VEO-Engine: Interfacing and Reasoning with an Emotion Ontology for Device Visual Expression

Muhammad Amith[1], Rebecca Lin[2], Chen Liang[3], Yang Gong[1], and Cui Tao[1](✉)

[1] University of Texas Health Science Center, Houston, TX 77030, USA
cui.tao@uth.tmc.edu
[2] Johns Hopkins University, Baltimore, MD 21218, USA
[3] Lousiana Tech University, Ruston, LA 71272, USA

Abstract. In order for machines to understand or express emotion to users, the specific emotions must be formally defined and the software coded to how those emotions are to be expressed. This is particularly important if devices or computer-based tools are utilized in clinical settings, which may be stressful for patients and where emotions can dominate their decision making. We have reported our development and feasibility results of an ontology, Visualized Emotion Ontology (VEO), that links abstract visualizations that express specific emotions. Here, we used VEO with the VEO-Engine, a software API package that interfaces with the VEO. The VEO-Engine was developed in Java using Apache Jena and OWL-API. The software package was tested on a Raspberry Pi machine with a small touchscreen display that linked each visualization to an emotion. The VEO-Engine stores input parameters of emotional situations and valences to reason and interpret users' emotions using the ontology-based reasoner. With this software, devices can interfaced wirelessly, so smart devices with visual displays can interact with the ontology. By means of the VEO-Engine, we show the portability and usability of the VEO in human-computer interaction.

Keywords: Ontology · Emotions · Semantic web · mHealth
Affective computing

1 Introduction

Emotions affect our intentions, perceptions, behaviors, and decision-making. From a patient's perspective, emotions can influence decisions that impact his/her health. Such decisions may involve care situations wherein a patient's loved ones must choose to continue or to end care, or they may influence whether or not a patient accepts immunizations So, when interacting with patients or health consumers, it is important to account for the role emotions play. Providers must be conscious of emotional contagion [7], where he or she could sway the patient by expressing an emotion that could affect or inspire the patient's own emotions.

C. Stephanidis (Ed.): HCII Posters 2018, CCIS 851, pp. 349–355, 2018.
https://doi.org/10.1007/978-3-319-92279-9_47

Inspired by the use of intelligent agents in health care, we surmised that if such tools were to be used with patients, they would need to include emotions as a factor. In this paper, we discuss and demonstrate a proof-of-concept software engine, VEO-Engine, that could add emotional responses to intelligent agents using ontologies and semantic web technologies.

Briefly, an ontology is a semantic-driven electronic artifact that formally represents concepts, links between concepts, and domain knowledge in a machine-readable format. This artifact is published in a machine-based syntax that assists machines to structure domain knowledge and manifests the knowledge in a format that can be shared and processed by machines. One of these syntax languages is Web Ontology Language (OWL) [13], which is the language we used in this work. OWL provides language features that perform high-level machine reasoning based on coding. Theoretically, when a machine can define and structure knowledge and concepts from a specific domain, it can further understand the domain.

1.1 Summary of Previous Work

We investigated a spectrum of emotions and how to define them for machines to understand. We translated the Ortony, Clore, and Collins' (OCC) model of emotions [11] and the proposed revised version by Steunebrink et al. [12] into what we called the Visualized Emotion Ontology (VEO) using OWL [10]. In addition, while all but one emotion overlapped with the Paul Ekman classifications of emotions [4], we also included *surprise* into the ontology. In brief, the OCC model defines emotions based on emergent conditions using a composite of behavior and situations. For example, the emotion of *fear* is defined as a negative feeling that involves a situation pertaining to displeasure of a probable consequence. Further, OCC utilizes some semantics and logical structures that can be easily rendered to create an ontology using OWL. Lastly, we created visualizations based on evidence from published research for each of the emotions described, and each visualization linked to an emotion using the ontology. Overall, the VEO semantically defined and visualized 25 emotions [9].

We then assessed the representation of emotions by evaluating the VEO structure using semiotic theory-driven metrics and assessed the visual representations of the emotions using Amazon Mechanical Turk. The initial assessments yielded a structurally and semantically sound ontology, compared with other cognitive-related ontologies, and the individuals surveyed ($n = 1082$) agreed that most visualizations represented specific emotions [9].

Next, we endeavored to use the VEO in machines. This could demonstrate the usefulness of the ontology, and semantic web technologies in general, in machines that could host intelligent agents.

1.2 Research Objectives

The objective of this study was to show that, for small devices, we could use an emotion ontology to reason and query emotions. This study could further our work in developing conversational agents that include emotions in interactions with humans. Also, this may further interest in using ontologies and the semantic web to help machines express and interpret emotions with humans users.

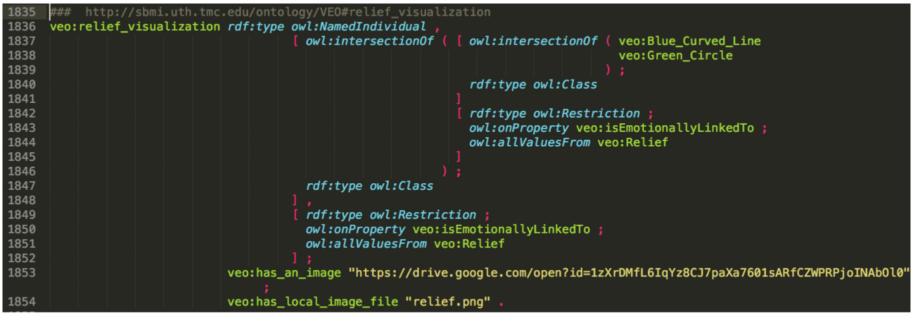
To support our objective, we performed the following:

1. Developed the proof-of-concept engine that harnessed the VEO to allow for querying and interpretation of emotions using an application programming interface (API)
2. Tested the VEO-Engine's functionality to query and perform reasoning for emotions.

2 Materials and Method

The VEO-Engine was developed in Java and employs the following libraries: Apache Jena [2], OWL-API [8], and the HermiT reasoner [5]. The VEO-Engine software library carries an application-specific version of the VEO, which is the core knowledge base without the imported ontologies from our previous studies. With a simpler form of the VEO, this would make it easier to test and to experiment. Also, the VEO-Engine hosts local versions of the visualized emotion images. It was deployed as a distributable JAR[1] file that could be integrated with existing software applications.

We also added sample Java GUI[2] that enabled a demonstration of two basic functionalities of the VEO-Engine-querying emotion visualizations and machine-based reasoning. SPARQL [6] was used to query emotion visualizations. Each SPARQL query was executed on the VEO, and each VEO emotion was linked to an image in the JAR file. Figure 1 shows the VEO emotion visualization for *relief* defined in Turtle syntax [3]. The link to the image file is handled by the property $veo : has\_local\_image\_file$. In Fig. 1, *relief* visualization is assigned to the image file "relief.png".



```
###  http://sbmi.uth.tmc.edu/ontology/VEO#relief_visualization
veo:relief_visualization rdf:type owl:NamedIndividual ,
                                  [ owl:intersectionOf ( [ owl:intersectionOf ( veo:Blue_Curved_Line
                                                                                veo:Green_Circle
                                                                              ) ;
                                                           rdf:type owl:Class
                                                         ]
                                                         [ rdf:type owl:Restriction ;
                                                           owl:onProperty veo:isEmotionallyLinkedTo ;
                                                           owl:allValuesFrom veo:Relief
                                                         ]
                                                       ) ;
                                    rdf:type owl:Class
                                  ] ,
                                  [ rdf:type owl:Restriction ;
                                    owl:onProperty veo:isEmotionallyLinkedTo ;
                                    owl:allValuesFrom veo:Relief
                                  ] ;
                         veo:has_an_image "https://drive.google.com/open?id=1zXrDMfL6IqYz8CJ7paXa7601sARfCZWPRPjoINAbOl0"
                              ;
                         veo:has_local_image_file "relief.png" .
```

Fig. 1. Sample encoding (in Turtle) of VEO's emotion concept of *relief*. The last three lines denote the linked image and web files.

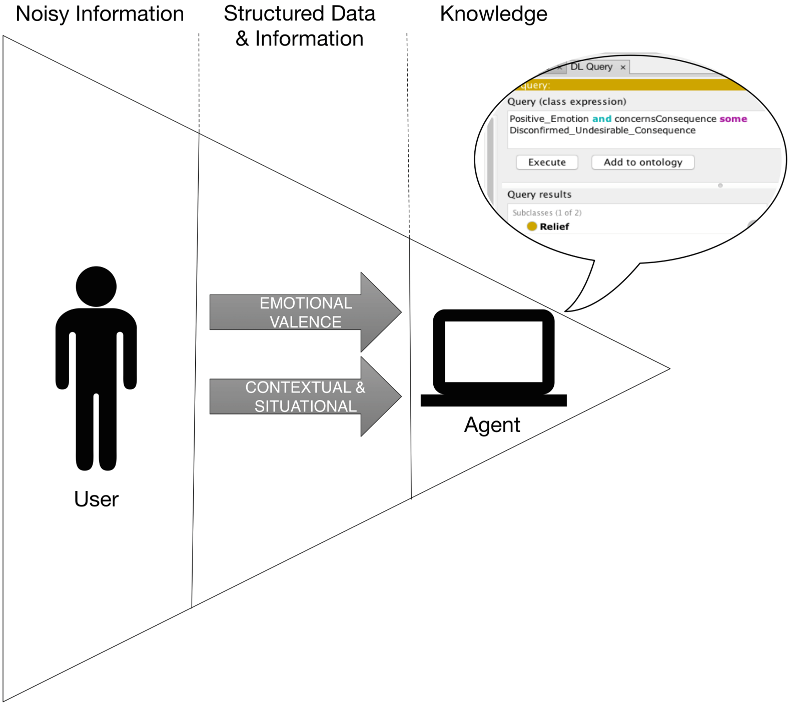
[1] executable Java archive file.

[2] Graphical User Interface.

The second VEO-Engine function involved machine-based reasoning that harnessed the HermiT API. In order to interpret the emotion, the VEO-Engine required an input for the emotional valence (positive or negative emotion type) and the contextual data for the situation or the psychological state. Formula (1) describes the format for the reasoner to define an emotion.

$$[\mathit{positive} \mid \mathit{negative}]\ \mathit{and}\ [\mathit{concept\_property}_1\ [\ldots \mathit{and}\ \mathit{concept\_property}_n]] \qquad (1)$$

For example, *love* is defined by VEO as a [*positive*] emotion that involves liking something familiar *[concept property := "concernsAspect some Familiar_Aspect"]*. In order for the software to determine whether love is being expressed, it would need data of *positive* for its emotional valence parameter and the parameter for a VEO concept property(ies) of *concernsAspect some Familar_Aspect*.



Positive_Emotion and concernsConsequence some Disconfirmed_Undesirable_Consequence

● Relief

Fig. 2. Interpreting user's emotional information.

Figure 2 shows the broad process wherein an intelligent agent consumes the emotional valence data and the contextual situational data from a human user. Using the entered parameters, the HermiT reasoner enabled the VEO-Engine to determine the precise emotion based on what has been defined in the VEO.

To test the software library, we used a Raspberry Pi 3 Model 3 board with Raspian version 9. Specific to Raspberry Pi, the device was also connected to 7" touchscreen display with 800 × 400 pixel screen resolution. The VEO-Engine was deployed to the device, and we executed sample tests through the command line to assess both the visualization query and the emotion reasoning of the library.

3 Results and Discussion

Aside from the input parameters we provided through the command line, the entire library was executed locally on the Raspberry Pi device and performed its functions without any connection to external software services.

Through a command line input for a specific emotion, the VEO-Engine queried for the corresponding image file and displayed a sample window showing that the visualization was linked to the emotion. Figure 3 shows *anger* displayed from the VEO-Engine on a Raspberry Pi device.

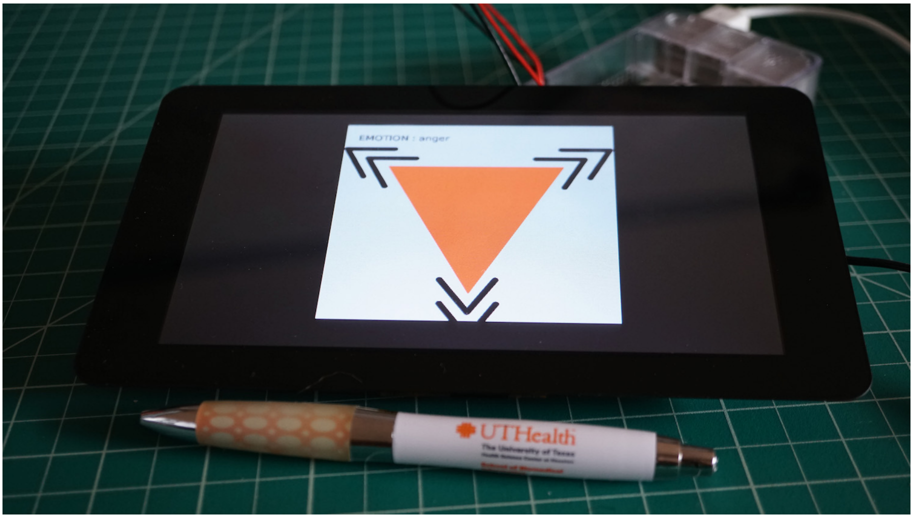
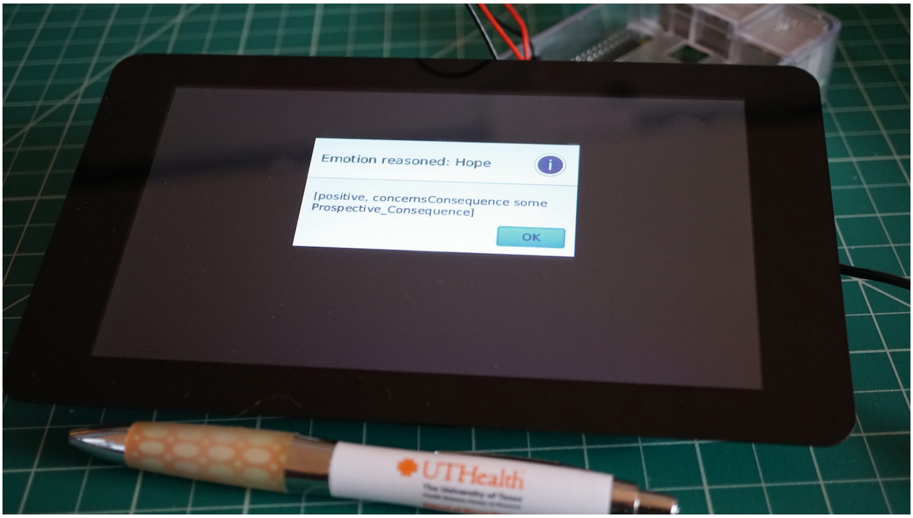


Fig. 3. Touchscreen device displaying results of a visualization query for the emotion *anger*.

We tested the VEO-Engine's reasoner by feeding a string of data describing an emotion. To test, the input required:

$$reason\ [positive|negative]\ [concept\_property_n]$$

For example, the input parameters of *reason [positive] [concernsConsequence some Prospective_Consequence]* was revealed to be the emotion of *hope*. Figure 4



Emotion reasoned: Hope

[positive, concernsConsequence some Prospective_Consequence]

Fig. 4. VEO-Engine performed a reasoning task based on parameters for the emotion of *hope*.

displays the result of a sample parameter input for *hope* to demonstrate the reasoning capability of VEO-Engine on a small device.

While our results show promise for semantic driven technologies, there are still opportunities for improvement. One would be to allow for synonymous emotion input in visualization queries, for example, *fondness* in place of *love*. To permit this, we would need to expand the ontology to link similar terms with each emotion and then modify the SPARQL queries. These improvements are possible because ontologies are graph-like, and therefore they can be changed easier than, say, a relational database [1].

To perform reasoning functions, the VEO-Engine required structured data input, so for this technology to be further applicable, it must map or translate the noisy contextual information from the human user into structured data. Therefore, if we looked at unstructured, free text from a person's utterances, we would need to parse out the information and then map that information to the appropriate parameters for emotional valences and concept properties to then input into the VEO-Engine. In this scenario, natural language processing might offer a direction.

4 Conclusion and Future Work

Our work exemplifies how semantic-encoded emotions could be utilized by software and small devices to assist machines in understanding human emotions. Based on our previous VEO work [9,10], we developed VEO-Engine, a software library that interfaces with the emotion ontology. The VEO-Engine was able to query for visualizations associated with an emotion, and it was able to deduce an emotion based on sample input parameters. The combination of having emotions semantically defined and a software wrapper to interface with the ontology makes semantic web technologies a feasible option for affective computing. In the future, we will look to incorporating this work into conversational agents for health care applications. Specifically, this could enhance how machines react and respond to patients' or health consumers' utterances to improve their outlook and well-being.

Acknowledgments. This research was supported by the UTHealth Innovation for Cancer Prevention Research Training Program (Cancer Prevention and Research Institute of Texas grant #RP160015), the National Library of Medicine of the National Institutes of Health award #R01LM011829, and the National Institute of Allergy and Infectious Diseases of the National Institutes of Health award #R01AI130460.

References

1. Allemang, D., Hendler, J.: Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL. Elsevier, New York City (2011)
2. Apache Software Foundation: Apache Jena. https://jena.apache.org
3. Beckett, D., Berners-Lee, T., Prudhommeaux, E.: Turtle-terse RDF triple language. W3C Team Submiss. **14**, 7 (2008)
4. Ekman, P., Friesen, W.V., Ellsworth, P.: Emotion in the Human Face: Guidelines for Research and an Integration of Findings: Guidelines for Research and an Integration of Findings. Pergamon, Oxford (1972)
5. Glimm, B., Horrocks, I., Motik, B., Stoilos, G., Wang, Z.: HermiT: an OWL 2 reasoner. J. Autom. Reason. **53**(3), 245–269 (2014)
6. Harris, S., Seaborne, A.: SPARQL (2013). http://www.w3.org/TR/sparql11-overview/
7. Hatfield, E., Cacioppo, J.T., Rapson, R.L.: Emotional contagion. Curr. Dir. Psychol. Sci. **2**(3), 96–100 (1993)
8. Horridge, M., Bechhofer, S.: The OWL API: a java API for OWL ontologies. Semant. Web **2**(1), 11–21 (2011)
9. Lin, R., Amith, M., Liang, C., Duan, R., Chen, Y., Tao, C.: Visualized emotion ontology: a model for representing visual cues of emotions. BMC Medical Informatics and Decision Making (In Press)
10. Lin, R., Amith, M., Liang, C., Tao, C.: Designing an ontology for emotion-driven visual representations. In: 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 1280–1283. IEEE (2017)
11. Ortony, A., Clore, G.L., Collins, A.: The cognitive structure of emotions. Cambridge University Press, Cambridge (1990)
12. Steunebrink, B.R., Dastani, M., Meyer, J.J.C.: The OCC model revisited. In: Proceedings of the 4th Workshop on Emotion and Computing (2009)
13. W3C Owl Working Group and others: OWL 2 Web Ontology Language Document Overview (2nd edn) (2012). http://www.w3.org/TR/owl2-overview/