Detecting Emotions for Preventing Aggression, Bullying, and Femicide: A Description Logic based Approach

Alessandra Cincotti[1,†], Antonio Lieto[2,3,†], Gian Luca Pozzato[1,*] and Stefano Zoia[1,†]

[1]Dipartimento di Informatica, Università degli Studi di Torino, c.so Svizzera 185, 10149 Turin, Italy
[2]Cognition, Interaction and Intelligent Technologies Lab/DISPC, Università di Salerno, Via Giovanni Paolo II, 132 84084 Fisciano (SA), Italy
[3]Cognitive Systems for Robotics, ICAR-CNR Institute, Palermo, Italy

Abstract

In this work we present SERC (Symbolic AI System for Emotion Recognition in Conversation), a cognitively inspired tool based on the Description Logic of typicality $\mathbf{T}^{\text{CL}}$ for the automatic attribution of emotions to speeches, with the main objective of providing a system to support the automatic detection of situations in which a dialogue may contain aggressive messages, expressions of anger, and indicators of (cyber)bullying behaviors. The SERC system exploits a commonsense reasoning framework based on the logic $\mathbf{T}^{\text{CL}}$, a probabilistic extension of Description Logics of typicality able to deal with the conceptual combination of prototypical descriptions, i.e. commonsense representations of given concepts. The tool SERC exploits an ontological formalization of emotions based on the Plutchik model for emotion attribution, in order to classify transcriptions of speeches in the corresponding emotions. As a difference with existing approaches based on subsymbolic methods (neural networks, deep learning), that typically take basic emotions into account, SERC exploits the logic $\mathbf{T}^{\text{CL}}$ to automatically generate novel commonsense semantic representations of compound emotions (e.g. Love as derived from the combination of Joy and Trust according to Plutchik). We show an application of the tool SERC to messages extracted from the legal documents concerning a femicide that actually occurred in Italy in 2023 and we provide an evaluation of the proposed approach by comparing its results with those of DailyDialog, a dataset of conversations in which each message is associated with an emotionally grounded annotation obtained through manual labeling. The results provided by SERC are promising and witness the potential of a symbolic approach to emotion recognition in text, which ensures transparency and explainability in the assessment process.

Keywords

Description Logics, Commonsense Reasoning, Concept combination, Affective Computing

1. Introduction

The rapid deterioration of social dynamics has led to a marked increase in reported violent incidents in recent years, ranging from brutal crimes committed by youth gangs to suicides among adolescents who are victims of (cyber)bullying, as well as the alarming number of femicides perpetrated by men who adhere to a deeply rooted notion of women as possessions or property. In many cases, the individuals involved leave multiple warning signs of the impending tragedy within the large volume of messages they exchange across social media platforms and private or group chats.

Sentiment analysis, an important area in natural language processing, has evolved from traditional techniques to deep learning approaches, notably transformer-based models. The field now extends beyond polarity classification to the recognition of specific emotions like joy, anger, and fear, despite challenges such as ambiguity, cultural differences, and the scarcity of fine-grained annotated data. Emotions have long been recognized as fundamental to human experience across cultures. Two primary modeling approaches exist: dimensional models, which explore continuous and physiological aspects

CILC 2026: 41th Italian Conference on Computational Logic, June 25–27, 2026, Ferrara, Italy
*Corresponding author.
†These authors contributed equally.
alessandra.cincotti@edu.unito.it (A. Cincotti); alieto@unisa.it (A. Lieto); gianluca.pozzato@unito.it (G. L. Pozzato); stefano.zoia@unito.it (S. Zoia)
https://www.antoniolieto.net (A. Lieto); http://www.di.unito.it/~pozzato/ (G. L. Pozzato); https://stefanozoia.github.io/ (S. Zoia)
0000-0002-8323-8764 (A. Lieto); 0000-0002-3952-4624 (G. L. Pozzato); 0000-0002-5797-9542 (S. Zoia)

of emotions [1, 2, 3], and categorical models, which capture conscious emotional states [4, 5, 6]. Recent advances include deep neural networks for affective state detection, leading to the development of annotated corpora, ontologies, and lexicons [7, 8, 9]. Attention-based neural models, such as the bidirectional CNN-RNN with attention [10], have shown state-of-the-art results on datasets ranging from long reviews to short texts like tweets. The affective content of text encompasses emotion, sentiment, mood, personality, and interpersonal stance [11]. This complexity has driven recent work toward fine-grained analysis, including the prediction of emotion and sentiment intensity or activation [12, 13], often using ensemble methods that combine deep learning with traditional features.

Depending on the specific research goals addressed, one could be interested in issuing a discrete label describing the affective state expressed (frustration, anger, joy, etc.) to address different contexts of interaction and tasks. Both basic emotion theories, in the Plutchik-Ekman [5] tradition, and dimensional models of emotions, have provided a precious theoretical grounding for the development of lexical resources [14, 15, 16, 17, 18] and computational models for emotion extraction. However, there is a general tendency to move towards richer, finer-grained models, possibly including complex emotions, especially in the context of data-driven and task-driven approaches, where restricting the automatic detection to a small set of basic emotions would fall short to achieve the objective.

This work is motivated by the hypothesis that symbolic artificial intelligence techniques may provide a partial solution to the problem of early and timely identification—directly from textual messages—of situations characterized by aggressive behaviors, expressions of anger, and patterns of coercion or dominance, thereby enabling an automated system to issue alerts. The increasing volume of textual conversations has rendered manual analysis impractical in contexts such as content moderation and forensic investigation. Although automated systems offer substantial processing speed, the primary challenge lies in ensuring the reliability and transparency of emotion recognition.

In [19, 20], the authors introduce a system for emotion attribution and recommendation, employing a white box approach to emotion classification based on the human-like conceptual combination framework proposed in the $\mathbf{T}^{\mathrm{CL}}$ logic [21, 22]. In this work, we adopt a similar approach in order to perform emotion attribution to messages extracted from a chat by exploiting the ontology of the Plutchik emotion model introduced in [19], with the main aim of detecting the adoption of an aggressive language. This work is inspired by the tool COGDELEMOS [23], whose aim was the automatic attribution of emotions to speeches and its application to astronauts' speeches from the NASA Mission Transcript Collection. The objective of COGDELEMOS is to provide an intelligent system for monitoring the psycho-physical condition of an astronaut during an aerospace mission.

Here we introduce a tool called SERC– a Symbolic AI System for Emotion Recognition in Conversation – , which refers to the Plutchik's emotion Ontology and to the NRC Emotion Intensity Lexicon [12], that provides a list of English words, each with real-values representing intensity scores for the eight basic emotions of Plutchik's theory, namely: *Joy*, *Trust*, *Fear*, *Surprise*, *Sadness*, *Disgust*, *Anger*, *Anticipation* by exploiting the logic $\mathbf{T}^{\mathrm{CL}}$. In $\mathbf{T}^{\mathrm{CL}}$, "typical" properties can be directly specified by means of a "typicality" operator $\mathbf{T}$ enriching the underlying Description Logic (from now on, DL for short), and a TBox can contain inclusions of the form $\mathbf{T}(C) \sqsubseteq D$ to represent that "typical Cs are also Ds". As a difference with standard DLs, in the logic $\mathbf{T}^{\mathrm{CL}}$ one can consistently express exceptions and reason about defeasible inheritance as well. Typicality inclusions are also equipped with a real number $p \in (0.5, 1]$ representing the probability/degree of belief in such a typical property: this allows us to define a semantics inspired to the DISPONTE semantics [24] characterizing probabilistic extensions of DLs, which in turn is used in order to describe different *scenarios* where only some typicality properties are considered. The generation process building the knowledge base of emotions in the $\mathbf{T}^{\mathrm{CL}}$ logic unfolds in two main steps. First, the system constructs a prototypical description of basic emotions using the logic $\mathbf{T}^{\mathrm{CL}}$, with typicality inclusions of the form

$$\mathbf{T}(BasicEmotion) \sqsubseteq Property.$$

The basic emotions are the eight defined in Plutchik's model. The logic $\mathbf{T}^{\mathrm{CL}}$ is also able to deal with *concept combination*, whose basic idea is as follows: given a KB and two concepts C_H (HEAD) and C_M

(MODIFIER) occurring in it, we can then dynamically extend KB by providing a prototypical definition of combined concept $C \sqsubseteq C_H \sqcap C_M$, that we exploit in order to provide an ontological description of complex concepts. In this respect, we obtain a prototypical descriptions of pairs of basic emotions, generating the prototypical description of compound emotions, for instance *Guilt* is obtained by means of the combination of the basic emotions Joy and $Fear$.

In the second step, the system analyzes a message to assign the most appropriate emotion. The message is first processed by a tokenizer, which extracts relevant words. The system then checks whether the message contains terms associated with basic emotions. If such terms are found, their individual scores are retrieved and summed for each of the eight basic emotions. This results in a score distribution that indicates the presence of specific emotions. A similar method is used to detect combined emotions: the system searches for terms derived from the combination of two basic emotions, and aggregates their scores accordingly.

The novelties introduced by the SERC system in the landscape of systems for emotion assignment to short texts are, on the one hand, that existing systems only refer to the eight basic emotions of the Plutchick's proposal for efficiency reasons, whereas the SERC system exploits the $\mathbf{T}^{\text{CL}}$ logic and is therefore able to handle complex emotions as well; on the other hand, existing systems are subsymbolic systems based on neural networks and, as such, are not able to provide adequate explanations for the emotions identified in order to label a single text message; the SERC system, instead, is able to exploit the $\mathbf{T}^{\text{CL}}$ logic to construct suitable explanations of the inferred properties and of the selection of the emotion to be assigned, based on the number (and probabilities) of properties identified in the text, corresponding to the properties that the logic attributes to the emotions themselves.

In order to provide a case study, we have applied the tool SERC to messages extracted from conversations between the 22-year-old student Giulia Cecchettin and her ex-boyfriend, Filippo Turetta, who later murdered her. The case gained significant media attention in 2023 and brought public awareness to aggressive behaviors that, too often, culminate in femicide. Last, we provide an evaluation of the proposed approach by comparing its results with those of DailyDialog, a dataset of conversations in which each message is associated with an emotionally grounded annotation obtained through manual labeling. The results provided by SERC are promising and witness the potential of a symbolic approach to emotion recognition in text, which ensures transparency and explainability in the assessment process.

2. $\mathbf{T}^{\text{CL}}$: a Typicality Description Logic for Concept Combination

The automatic generation of novel concepts within a knowledge base (also known as *knowledge invention* process) can be obtained, as happens in humans [22, 21], by exploiting a process of commonsense conceptual combination. Such ability is associated to creative thinking and problem solving, however, it still represents an open challenge in Artificial Intelligence [25]. Dealing with this problem, indeed, requires, from an AI perspective, the harmonization of two conflicting requirements that are hardly accommodated in symbolic systems [26]: the need for a syntactic and semantic compositionality (typical of logical systems) and the one concerning the exhibition of typicality effects. According to a well-known argument [27], in fact, prototypes (i.e. commonsense conceptual representations based on typical properties) are not compositional. The argument runs as follows: consider a concept like *pet fish*. It results from the composition of the concept *pet* and of the concept *fish*. However, the prototype of *pet fish* cannot result from the composition of the prototypes of a pet and a fish: e.g., a typical pet is furry and warm, a typical fish is grayish, but a typical pet fish is neither furry and warm nor grayish (typically, it is red). The *pet fish* phenomenon is a paradigmatic example of the difficulty to address when building formalisms and systems trying to imitate this combinatorial human ability.

In this work, we exploit the nonmonotonic Description Logic $\mathbf{T}^{\text{CL}}$ (typicality-based compositional logic), introduced in [21, 22], which is able to account for this type of human-like concept combination. Other works have already shown how such logic can be used to model complex cognitive phenomena [22], goal-directed creative problem solving [28, 29, 30] and to build intelligent applications for computational creativity [31, 32]. In particular, we show how it can be used as a tool for the generation of novel compound emotions and, as a consequence, for the suggestion of novel emotion-related contents. In $\mathbf{T}^{\text{CL}}$, "typical" properties can be directly specified by means of a "typicality" operator $\mathbf{T}$ enriching the underlying DL, namely a TBox can contain inclusions of the form $\mathbf{T}(C) \sqsubseteq D$ to represent that "typical Cs are also Ds". As a difference with standard DLs, in the logic $\mathbf{T}^{\text{CL}}$ one can consistently express exceptions and reason about defeasible inheritance as well. Typicality inclusions are also equipped by a real number $p \in (0.5, 1]$ representing the probability/degree of belief in such a typical property: this allows us to define a semantics inspired to the DISPONTE semantics [24] characterizing probabilistic extensions of DLs, which in turn is used in order to describe different *scenarios* where only some typicality properties are considered. Given a KB containing the description of two concepts C_H and C_M occurring in it, we then consider only *some* scenarios in order to define a revised knowledge base, enriched by typical properties of the combined concept $C \sqsubseteq C_H \sqcap C_M$ by also implementing a heuristics coming from the cognitive semantics.

The logic $\mathbf{T}^{\text{CL}}$ [22] combines three main ingredients. The first one relies on the DL of typicality $\mathcal{ALC} + \mathbf{T_R}$ introduced in [33], which allows to describe the *prototype* of a concept. In this logic, "typical" properties can be directly specified by means of a "typicality" operator $\mathbf{T}$ enriching the underlying DL, and a TBox can contain inclusions of the form $\mathbf{T}(C) \sqsubseteq D$ to represent that "typical Cs are also Ds". As a difference with standard DLs, in the logic $\mathcal{ALC} + \mathbf{T_R}$ one can consistently express exceptions and reason about defeasible inheritance as well. For instance, a knowledge base can consistently express that "normally, singers can sing", whereas "trappers usually cannot sing" by $\mathbf{T}(\mathit{Singer}) \sqsubseteq \mathit{CanSing}$ and $\mathbf{T}(\mathit{Trapper}) \sqsubseteq \neg \mathit{CanSing}$, given that $\mathit{Trapper} \sqsubseteq \mathit{Singer}$. The semantics of the $\mathbf{T}$ operator is characterized by the properties of *rational logic* [34], recognized as the core properties of nonmonotonic reasoning. $\mathcal{ALC} + \mathbf{T_R}$ is characterized by a minimal model semantics corresponding to an extension to DLs of a notion of *rational closure* as defined in [34] for propositional logic: the idea is to adopt a preference relation over $\mathcal{ALC} + \mathbf{T_R}$ models, where intuitively a model is preferred to another one if it contains less exceptional elements, as well as a notion of *minimal entailment* restricted to models that are minimal with respect to such preference relation. As a consequence, $\mathbf{T}$ inherits well-established properties like *specificity* and *irrelevance*: in the example, the logic $\mathcal{ALC} + \mathbf{T_R}$ allows us to infer $\mathbf{T}(\mathit{Singer} \sqcap \mathit{LongHair}) \sqsubseteq \mathit{CanSing}$ (having long hair is irrelevant with respect to being able to sing or not) and, if one knows that Tony is a typical trapper, to infer that he cannot sing, giving preference to the most specific information.

A second ingredient consists of a distributed semantics similar to the one of probabilistic DLs known as DISPONTE [35], which allows labeling inclusions $\mathbf{T}(C) \sqsubseteq D$ with a real number between 0.5 and 1, which represents its degree of belief/probability, under the assumption that each axiom is independent from each other. Degrees of belief in typicality inclusions allow defining a probability distribution over *scenarios*: roughly speaking, a scenario is obtained by choosing, for each typicality inclusion, whether it is considered as true or false. In a slight extension of the above example, we could have the need to represent that both the typicality inclusions about singers and trappers have a degree of belief of 80%, whereas we also believe that, normally, singers are also able to play an instrument with a higher degree of 95%, with the following KB:

(1) $\mathit{Trapper} \sqsubseteq \mathit{Singer}$
(2) $0.8 \;::\; \mathbf{T}(\mathit{Singer}) \sqsubseteq \mathit{CanSing}$
(3) $0.8 \;::\; \mathbf{T}(\mathit{Trapper}) \sqsubseteq \neg \mathit{CanSing}$
(4) $0.95 \;::\; \mathbf{T}(\mathit{Singer}) \sqsubseteq \mathit{PlayInstrument}$

In this case, we consider eight different scenarios, representing all possible combinations of typicality inclusion: as an example, $\{((2), 1), ((3), 0), ((4), 1)\}$ represents the scenario in which (2) and (4) hold, whereas (3) does not. Obviously, (1) holds in every scenario, since it represents a rigid property, not admitting exceptions. We equip each scenario with a probability depending on those of the involved inclusions: the scenario of the example has probability 0.8×0.95 (since 2 and 4 are involved) $\times(1-0.8)$ (since 3 is not involved) $= 0.152 = 15.2\%$. Such probabilities are then taken into account in order to choose the most adequate scenario describing the prototype of the combined concept.

As a third element of the proposed formalization is a method inspired by cognitive semantics [36] for the identification of a dominance effect between the concepts to be combined: for every combination, we distinguish a HEAD, representing the stronger element of the combination, and a MODIFIER. The basic idea is: given a KB and two concepts C_H (HEAD) and C_M (MODIFIER) occurring in it, we consider only *some* scenarios in order to define a revised knowledge base, enriched by typical properties of the combined concept $C \sqsubseteq C_H \sqcap C_M$.

The language of $\mathbf{T}^{\text{CL}}$ extends the basic DL $\mathcal{ALC}$ by *typicality inclusions* of the form $\mathbf{T}(C) \sqsubseteq D$ equipped by a real number $p \in (0.5, 1]$, representing its degree of belief, whose meaning is that "we believe with degree/probability p that, normally, Cs are also Ds" [1]

Definition 2.1 (Language of $\mathbf{T}^{\text{CL}}$). *We consider an alphabet of concept names* $\mathtt{C}$*, of role names* $\mathtt{R}$*, and of individual constants* $\mathtt{O}$*. Given* $A \in \mathtt{C}$ *and* $R \in \mathtt{R}$*, we define:*

$$C, D := A \mid \top \mid \bot \mid \neg C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C$$

We define a knowledge base $\mathcal{K} = \langle \mathcal{R}, \mathcal{T}, \mathcal{A} \rangle$ *where: (i)* $\mathcal{R}$ *is a finite set of rigid properties of the form* $C \sqsubseteq D$*; (ii)* $\mathcal{T}$ *is a finite set of typicality properties of the form* $p \ :: \ \mathbf{T}(C) \sqsubseteq D$ *where* $p \in (0.5, 1] \subseteq \mathbb{R}$ *is the degree of belief of the typicality inclusion; (iii)* $\mathcal{A}$ *is the ABox, i.e. a finite set of formulas of the form either* $C(a)$ *or* $R(a, b)$*, where* $a, b \in \mathtt{O}$ *and* $R \in \mathtt{R}$*.*

A model $\mathcal{M}$ in $\mathbf{T}^{\text{CL}}$ extends standard $\mathcal{ALC}$ ones by a preference relation among domain elements as in the DL of typicality [33]. In this respect, $x < y$ means that x is "more normal" than y, and that typical members of a concept C are the minimal elements of C with respect to $<$. An element $x \in \Delta^{\mathcal{I}}$ is a *typical instance* of C if $x \in C^{\mathcal{I}}$ and there is no C-element in $\Delta^{\mathcal{I}}$ *more normal* than x. Formally:

Definition 2.2 (Model of $\mathbf{T}^{\text{CL}}$). *A model* $\mathcal{M}$ *is any structure* $\langle \Delta^{\mathcal{I}}, <, .^{\mathcal{I}} \rangle$ *where: (i)* $\Delta^{\mathcal{I}}$ *is a non empty set of items called the domain; (ii)* $<$ *is an irreflexive, transitive, well-founded and modular (for all* x, y, z *in* $\Delta^{\mathcal{I}}$*, if* $x < y$ *then either* $x < z$ *or* $z < y$*) relation over* $\Delta^{\mathcal{I}}$*; (iii)* $.^{\mathcal{I}}$ *is the extension function that maps each atomic concept* C *to* $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$*, and each role* R *to* $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$*, and is extended to complex concepts as follows:* $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$*;* $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$*;* $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$*;* $(\exists R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \exists (x, y) \in R^{\mathcal{I}} \text{ such that } y \in C^{\mathcal{I}}\}$*;* $(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall (x, y) \in R^{\mathcal{I}} \text{ we have } y \in C^{\mathcal{I}}\}$*;* $(\mathbf{T}(C))^{\mathcal{I}} = Min_{<}(C^{\mathcal{I}})$*, where* $Min_{<}(C^{\mathcal{I}}) = \{x \in C^{\mathcal{I}} \mid \nexists y \in C^{\mathcal{I}} \text{ s.t. } y < x\}$*.*

A model $\mathcal{M}$ can be equivalently defined by postulating the existence of a function $k_{\mathcal{M}} : \Delta^{\mathcal{I}} \longmapsto \mathbb{N}$, where $k_{\mathcal{M}}$ assigns a finite rank to each domain element [33]: the rank of x is the length of the longest chain $x_0 < \cdots < x$ from x to a minimal x_0, i.e. such that there is no x' such that $x' < x_0$. The rank function $k_{\mathcal{M}}$ and $<$ can be defined from each other by letting $x < y$ if and only if $k_{\mathcal{M}}(x) < k_{\mathcal{M}}(y)$.

Definition 2.3. *Let* $\mathcal{K} = \langle \mathcal{R}, \mathcal{T}, \mathcal{A} \rangle$ *be a KB. Given a model* $\mathcal{M} = \langle \Delta^{\mathcal{I}}, <, .^{\mathcal{I}} \rangle$*, we assume that* $.^{\mathcal{I}}$ *is extended to assign a domain element* $a^{\mathcal{I}}$ *of* $\Delta^{\mathcal{I}}$ *to each individual constant* a *of* $\mathtt{O}$*. We say that: (i)* $\mathcal{M}$ *satisfies* $\mathcal{R}$ *if, for all* $C \sqsubseteq D \in \mathcal{R}$*, we have* $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$*; (ii)* $\mathcal{M}$ *satisfies* $\mathcal{T}$ *if, for all* $q \ :: \ \mathbf{T}(C) \sqsubseteq D \in \mathcal{T}$*, we have that* [2] $\mathbf{T}(C)^{\mathcal{I}} \subseteq D^{\mathcal{I}}$*, i.e.* $Min_{<}(C^{\mathcal{I}}) \subseteq D^{\mathcal{I}}$*; (iii)* $\mathcal{M}$ *satisfies* $\mathcal{A}$ *if, for each assertion* $F \in \mathcal{A}$*, if* $F = C(a)$ *then* $a^{\mathcal{I}} \in C^{\mathcal{I}}$*, otherwise if* $F = R(a, b)$ *then* $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$*.*

Even if the typicality operator $\mathbf{T}$ itself is nonmonotonic (i.e. $\mathbf{T}(C) \sqsubseteq E$ does not imply $\mathbf{T}(C \sqcap D) \sqsubseteq E$), what is inferred from a KB can still be inferred from any KB' with KB $\subseteq$ KB', i.e. the resulting logic is monotonic. As already mentioned, in order to perform useful nonmonotonic inferences, in [33] the

[1]The reason why we only allow typicality inclusions equipped with probabilities $p > 0.5$ is due to our effort of integrating two different semantics: typicality based logic and DISPONTE. In particular, as detailed in [22] this choice seems to be the only one compliant with both formalisms. On the contrary, it would be misleading to also allow low degrees of belief for typicality inclusions, since typical knowledge is known to come with a low degree of uncertainty.

[2]It is worth noticing that here the degree q does not play any role. Indeed, a typicality inclusion $\mathbf{T}(C) \sqsubseteq D$ holds in a model only if it satisfies the semantic condition of the underlying DL of typicality, i.e. minimal (typical) elements of C are elements of D. The degree of belief q will have a crucial role in the application of the distributed semantics, allowing the definition of scenarios as well as the computation of their probabilities.

authors have strengthened the above semantics by restricting entailment to a class of minimal models. Intuitively, the idea is to restrict entailment to models that *minimize the atypical instances of a concept.* The resulting logic corresponds to a notion of *rational closure* on top of $\mathcal{ALC} + \mathbf{T_R}$. Such a notion is a natural extension of the rational closure construction provided in [34] for the propositional logic. This nonmonotonic semantics relies on minimal rational models that minimize the *rank of domain elements.* Informally, given two models of KB, one in which a given domain element x has rank 2 (because for instance $z < y < x$), and another in which it has rank 1 (because only $y < x$), we prefer the latter, as in this model the element x is assumed to be "more typical" than in the former. Query entailment is then restricted to minimal *canonical models.* The intuition is that a canonical model contains all the individuals that enjoy properties that are consistent with KB. This is needed when reasoning about the rank of the concepts: it is important to have them all represented.

Given a KB $\mathcal{K} = \langle \mathcal{R}, \mathcal{T}, \mathcal{A} \rangle$ and given two concepts C_H and C_M occurring in $\mathcal{K}$, the logic $\mathbf{T}^{\text{CL}}$ allows defining a prototype of the combined concept C as the combination of the HEAD C_H and the MODIFIER C_M, where the typical properties of the form $\mathbf{T}(C) \sqsubseteq D$ (or, equivalently, $\mathbf{T}(C_H \sqcap C_M) \sqsubseteq D$) to be ascribed to the concept C are obtained by considering blocks of scenarios with the same probability, in decreasing order starting from the highest one. We first discard all the inconsistent scenarios, then:

- we discard those scenarios considered as *trivial*, consistently inheriting all the properties from the HEAD from the starting concepts to be combined. This choice is motivated by the challenges provided by task of commonsense conceptual combination itself: in order to generate plausible and creative compounds, it is necessary to maintain a level of surprise in the combination. Thus both scenarios inheriting all the properties of the two concepts and all the properties of the HEAD are discarded, since they prevent this surprise;
- among the remaining ones, we discard those inheriting properties from the MODIFIER which are in conflict with properties that could be consistently inherited from the HEAD;
- if the set of scenarios of the current block is empty, i.e. all the scenarios have been discarded either because trivial or because the MODIFIER is preferred, we repeat the procedure by considering the block of scenarios having the immediately lower probability.

Remaining scenarios are those selected by the logic $\mathbf{T}^{\text{CL}}$. The ultimate output of our mechanism is a knowledge base in the logic $\mathbf{T}^{\text{CL}}$ whose set of typicality properties is enriched by those of the compound concept C. Given a scenario w satisfying the above properties, we define the properties of C as the set of inclusions $p \ :: \ \mathbf{T}(C) \sqsubseteq D$, for all $\mathbf{T}(C) \sqsubseteq D$ that are entailed from w in the logic $\mathbf{T}^{\text{CL}}$. The probability p is such that:

- if $\mathbf{T}(C_H) \sqsubseteq D$ is entailed from w, that is to say D is a property inherited either from the HEAD (or from both the HEAD and the MODIFIER), then p corresponds to the degree of belief of such inclusion of the HEAD in the initial knowledge base, i.e. $p \ : \ \mathbf{T}(C_H) \sqsubseteq D \in \mathcal{T}$;
- otherwise, i.e. $\mathbf{T}(C_M) \sqsubseteq D$ is entailed from w, then p corresponds to the degree of belief of such inclusion of a MODIFIER in the initial knowledge base, i.e. $p \ : \ \mathbf{T}(C_M) \sqsubseteq D \in \mathcal{T}$.

The knowledge base obtained as the result of combining concepts C_H and C_M into the compound concept C is called C*-revised* knowledge base, and it is defined as follows:

$$\mathcal{K}_C = \langle \mathcal{R}, \mathcal{T} \cup \{p \ : \ \mathbf{T}(C) \sqsubseteq D\}, \mathcal{A} \rangle,$$

for all D such that either $\mathbf{T}(C_H) \sqsubseteq D$ is entailed in w or $\mathbf{T}(C_M) \sqsubseteq D$ is entailed in w, and p is defined as above. As an example, consider the following version of the above mentioned *Pet-Fish* problem. Let KB contain the following inclusions:

$$\mathit{Fish} \sqsubseteq \mathit{LivesInWater} \quad (1)$$
$$0.6 \ :: \ \mathbf{T}(\mathit{Fish}) \sqsubseteq \mathit{Greyish} \quad (2)$$
$$0.8 \ :: \ \mathbf{T}(\mathit{Fish}) \sqsubseteq \mathit{Scaly} \quad (3)$$
$$0.8 \ :: \ \mathbf{T}(\mathit{Fish}) \sqsubseteq \neg \mathit{Affectionate} \quad (4)$$

$$0.9 :: \mathbf{T}(Pet) \sqsubseteq \neg LivesInWater \quad (5)$$
$$0.9 :: \mathbf{T}(Pet) \sqsubseteq LovedByKids \quad (6)$$
$$0.9 :: \mathbf{T}(Pet) \sqsubseteq Affectionate \quad (7)$$

representing that a typical fish is greyish (2), scaly (3) and not affectionate (4), whereas a typical pet does not live in water (5), is loved by kids (6) and is affectionate (7). Concerning rigid properties, we have that all fishes live in water (1). The logic $\mathbf{T}^{\mathrm{CL}}$ combines the concepts Pet and $Fish$, by using the latter as the HEAD and the former as the MODIFIER. The prototypical Pet-Fish inherits from the prototypical fish the fact that it is scaly and not affectionate, the last one by giving preference to the HEAD since such a property conflicts with the opposite one in the modifier (a typical pet is affectionate). The scenarios in which all the three typical properties of a typical fish are inherited by the combined concept are considered as trivial and, therefore, discarded, as a consequence the property having the lowest degree ($Greyish$ with degree 0.6) is not inherited. The prototypical Pet-Fish inherits from the prototypical pet only property (6), since (5) conflicts with the rigid property (1), stating that all fishes (then, also pet fishes) live in water, whereas (7) is blocked, as already mentioned, by the HEAD/MODIFIER heuristics. Formally, the $Pet \sqcap Fish$-revised knowledge base contains, in addition to the above inclusions, the following ones:

$$0.8 :: \mathbf{T}(Pet \sqcap Fish) \sqsubseteq Scaly \quad (3')$$
$$0.8 :: \mathbf{T}(Pet \sqcap Fish) \sqsubseteq \neg Affectionate \quad (4')$$
$$0.9 :: \mathbf{T}(Pet \sqcap Fish) \sqsubseteq LovedByKids \quad (6')$$

In [22] it has been also shown that reasoning in $\mathbf{T}^{\mathrm{CL}}$ remains in the same complexity class of standard $\mathcal{ALC}$ Description Logics.

3. An Ontological Description of Emotions

In this work, we have applied the $\mathbf{T}^{\mathrm{CL}}$ reasoning framework to the generation of new compound emotions by starting from the affective ontological knowledge base named ArsEmotica. Intuitively, given the ArsEmotica knowledge base equipped with the prototypical descriptions of basic emotions, we exploit the reasoning capabilities of the logic $\mathbf{T}^{\mathrm{CL}}$ in order to generate new *derived* emotions as the result of the creative combination of two (or even more) basic or derived ones. Moreover, an item of the tested dataset belongs to the new generated emotion if it contains all the rigid properties as well as at least 30% of the typical properties of such a derived emotion.

More in detail, the affective knowledge leveraged by the $\mathbf{T}^{\mathrm{CL}}$ logic is encoded in an ontology of emotional categories based on Plutchik's psychological circumplex model [4], called ArsEmotica[3] and includes also concepts from the Hourglass model [37]. The ontology structures emotional categories in a taxonomy, which currently includes 32 emotional concepts. The design of the taxonomic structure of emotional categories, of the disjunction axioms and of the object and data properties mirrors the main features of Plutchik model. As already mentioned, such model can be represented as a wheel of emotions (see Figure 1) and encodes the following elements: (i) basic or primary emotions: *Joy*, *Trust*, *Fear*, *Surprise*, *Sadness*, *Disgust*, *Anger*, *Anticipation*; in the color wheel, this is represented by differently colored sectors; (ii) opposites: basic emotions can be conceptualized in terms of polar opposites: *Joy* versus *Sadness*, *Anger* versus *Fear*, *Trust* versus *Disgust*, *Surprise* versus *Anticipation*; (iii) Intensity: each emotion can exist in varying degrees of intensity; in the wheel, this is represented by the vertical dimension; (iv) Similarity: emotions vary in their degree of similarity to one another; in the wheel, this is represented by the radial dimension; (v) Complex emotions: a complex emotion is a composition of two basic emotions; the pair of basic emotions involved in the composition is called a dyad. Looking at the Plutchik wheel, the eight emotions in the blank spaces are compositions of similar basic emotions, called primary dyads. Pairs of less similar emotions are called secondary dyads (if the radial distance between them is 2) or tertiary dyads (if the distance is 3), while opposites cannot be combined.

[3]The ArsEmotica ontology is available here: http://130.192.212.225/fuseki/ArsEmotica-core and queryable via SPARQL endpoint at: http://130.192.212.225/fuseki/dataset.html?tab=query&ds=/ArsEmotica-core

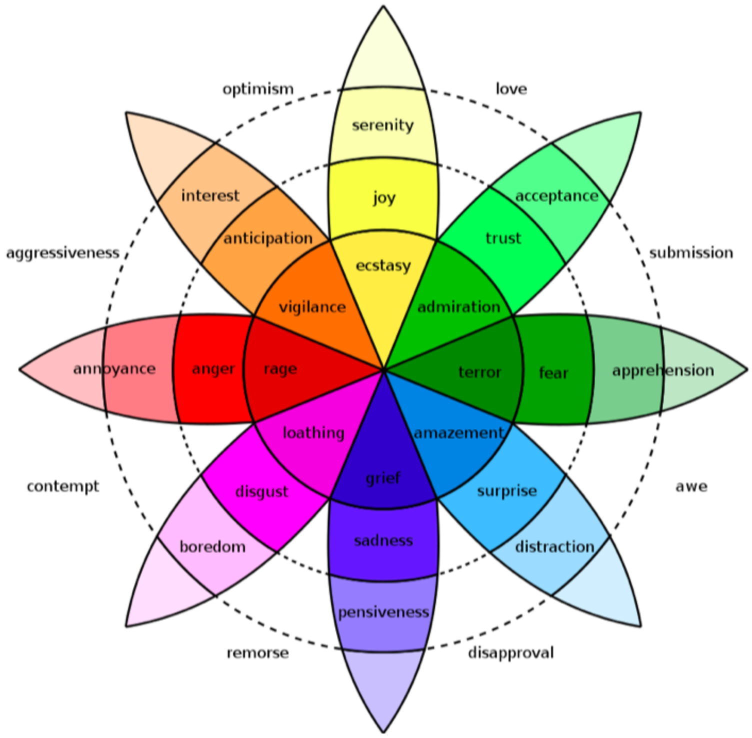

Figure 1: The Wheel of Emotion of the Plutchik Model

We have chosen to encode the Plutchik model in the ontology for several reasons: (i) it is well-grounded in psychology and general enough to guarantee a wide coverage of emotions; (ii) the Plutchik wheel of emotions is perfectly compliant with the generative model underlying the $\mathbf{T}^{\text{CL}}$ logic; (iii) it encodes interesting notions, e.g. emotional polar opposites, which can be exploited for finding novel, non obvious relations among contents.

Within the ArsEmotica ontology, the class Emotion is the root for all the emotional concepts. The Emotions hierarchy includes all the 32 emotional categories presented as distinguished labels in the model. In particular, the Emotion class has two disjoint subclasses: BasicEmotion and ComplexEmotion. Basic emotions of the Plutchik model are direct sub-classes of BasicEmotion. Each of them is specialized again into two subclasses representing the same emotion with weaker or stronger intensity (e.g. the basic emotion *Joy* has *Ecstasy* and *Serenity* as sub-classes). Therefore, we have 24 emotional concepts subsumed by the BasicEmotion concept. Instead, the class CompositeEmotion has 24 subclasses, corresponding to the primary (*Love, Submission, Awe, Disapproval, Remorse, Contempt, Aggressiveness and Optimism*), secondary (*Hope, Guilt, Curiosity, Despair, Unbelief, Envy, Cynicism and Pride*) and tertiary (*Anxiety, Delight, Sentimentality, Shame, Outrage, Pessimism, Morbidness, Dominance*) dyads. Other relations in the Plutchik model have been expressed in the ontology by means of object properties: the hasOpposite property encodes the notion of polar opposites; the hasSibling property encodes the notion of similarity and the isComposedOf property encodes the notion of composition of basic emotions. Moreover, a data type property hasScore was introduced to link each emotion with an intensity value mapped into the Hourglass model.

The devised model allows inferring complex emotions from basic ones by exploiting simple SWRL rules (i.e. if-then clauses) allowing to infer, from the isComposedOf property connecting Basic and Composite Emotions, the fact that if an agent feels two emotions (suppose for a given item), and if these emotions jointly constitute a Composite Emotion, then the latter emotion will be automatically assigned to the agent in order to better describe his/her aesthetic experience.

Due to the need of modeling the links between words in a language and the emotions they refer to, the ArsEmotica Ontology is also integrated with the ontology framework LExicon Model for ONtologies (LEMON) [38]. In particular, such integration allows differentiating explicitly between the language level (lexicon-based) and the conceptual one in representing the emotional concepts [39]. Within this enriched framework, it is possible to associate a plethora of emotional words, with the encoding of language information, to the corresponding emotional concepts. In this work, we have used the ArsEmotica model of emotional concepts with the NRC Emotion Intensity Lexicon mentioned above [12]. Such lexicon provides a list of English words, each with real-values representing intensity scores for the eight basic emotions of Plutchik's theory. The lexicon includes close to $10,000$ words including terms already known to be associated with emotions as well as terms that co-occur in Twitter posts that convey emotions. The intensity scores were obtained via crowdsourcing, using best-worst scaling annotation scheme. For our purposes, we considered the most frequent terms available in such lexicon (and associated to the basic emotions of the Plutchik wheel) as typical features of such emotions. In this way, once the prototypes of the basic emotional concepts were formed, the $\mathbf{T}^{\text{CL}}$ reasoning framework was used to generate the compound emotions.

Let us now provide some details about the construction of the ontology of emotions in $\mathbf{T}^{\text{CL}}$. As in [19], prototypes generation proceeds in two steps: in the first one, the system builds a prototypical description of basic emotions in the language of the logic $\mathbf{T}^{\text{CL}}$, in order to describe their typical properties. In detail, a knowledge base in the logic $\mathbf{T}^{\text{CL}}$ characterized by typicality inclusions of the form

$$p \ :: \ \mathbf{T}(\mathit{BasicEmotion}) \sqsubseteq \mathit{Property}$$

where $\mathit{BasicEmotion}$ is one of the eight basic emotions of the Plutchik model: Joy, Trust, Fear, Surprise, Sadness, Disgust, Anger, and Anticipation. As an example, consider the basic emotion Joy. The words having the highest scores are happiness (0.98), bliss (0.97), to celebrate (0.97), jubilant (0.97), ecstatic (0.95), and euphoria (0.94). Therefore, the knowledge base generated will contain, among others, the following inclusions:

$$\begin{array}{ll}
\mathit{Joy} \sqsubseteq \neg \mathit{Holocaust} & \\
0.98 \ :: \ \mathbf{T}(\mathit{Joy}) \sqsubseteq \mathit{Happiness} & 0.97 \ :: \ \mathbf{T}(\mathit{Joy}) \sqsubseteq \mathit{Bliss} \\
0.97 \ :: \ \mathbf{T}(\mathit{Joy}) \sqsubseteq \mathit{Celebrating} & 0.97 \ :: \ \mathbf{T}(\mathit{Joy}) \sqsubseteq \mathit{Jubilant} \\
0.95 \ :: \ \mathbf{T}(\mathit{Joy}) \sqsubseteq \mathit{Ecstatic} & 0.94 \ :: \ \mathbf{T}(\mathit{Joy}) \sqsubseteq \mathit{Elation}
\end{array}$$

As a second step, the system exploits the above described reasoning mechanism of such a Description Logic in order to combine the prototypical descriptions of pairs of basic emotions, generating the prototypical description of compound emotions by using the same logical procedure of the *pet-fish* problem. As an example, let us consider the combination of the above basic emotion Joy with Fear, whose prototypical description is as follows:

$$\begin{array}{ll}
0.96 \ :: \ \mathbf{T}(\mathit{Fear}) \sqsubseteq \mathit{Kill} & 0.95 \ :: \ \mathbf{T}(\mathit{Fear}) \sqsubseteq \mathit{Annihilate} \\
0.95 \ :: \ \mathbf{T}(\mathit{Fear}) \sqsubseteq \mathit{Terror} & 0.98 \ :: \ \mathbf{T}(\mathit{Fear}) \sqsubseteq \mathit{Torture} \\
0.97 \ :: \ \mathbf{T}(\mathit{Fear}) \sqsubseteq \mathit{Terrorist} & 0.97 \ :: \ \mathbf{T}(\mathit{Fear}) \sqsubseteq \mathit{Horrific}
\end{array}$$

In order to obtain a description of the compound emotion Guilt as the result of the combination of the two basic emotions ($\mathit{Joy} \sqcap \mathit{Fear}$) in the logic $\mathbf{T}^{\text{CL}}$, the system combines the two basic emotions by implementing a variant of CoCoS [40], a Python implementation of reasoning services for the logic $\mathbf{T}^{\text{CL}}$ in order to exploit efficient DLs reasoners for checking both the consistency of each generated scenario and the existence of conflicts among properties, following the line of the system DENOTER [41] and DEGARI [19]. CoCoS generates scenarios and chooses the selected one(s) by exploiting the translation of an $\mathcal{ALC} + \mathbf{T_R}$ knowledge base into standard $\mathcal{ALC}$ introduced in [33] and adopted by the system RAT-OWL [42]. CoCoS makes use of the above mentioned library `owlready2`[4], which allows relying on the services of efficient DL reasoners, e.g. the HermiT reasoner.

CoCoS is embedded in DEGARI and allows one: i) to include the logical descriptions of the concepts to be combined; ii) to select which among the concepts has to be intended as HEAD and as MODIFIER(s); iii) to choose how many typical properties one wants to inherit in the scenarios that will be selected by

[4]https://pythonhosted.org/Owlready2/

$\mathbf{T}^{\text{CL}}$. In addition to presenting the selected scenario with typical properties of the combined concept, CoCoS also allows the users to select alternative scenarios, ranging from more trivial to more surprising ones. More in detail, the system considers both the available choices for the HEAD and the MODIFIER, and it allows restricting its concern to a given and fixed number of inherited properties. The combined emotion Guilt has the following $\mathbf{T}^{\text{CL}}$ description (concept $\mathit{Joy} \sqcap \mathit{Fear}$):

$$
\begin{array}{ll}
0.98 :: \mathbf{T}(\mathit{Joy} \sqcap \mathit{Fear}) \sqsubseteq \mathit{Happiness} & 0.97 :: \mathbf{T}(\mathit{Joy} \sqcap \mathit{Fear}) \sqsubseteq \mathit{Celebrating} \\
0.97 :: \mathbf{T}(\mathit{Joy} \sqcap \mathit{Fear}) \sqsubseteq \mathit{Bliss} & 0.98 :: \mathbf{T}(\mathit{Joy} \sqcap \mathit{Fear}) \sqsubseteq \mathit{Torture} \\
0.97 :: \mathbf{T}(\mathit{Joy} \sqcap \mathit{Fear}) \sqsubseteq \mathit{Terrorist} & 0.97 :: \mathbf{T}(\mathit{Joy} \sqcap \mathit{Fear}) \sqsubseteq \mathit{Horrific}
\end{array}
$$

Obviously, rigid properties of basic emotions (if any) are inherited by the compound emotion (in the example, $\mathit{Joy} \sqcap \mathit{Fear} \sqsubseteq \neg \mathit{Holocaust}$), and this retains the system from considering any inconsistent typical properties even if they have the highest probability.

It is worth noticing that the properties of the derived emotion are still expressed in the language of the logic $\mathbf{T}^{\text{CL}}$, therefore the combined emotion, Guilt in the example, can be further combined with another emotion, in order to iterate the procedure.

4. The Tool SERC

We describe the system SERC exploiting the logic $\mathbf{T}^{\text{CL}}$ in order to detect anger and aggressive language in conversations. The system analyzes each message to identify the dominant emotion – whether basic or complex – that characterizes it. Moreover, SERC then considers all messages from an individual participant in a conversation, as well as the conversation as a whole. The tool is implemented in Python and it is available for free download, together with all the data used for validation and for the case study, at https://github.com/AleCi00/SERC. We have also implemented EmotionChat, an interactive web application that allows us to show the potential use of SERC in an active chatroom for real-time analysis, illustrating its applicability to instant messaging rather than solely post hoc on transcripts.

In order to detect the most adequate emotion to assign to a given message, SERC computes the following steps: (i) messages are first processed by a tokenizer, whose objective is to extract relevant words. The first step is about determining whether it contains (or not) terms used to describe basic emotions and identifying them; (ii) when the system finds some of these terms in a message, it considers the score each term has paired, and it sums up all the scores for each of the eight basic emotions. This way we get indicative values concerning which basic emotions are found in each message. It is possible to implement a more suitable function rather than the sum, for example a weighted mean considering the total number of words in the message we are analyzing; (iii) a similar process is carried out in order to identify combined emotions: instead of looking for the basic emotion's terms, this time the system looks for the terms obtained from the combination of the two basic emotions, considering their scores. To enhance the sensitivity of the analysis, some refinements are implemented to handle conventions typical of text messaging, such as the use of uppercase characters, punctuation, and quotation marks.

In informal contexts, the use of emojis helps to add nuance to otherwise flat text, clarifying the tone of the communication, even more than punctuation. The SERC tool therefore takes into account the emoji classification provided by GoEmotions, a large emotion lexicon that associates 28 labels with descriptive terms, as shown in Figure 2.

As an example, let us consider the transcription of a message from the case study described in the next section. The system produces the following json:


```
{
    "speaker": "Filippo",
    "text": "Listen... stop saying \"eh wake up\" \"ridiculous\" \"you need to study more
      and better\" and all this stuff fuck... I’m in such a shitty situation that I can’t do anything,
         I’m depressed and in total solitude ... I want to kill myself ...",
    "occurrences": [
      "shitty",
```

Positive		Negative		Ambiguous
admiration 👏	joy 😃	anger 😡	grief 😢	confusion 😕
amusement 😂	love ❤️	annoyance 😒	nervousness 😬	curiosity 🤔
approval 👍	optimism 🤞	disappointment	remorse 😔	realization 💡
caring 🤗	pride 😌	disapproval 👎	sadness 😞	surprise 😲
desire 😍	relief 😅	disgust 🤮		
excitement 🤩		embarrassment 😳		
gratitude 🙏		fear 😨		

Figure 2: Words strongly related to emojis.


```
    "depressed",
    "solitude",
    "like",
    "food",
    "kill",
    "living",
    "fault"
  ],
  "word_count": 75,
  "stop_word_count": 40,
  "occurrences_count": 8,
  "anger-NRC-Emotion-Intensity-Lexicon-v1": 0.297,
  "anger-NRC-Emotion-Intensity-Lexicon-v1_computed": 0.0085,
  "fear-NRC-Emotion-Intensity-Lexicon-v1": 2.393,
  "fear-NRC-Emotion-Intensity-Lexicon-v1_computed": 0.0684,
  "joy-NRC-Emotion-Intensity-Lexicon-v1": 1.265,
  "joy-NRC-Emotion-Intensity-Lexicon-v1_computed": 0.0361,
  "sadness-NRC-Emotion-Intensity-Lexicon-v1": 3.921,
  "sadness-NRC-Emotion-Intensity-Lexicon-v1_computed": 0.112,
  "trust-NRC-Emotion-Intensity-Lexicon-v1": 1.156,
  "trust-NRC-Emotion-Intensity-Lexicon-v1_computed": 0.033,
  "anger": true,
  "fear": true,
  "sadness": true,
  "fear-joy": 1.924,
  "fear-joy_computed": 0.055,
  ...
  "joy-fear": 1.924,
  "joy-fear_computed": 0.055,
  "main_emotion": [
    [
      "sadness",
      "11%"
    ],
    [
      "fear",
      "6%"
    ],
    [
      "fear-joy",
      "5%"
    ]
  ],
  "combined_emotions": [
    {
      "emotion": "anger-fear",
      "matched_words": [
      ...
      ]
```

```
      },
      {
        "emotion": "fear-joy",
        "matched_words": [
          {
            "word": "kill",
            "score": 0.962
          }
        ]
      }
}
```

The first two fields exhibit the information extracted by the transcription under consideration, namely "speaker" and "text", then the "occurrences" field shows the relevant terms found in the text. Next, we have some word counters regarding the "text" field. After that, the system shows the score obtained for the message by each of the eight basic emotions, as well as by combined emotions, along with the matched words. In this example, the basic emotion with the highest score is *Sadness*, given the presence of words "depressed" and "solitude" in the analyzed text, however, the fact that also the score for the emotion *Fear* is significant, it follows that system SERC also suggests the compound emotion *Guilt*, obtained by the combination of *Fear* and *Joy*, whose descriptions in the Plutchik's ontology are available at the end of the previous section.

5. Validation and Case Study: the Homicide of Giulia Cecchettin

We have tested the system SERC in two different ways and the obtained preliminary results seem to be promising. On the one hand, we have tested it on the dataset[5] Daily Dialog [43], a corpus of conversations written by humans, in English, covering a range of topics (relationships, daily life, work, and school), manually annotated with emotional labels. The dataset contains a total of over 100.000 messages, each labeled with one of the following categories: anger, disgust, sadness, happiness, fear, surprise, and contempt, to which a "neutral" category is added. As a first, promising, result, we have observed that the system SERC is able to correctly identify the emotions for the 61.9% of the processed text, against the 17.3% of the system provided by Daily Dialog itself.

We have then applied the SERC system to messages extracted from conversations between the 22-year-old student Giulia Cecchettin and her ex-boyfriend, Filippo Turetta. In November 2023, Giulia was killed by her former partner, who had continued to display possessive behavior toward her even after the end of their relationship. Filippo Turetta was later sentenced to life imprisonment for intentional homicide with the aggravating circumstance of premeditation. The case gained significant media attention in 2023 and brought public awareness to aggressive behaviors that, too often, culminate in femicide. In order to witness the potential usefulness of our system SERC, consider the following messages, belonging to conversation #12: we have obtained the following outcome for some messages:

Speaker	Text	Detected emotions
Filippo	Okay, ok, but very brief complaints...two phrases maybe...I feel very bad, ok? ...	Anger (4%), Disgust (2%)
Giulia	Common sense? You come to talk to me about common sense? Pippo, I've been standing close to you for three months...There wasn't a person who agreed with me...I didn't want you to be worse and you come to talk about common sense now...are we kidding?	Sadness (4%), Fear (2%)
Filippo	I mean. You might as well tell me to kill myself	Awe: Surprise-Fear (19%)
Giulia	Do you think that by telling me you're killing yourself I'll get back with you?	Fear (15%)

[5] The dataset Daily Dialog is available on http://yanran.li/dailydialog.

The analyzed messages were collected from both (i) news reports published at the time of the events and (ii) the official sentencing document of Filippo Turetta, available on the website giurisprudenzapenale.com. The above example allows us to observe how the application of SERC to fragments of the conversations between the two individuals could have enabled the identification of early warning signals and, consequently, might have prevented the tragic outcome.

Let us finally observe that all the derived emotions, obtained through combinations of basic ones, are used to label at least one message, which supports the idea that the proposed approach may be useful for the intended purpose of yielding meaningful results when applied to real-world conversations characterized by high emotional content, highlighting its potential in forensic contexts and, in particular, for the implementation of an automatic – preventive – detection of situations involving excessively aggressive or coercive behaviors.

6. Conclusions and Future Works

In this work, we have introduced a tool called SERC for the automatic identification of emotions in messages. The system is based on the Description Logic of typicality $\mathbf{T}^{\mathrm{CL}}$, introduced in [21, 22] as a probabilistic extension of the basic formalism able to deal with the combination of concepts. Intuitively, the system checks words/concepts belonging to a given message with the properties characterizing both basic and derived emotions described in the language of the $\mathbf{T}^{\mathrm{CL}}$ logic, by also considering probabilities equipping each typicality inclusion as suitable weights of such properties. The ontology with typicality and probabilities describing emotions and related properties [19], is based on Plutchik's psychological circumplex model [4]. Inspired by the tool introduced in [23], the system SERC introduces two main innovations in the field of emotion assignment for short texts. First, unlike existing approaches that, for efficiency reasons, focus almost exclusively on the eight basic emotions, SERC exploits the $\mathbf{T}^{\mathrm{CL}}$ logic and is therefore able to represent and manage complex emotions, modelled as combinations of fundamental ones. Second, while most state-of-the-art systems are subsymbolic neural-network-based models that do not provide transparent justifications for their predictions, SERC leverages $\mathbf{T}^{\mathrm{CL}}$ to generate explicit explanations. In particular, the selected emotion is justified on the basis of the number and probabilities of textual properties detected in the input and associated, by the logic, with specific emotions.

In future research, we aim at extending our validation by comparing the results provided by our tool SERC with those produced by a second system. For this purpose, we are currently employing a *Large Language Model* (LLM), i.e., a model capable of generating natural language by leveraging *self-supervised learning* (SSL). We are currently setting up a comparison by using a well-established LLM, namely BART large MNLI[6] performing a *Zero-Shot Classification*. In this paradigm, the model learns from the input data without relying on external labels, as typically occurs in supervised learning. In supervised learning, a system is provided with a set of examples so that it can learn how to correctly classify instances. Conversely, in SSL the system can learn directly from the data it processes. The goal of the LLM employed in this work is to classify the input instances, i.e., to determine to which classes they belong. More concretely, given a set of instances and a list of classes, the model determines which instances fall under each provided class. This task is known as *classification* and is a central topic in Machine Learning. In our specific case, the objective is to determine which emotions the LLM is able to detect within the input sentences, in order to compare its outputs with those of the tool SERC.

We also plan to compare the results produced by the SERC system in assigning emotions to messages with those obtained by the LLM by a user study involving human participants of different ages, genders, and educational backgrounds: we aim to ask each participant to evaluate messages and conversations processed by both SERC and the LLM, selecting the emotion they consider most significant for each sentence. Finally, we intend to test the SERC tool as soon as possible on real chat data, both private and group conversations, with the aim of evaluating the effectiveness of our system in the early detection of aggressive behavior and (cyber)bullying.

[6]https://huggingface.co/facebook/bart-large-mnli

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

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

[2] D. Watson, L. A. Clark, A. Tellegen, Development and validation of brief measures of positive and negative affect: the PANAS scales., Journal of personality and social psychology 54 (1988) 1063.

[3] A. Mehrabian, Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament, Current Psychology 14 (1996) 261–292.

[4] R. Plutchik, A general psychoevolutionary theory of emotion, in: Theories of emotion, Elsevier, 1980, pp. 3–33.

[5] P. Ekman, Basic emotions, in: M. J. P. Tim Dalgleish (Ed.), Handbook of cognition and emotion, volume 98, Wiley Online Library, 1999, pp. 45–60. doi:10.1002/0470013494.

[6] T. Bänziger, K. R. Scherer, Introducing the geneva multimodal emotion portrayal (gemep) corpus, Blueprint for affective computing: A sourcebook 2010 (2010) 271–94.

[7] D. Jurafsky, J. H. Martin, Lexicons for sentiment, affect, and connotation, in: Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition, 3nd Edition, Prentice Hall Series in artificial intelligence, Prentice Hall, Pearson Education International, 2019. Draft, available online: https://web.stanford.edu/ jurafsky/slp3/21.pdf. Accessed on December 2020.

[8] M. Nissim, V. Patti, Chapter 3 - semantic aspects in sentiment analysis, in: F. A. Pozzi, E. Fersini, E. Messina, B. Liu (Eds.), Sentiment Analysis in Social Networks, Morgan Kaufmann, Boston, 2017, pp. 31 – 48. URL: http://www.sciencedirect.com/science/article/pii/B9780128044124000036. doi:https://doi.org/10.1016/B978-0-12-804412-4.00003-6.

[9] Z. Wang, S. Ho, E. Cambria, A review of emotion sensing: categorization models and algorithms, Multimedia Tools and Applications 79 (2020). doi:10.1007/s11042-019-08328-z.

[10] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, U. R. Acharya, Abcdm: An attention-based bidirectional cnn-rnn deep model for sentiment analysis, Future Generation Computer Systems 115 (2021) 279–294. URL: https://www.sciencedirect.com/science/article/pii/S0167739X20309195. doi:https://doi.org/10.1016/j.future.2020.08.005.

[11] S. M. Mohammad, Sentiment analysis: Detecting valence, emotions, and other affectual states from text, CoRR abs/2005.11882 (2020). URL: https://arxiv.org/abs/2005.11882. arXiv:2005.11882.

[12] S. Mohammad, Word affect intensities, in: N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, T. Tokunaga (Eds.), Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018, European Language Resources Association (ELRA), 2018. URL: http://www.lrec-conf.org/proceedings/lrec2018/summaries/329.html.

[13] M. S. Akhtar, A. Ekbal, E. Cambria, How intense are you? predicting intensities of emotions and sentiments using stacked ensemble [application notes], IEEE Computational Intelligence Magazine 15 (2020) 64–75. doi:10.1109/MCI.2019.2954667.

[14] C. Strapparava, R. Mihalcea, Semeval-2007 task 14: Affective text, in: E. Agirre, L. M. i Villodre, R. Wicentowski (Eds.), Proceedings of the 4th International Workshop on Semantic Evaluations, SemEval '07, Association for Computational Linguistics, Stroudsburg, PA, USA, 2007, pp. 70–74. URL: https://www.aclweb.org/anthology/volumes/S07-1/.

[15] S. M. Mohammad, P. D. Turney, Crowdsourcing a word-emotion association lexicon, Computational Intelligence 29 (2013) 436–465.

[16] S. Mohammad, Obtaining reliable human ratings of valence, arousal, and dominance for 20,000

English words, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 174–184. URL: https://www.aclweb.org/anthology/P18-1017. doi:10.18653/v1/P18-1017.

[17] E. Cambria, Y. Li, F. Z. Xing, S. Poria, K. Kwok, Senticnet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis, in: M. d'Aquin, S. Dietze, C. Hauff, E. Curry, P. Cudré-Mauroux (Eds.), CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020, ACM, 2020, pp. 105–114. URL: https://doi.org/10.1145/3340531.3412003. doi:10.1145/3340531.3412003.

[18] C. Strapparava, A. Valitutti, WordNet affect: an affective extension of WordNet, in: Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04), European Language Resources Association (ELRA), Lisbon, Portugal, 2004. URL: http://www.lrec-conf.org/proceedings/lrec2004/pdf/369.pdf.

[19] A. Lieto, G. L. Pozzato, S. Zoia, V. Patti, R. Damiano, A commonsense reasoning framework for explanatory emotion attribution, generation and re-classification, Knowledge Based Systems 227 (2021) 107166. URL: https://doi.org/10.1016/j.knosys.2021.107166. doi:10.1016/J.KNOSYS.2021.107166.

[20] A. Lieto, G. L. Pozzato, M. Striani, S. Zoia, R. Damiano, DEGARI 2.0: A diversity-seeking, explainable, and affective art recommender for social inclusion, Cognitive Systems Research 77 (2023) 1–17. URL: https://doi.org/10.1016/j.cogsys.2022.10.001. doi:10.1016/J.COGSYS.2022.10.001.

[21] A. Lieto, G. L. Pozzato, A description logic of typicality for conceptual combination, in: M. Ceci, N. Japkowicz, J. Liu, G. A. Papadopoulos, Z. W. Ras (Eds.), Foundations of Intelligent Systems - 24th International Symposium, ISMIS 2018, Limassol, Cyprus, October 29-31, 2018, Proceedings, volume 11177 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 189–199. URL: https://doi.org/10.1007/978-3-030-01851-1_19. doi:10.1007/978-3-030-01851-1_19.

[22] A. Lieto, G. L. Pozzato, A description logic framework for commonsense conceptual combination integrating typicality, probabilities and cognitive heuristics, Journal of Experimental and Theoretical Artificial Intelligence 32 (2020) 769–804. URL: https://doi.org/10.1080/0952813X.2019.1672799. doi:10.1080/0952813X.2019.1672799.

[23] A. Lieto, M. Orusa, G. L. Pozzato, S. Zoia, An application of the TCL logic to aerospace missions, in: D. Guidotti, L. Pandolfo, L. Pulina (Eds.), Proceedings of the 40th Italian Conference on Computational Logic, Alghero, Italy, June 25-27, 2025, volume 4003 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2025. URL: https://ceur-ws.org/Vol-4003/paper24.pdf.

[24] F. Riguzzi, E. Bellodi, E. Lamma, R. Zese, Probabilistic description logics under the distribution semantics, Semantic Web 6 (2015) 477–501. URL: http://dx.doi.org/10.3233/SW-140154. doi:10.3233/SW-140154.

[25] M. A. Boden, Creativity and artificial intelligence, Artificial Intelligence 103 (1998) 347–356.

[26] M. Frixione, A. Lieto, Representing and reasoning on typicality in formal ontologies, in: C. Ghidini, A. N. Ngomo, S. N. Lindstaedt, T. Pellegrini (Eds.), Proceedings of the 7th International Conference on Semantic Systems, ACM International Conference Proceeding Series, ACM, 2011, pp. 119–125. URL: https://doi.org/10.1145/2063518.2063534. doi:10.1145/2063518.2063534.

[27] D. N. Osherson, E. E. Smith, On the adequacy of prototype theory as a theory of concepts, Cognition 9 (1981) 35–58.

[28] A. Lieto, G. L. Pozzato, F. Perrone, E. Chiodino, Knowledge capturing via conceptual reframing: A goal-oriented framework for knowledge invention, in: M. Kejriwal, P. A. Szekely, R. Troncy (Eds.), Proceedings of the 10th ACM Conference on Knowledge Capture, K-CAP 2019, Marina del Rey, ACM, 2019, pp. 109–114. URL: https://doi.org/10.1145/3360901. doi:10.1145/3360901.

[29] A. Lieto, F. Perrone, G. L. Pozzato, E. Chiodino, Beyond subgoaling: A dynamic knowledge generation framework for creative problem solving in cognitive architectures, Cognitive Systems Research 58 (2019) 305–316.

[30] E. Chiodino, A. Lieto, F. Perrone, G. L. Pozzato, A goal-oriented framework for knowledge invention and creative problem solving in cognitive architectures, in: G. D. Giacomo, A. Catalá,

B. Dilkina, M. Milano, S. Barro, A. Bugarín, J. Lang (Eds.), Proceedings of ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, 2020, volume 325 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2020, pp. 2893–2894. URL: https://doi.org/10.3233/FAIA200440. doi:10.3233/FAIA200440.

[31] A. Lieto, G. L. Pozzato, Applying a description logic of typicality as a generative tool for concept combination in computational creativity, Intelligenza Artificiale 13 (2019) 93–106.

[32] A. Lieto, G. L. Pozzato, S. Zoia, The delta of thought: Channeling rivers of commonsense knowledge in the sea of metaphorical interpretations, in: J. Kwok (Ed.), Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25, International Joint Conferences on Artificial Intelligence Organization, 2025, pp. 10316–10324. URL: https://doi.org/10.24963/ijcai.2025/1146. doi:10.24963/ijcai.2025/1146, human-Centred AI.

[33] L. Giordano, V. Gliozzi, N. Olivetti, G. L. Pozzato, Semantic characterization of Rational Closure: from Propositional Logic to Description Logics, Artificial Intelligence 226 (2015) 1–33. doi:10.1016/j.artint.2015.05.001.

[34] D. Lehmann, M. Magidor, What does a conditional knowledge base entail?, Artificial Intelligence 55 (1992) 1–60. doi:http://dx.doi.org/10.1016/0004-3702(92)90041-U.

[35] F. Riguzzi, E. Bellodi, E. Lamma, R. Zese, Reasoning with probabilistic ontologies, in: Q. Yang, M. Wooldridge (Eds.), Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015, AAAI Press, 2015, pp. 4310–4316. URL: http://ijcai.org/proceedings/2015.

[36] J. A. Hampton, Inheritance of attributes in natural concept conjunctions, Memory & Cognition 15 (1987) 55–71.

[37] E. Cambria, A. Livingstone, A. Hussain, The hourglass of emotions, in: A. Esposito, A. M. Esposito, A. Vinciarelli, R. Hoffmann, V. C. Müller (Eds.), Cognitive Behavioural Systems - COST 2102 International Training School, Dresden, Germany, February 21-26, 2011, Revised Selected Papers, volume 7403 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 144–157. URL: https://doi.org/10.1007/978-3-642-34584-5_11. doi:10.1007/978-3-642-34584-5_11.

[38] J. P. McCrae, D. Spohr, P. Cimiano, Linking lexical resources and ontologies on the semantic web with lemon, in: G. Antoniou, M. Grobelnik, E. P. B. Simperl, B. Parsia, D. Plexousakis, P. D. Leenheer, J. Z. Pan (Eds.), The Semantic Web: Research and Applications - 8th Extended Semantic Web Conference, ESWC 2011, Heraklion, Crete, Greece, May 29-June 2, 2011, Proceedings, Part I, volume 6643 of *Lecture Notes in Computer Science*, Springer, 2011, pp. 245–259. URL: https://doi.org/10.1007/978-3-642-21034-1_17. doi:10.1007/978-3-642-21034-1_17.

[39] V. Patti, F. Bertola, A. Lieto, Arsemotica for arsmeteo. org: Emotion-driven exploration of online art collections, in: I. Russell, W. Eberle (Eds.), The Twenty-Eighth International Florida Artificial Intelligence Research Society Conference (FLAIRS 2015), Association for the Advancement of Artificial Intelligence, AAAI Press, 2015, pp. 288–293. URL: http://www.aaai.org/Library/FLAIRS/flairs15contents.php.

[40] A. Lieto, G. Pozzato, A. Valese, COCOS: a typicality based COncept COmbination System , in: M. Montali, P. Felli (Eds.), Proceedings of the 33rd Italian Conference on Computational Logic (CILC 2018), CEUR Workshop Proceedings, Bozen, Italy, 2018, pp. 55–59.

[41] E. Chiodino, D. D. Luccio, A. Lieto, A. Messina, G. L. Pozzato, D. Rubinetti, A knowledge-based system for the dynamic generation and classification of novel contents in multimedia broadcasting, in: G. D. Giacomo, A. Catalá, B. Dilkina, M. Milano, S. Barro, A. Bugarín, J. Lang (Eds.), Proceedings of ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, 2020, volume 325 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2020, pp. 680–687. URL: https://doi.org/10.3233/FAIA200154. doi:10.3233/FAIA200154.

[42] L. Giordano, V. Gliozzi, G. L. Pozzato, R. Renzulli, An efficient reasoner for description logics of typicality and rational closure, in: A. Artale, B. Glimm, R. Kontchakov (Eds.), Proceedings of the 30th International Workshop on Description Logics, Montpellier, France, July 18-21, 2017, volume 1879 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017. URL: http://ceur-ws.org/Vol-1879/

paper25.pdf.
[43] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, S. Niu, DailyDialog: A manually labelled multi-turn dialogue dataset, in: G. Kondrak, T. Watanabe (Eds.), Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Asian Federation of Natural Language Processing, Taipei, Taiwan, 2017, pp. 986–995. URL: https://aclanthology.org/I17-1099/.