavallaro, A. (2014). Automatic analysis of facial affect: A survey of registration, representation, and recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(6):1113–1133.

[122] Scherer, K. R. and Wallbott, H. G. (1994). Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66(2):310.

[123] Schuller, B., Rigoll, G., and Lang, M. (2003). Hidden markov model-based speech emotion recognition. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, volume 2, pages II–1. IEEE.

[124] Schuller, B., Villar, R. J., Rigoll, G., and Lang, M. (2005). Meta-classifiers in acoustic and linguistic feature fusion-based affect recognition. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 1, pages I–325. IEEE.

[125] Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

[126] Serafini, L., Donadello, I., and Garcez, A. d. (2017). Learning and reasoning in logic tensor networks: theory and application to semantic image interpretation. In *Proceedings of the Symposium on Applied Computing*, pages 125–130.

[127] Silva, H., Lourenço, A., and Fred, A. (2012). In-vehicle driver recognition based on hand ecg signals. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, pages 25–28.

[128] Strapparava, C. and Mihalcea, R. (2007). Semeval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 70–74.

[129] Strapparava, C. and Mihalcea, R. (2008). Learning to identify emotions in text. In *Proceedings of the ACM symposium on Applied computing*, pages 1556–1560. ACM.

[130] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

[131] Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

[132] Swain, M., Routray, A., and Kabisatpathy, P. (2018). Databases, features and classifiers for speech emotion recognition: a review. *International Journal of Speech Technology*, 21(1):93–120.

[133] Tarantino, L., Garner, P. N., and Lazaridis, A. (2019). Self-attention for speech emotion recognition. In *INTERSPEECH*, pages 2578–2582.

[134] Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., and Zafeiriou, S. (2016). Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5200–5204. IEEE.

[135] Truong, K. P. and Leeuwen, D. A. v. (2005). Automatic detection of laughter. In *Ninth European Conference on Speech Communication and Technology*.

[136] Tzirakis, P., Trigeorgis, G., Nicolaou, M. A., Schuller, B. W., and Zafeiriou, S. (2017). End-to-end multimodal emotion recognition using deep neural networks. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1301–1309.

[137] Valstar, M. and Pantic, M. (2010). Induced disgust, happiness and surprise: an addition to the mmi facial expression database. In *Proc. 3rd Intern. Workshop on EMOTION (satellite of LREC): Corpora for Research on Emotion and Affect*, page 65. Paris, France.

[138] Valstar, M. F., Pantic, M., Ambadar, Z., and Cohn, J. F. (2006). Spontaneous vs. posed facial behavior: automatic analysis of brow actions. In *Proceedings of the 8th international conference on Multimodal interfaces*, pages 162–170.

[139] Van Der Schalk, J., Hawk, S. T., Fischer, A. H., and Doosje, B. (2011). Moving faces, looking places: validation of the amsterdam dynamic facial expression set (adfes). *Emotion*, 11(4):907.

[140] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

[141] Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE.

[142] Wang, K., Peng, X., Yang, J., Meng, D., and Qiao, Y. (2020). Region attention networks for pose and occlusion robust facial expression recognition. *IEEE Transactions on Image Processing*, 29:4057–4069.

[143] Wang, Y. and Pal, A. (2015). Detecting emotions in social media: A constrained optimization approach. In *IJCAI*, pages 996–1002.

[144] Williams, R. M. and Gilbert, J. E. (2020). Perseverations of the academy: A survey of wearable technologies applied to autism intervention. *International Journal of Human-Computer Studies*, page 102485.

[145] Wu, S., Falk, T. H., and Chan, W.-Y. (2009). Automatic recognition of speech emotion using long-term spectro-temporal features. In *2009 16th international conference on digital signal processing*, pages 1–6. IEEE.

[146] Yadegaridehkordi, E., Noor, N. F. B. M., Ayub, M. N. B., Affal, H. B., and Hussin, N. B. (2019). Affective computing in education: A systematic review and future research. *Computers & Education*, 142:103649.

[147] Yeasin, M., Bullot, B., and Sharma, R. (2006). Recognition of facial expressions and measurement of levels of interest from video. *IEEE Transactions on Multimedia*, 8(3):500–508.

[148] Zadeh, L. A. (1975). Fuzzy logic and approximate reasoning. *Synthese*, 30(3-4):407–428.

[149] Zadeh, L. A. (1988). Fuzzy logic. *Computer*, 21(4):83–93.

[150] Zhang, J., Yin, Z., Chen, P., and Nichele, S. (2020). Emotion recognition using multi-modal data and machine learning techniques: A tutorial and review. *Information Fusion*, 59:103–126.

[151] Zhang, K., Huang, Y., Du, Y., and Wang, L. (2017). Facial expression recognition based on deep evolutional spatial-temporal networks. *IEEE Transactions on Image Processing*, 26(9):4193–4203.

[152] Zhang, Z., Luo, P., Loy, C. C., and Tang, X. (2018). From facial expression recognition to interpersonal relation prediction. *International Journal of Computer Vision*, 126(5):550–569.

[153] Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232.

[154] Zucco, C., Calabrese, B., and Cannataro, M. (2017). Sentiment analysis and affective computing for depression monitoring. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1988–1995. IEEE.