


An Interoperable Framework for Computational Models of Emotion

Enrique Osuna, Instituto Tecnológico de Sonora, Mexico

Sergio Castellanos, Instituto Tecnológico de Sonora, Mexico

Jonathan Hernando Rosales, Universidad Autónoma de Guadalajara, Mexico

Luis-Felipe Rodríguez, Instituto Tecnológico de Sonora, Mexico*

 <https://orcid.org/0000-0001-8114-0299>

ABSTRACT

Computational models of emotion (CMEs) are software systems designed to emulate specific aspects of the human emotions process. The underlying components of CMEs interact with cognitive components of cognitive agent architectures to produce realistic behaviors in intelligent agents. However, in contemporary CMEs, the interaction between affective and cognitive components occurs in an ad-hoc manner, which leads to difficulties when new affective or cognitive components should be added in the CME. This paper presents a framework that facilitates taking into account in CMEs the cognitive information generated by cognitive components implemented in cognitive agent architectures. The framework is designed to allow researchers to define how cognitive information biases the internal workings of affective components. This framework is inspired in software interoperability practices to enable communication and interpretation of cognitive information and standardize the cognitive-affective communication process by ensuring semantic communication channels used to modulate affective mechanisms of CMEs.

KEYWORDS

Affective Component, Agent Architecture, Cognitive Component, Computational Modeling, Emotion Modeling, Intelligent Agent, Interoperability, Software Framework

INTRODUCTION

Computational models of emotion (CMEs) are software systems designed to imitate some aspects of the process of human emotions (Sergio Castellanos & Rodríguez, 2018). This type of computational model is usually developed to be included in the cognitive architecture of virtual agents so that this type of intelligent system is capable of exhibiting affective behaviors in specific application domains (Caro et al., 2019; Rath et al., 2021). In general, CMEs are designed and implemented to provide virtual agents with mechanisms for evaluating a stimulus, eliciting synthetic emotions, and generating emotional behaviors (Huang et al., 2017; Rodríguez & Ramos, 2014). It is common practice that the internal mechanisms of CMEs are inspired by

DOI: 10.4018/IJCINI.296257

*Corresponding Author

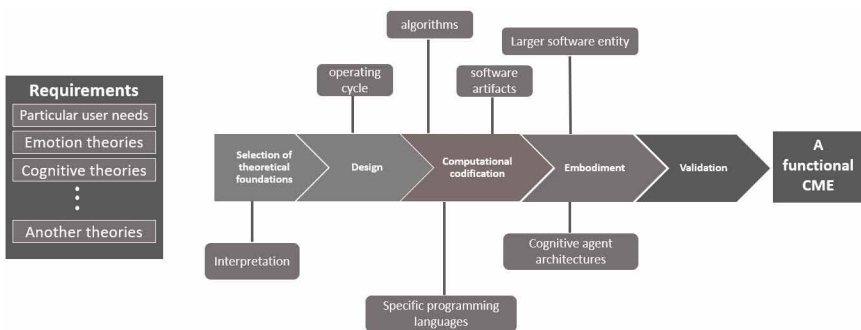
This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

theories about human emotions originated in areas such as psychology and neuroscience. Thus, the development process of CMEs is supported by theoretical and computational aspects. First, emotion theory provides explanations about the workings of human emotions that serve as guidelines underlying the design of the internal mechanisms, processes, phases, architectures, among other elements of CMEs. Second, computational artifacts and practices from areas such as software engineering are utilized to achieve a working computational software of such a human emotion model and ensure a correct technical functioning. The development process of contemporary CMEs reported in the literature follows, in general, the procedure depicted in Figure 1, which reflects an effort of researchers in obtaining the requirements from emotion theories and the generation of a functional model (Rodríguez & Ramos, 2014).

According to emotion theory, the underlying mechanisms of emotion processing are largely influenced by cognitive information that results from cognitive functions such as attention, as well as by psychological constructs (e.g., individual's personality and culture) (Jain & Asawa, 2015; Jha et al., 2013; Rath et al., 2021). Based on this evidence, the components of a CME are required to be designed so that cognitive information from components in cognitive agent architectures are considered. It is assumed that this strategy leads to imitating closely the process of human emotion and ultimately allowing the virtual agent to exhibit very realistic affective behavior (Jha et al., 2013; Xie et al., 2012; Yalcin & Dipaola, 2018). From a software system perspective, affective and cognitive components must therefore interact with each other in order to generate realistic emotions and, in turn, these emotions influence the functioning of cognitive processes such as the agent's decision-making and planning (Gavirangaswamy et al., 2019; Tieck et al., 2019). Nevertheless, this cognitive-affective relationship becomes highly complex since sharing information between cognitive and affective components that presumably may have been developed independently involves an important technical challenge. For instance, enabling the data exchange between affective components in CMEs and cognitive components in cognitive agent architectures is not enough, it is also necessary to resolve semantic issues to enable the accurate interpretation of the data that is being exchanged.

The interaction among systems developed independently is an issue that has been widely addressed in the field of software engineering and is associated with the concept of interoperability (Osuna et al., 2021). In fact, in diverse research areas this concept has been adopted for the design of scientific software (Horsch et al., 2019). In the context of emotion modeling, interoperability involves the design of mechanisms that enable cognitive and affective components to communicate utilizing an agreed model description and thus collaborate in the processing of data in the context of a broader cognitive-affective model, such as in the context of cognitive agent architectures. However, despite the potential benefits of adopting interoperability in the design of CMEs, there are still a series of challenges associated to the following:

Figure 1. Development process of CMEs



- Establishing communication channels for the exchange of data between affective components in CMEs and cognitive components in cognitive architectures (i.e., syntactic interoperability).
- Establishing mechanisms to accurately interpret the data exchanged between these affective and cognitive components (i.e., semantic interoperability).

The first, syntactic interoperability, refers to the technical capacity of the model to establish a communication channel for data exchange with other systems such as cognitive models. This implies providing an adequate data structure and syntax for communication. The second, semantic interoperability, addresses the question of the exact meaning of the information exchanged and that this is understandable for the CME and the cognitive components. In particular, this latter point represents a significant challenge in the field of emotion modeling, since a CME, similar to cognitive models, is largely based on very diverse and numerous theories of emotion that assign unique and specific meanings to the data processed (Aydin & Aydin, 2020).

In this paper, it is proposed a layered framework that incorporates software engineering artifacts for the interaction between cognitive components implemented in cognitive agent architectures and affective components of a CME. In particular, it addresses the problems described above by providing a syntactic interoperability solution that allows establishing a communication channel for the exchange of data between cognitive and affective components, as well as a semantic interoperability solution to assign a specific meaning to the data being exchanged. It is noteworthy that the cognitive components that seek to interact with the proposed framework do not require modifications in their underlying mechanisms, on the contrary, the cognitive components can be extended to include message exchange methods to use an adapter design pattern that allows their communication. The paper is structured as follows. Section 2 presents related work. In section 3 the proposed approach is introduced. Section 4 presents an example detailing the implications of connecting a cognitive component to the framework. Finally, Section 4 and Section 5 present a discussion and some concluding remarks, respectively.

RELATED WORK

This section discusses literature focused on modeling the interaction between emotion and cognition. The analysis focus on the communication channels for the exchange of data with cognitive entities (syntactic interoperability) and the mechanisms designed to provide a meaning to these data (semantic interoperability). The following keywords were used to search for related work: *computational model of emotion, or mood, or feeling, or sentiment, or cognition, and agent*, in databases such as IEEEExplore, ACM Digital library, ScienceDirect, and SpringerLink. Only those articles that propose CMEs to be included in cognitive architectures of agents were considered. Furthermore, only those articles that provided details about the interaction of affective processes of the CMEs with cognitive components were analyzed.

An Integrative Framework (InFra) was proposed by (Rodriguez et al., 2016) to facilitate the construction of CMEs capable of interacting with components of cognitive agent architectures. This proposal presents an input interface that makes use of the fuzzy logic technique to translate the data sent by cognitive components into production rules that adjust membership functions associated with appraisal variables. In particular, this input interface enables the use of cognitive information in the CME to amplify, attenuate or maintain the emotional meaning of the stimuli perceived by the agent. This proposal by (S. Castellanos et al., 2019), assumes that the representation coming from the cognitive components can be manipulated using membership functions. However, the input interface does not support syntactic interoperability and technical details are not provided. Instead, the InFra assumes that a specific communication module enables the communication between the CME and the cognitive components. In addition, the InFra has an output interface that allows to organize

the affective information produced by the internal components of the CME using an XML based data structure, so that this information is used by the cognitive components to influence the social behavior of the agent. However, the InFra does not provide implementation details about the output interface, so developers must build a communication channel between the emotion model and the cognitive components on their own in order to manipulate the XML files produced by InFra. Likewise, the output interface does not provide semantic interoperability, since the InFra does not guarantees that affective representations in XML format are understandable for cognitive components.

Another related proposal is ArchitecturalEmotion (Osuna et al., 2021), an architectural software design pattern for designing a structural organization for CMEs and their interrelationships with cognitive components. ArchitecturalEmotion has a degree of syntactic interoperability, as it uses design patterns to manage communications with cognitive entities. For example, it uses the adapter design pattern to provide a common communication interface between the CME and cognitive components that were not explicitly designed to interact with a CME. However, ArchitecturalEmotion does not support semantic interoperability, since it is not capable of constructing a propositional meaning from the cognitive representation.

The work by (Marinier & Laird, 2008) presents another CME that addresses the integration of emotion and cognition. It uses the PEACTIDM cognitive control theory that establishes a series of abstract functional operations that agents can perform to generate specific behaviors. However, this proposed model seems to lack interoperable mechanisms regarding the communication with cognitive components. In particular, affective processes were embedded within a cognitive model (PEACTIDM) and were subsequently evaluated in the cognitive architecture Soar, proposed by (Laird, 2019). Therefore, syntactic and semantic interoperability were established in an ad hoc manner. Thus, it is able to interact with only cognitive components that have been constructed under the criteria established by the theory of cognitive control.

MAMID (Methodology for Analysis and Modeling of Individual Differences) is a methodology for modeling the influences of emotions and individual differences in cognitive processing (Hudlicka, 2020). The emotion model presents two main contributions: (1) an architecture that executes an appraisal process to produce an affective state, and (2) a means to represent the effects of the resulting emotions on cognitive processing. Similar to PEACTIDM, the emotion model is embedded within the cognitive architecture of MAMID. Therefore, syntactic and semantic interoperability were established in an ad hoc manner. In this sense, MAMID (as PEACTIDM) does not presents interoperable mechanisms, since it is not capable of interacting with other cognitive architectures or with other independent cognitive components.

In spite of the extensive interaction between affective and cognitive components modeled in CMEs, contemporary models barely address issues associated with the concept of interoperability (e.g., syntactic and semantic aspects). For instance, although some models have attempted to devise dedicated interfaces to communicate independently developed cognitive components with components in a CMEs (Osuna et al., 2021; Rodríguez & Ramos, 2012; Steunebrink et al., 2012), very few or no technical detail on how communication channels are designed is provided nor evaluations of the benefits of such communication interfaces. Furthermore, no descriptions for assigning accurate and agreed meanings (between the two types of components) to the data being exchanged are presented. This evidence emphasizes the relevance of research questions associated with the extensive interaction and interdependence between cognitive components in cognitive agent architectures and affective components in CMEs, which could even be components developed by independent parties. Importantly, the previous analysis makes evident the importance of adopting methods and techniques to achieve the interoperability required in this type of computational model and address some of the challenges in contemporary CMEs.

A FRAMEWORK FOR THE INTEROPERABILITY OF AFFECTIVE AND COGNITIVE COMPONENTS

This section presents a framework aimed at facilitating the communication between cognitive and affective components so that cognitive data is made available to the emotional mechanisms in CMEs. Particularly, the framework's design is based on evidence indicating that cognitive information influences human emotions and that, therefore, the mechanisms implemented in CME should be able to receive, understand, and process any type of cognitive information that affects its internal operation. In this context, a CME must take as input different types of information available from agent architectures in order to accurately assess perceived emotional stimuli and generate consistent emotional states and emotional behaviors. From a computational point of view, dealing with diverse cognitive components and varied types of cognitive information represents a technical challenge.

The framework (Figure 2 and Table 1) is organized into three layers: i) Model of Emotion Layer, composed of affective components, ii) Cognitive Layer, composed of cognitive components of a cognitive agent architecture, iii) Interoperability Layer (syntactic and semantic). This proposal assumes that cognitive components could be developed by third parties, independent of the development of affective components in a CME. Therefore, data normalization procedures are considered to translate

Figure 2. Proposed framework for syntactic and semantic interoperability in CMEs

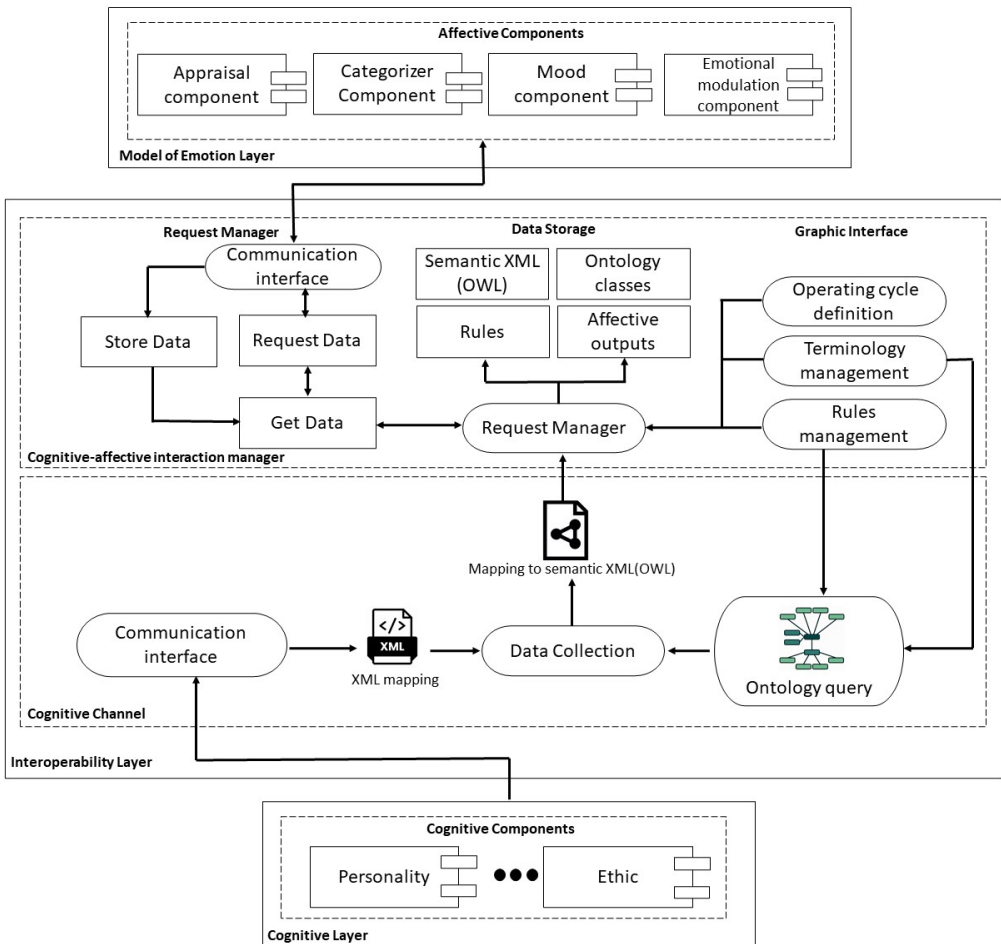


Table 1. Components and processes of the proposed framework

Component/Process	Description
Cognitive component	Cognitive models that establish communication with the proposed framework
Communication interface	Provides a common interface between the cognitive components and the interoperability layer through adapter design pattern
XML mapping	Mapping of the output data of a cognitive component to a valid XML-based data representation for the framework
Data Collection	Collects the information generated by the communication interface, as well as the classes and rules provided by the ontology
Ontology query	Performs a class search in the ontology using as reference the XML file produced in the syntactic interoperability
Mapping to semantic XML(OWL)	Mapping the data of the output of a cognitive component, its class in the ontology and the associated rules to a semantic XML-based data representation (OWL)
Request Manager	Serves requests related to obtaining and delivering data such as semantic XML, ontology classes, rules and outputs of affective components
Semantic XML (OWL)	Data entity that persists in the Data Storage and that refers to the semantic XMLs provides by interoperability layers
Ontology classes	Data entity that persists in the Data Storage and that refers to the ontology classes
Affective outputs	Data entity that persists in the Data Storage and that refers to the output provides by affective components
Rules	Data entity that persists in the Data Storage and that refers to the rules associated with ontology classes
Operating cycle definition	Determines the order of execution of the affective components through a graphic interface
Ontology management	Provides new classes associated with ontology through a graphic interface
Rules management	Provides new rules associated with ontology classes through a graphic interface
Request Data	Represents a request for data by an affective component.
Store Data	Represents a request to store data (output) of an affective component
Get Data	Request a specific data to the Data Storage layer
Send Request	Send the requested data to a specific affective component
Affective components	Components that model a specific phase of the emotion process and that interact with the proposed framework

data to appropriate formats understandable by the internal mechanisms of CMEs. It is important to note that cognitive and affective components are treated separately: the first considered as part of cognitive agent architectures and the second as part of CMEs. Furthermore, cognitive components are treated as source of information that biases the underlying mechanisms of the emotion model.

Cognitive Channel

The Cognitive Channel comprises a series of components to communicate the information produced by cognitive components that belong to the agent's cognitive architecture, which according to the literature influence emotion processes and therefore lead to more realistic emotions in virtual agents (see Figure 2). In order to communicate the output information, cognitive components must implement a generic interface based on the adapter design pattern, a common practice used in the software engineering field that allows two incompatible components to work together. It is assumed that information from

cognitive components may not be fully compatible with affective components. Therefore, interfaces may be incompatible, but the inner functionality should suit the need. The adapter design pattern allows incompatible components to work together by converting the interface of one component into an interface expected by the others components (Hamlaoui et al., 2018).

Cognitive components must be extended to include message exchange methods that enable communication with the Cognitive Channel. A message exchange pattern establishes the message characteristics required by the communication protocol used by the communication channel. The *communication protocol* is a format used to represent the message that all communicating parties agree on (or are capable to process). The *communication channel* is the infrastructure that enables messages to “travel” between the communicating parties. The *message exchange patterns* describe the message flow between parties in the communication process, using a one-way pattern where each cognitive component takes the role of client sending its information as a message that is then processed by the adapter pattern in the communication interface. The information projected by the cognitive component is encapsulated and represented in an XML file that ensures the establishment of a common communication interface so that the component can be processed in later phases (Taylor & Taylor, 2021).

The representation of the cognitive information begins in the communication interface, which ensures that the cognitive component’s information can be processed by the Cognitive-Affective Interaction Manager (CAIM) and affective components, transforming the information and mapping it into an XML file representing the component. However, the file in the first instance is meaningless since at this point it is not possible to establish a relationship that determines its impact on the affective components. Depending on the nature of the cognitive process that is modeled on the first step, the output of a cognitive component may consist of one or more data values. The Interoperability Layer maps all these data values into dimensions conformed by a name and its associated value in an XML format. The XML representation of the cognitive component is complemented by an ontology that encompasses a representation, formal naming and definition of the categories, properties, and relations between the cognitive and affective concepts. Regarding the semantic interoperability, it is responsible for assigning a coherent meaning to the data received from cognitive components (and previously pre-processed by the communication interface achieving syntactic and semantic interoperability). This component uses ontologies to map the outputs of the cognitive components into relevant meaningful data that can be understandable and used by affective components. Furthermore, this layer is responsible for publishing this information so that the CAIM makes it available to all interested affective components in the Model of Emotion Layer.

The main objective of building and using an ontology is to create a shared common vocabulary and provide a format to standardize a general understanding regarding the relationship of the emotions model with specific cognitive components. The proposed ontology is based on the Web Ontology Language (OWL), a markup language for publishing and sharing data through a model built on RDF and encoded in XML (Bechhofer et al., 2004). A top-down approach was utilized to develop the ontology (Bechhofer et al., 2004; Zheng & Qin, 2011), which establishes that the ontology concept modeling process must start at the higher level, followed by more specific concepts at the lower levels. The proposed approach allows manipulating the ontology to add new classes and sub classes, which represent the cognitive components that are connected to the framework. Examples of classes that could be added to the ontology and that correspond to cognitive components in an agent architecture are perception, learning, personality, memory, attention, among others (Bosse et al., 2013; Bozkurt, 2015; Cherukuri et al., 2021; Faghihi et al., 2012; Lotfi & Akbarzadeh-T., 2014; Mavis et al., 2021).

The ontology represents a first step in assigning semantic meaning to cognitive data. The taxonomy of concepts provided by the ontology is also utilized to identify cognitive components, their theoretical approach, the different values that the component delivers, among other aspects. Subsequently, a rule-based approach is used to assign an appropriate meaning to the cognitive data based on available rules associated with specific classes of the ontology (consulted to the CAIM).

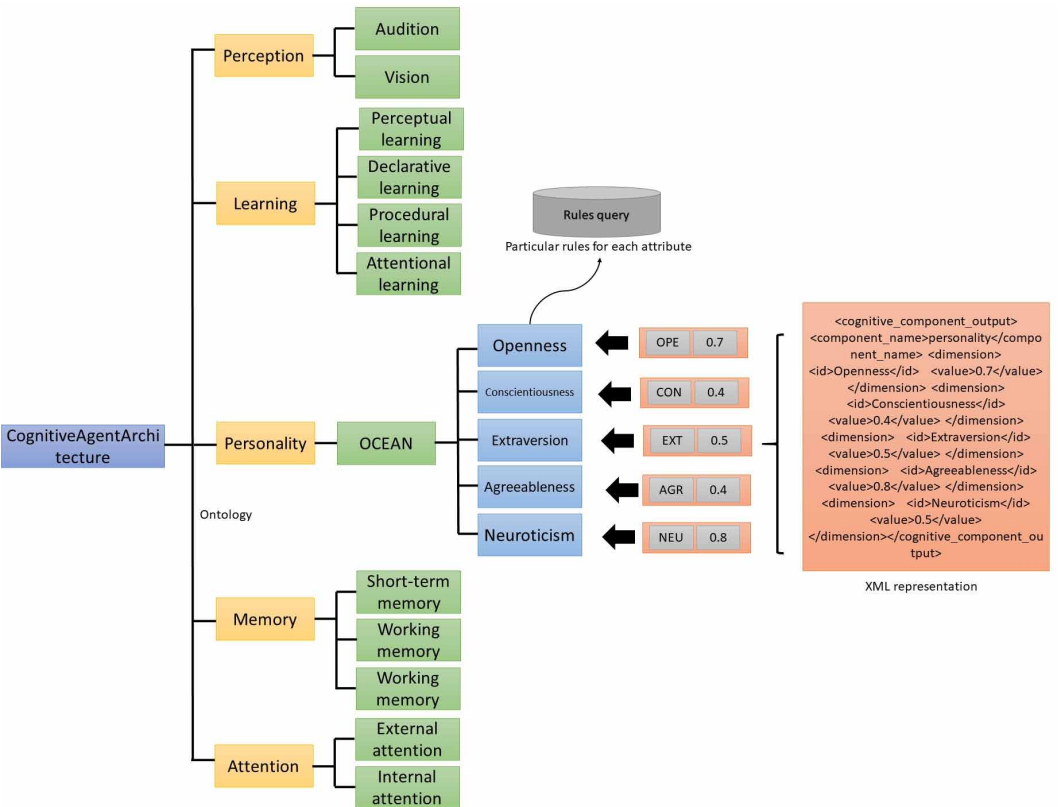
These rules determine how and to what extent the affective components are affected by cognitive data. In this manner, cognitive data is utilized to influence the internal function of affective components. The particular rules associated to each cognitive component are introduced by the researcher using a graphical interface. See Figure 3 for an explanation of how cognitive data is translated to rules using the ontology. The rules are separated from the other components and can be added and edited by experts.

Cognitive-Affective Interaction Manager

This layer is composed of affective components of a given CME. These affective components have a dependency with the Cognitive-affective Interaction Manager (CAIM) of the Interoperability Layer, since it provides input data that allows performing their internal functions. Affective components must use a generic interface based on the Adapter design pattern. However, unlike the cognitive components, these must include methods of sending and receiving data to maintain a constant interaction with the CAIM.

The CAIM is composed of modules that implement mechanisms in charge of managing and coordinating requests from the affective components and the Interoperability Layer. Particularly, the CAIM regulates and manages the interactions between the affective and cognitive components, stores information that results from these interactions, and provides a graphic interface that allow researchers to i) establish an execution flow between affective and cognitive components, ii) manage an appropriate terminology in the form of an ontology to homogenize concepts, dimensions and ranges of values that come from the cognitive components outputs, and iii) assign rules to the ontology classes. The CAIM implements three main mechanisms: a Request Manager, a Data Storage and the Graphic Interface.

Figure 3. Ontology details



The proposed framework assumes that cognitive and affective components may have not been designed to interact with each other or share information between them. To address this issue, a flexible Data Storage mechanism is designed to serve as a central entity for storing and sharing information among affective and cognitive components, thus freeing the components from having to maintain internal storage and manage complex data. This mechanism is capable of storing four types of data: i) Semantic XML, a product of the interaction of cognitive components with the interoperability layer, ii) affective outputs that are generated by affective components, iii) ontology classes that correspond to the different classes that make up the cognitive component ontology, and iv) rules associated with the different classes of the ontology.

In the proposed framework, a Request Manager mechanism is provided to help mediate between the different requests of the affective components. In this manner, affective components focus only on performing two types of interactions with the CAIM. The first is related to the data that each affective component requires to perform its internal functions. In this case, the Request Manager directly obtains the requested data from the Data Storage mechanism and delivers it to the requesting affective component. The second type of interaction occurs when an affective component generates a data structure as output. The Data Manager is responsible for directly sending the output of the affective component to the Data Storage so that this information is available to the rest of affective components. In this manner, the affective components do not need to communicate directly with each other to request or store data, instead they interact with the Data Manager, thus providing a flexible interaction between this type of component.

The CAIM includes graphical interfaces that allow researchers to input and manage information about the classes and the rules associated with the ontology concepts (utilized in the interoperability process). The ontology classes can be mapped into high-level abstract models can be created in any object-oriented programming language. In particular, a tree data structure is used to represent each concept of the ontology, thus, the graphical interface only focuses on adding or removing new nodes in the tree (including properties and constraints). The first level of the tree corresponds to the concepts related to the cognitive components (e.g., personality), the next lower level corresponds to the properties associated with it (e.g., Openness, Conscientiousness, Extraversion, Agreeableness and Neuroticism according to the OCEAN theory proposed (Wiggins & Trapnell, 1997) and in turn, each property has attributes (e.g., high, medium, and low).

The graphical interface was also designed to manage the rules associated with the elements of the ontology. The interface allows the researcher to select an element of the ontology and define how an affective component is influenced by cognitive information. For instance, the appraisal variables in the appraisal component of a CME could be positively or negatively influenced, according to what the researcher defines in rules of the ontology captured using the available interface (see Figure 4). In this manner, the framework provides flexibility so that cognitive components are enabled to make available cognitive information to affective components in CMEs. Thus, as long as the necessary classes and rules are added to the ontology, any cognitive component is able to interact with affective components through the proposed framework.

PROOF OF CONCEPT

This section illustrates how the proposed framework allows the components of a CME to establish syntactic and semantic interoperability with cognitive components that are developed independently. First, assume that the cognitive component modeling the agent's personality is connected to the framework and that this component was designed based on the OCEAN personality theory (Wiggins & Trapnell, 1997), which utilizes five dimensions to represent the agent's personality (i.e. openness, conscientiousness, extraversion, agreeableness, and neuroticism). As a result, a common interface is adopted by this cognitive component that enables the communication with the framework. This interface helps translate and organize the personality output data into an XML based file that serves as input to the framework. In particular, the Interoperability Layer receives an XML similar to the one presented below:

Figure 4. User interface to enter ontology classes and rules

Create Condition

Cognitive Component:

Sub Class:

Condition:

Property: Attribute:

Create Conclusion

Affective Component:

Affective Property:

Impact:

Condition:

Final Rules

IF Extraversion IS high AND Neuroticism IS low AND Agreeableness IS low AND Conscientiousness IS low AND Neuroticism IS low THEN emotionModule
 IF Extraversion IS low THEN moodComponent-arousal IS low.

```
<cognitive\_component\_output>
  <component\_name>personality</component\_name>
  <dimension>
    <id>Openness</id>
    <value>0.7</value>
  </dimension>
  <dimension>
    <id>Conscientiousness</id>
    <value>0.4</value>
  </dimension>
  <dimension>
    <id>Extraversion</id>
    <value>0.5</value>
  </dimension>
  <dimension>
    <id>Agreeableness</id>
    <value>0.8</value>
  </dimension>
  <dimension>
    <id>Neuroticism</id>
    <value>0.5</value>
  </dimension>
</cognitive\_component\_output>
```

This sequence indicates that the personality component identified with the name “personality” uses a data representation of five dimensions with values for Openness = 0.7, Conscientiousness = 0.4, Extraversion = 0.5, Agreeableness = 0.8 and Neuroticism = 0.5. Subsequently, the Interoperability Layer uses the ontology about components of cognitive agent architectures to query the existence of a personality class, using the <component_name> tag of the XML file as a reference. It is important to note that the classes and the corresponding rules in the ontology must be previously introduced by the researcher through the graphical interface provided by the framework so that when the ontology identifies the class with the concept of personality, it consults the rules associated with this concept. In this case, the XML file that receives the Interoperability Layer contains five values that match with the values of the OCEAN class in the ontology. Then, the rules associated with each dimension of the OCEAN class are extracted and a new semantic XML file is generated, which is inspired by Ontology Web Language (OWL). It is important to note rules are used to define the impact of cognitive information on affective components, in a specific way and to a specific degree (See Figure 4). In this sense, it may be assumed that the personality component is connected to the framework to bias the internal workings of the Mood component. Thus, when consulting the associated rules, it is determined that the first and third rule are relevant.

```
IF Extraversion IS high AND Neuroticism IS low AND Agreeableness
IS low THEN moodComponent-pleasure IS positive.
```

```
IF Conscientiousness IS low AND Neuroticism IS low THEN
emotionModulationComponent-coping IS negative.
```

```
IF Extraversion IS low THEN moodComponent-arousal IS low.
```

The semantic interoperability layer maps the information from the personality cognitive component and the associated rules to a new format based on semantic XML. Finally, the interoperability layer requests the CAIM to store the semantic XML (which has all the information necessary to modulate the affective component of Mood) and make it available. When the Data Storage receives a new data representation, it consults the operating cycle defined in the framework to execute the consequent affective components. Under this line, the framework determines that it must execute the Mood Component in order to update its state, so it requests the Data Manager layer to send the semantic XML file to the Mood component. The Data Manager receives the request, obtains the semantic XML, and sends it to the Mood component. Then, the Mood component can receive the semantic XML file as input and be executed taking into consideration the impact of the personality component data.

DISCUSSION

Due to the growing interest in the design of computational systems that model affective processes, proposals are required to focus on modular architectures capable of interoperating with other cognitive systems through well-defined and standardized communication interfaces. These types of interoperable proposals make sense when it is necessary to couple computational models of emotions with cognitive systems that provide a bias to the emotional behavior exhibited by an agent. To contribute to the development of interoperable architectures between emotion models and cognitive components, this article constitutes an effort to design a framework for CMEs. The proposed framework provides researchers with an option to achieve interoperability between cognitive and affective components through the use of software design patterns that have been used in other areas of research, whose usefulness has been widely demonstrated and proven. Based on the use of design patterns, the interoperability layer of the proposed approach is capable of providing a degree of syntactic interoperability, since it allows the affective components of an emotions model to be

flexibly connected to other cognitive components that could have been developed by third parties. So, these cognitive components should be extended slightly to include data exchange methods with the interoperability layer. On the other hand, in addition to providing a common and open interface between cognitive systems, a common semantics is required to give meaning to the exchanged data. In this sense, an ontology is utilized to assign unique meanings to the data that the components exchange, so the framework is capable of providing a degree of semantic interoperability. However, it is recognized that more extensive work on semantic interoperability is required to increase the precision of the meaning of the data that come from the numerous theories on which the models of emotions and cognitive components are based. Therefore, the proposed framework can be improved by using existing robust ontologies (such as those used in (Neerinx et al., 2016) or (Aydin & Aydin, 2020)), using ontology design patterns (such as those used in (Hitzler et al., 2016)) and using an ontology merging module (like the one used in (Raunich & Rahm, 2012)).

CONCLUSION AND FUTURE WORK

This study proposes a framework to integrate the data of cognitive components that were developed independently in a model of emotions from the perspective of syntactic and semantic interoperability. The proposed approach contributes to the understanding in relation to the technical and semantic aspects that must be taken into account so that the cognitive components can be integrated into a model of emotions. To achieve this integration, an ontology is used to assign specific meanings to the cognitive components that are connected to the proposed framework and through pre-established rules, it is possible to influence the internal functions of the affective components in an emotions model. As a result of this research, three main findings were identified: i) in order to integrate emotion and cognition in a software system, it is necessary to explore aspects related to software interoperability; ii) semantic interoperability between affective and cognitive entities can be carried out through the use of ontologies of cognitive agent architectures and the use of rules that assign specific meanings; iii) software engineering can provide methods, techniques and strategies to solve syntactic integration problems between affective and cognitive entities. A clear example of this is the use of the adapter design pattern that has been used as inspiration in the syntactic interoperability layer of the proposed approach. Likewise, the proposed framework represents the cornerstone of a research effort that aims to integrate CMEs with cognitive entities that are developed by third parties. Finally, four lines of future research are identified. First, it is required to explore new methods of software engineering to provide a better light on the technical implications of connecting cognitive components to CMEs. Second, it is required to define ontology patterns that support efforts to achieve semantic interoperability between CMEs and cognitive components. Third, to achieve a better degree of interoperability, a scalable model of emotions is required that be capable of connecting third-party affective components and that use different theoretical approaches. Finally, the present study has only investigated syntactic and semantic interoperability, so research on other layers of interoperability can potentially improve the interaction between CMEs and cognitive components.

REFERENCES

- Aydin, S., & Aydin, M. N. (2020). Semantic and syntactic interoperability for agricultural open-data platforms in the context of IoT using crop-specific trait ontologies. *Applied Sciences (Basel, Switzerland)*, 10(13), 4460. doi:10.3390/app10134460
- Bechhofer, S., Van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F., Stein, L. A., & others. (2004). OWL web ontology language reference. *W3C Recommendation*, 10(2), 1–53.
- Bosse, T., Gerritsen, C., De Man, J., & Treur, J. (2013). Learning emotion regulation strategies: A cognitive agent model. *Proceedings - 2013 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2013*, 2, 245–252. doi:10.1109/WI-IAT.2013.116
- Bozkurt, I. (2015). Developing an agent-based model on how different individuals solve complex problems. *Journal of Industrial Engineering and Management*, 8(1), 233–266. doi:10.3926/jiem.1197
- Caro, M. F., Josyula, D. P., Madera, D. P., Kennedy, C. M., & Gómez, A. A. (2019). The Carina metacognitive architecture. *International Journal of Cognitive Informatics and Natural Intelligence*, 13(4), 71–90. doi:10.4018/IJCINI.2019100104
- Castellanos, S., & Rodríguez, L.-F. (2018). A Flexible Scheme to Model the Cognitive Influence on Emotions in Autonomous Agents. *International Journal of Cognitive Informatics and Natural Intelligence*, 12(4), 81–100. doi:10.4018/IJCINI.2018100105
- Castellanos, S., Rodríguez, L.-F., & Gutierrez-Garcia, J. O. (2019). A mechanism for biasing the appraisal process in affective agents. *Cognitive Systems Research*, 58, 351–365. Advance online publication. doi:10.1016/j.cogsys.2019.08.008
- Cherukuri, A. K., Shivhare, R., Abraham, A., Li, J., & Jonnalagadda, A. (2021). A Pragmatic Approach to Understand Hebbian Cell Assembly. *International Journal of Cognitive Informatics and Natural Intelligence*, 15(2), 73–95. doi:10.4018/IJCINI.20210401.oa6
- El Hamlaoui, M., Bennani, S., Nassar, M., Ebersold, S., & Coulette, B. (2018). Heterogeneous design models alignment: from matching to consistency management. *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, 1695–1697. doi:10.1145/3167132.3167425
- Faghihi, U., Fournier-Viger, P., & Nkambou, R. (2012). A computational model for causal learning in cognitive agents. *Knowledge-Based Systems*, 30, 48–56. doi:10.1016/j.knsys.2011.09.005
- Gavirangaswamy, V., Gupta, A., Terwilliger, M., & Gupta, A. (2019). RDMTk: A Toolkit for Risky Decision Making. [IJCINI] *International Journal of Cognitive Informatics and Natural Intelligence*, 13(4), 1–38. doi:10.4018/IJCINI.2019100101
- Hitzler, P., Gangemi, A., Janowicz, K., Krisnadhi, A. A., & Presutti, V. (2016). *Ontology Engineering with Ontology Design Patterns: Foundations and Applications*. IOS Press.
- Horsch, M. T., Niethammer, C., Boccardo, G., Carbone, P., Chiacchiera, S., Chiricotto, M., Elliott, J. D., Lobaskin, V., Neumann, P., & Schiffels, P. (2019). Semantic interoperability and characterization of data provenance in computational molecular engineering. *Journal of Chemical & Engineering Data*, 65(3), 1313–1329.
- Huang, X., Zhang, S., Shang, Y., Zhang, W., & Liu, J. (2017). Creating Affective Autonomous Characters Using Planning in Partially Observable Stochastic Domains. *IEEE Transactions on Computational Intelligence and AI in Games*, 9(1), 42–62. doi:10.1109/TCIAIG.2015.2494599
- Hudlicka, E. (2020). The Case for Cognitive-Affective Architectures as Affective User Models in Behavioral Health Technologies. *International Conference on Human-Computer Interaction*, 191–206. doi:10.1007/978-3-030-50439-7_13
- Jain, S., & Asawa, K. (2015). EMIA: Emotion Model for Intelligent Agent. *Journal of Intelligent Systems*, 24(4), 449–465. doi:10.1515/jisys-2014-0071
- Jha, S. S., Shrinivasa, S. N., & Nair, S. B. (2013). Driving robots using emotions. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 8160, pp. 64–89). Springer. doi:10.1007/978-3-642-45318-2_3

- Laird, J. E. (2019). *The Soar cognitive architecture*. MIT Press.
- Lotfi, E., & Akbarzadeh-T, M. R. (2014). Practical emotional neural networks. *Neural Networks*, 59, 61–72. doi:10.1016/j.neunet.2014.06.012 PMID:25078111
- Marinier, R. P., & Laird, J. E. (2008). Emotion-driven reinforcement learning. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 30(30).
- Mavis, G., Toroslu, I. H., & Karagoz, P. (2021). Personality Analysis Using Classification on Turkish Tweets. [IJCINI]. *International Journal of Cognitive Informatics and Natural Intelligence*, 15(4), 1–18. doi:10.4018/IJCINI.287596
- Neerincx, Kaptein, F., van Bekkum, M. A., Krieger, H.-U., Kiefer, B., Peters, R., Broekens, J., Demiris, Y., & Sapelli, M. L. S. (2016). Ontologies for Social, Cognitive and Affective Agent-Based Support of Child's Diabetes Self Management. *ECAI 2016*.
- Osuna, E., Rodríguez, L.-F., & Gutierrez-Garcia, J. O. (2021). Toward integrating cognitive components with computational models of emotion using software design patterns. *Cognitive Systems Research*, 65, 138–150. doi:10.1016/j.cogsys.2020.10.004
- Rath, M., Rodrigues, J. J. P. C., & Oreku, G. S. (2021). Applications of Cognitive Intelligence in the Information Retrieval Process and Associated Challenges. *International Journal of Cognitive Informatics and Natural Intelligence*, 15(1), 26–38. doi:10.4018/IJCINI.2021010103
- Raunich, S., & Rahm, E. (2012). Towards a Benchmark for Ontology Merging. In P. Herrero, H. Panetto, R. Meersman, & T. Dillon (Eds.), *On the Move to Meaningful Internet Systems: OTM 2012 Workshops* (pp. 124–133). Springer. doi:10.1007/978-3-642-33618-8_20
- Rodriguez, L.-F., Gutierrez-Garcia, J. O., & Ramos, F. (2016). Modeling the interaction of emotion and cognition in Autonomous Agents. *Biologically Inspired Cognitive Architectures*, 17, 57–70. doi:10.1016/j.bica.2016.07.008
- Rodríguez, L. F., & Ramos, F. (2012). Computational models of emotions for autonomous agents: Major challenges. *Artificial Intelligence Review*, 43(3), 437–465. doi:10.1007/s10462-012-9380-9
- Rodríguez, L. F., & Ramos, F. (2014). Development of Computational Models of Emotions for Autonomous Agents: A Review. *Cognitive Computation*, 6(3), 351–375. doi:10.1007/s12559-013-9244-x
- Steunebrink, B. R., Dastani, M., & Meyer, J. J. C. (2012). A formal model of emotion triggers: An approach for BDI agents. *Synthese*, 185(S1), 83–129. doi:10.1007/s11229-011-0004-8
- Taylor, J. T., & Taylor, W. T. (2021). Software architecture. In *Patterns in the Machine* (pp. 63–82). Springer. doi:10.1007/978-1-4842-6440-9_5
- Tieck, J. C. V., Steffen, L., Kaiser, J., Reichard, D., Roennau, A., & Dillmann, R. (2019). Combining motor primitives for perception driven target reaching with spiking neurons. *International Journal of Cognitive Informatics and Natural Intelligence*, 13(1), 1–12. doi:10.4018/IJCINI.2019010101
- Wiggins, J. S., & Trapnell, P. D. (1997). Personality structure: The return of the Big Five. In *Handbook of personality psychology* (pp. 737–765). Elsevier. doi:10.1016/B978-012134645-4/50029-9
- Xie, L.-J., Wen, H., & Xiao, N.-F. (2012). Affective computing model based on HMM for home-service robot. *Computer Engineering and Design*, 33(1), 322–327.
- Yalcin, O. N., & Dipaola, S. (2018). A computational model of empathy for interactive agents. *Biologically Inspired Cognitive Architectures*, 26(July), 20–25. doi:10.1016/j.bica.2018.07.010
- Zheng, Q., & Qin, J. (2011). Evaluating the emotion based on ontology. *IEEE Symposium on Web Society*, 32–36. doi:10.1109/SWS.2011.6101266

Enrique Osuna is a PhD student working on a formal methodology for the development of computational models of emotions (CMEs). Specifically, he is making an intersection of affective computing and software engineering to create such methodology. The main objective is to facilitate the CMEs development process with the help of software engineering practices.

Sergio Castellanos is a PhD Student in Engineering Sciences at ITSON, working in the area of cognitive computing and affective computing for intelligent agents. Experience in computational modeling of emotions in agents, learning and behavior.

Luis-Felipe Rodríguez is currently a full time Professor of Computer Science at Sonora Institute of Research, México. He received a PhD in Computer Science from the Center for Research and Advanced Studies of the National Polytechnic Institute (CINVESTAV-IPN), campus GDL, México in 2013. Dr. Rodríguez has served as a reviewer for various international journals and conferences. His research interests include cognitive computing, agent-based modeling, affective human-computer interaction, and decision support systems. Dr. Rodríguez is member since 2015 of the National System of Researchers from the National Council of Science and Technology in Mexico (SNI-CONACYT).