Social signs processing in a cognitive architecture for an humanoid robot.

Agnese Augello[1], Emanuele Cipolla[1], Ignazio Infantino[1], Adriano Manfré[1], Giovanni Pilato[1], and Filippo Vella[1]

Institute for High Performance Computing and Networking, National Research Council of Italy (ICAR-CNR), Palermo, Italy
name.surname@icar.cnr.it

Abstract
A social robot has to recognize human social intention in order to fully interact with him/her. People intention can be inferred by processing verbal and non-verbal communicative signs. In this work we describe an actions classification module embedded into a robot's cognitive architecture, contributing to the interpretation of users behavior.

Keywords: Cognitive Architecture, Recurrent Neural Networks, Actions Classification, Pepper Robot

1 Introduction

Investing in social robotics and proposing cognitive models that provide robots with a form of social intelligence is becoming more and more important nowadays. Thanks to improvement of their expressive and communication skills, robots are assuming an increasingly integrated role in society[1]. There are several discussions about the abilities that robots must exhibit to be considered socially adaptive [2]. They must be able to engage a social interaction with people, understand their intentions by recognizing social signs, express emotions and adapt their behavior to the different social situations.

A proper cognitive architecture is needed to model these features. Some proposals have been made, such as in [3], [4] where a key role in the robot's decisional processes is given to its needs and motivations, or in [5], where the robot's perceptions and mental states trigger its emotions. Coping with the issue of modeling a form of social intelligence in a robot means properly model its identity (with its baggage of knowledge and its point of view about users and environment) and its knowledge about socio-cultural practices and provide him with the ability of understanding social situations and, as consequence, planning and carrying on the interaction with human beings [6]. The success of a social interaction strongly depends on the ability to recognize and properly understand the intention of people [7]. People intention can be inferred by processing the multi-modal signals conveying information about social actions, emotions, attitudes and relationships [8] [9]. In particular, non verbal behavior, such as face expressions,

postures and gestures, are particularly meaningful in social interaction and they reveal the attitude of people towards others and their affective states [9]. Some example of gestures recognition modules has been proposed in [10] [11]. In this work we focus on the abilities of a robot to interpret the non-verbal signs communicated by the users interacting him. We propose an action classification module that allows a robot to distinguish normal actions from aggressive actions. In particular, the robot percepts human postures by an RGB-Depth camera, and stores data as 3D skeletons. Then, a recurrent neural network classifies the detected actions. Section 2 describes how the module, together with other signs processing modules, contributes to the interpretation of the user behavior and influences the robot motivation, according to a Psi-inspired [12] cognitive architecture. Section 3 describes in detail the process of the classification of the actions performed in front of the robot while Section 4 show some experimental results. Last section summarizes the work and describes how we are working on the overall architecture to the interaction with the user, according to the interpretation of the human behavior.

2 Cognitive Architecture

The cognitive architecture (see Figure 1) is inspired to the PSI architecture [12]. The motivation and behavior of the robot depends on different demands, such as Physiological needs influenced by the robot's physical conditions, and cognitive demands, named *Competence* and *Certainty*, considering respectively the capability to perform a task and the confidence to successfully execute a task, influenced by internal and external evaluations. In the specific case of a social robot, a key role is given by the *Affiliation* demand, that is the need for the acceptance by other agents and it is influenced by the behavior of the users interacting with the robot. An appropriate behavior of the interlocutor, characterized by non aggressive actions and a kind verbal interaction is very important to motivate the robotic receptionist in giving support to the visitor. The architecture includes proper modules for internal and external sensing. In particular modules for the recognition of faces, postures, and emotions conveyed in the verbal interaction can be exploited to interpret the user's behavior. An instructor organizes the knowledge of the robot in its Long Term-Memory (LTM) such as the rules needed to manage different situations, an AIML-KB to manage the verbal interaction AIML-KB by means of a chatbot module [1], and collections of postures and verbal expressions that can be used according to the current situation.

3 Classification of the social signs conveyed by actions

As discussed in the introduction, people communicate and reveal their attitude and affective states through both verbal and non verbal social signs. In this work we are focusing the attention on social signs conveyed by non verbal behavior, and in particular by postures and gestures. To recognize these social signs we propose the use of a deep neural network used for the classification of the user actions.

The great advantage of the deep networks is that the first layers of the network, if suitably trained, can automatically extract features allowing a robust representation of the input pattern. In this way, instead of using hand crafted features, the raw data can be processed creating a good classifier with features learned from data [13]. Beyond the possibility of extracting spatial features for the recognition of patterns in bidimensional inputs, deep networks can also be used for the processing and classification of sequence of data. In these cases, the network,

[1] `https://www.alicebot.org`

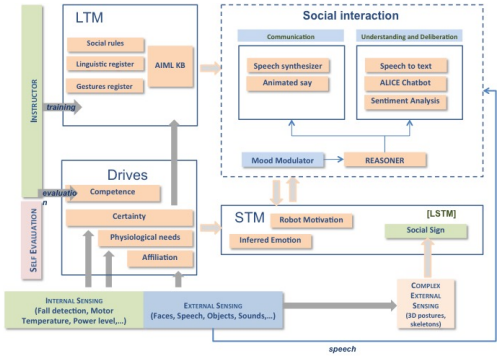


Figure 1: The cognitive architecture

through recurrent connections can maintain memory of the past history and compute the input accordingly. An example of these kind of networks are the Long Short Term Memory networks that are used in the proposed network architecture 2.

LSTMs have been designed by Hochreiter and Schmidhuber [14] with the aim of avoiding the long-term dependency problem, at the price of a more complex cell structure. The key feature of LSTMs is the "cell state" that is propagated from a cell to another. State modifications are regulated by three structures called gates, composed out of a sigmoid neural net layer and a pointwise multiplication operation. Let $C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$ the cell state at time t; the first gate, called "forget gate layer", considers both the input x_t and the output from the previous step h_{t-1}, and returns values between 0 and 1, describing how much of each component of the old cell state C_{t-1}, where should be left unaltered: if the output is 0, no modification is made; if the output is one, the component is completely replaced. New information to be stored in the state is processed afterwards. The second sigmoid layer, called the input gate layer decides which values will be updated. Next, a *tanh* layer creates a vector of new candidate values, $\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_c)$, that could be added to the state. To perform a state update, C_{t-1} is first multiplied by the output of the forget gate $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$, and the result is added to the pointwise multiplication of the input gate output $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$ and $\tilde{C}_t$. Finally, the output $h_t = o_t * \tanh(C_t)$ can be generated, where $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$.

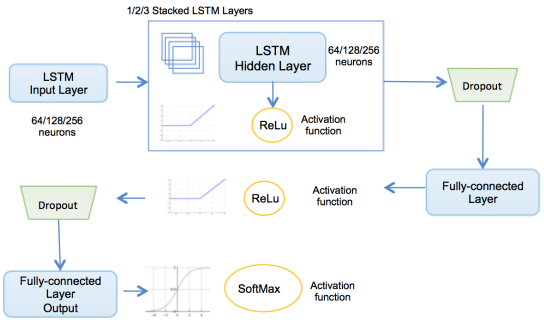


Figure 2: Proposed network architecture

First, a sigmoid is applied, taking into account both h_{t-1} and x_t; its output is then multiplied by a constrained version of C_t, so that we only output the parts we decided to. In [15] it is given a detailed theoretical explanation of the reasons behind the advantages of using a network made of multiple layers. In our scenario we have chosen to gradually stack LSTM layers and measure the trend of the F1-score to determine what the correct number of layers can be. Each LSTM layer is separated from the next one by a ReLU function. In addition, given a sequence length, we strived to determine how many neurons are needed for the representation to be of good quality. To speed up the information acquisition task, to train the network, has been used a dataset formed by a set of actions divided in two macro classes dealing with aggressive or non aggressive behavior [16]. The dataset has been created by monitoring, by using proper sensors, placed in several body parts, the free movements of ten people in front of a camera or in front of a standing bag. The actions of the dataset, with twenty different labels, are divided between actions for the *normal* behaviour and actions for *aggressive* behaviour.

4 Experimental Results

The action classification module will be employed by Pepper, a humanoid robot designed by Aldebaran Robotics and SoftBank [2]. The robot is equipped with an Asus Xtion RGB-D camera able to capture the skeleton data of the human and obtain in real-time the space position of each joint along the three axis (x,y,z). For the experiments we have used a subset of the Vicon Physical Action dataset first used in [16] and made available through [17]. 10 subjects (7 men and 3 women) have been recorded while performing 20 actions, each accounting for 10 normal and 10 aggressive activities. For our setup, a subset of the actions, more inherent to the considered context, has been selected, composed of three normal actions (Bowing, Clapping and Handshaking) and three aggressive actions (Punching, Slapping and Frontkicking). The training has been performed on nine subjects, while testing is done on the tenth, last subject. We have tested with the following variations: the number of neurons in the LSTM layers have been set to the values 64, 128 or 256; one, two or three stacked LSTM levels have been considered and the sliding window have been set to a value from 2 to 20. The training has been performed for 10 epochs. After 10 epochs, the accuracy has already approached 1, so we choose to stop. The F_1 score function has been chosen as a performance index as it balances *precision* and *recall*, where the *Precision* is the fraction of classified actions that have a given label, while *Recall* is the fraction of the instances having a given label that are successfully classified. Some form of averaging is needed to use F1 in a multi-class context: we chose not to take into account the slight imbalance in labels found for test subject, so we did a so-called *macro-averaging*. Throughout the rest of the section we will call this metric *F1 macro*.

F1 macro has been computed for every combination of sequence length, LSTM layers and neurons per layer combination previously detailed. 1 contains the maximum values of F1 macro found.

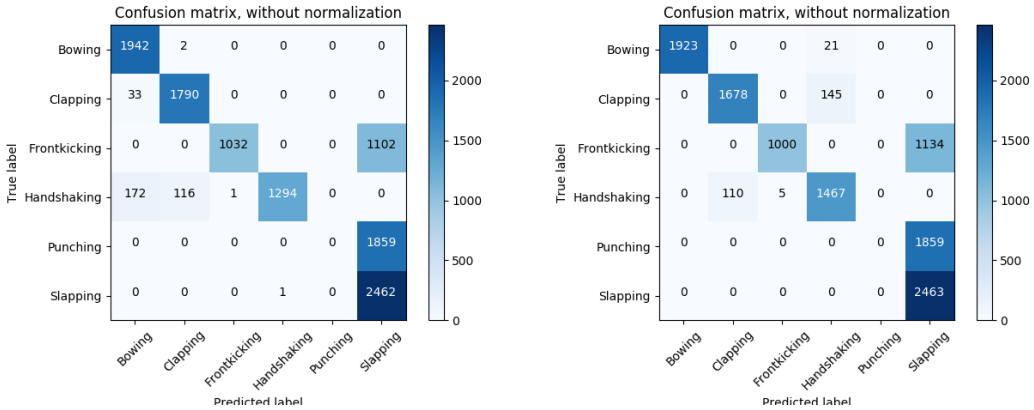
Some conclusions may be drawn from table 1. The best score is achieved with 10-frame long sequence, using 2 LSTM layers with 64 neurons each; as a comparison, the F1 score for the same sequence length with a single LSTM layer having 128 neurons is much lower (0.29). This means that simpler, stacked layer configurations are to be preferred.

It seems that a compact representation more aptly represents the features of the test set, so let us consider the 64-neuron configurations. In particular, a network with single hidden layer has a very similar performance to our best candidate.

[2] `https://www.ald.softbankrobotics.com/en/cool-robots/pepper`

Table 1: Maximum F1 macro score values and sequence size per each network; the best result is in bold

LSTM layers	Neurons per layer	Sequence size	F1 macro score
1	64	9	0.680
1	128	14	0.496
1	256	7	0.504
2	**64**	**10**	**0.682**
2	128	9	0.500
3	64	8	0.478
3	128	7	0.528



(a) Single LSTM, 9 frame sequence

(b) Double LSTM, 10 frame sequence

Figure 3: Comparing confusion matrices for 64-neuron networks

It may thus be useful to consider the confusion matrices for these two configurations, as shown in figure 3, to introduce a criterion based on real-world uses of this classifier within the context of social interactions. The results for the aggressive action Punching are always incorrectly classified as Slapping; the double-LSTM network has a marginally better score, in a way: it does not ever mistake Slapping for Handshaking, and thus is more able to identify a dangerous situation. As for Front kicking, in both cases is correctly detected only about 50% of the times, if it is not mistaken for Slapping. Handshaking, one of the normal actions a well-behaved user would do if the robot were human, is almost always correctly classified, and the double-LSTM manages to do it much better. The single-layered network, instead, seems more fit to detect some more complex positive actions, such as Bowing and Clapping. These interesting information may be used in a future work to arrange for *transfer learning*, as to benefit from the best capabilities learned from the first network in the second one, and have an overall better behaving classifier.

5 Conclusion and Future Works

The proposed work discussed an actions recognition module that can be exploited by a a robotic concierge to interpret the social signs communicated by the users. The results are promising

and actions can be classified to have an important hint for the fore-coming interaction. The ongoing work regards: 1) the implementation of the other modules for social signs processing 2) the evaluation of the Affiliation according to the interpreted user behavior and the influence of all demands to establish the motivation of the robot 3) the definition of the KB and the reasoning rules of the robot according to a proper model of social interaction.

References

[1] Adriana Tapus, Maja J Mataric, and Brian Scassellati. Socially assistive robotics. *IEEE Robotics and Automation Magazine*, 14(1):35, 2007.

[2] Kerstin Dautenhahn. Socially intelligent robots: dimensions of human–robot interaction. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1480):679–704, 2007.

[3] María Malfaz, Álvaro Castro-González, Ramón Barber, and Miguel A Salichs. A biologically inspired architecture for an autonomous and social robot. *IEEE Transactions on Autonomous Mental Development*, 3(3):232–246, 2011.

[4] Seng-Beng Ho. Cognitive architecture for adaptive social robotics. In *International Conference on Intelligent Robotics and Applications*, pages 549–562. Springer, 2016.

[5] Carole Adam, Wafa Johal, Damien Pellier, Humbert Fiorino, and Sylvie Pesty. Social human-robot interaction: A new cognitive and affective interaction-oriented architecture. In *International Conference on Social Robotics*, pages 253–263. Springer, 2016.

[6] Agnese Augello, Manuel Gentile, and Frank Dignum. Social agents for learning in virtual environments. In *Games and Learning Alliance*, pages 133–143. Springer, 2016.

[7] Sebastian Loth and Jan P De Ruiter. Editorial: Understanding social signals: How do we recognize the intentions of others? *Frontiers in psychology*, 7, 2016.

[8] Isabella Poggi and D'Errico Francesca. Cognitive modelling of human social signals. In *Proceedings of the 2nd international workshop on Social signal processing*, pages 21–26. ACM, 2010.

[9] Alessandro Vinciarelli, Maja Pantic, and Hervé Bourlard. Social signal processing: Survey of an emerging domain. *Image and vision computing*, 27(12):1743–1759, 2009.

[10] Pablo Barros, German I Parisi, Doreen Jirak, and Stefan Wermter. Real-time gesture recognition using a humanoid robot with a deep neural architecture. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*, pages 646–651. IEEE, 2014.

[11] Gerard Canal, Sergio Escalera, and Cecilio Angulo. A real-time human-robot interaction system based on gestures for assistive scenarios. *Computer Vision and Image Understanding*, 149:65–77, 2016.

[12] Zhenhua Cai, Ben Goertzel, and Nil Geisweiller. Openpsi: Realizing dörner's psi cognitive model in the opencog integrative agi architecture. In *Artificial General Intelligence*, pages 212–221. Springer, 2011.

[13] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

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

[15] Razvan Pascanu, Guido Montufar, and Yoshua Bengio. On the number of response regions of deep feed forward networks with piece-wise linear activations. *arXiv preprint arXiv:1312.6098*, 2013.

[16] Theodoros Theodoridis and Huosheng Hu. Action classification of 3d human models using dynamic anns for mobile robot surveillance. In *Robotics and Biomimetics, 2007. ROBIO 2007. IEEE International Conference on*, pages 371–376. IEEE, 2007.

[17] M. Lichman. UCI machine learning repository, 2013.