Cognitive Architecture for a Companion Robot: Speech Comprehension and Real-World Awareness

Artemiy Kotov[1, 2] [0000-0003-3353-5549], Nikita Arinkin[1, 2] [0000-0003-2303-2817], Alexander Filatov[2], Kirill Kivva[2], Liudmila Zaidelman[1, 2] [0000-0002-2941-144X] and Anna Zinina [1, 2] [0000-0001-9575-1875]

[1] National Research Center “Kurchatov Institute”, Moscow, Russia
[2] Russian State University for the Humanities, Moscow, Russia
kotov@harpia.ru

Abstract. Companion robots should support natural communication with humans referencing the events in the environment and should respond with compound communicative reactions, reacting to incoming utterances, actions, gazes and other surrounding events. We represent a project of F-2 companion robot, where we implement unified representation types for (a) incoming speech semantics and (b) real events in the environment, registered by computer vision system. We rely of a classic linguistic notion of “case-frame” and represent each event as a predicate and a number of valencies: agent, patient, instrument, etc. Predicate and each valency refer to a referent (“id” of event or object) and is also represented by a list of semantic markers. These unified representations allow the robot to react to speech semantics and real events in a unified way. To simulate communicative reactions, we use a number of scripts (productions) which are activated by incoming case-frames, generate behavioral cues and get disactivated, once the cue is executed on the robot. The balance of activations allows the robot to select the most significant reactions or to simulate emotional behavior, expressively reacting to some minor stimulus. This architecture allows to eliminate the bottleneck on the stage of robot reactions, effectively process diverse incoming stimuli and simulate rich and compound communicative behavior on the robot.

Keywords: Emotional Computer Agents, Cognitive Models, Semantic Representation.

1 Introduction

Companion robots should process incoming natural speech, recognize surrounding objects, and talk about their internal states, memory events and real-world situations, using speech and nonverbal cues. Robots should also remember the history of interaction and accumulate knowledge from different text sources: personal communication of incoming texts. The central point in the design of cognitive architectures for robots is the representation of semantic information (both from speech and visual sources) and its processing architecture, offering rich and believable behavior for the robot.

Following classic approach by M. Minsky, an artificial cognitive architecture should consist of modules, competing or cooperating in various cognitive tasks [1]. Minsky has suggested, that basic processing should be performed by *proto-specialists*, each of those is a simple “agent” or procedure, designed to react to some dangerous or lucrative stimulus (external or internal), and thus, modelling a primary emotion (*fear, aggression*), reflex (*withdrawal*) or drive (*hunger*). The resolution of conflicts between the proto-specialists, active within the current moment of operation, constitutes current behavior of the agent – possibly, a mixture of expressive cues, suggested by diverse proto-specialists. We rely on further elaboration of this architecture, suggested within *Cognition and Affect Project – CogAff* model [2]. CogAff includes three levels of processing: (a) basic *reactive* level, withholding primary emotions and drives, (b) middle *deliberative* level, responsible for rational inferences and (c) upper *meta-management* level, engaged in reflexive processing. CogAff via SimAgent Toolkit has passed numerous experiments and is implemented in various artificial emotional agents. Although CogAff offers an extended view on the architecture of human cognitive processing, its internal cognitive representations are quite simple, as designed to represent simple events of virtual environments within SimAgent Toolkit. In our study we suggest an extended architecture, which implements the principle of script competition, typical for *proto-specialists* and *CogAff* architectures, but handles real speech semantics and representations of real objects and events, as recognized by computer vision modules.

2 F-2 architecture

We represent a design of a cognitive architecture for a companion robot F-2. The architecture combines (a) an advanced speech processing system with morphological, syntactic and semantic levels, (b) visual recognition system, (e) central processor, which is operating with *scripts* or productions, and (d) robot controller, which manages speech synthesis and executes behavioral patterns on the real F-2 robot (Figure 1).

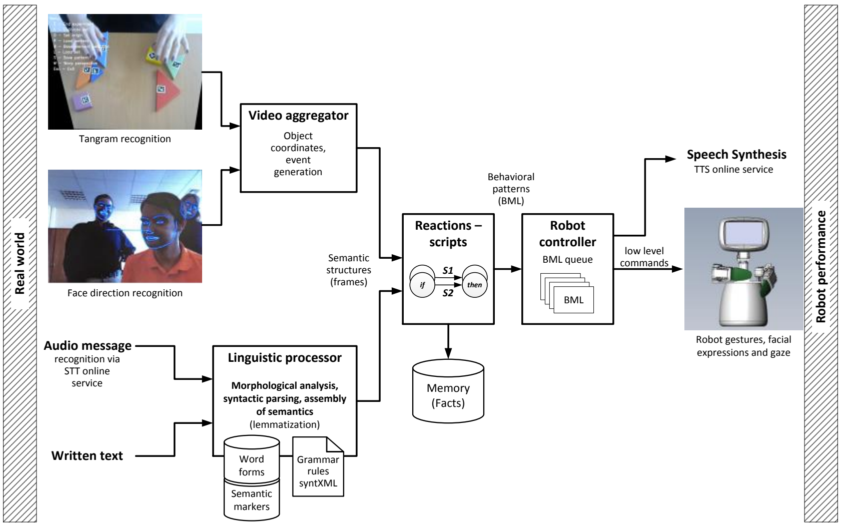


Fig. 1. General architecture of the F-2 companion robot.

The system is designed as a composition of modules (pipeline), which may run on the server without robot and computer vision (CV) subsystem and process daily news, blogs or novels to accumulate the extracted case-frames (facts) to a memory base. The system may also run on the robot to process (a) incoming written speech or oral speech after the external speech-to-text recognition and (b) the events from CV-subsystem. The processing is executed as the activation of *scripts* in the central processor module by the incoming representations. Scripts are subdivided into groups, sensitive to (a) emotional semantic representations (*d-scripts*, n = 79) and (b) rational semantic representations (*r-scripts*, n = 1600). In this sense, the distribution of scripts implements the two basic levels of CogAff architecture, while meta-management of reflexive processes are not covered. In case the system runs on the robot, the activated scripts send the corresponding behavioral patterns to the queue of the robot controller. These patterns can be executed depending on the activation of scripts and the availability of the robot's actuators. To describe the format of semantic representations within F-2 architecture we further examine the functions of speech processor and CV-subsystem as well as their integration within the component of scripts.

3 Speech processor

Speech processor is designed for the Russian language, but after the replacement of grammar and dictionary it may work with other languages, as the semantic representations will remain mostly intact. It receives written text or transcribed oral speech after a speech-to-text recognition service. The processor is based on traditional linguistic layered architecture with *morphological*, *syntactic*, and *semantic* processing levels. Wordforms are tagged with the help of a dictionary with 100,000 lemmata (1.5 mln wordforms), stored as a database. An automatic tagger (guesser) is used for unknown forms. Syntactic analysis relies on formal representation of the Russian grammar in SyntXML format [3]. The grammar contains over 600 syntactic rules, which define possible binding of two or more wordforms, or even virtual language segments. As for the output the syntactic component constructs a syntactic tree, as on Fig. 2, or a set of such trees with limited cardinality in case of lexical or syntactic ambiguity. Within a syntactic tree each noun (or pronoun) is assigned to a certain valency within the case-frame of the verb – as seen in SemVal variable in Fig. 2. The list of valencies is based on [4] and includes *agent* (**ag**), *patient* (**p**), *instrument* (**instr**) and other valencies (n = 22). A special *predicate* (**p**) meta-valency is assigned to verbs and predicatives; we assume, that this valency governs a predication, and with this assumption the semantics of a predication can be represented as a table (see Table 1). The semantic representation of each valency is a union of semantic markers for all the words within this valency (e. g. an *adjective* is joined with *noun*) retrieved from the semantic dictionary. Homonymy is marked by sub-division indexes ("1 1", "1 2" etc.).

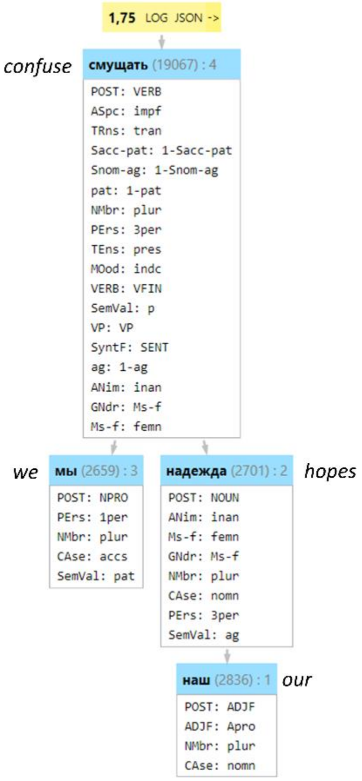


Fig. 2. Syntactic representation of a sentence *Our hopes confuse us*.

Table 1. Semantic representations of a sentence *Our hopes confuse us.*

P (*predicate*)	**Ag** (*agens*)	**Pat** (*patient*)
1 1 present tense	1 1 many	1 1 somebody
1 1 assertive	1 1 abstract-goal	1 1 egocentric – me
1 1 to cause emotions	1 1 abstract	1 1 other person
1 1 to cause negative emotions	1 1 @wish_NOUN	1 1 physical object
1 1 @to_ surprise	1 1 @343_NOUN	1 1 principal – speaker
1 1 @5_VERB	1 1 own	1 1 set of people

The semantic representation in this format is constructed for every clause for each of the homonymic syntactic trees. All the constructed representations arrive at the input of script component, which calculates distances between the semantic representations and the premises of scripts. The tree with minimal average distance to scenarios is chosen for processing. A similar operation is executed for visual representations, that resemble text semantics but do not have homonymic variants.

4 Computer vision processor

Visual recognition subsystem should generate conceptual representations for the events, that are important enough to invoke the reactions of the robot. It also has to aggregate spatial information of the recognized events and objects, thus constructing the 3D-model of the robot's surrounding. The system consists of a set of CV modules and a *video aggregator*, which accumulates data from the modules and creates conceptual representations (events) for possible reactions of the robot. We assume, that personal interaction and communication regarding the problem space are the two most important areas for the CV-awareness. Thus, we use (a) face detection and face tracking modules to detect the aspects of personal communication and (b) tangram puzzle assembly submodule to detect operations in the problem space. The number of modules can be easily extended. The data on the recognized objects and events is transferred by each module into *video aggregator*, which constructs 3D-model of the environment and generates events for further processing by scripts (Figure 3). These stimuli events of the visual recognition subsystem are represented as case-frames, similar to semantic representations from the speech processor.

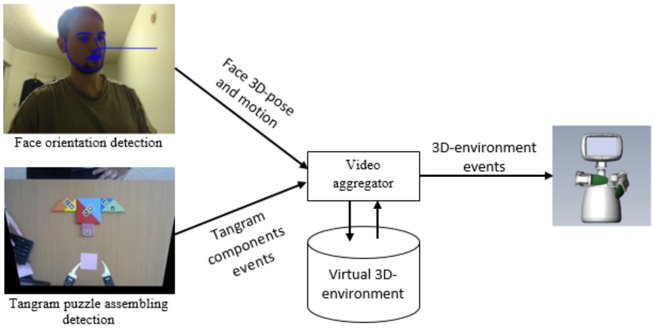

Face orientation detection
Face 3D-pose and motion
Video aggregator
3D-environment events
Tangram puzzle assembling detection
Tangram components events
Virtual 3D-environment

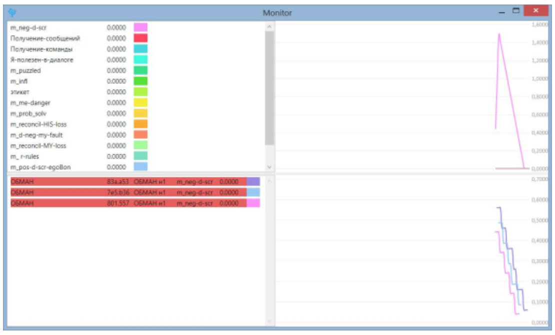
Fig. 3. Visual recognition subsystem principal scheme.

For face detection we apply pre-trained linear *SVM*-classifier in a sliding window within an image multiscale pyramid. This classifier uses advanced version of *HoG*-features [5] implemented in *Dlib* library [6]. We have decided to recognize user's face orientation as one of the key stimuli for user-robot interaction. In our experiments we assume the orientation of user's face as a rough evaluation of user's gaze or attention. The face orientation detection consists of two steps: (a) facial landmarks detection, (b) determination of relative 3D-position of the detected landmarks in the camera coordinate system. To detect facial landmarks, we apply an Ensemble of Regression Trees approach [7]. Then we associate these landmarks with 3D-points in an approximate face model and solve a Perspective-n-Point (PnP) problem to determine a 3D-orientation of the user's face in the camera coordinate system [8]. Video aggregator receives and updates the information on the location and orientation of the human face. It generates the events in the format as in Table 1, like, 'person (**ag**) looks at (**p**) you (**pat**)', to invoke a response from the robot, e. g. a response gaze. Also, it reports the current coordinates of the human face, so that the robot can update the angles of head and eyes, while trying to ‘look at the person’.

To model the interaction with a user while solving a problem in the real world, we have chosen the Tangram puzzle, where a user has to construct a figure with a given shape with a help of 7 game elements. This game represents a good example of a task, where a human and a robot construct something together. The robot registers the position of game elements and user moves, gives emotional feedback and suggests the required actions. We have developed a recognition and game support library, which records movements and evaluates, whether a move is a step forward to any of the possible puzzle solutions (each game has multiple solutions). The data from this library is also accumulated by the aggregator which further generates events in a form, similar to Table 1, like ‘Game element 5 (**pat**) has moved (**p**) to the correct position (**targ**) for the solution No 3 (**ben**)’. This allows the script component to react accordingly.

5 Competitive processing

Upon the receipt of each input event in a form of a case-frame the script component calculates its distance to the premises of scripts. The preferring scripts are chosen (a) proportionally to the similarity between input event and the premise of the script, and (b) proportionally to the activation of microstate (current emotional state) for the corresponding script (only for the emotional *d-scripts*). In other words, each input is evaluated as relevant, if it corresponds to some known situation frame (for *r-scripts*) or to the emotive representation (*d-script*) of a prevailing emotion (microstate). The activated script transfers its behavioral pattern on Behavior Markup Language (BML) [9] to *robot controller* for possible execution. The BML is executed if the script activation is high enough, or if the corresponding body parts of the robot are free. This allows to simulate blending emotions, when two opposite emotions are combined in behavior [10], and even irony, when a strong negative reaction is substituted by an ironic positive response from the opposing script [11]. The competition of scripts allows the robot to combine the behavior from several scripts, e. g. ‘head’ and ‘mouth’ can be engaged to execute the direct speech reply, while the underlying anxiety is expressed with ‘hands’ though automanipulation or scratching. Script loses its activation upon the execution of the corresponding BML. Figure 4 shows an example, where three incoming events (phrases) activate three instances of DECEPT (deception) d-script.



m_neg-d-scr	0.0000	🟪
Получение-сообщений	0.0000	🟥
Получение-команды	0.0000	🟦
Я-полезен-в-диалоге	0.0000	🟦
m_puzzled	0.0000	🟩
m_infl	0.0000	🟩
этикет	0.0000	🟩
m_me-danger	0.0000	🟨
m_prob_solv	0.0000	🟨
m_reconcil-HIS-loss	0.0000	🟧
m_d-neg-my-fault	0.0000	🟧
m_reconcil-MY-loss	0.0000	🟩
m_ r-rules	0.0000	🟩
m_pos-d-scr-egoBon	0.0000	🟦

ОБМАН	83a.a53	ОБМАН н1	m_neg-d-scr	0.0000	🟪
ОБМАН	7e5.b36	ОБМАН н1	m_neg-d-scr	0.0000	🟦
ОБМАН	801.557	ОБМАН н1	m_neg-d-scr	0.0000	🟪

Fig. 4. Competitive processing of scripts

The instances compete in time, incrementally losing their activation, when the corresponding BMLs are executed (lower part of the screen), while the general microstate (negative emotional state) is activated by the sum of negative incoming events and loses its activation, following the expression of the scripts (upper part of the screen).

6 Conclusion

While the general architecture of F-2 robot inherits the classic concepts of *proto-specialists* and *CogAff*, it has been greatly extended to handle the semantics of natural texts and real events following CV recognition. In this respect the representation of semantics in the form of linguistic case-frames allows the robot to process semantic and visual information in a unified way. The competition of scripts on the reaction stage makes the architecture flexible, allowing the robot to react to incoming speech and visual events with diverse (and even contradictory) internal states and generate rich and compound communicative behavior, including blending emotional patterns and irony.

Acknowledgments. The research is supported by the grant of the Russian Science Foundation (project №19-18-00547).

References

1. Minsky, M. L.: The Society of Mind. New-York, London: Touchstone Book (1988).
2. Sloman, A., Chrisley, R.: Virtual Machines and Consciousness. J. Conscious. Stud. 2003. vol. 10, № 4–5, pp. 133–172 (2003).
3. Kotov, A., Zinina, A., Filatov, A.: Semantic Parser for Sentiment Analysis and the Emotional Computer Agents. Proceedings of the AINL-ISMW FRUCT 2015, pp. 167–170 (2015).
4. Fillmore, C. J.: The Case for Case, Universals in linguistic theory, New York: Holt, Rinehart & Winston, pp. 1–68 (1968).
5. Felzenszwalb, P. F., Girshick, R. B., McAllester, D., Ramanan, D.: Object Detection with Discriminatively Trained Part-Based Models. In: IEEE Transactions on Pattern Analysis

and Machine Intelligence, vol. 32, № 9, pp. 1627-1645 (2010). doi: 10.1109/TPAMI.2009.167

6. *Dlib C++ Library – Image Processing* (2019). Available at: http://dlib.net/imaging.html (accessed 7 December 2019).
7. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, pp. 1867-1874 (2014). doi: 10.1109/CVPR.2014.241
8. Mallik, S.: Head Pose Estimation Using OpenCV and Dlib (2019). Available at: https://www.learnopencv.com/head-pose-estimation-using-opencv-and-dlib (accessed 7 December 2019).
9. Kopp, S, Krenn, B, Marsella, S, Marshall, A, Pelachaud, C, Pirker, H, Thórisson, K, Vilhjálmsson, H.: Towards a Common Framework for Multimodal Generation: The Behavior Markup Language. In: Intelligent Virtual Agents, pp. 205–217 (2006).
10. Ochs, M., Niewiadomski, R., Pelachaud, C., Sadek, D.: Intelligent Expressions of Emotions. In: Tao, J., Tan, T., Picard, R.W. (Eds.). ACII 2005, LNCS 3784. Springer-Verlag, Berlin, Heidelberg, pp. 707–714 (2005).
11. Kotov, A.: Accounting for irony and emotional oscillation in computer architectures. In: Proceedings of International Conference on Affective Computing and Intelligent Interaction ACII 2009. IEEE, Amsterdam, pp. 506–511 (2009).