Ref. Ares(2017)531934 - 31/01/2017

# D5.3 – Robot Sentiment Analysis

| | |
|---|---|
| Project Acronym: | **MARIO** |
| Project Title: | **Managing active and healthy aging with use of caring service robots** |
| Project Number: | **643808** |
| Call: | **H2020-PHC-2014-single-stage** |
| Topic: | **PHC-19-2014** |
| Type of Action: | **RIA** |

| D5.2 – MARIO Reading and Listening Component | |
|---|---|
| **Work Package** | WP5 |
| **Due Date** | M24 |
| **Submission Date** | 31/01/2017 |
| **Start Date of the Project** | 01/02/2015 |
| **Duration of the Project** | 36 Months |
| **Organisation Responsible of Deliverable** | CNR |
| **Version** | 1.0 |
| **Status** | Intermediate |
| **Author name(s)** | Valentina Presutti (CNR)<br>Aldo Gangemi (CNR)<br>Domenico Pisanelli (CNR)<br>Diego Reforgiato Recupero (R2M)<br>Paolo Ciancarini (CNR) |
| **Nature** | ☐ R – Report ☐ P – Prototype<br>☐ D – Demonstrator ☒ O – Other |
| **Dissemination Level** | ☒ PU – Public<br>☐ CO – Confidential, only for members of the Consortium (including the Commission)<br>☐ RE – Restricted to a group specified by the Consortium (including the Commission Services) |

| Revision History | | | |
|---|---|---|---|
| **Version** | Date | Modified by | Comments |
| **0.1** | 25/01/2017 | Valentina Presutti | First Draft |
| **0.2** | 30/01/2017 | Valentina Presutti | Changes according to internal reviews |
| | | | |
| | | | |

Copyright © 2016, MARIO Consortium

The MARIO Consortium (http://www.mario-project.eu/) grants third parties the right to use and distribute all or parts of this document, provided that the MARIO project and the document are properly referenced.

THIS DOCUMENT IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS DOCUMENT, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Outline and Contents

Overview of the report content

Overview of the task

Main components reused and developed in Task 5.3

- Opinion model ontology
- Sentiment lexical resources
- Frame-based sentiment analysis

# Overview of the report content

# Overview

This report briefly describes a set of resources and software components that are under development within WP5 and that contribute to the MARIO software system architecture. It is intended to provide the reviewers with an intermediate update on the progress of Task 5.3.

The main component aims at providing a formal representation and evaluation of the sentiment expressed in text sentences. It is built as an extension of FRED (see D5.2) and relies on novel resources developed in the context of this task. The software is available as REST services (and possibly associated with an online demo), which guarantees smooth integration with the rest of MARIO software architecture.

# Overview of the task

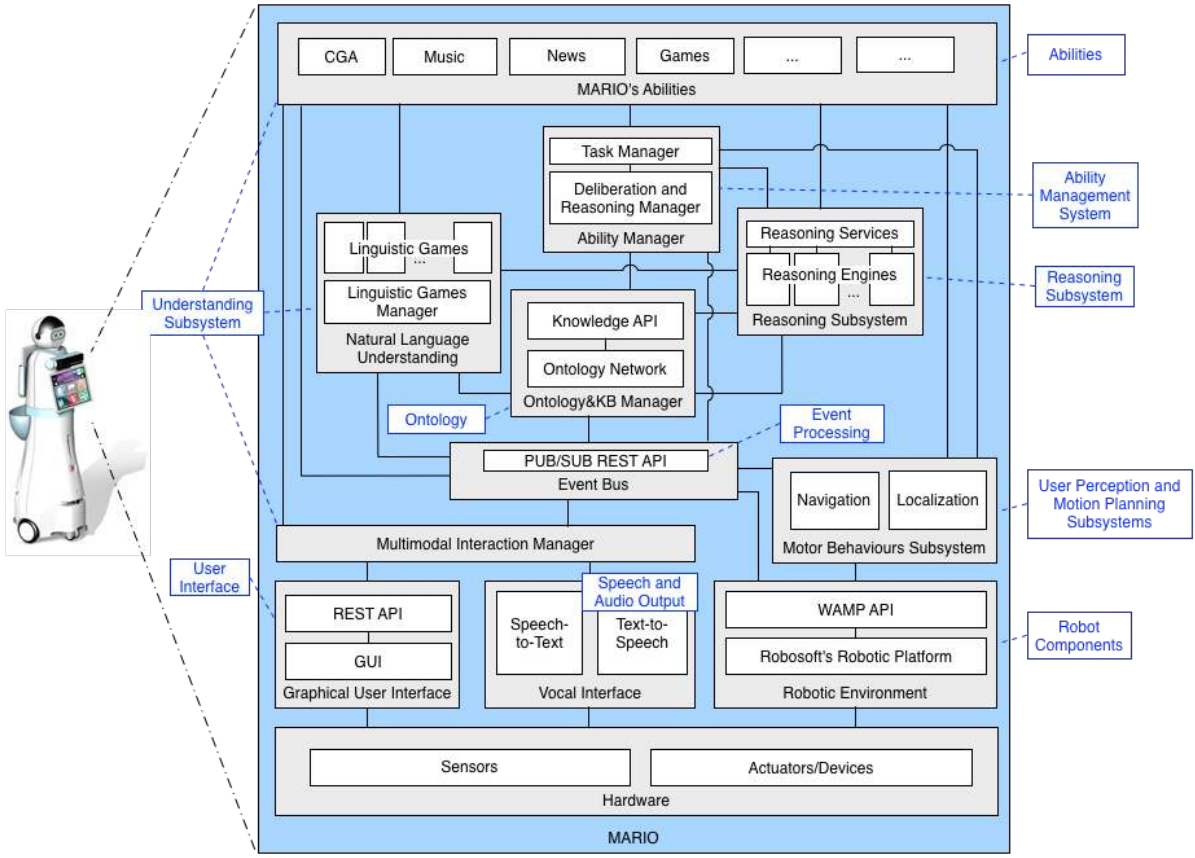
# Task 5.3

This task deals with MARIO's capability to extract sentiment information from natural language sentences expressed by a patient with dementia (PwD).

It relies on, and contributes to, the results of WP5 Task 5.1 (see deliverable D5.1) as far as the background knowledge and knowledge models that it uses are concerned. It relies on WP5 Task 5.2 as far as the capability of natural language understanding is concerned.

It provides support to tasks in WP4 and WP6 as far as natural-language based interaction is concerned, with specific focus on capturing and representing sentiment knowledge.

Referring to MARIO architecture, this task contributes to the *Natural Language Understanding* component



# Main components reused and developed in Task 5.3

# Overview

Sentiment analysis concerns  the study of intelligent algorithms capable of automatically mining opinions from natural language content.

In the context of MARIO, we are interested in applying sentiment analysis to sentences in natural language spoken by a patient, especially when they are expected to express feedback about some recent activity. This information may be stored and used in order to influence the robot's arbitration of behavior in later stages.

# Overview (cont.d)

An immediate application of the sentiment analysis module is to the Memory as well as the Play music app. When running such apps, MARIO interacts with patients by asking them to express a feedback e.g. whether they liked a song or not, to describe their mood after looking at a picture with few words or a sentence, etc.

This information expressed through speech is captured by the Understanding Component and processed by the Sentiment Analysis module so as to produce data expressing sentiment and emotional information associated to such a specific item view/listening event. At a later stage MARIO may use this information in order to decide whether to e.g. play a song, view a picture, according to specific goals and strategy it is implementing.

# Overview (cont.d)

In order to achieve the goal of this task we:

- Reuse an ontology for representing opinions
- Integrate sentiment lexical resources into MARIO background knowledge (Framester)
- Developed a sentence-polarity evaluation module
- Extended and implemented a frame-based sentiment analysis algorithm

# Opinion model ontology

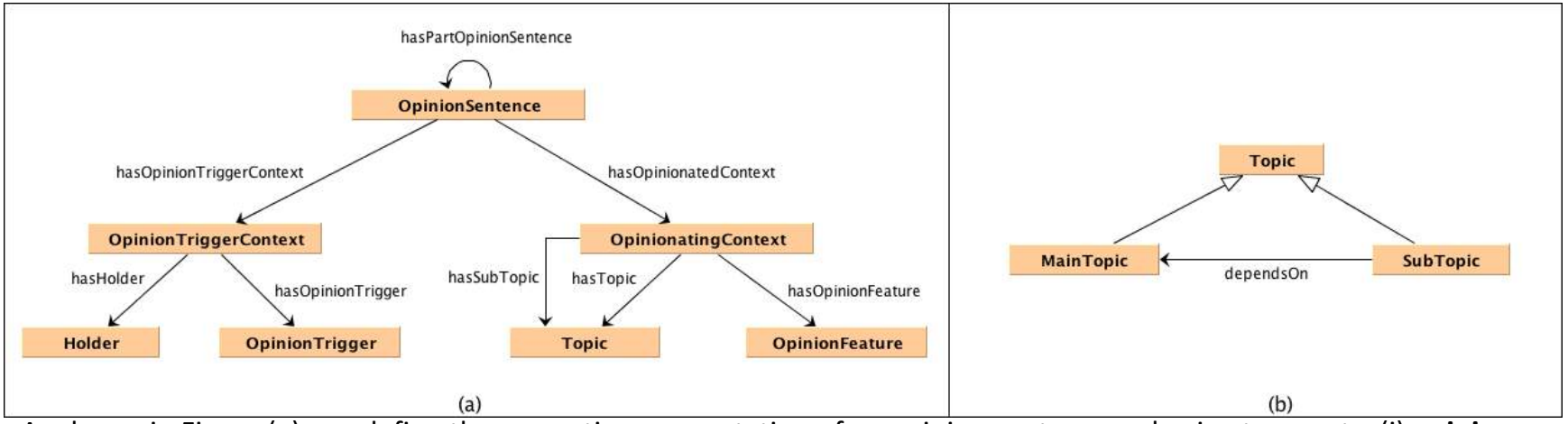
# Opinion model ontology

An opinion can be defined as an intentional statement by somebody (holder) on some fact (topic) that is expressed with a possible sentiment. A Sentiment Analysis system should be able to extract and characterize opinions by recognizing the attitude (positive, negative or objective) of an opinion holder on a certain topic, or by evaluating the overall tonality of a sentence.

# Opinion model ontology (cont.d)

Our opinion model defines the main concepts characterizing an opinion, and is used for annotating the semantic representation of a sentence, in order to identify its opinion holder and topic.

# Opinion model ontology (cont.d)



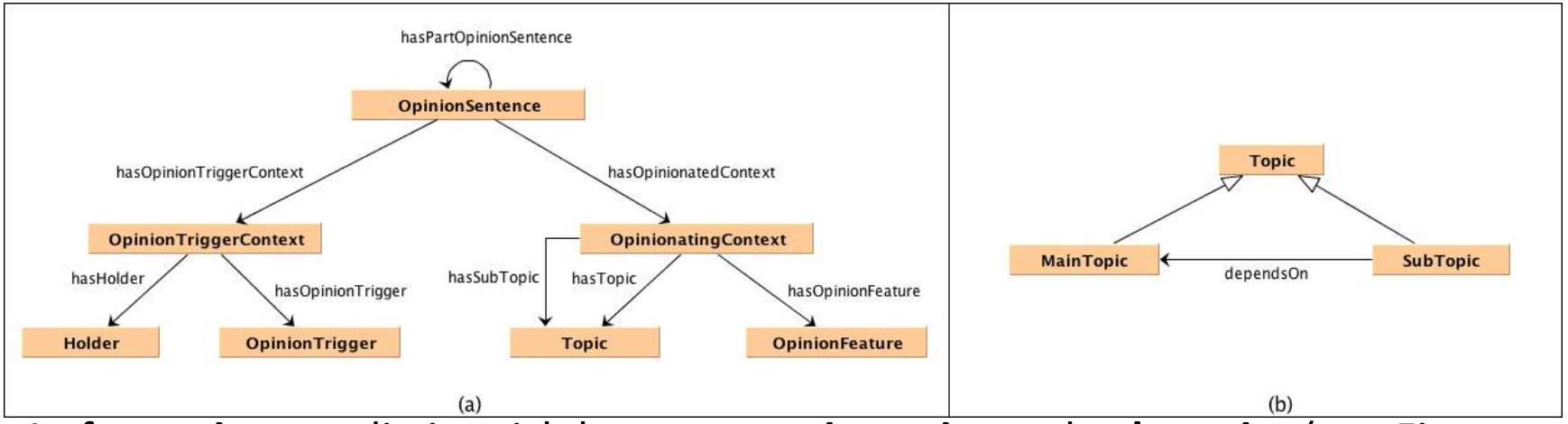

As shown in Figure (a), we define the semantic representation of an opinion sentence as having two parts: (i) **opinion trigger context** and (ii) **opinionated context**.

The **opinion trigger** context is optional and identifies concepts that indicate the presence of an opinion expressed in the sentence, and its holder. It is composed of two parts: the entities that allow the identification of the opinion **holders** (i.e. holders), and the events e.g., think, say, support, etc., that act as **triggers of an opinion** (i.e. opinion triggers).

The **opinionated context** identifies concepts that express an opinion (possibly including sentiments). It is composed of two parts: the entities that identify the **opinion topics** (i.e. topics), and those expressing the opinion and its possible associated sentiments (i.e. the **opinion features**).

In some cases, terms that activate an opinion can also convey the opinion itself or a possible sentiment, e.g. “approve” or “deny”. When this happens, such terms play the role of **opinion triggers** as well as that of **opinion features**.

# Opinion model ontology (cont.d)




As for **topics**, we distinguish between **main topics** and **sub-topics** (see Figure (b)).

In fact, a topic can be a complex structure. e.g. an event or a situation, including other complex structures. For the purpose of Sentiment Analysis, it is important to distinguish all main topics, that are the direct targets of an opinion, from sub-topics, which could be indirect targets of an opinion.

For example, the main topic of the opinion sentence: *"Anna says the weather will become beautiful"* is the event *become*, while *weather* is a sub-topic.

# Opinion model ontology (cont.d)

The following axioms formalize the concepts of main topic and sub-topic by using a standard description logic syntax, directly translatable into OWL (language used for representing all MARIO ontologies, see Deliverable 5.1).

$$(\text{MainTopic} \sqcup \text{SubTopic}) \sqsubseteq \text{Topic}$$

Topics (of opinion sentences) can be either main topics (direct targets of the opinion) or sub-topics (indirect targets of an opinion).

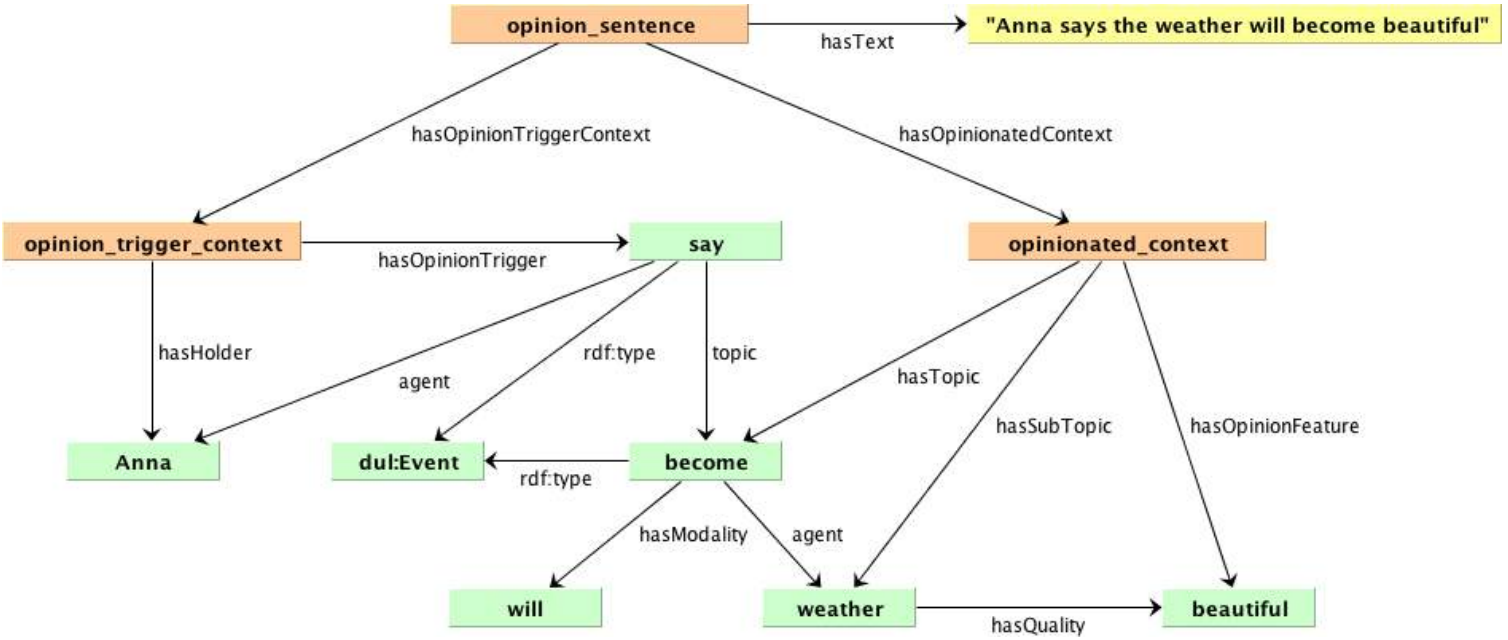
$$(\text{Topic} \sqcap (\exists \text{involvedIn}(\text{dul:Situation} \sqcap \text{MainTopic}))) \sqsubseteq \text{SubTopic}$$

When a main topic is a situation, its involved entities are subtopics.

$$(\text{Topic} \sqcap (\exists \text{dependsOn}(\text{dul:Event} \sqcap \text{MainTopic}))) \sqsubseteq \text{SubTopic}$$

When a main topic is an event, entities that have a dependency relation with it are sub-topics. Examples of dependency relations are: participation in the event (e.g., having a role in it), causality relations, temporal relations e.g., follows, precedes), etc.

# Opinion model ontology (cont.d)




Following the opinion model, the semantic representation of the sentence *"Anna says the weather will become beautiful"* is depicted in the Figure above.

# Sentiment Lexical resources

# Sentiment lexical resources

Lexical resources are key to annotate natural language sentence at the “lexical layer” with sentiment information. They provide the terms (and possibly a score for them) that act as triggers of an opinion or as opinion features. MARIO Sentiment Analysis module relies on the following resources:

- Sentic-Net [1]: a publicly available semantic and a affective resource for concept-level opinion and sentiment analysis. SenticNet is built by means of sentic computing, a paradigm that exploits both AI and Semantic Web techniques to better recognize, interpret, and process natural language opinions over the Web. We have transformed Sentic-Net to the RDF format and aligned it with Framester (D5.1) so as to exploit it in the Sentiment Analysis module.
- SentiWord-Net [2]: a lexical resource for opinion mining. SentiWordNet assigns to each synset of WordNet three sentiment scores: positivity, negativity, objectivity. This resource is integrated within Framester with the same approach and aim as the integration of SenticNet.
- DepecheMood [3]: an emotion lexicon built by harvesting crowdsourced affective annotation from a social news network.
- (a revision of) the Levin’s classification of verbs [4]: in this revision we have classified four classes of opinion verbs that imply the presence of an holder; we call them opinion trigger verbs. They are also included in MARIO background knowledge through alignment with Framester.
- Sentilo-Net [5]: a resource that annotates semantic roles of frames so as to enable the selection of subtopics that are indirectly affected by opinions expressed in a sentence, as well as the evaluation of their polarity. This resource is key to evaluate sentiment expressed complex structures such as events and distinguish the different sentiments associated with their participants.

# Frame-based sentiment analysis

# Simple sentiment polarity analysis

The Sentiment Analysis module is provided as a software component available through REST services. Hence, it can be used as a separate functionality. However, within MARIO architecture it is part of the Natural Language Understanding component (D5.2).

Within the NLU component a simple Sentiment extraction service is implemented. It allows to extract the sentiment of the terms in a sentence by returning their polarity score according to the different sentiment lexical resources included in MARIO background knowledge (see slide 23).

Currently we are working to implement different strategies for computing sentiment and emotional scores of a whole sentence, that combines the extracted values, and allows us to classify a sentence based on its polarity. We may also include within the architecture an off-the shelf component based on machine learning for classifying sentences according to their sentiment polarity e.g. Stanford CoreNLP [6].

# Frame-based sentiment analysis

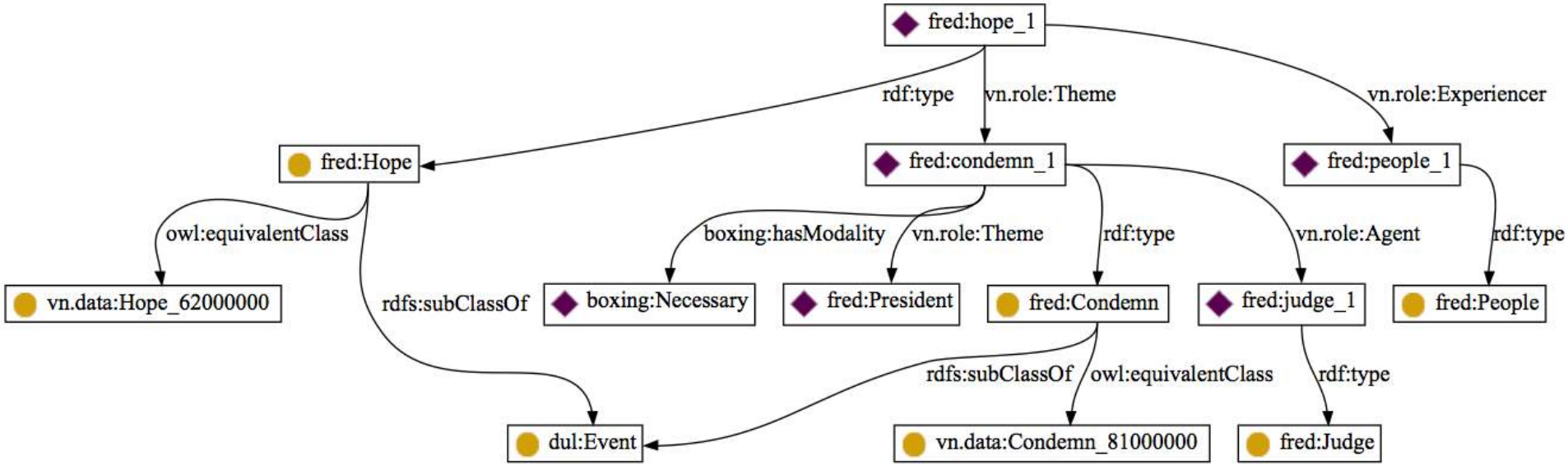
This module is implemented as an extension of FRED, hence it exploits a deep parsing analysis of a sentence and its frame-based representation.

FRED as well as the use of BabelNet as part of Framester (see D5.2) guarantee the applicability of our method to both Italian and English, which are the target languages for MARIO.

The frame-based representation of the sentence is further annotated with the opinion model ontology (cf. slide 13-17)

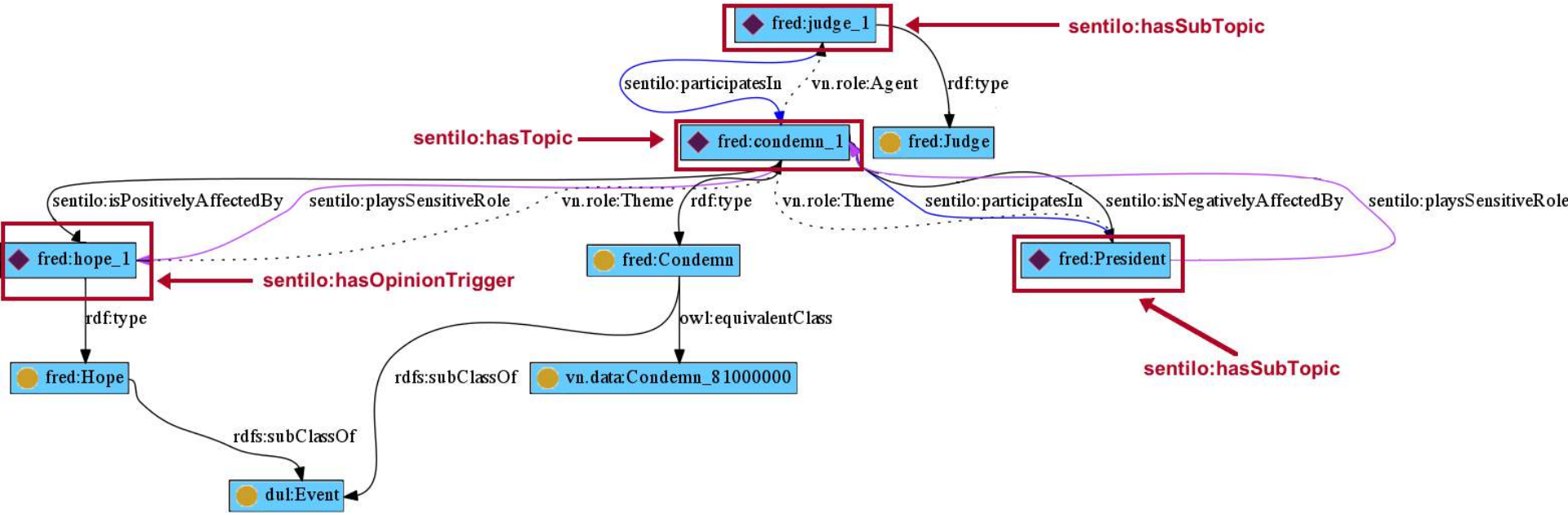
The core of the module is a sentiment propagation algorithm that relies on the Sentilo-Net resources (see slide 18). The algorithm computes the sentiment score associated with each specific identified topic in the sentence representation, according to the role that they play in their participating frame.

In the following slides we show through an example how the algorithm works and how the different lexical sentiment resources are involved.



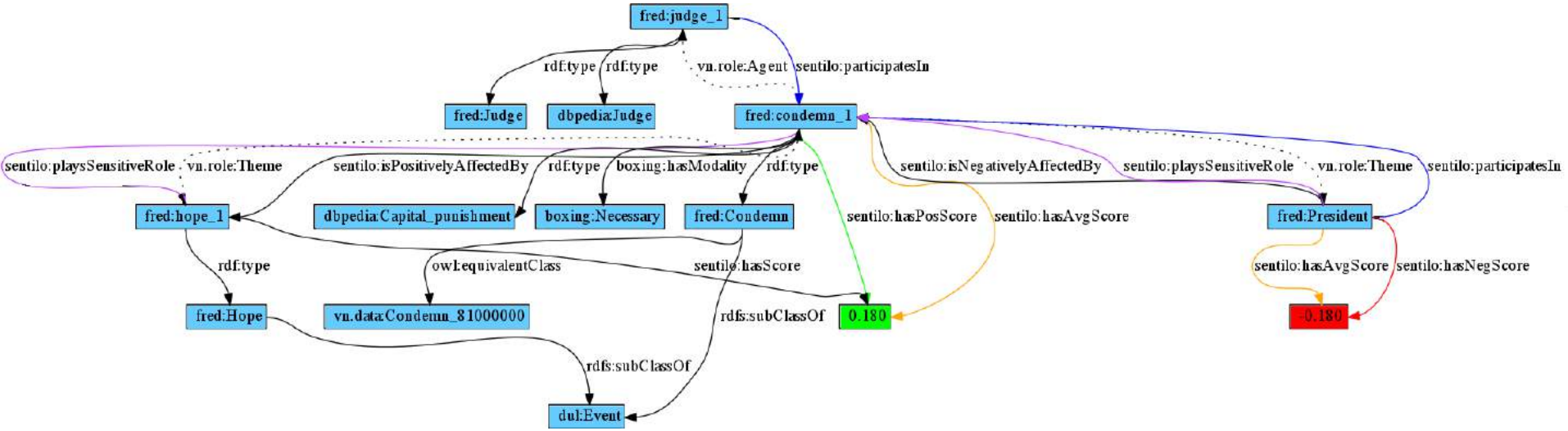

The graph above represents the sentence: *"People hope that the President will be condemned by the judges"*

The sentence is first processed by FRED (D5.2) which provides as output its frame-based graph representation.




Then the graph is passed to the frame-based sentiment analysis module, that annotates the resulting graph with the opinion model ontology. This annotation is performed by identifying triggering verbs, opinion holders and opinion topics according to rules associated with the revised Levin's verb classification [4].

Furthermore, the semantic roles identified in the sentence are annotated according to the Sentilo-Net resource in order to indicate for each sub-topic whether and how it is indirectly affected by the event (e.g. sentilo:playSentisitveRole).




All sentiment features (terms carrying a sentiment polarity) are identified and their scores represented in the model.

Finally, based on the Sentilo-Net-related annotations, the scores are propagated through the graph in order to assign them to the acutal entities they are referred to and with the correct sign.

# Sentiment propagation algorithm

It relies on a sub-algorithm named *CombinedScore* which assigns an individual sentiment score (if applicable) to each element in an opinion sentence graph. To this aim, it relies on SentiWordNet and SenticNet. The algorithm assigns a score to adjectives and adverbs that are identified by dul:hasQuality relation values, and to instances of dul:Event that are recognized as trigger events, i.e. identified by sentilo:hasOpinionTrigger relation values.

We have investigated and implemented two alternative approaches for score selection. The first approach assigns a score retrieved by querying the polarity attribute of a concept in SenticNet. The second one combines SenticNet and SentiWordNet scores.

# Sentiment propagation algorithm

From a set of empirical observations on using the Propagation algorithm with different approaches, we noticed that the method that combines the scores from the two resources shows to be more reliable. This confirmed our expectations based on the following rationale: sometimes, SenticNet misses a score value for a required concept. Moreover, it provides one score per concept without distinguishing its possible different nuances. Hence, SenticNet score approximates an average value for the scores of all possible senses, or possibly indicates the most probable one. For this reason, combining the SentiWordNet scores of most frequent senses and the SenticNet score can provide an appropriate balanced value. We have devised a simple heuristics for computing this combined score

In the next slide the CombinedScore algorithm is sketched.

# CombinedScore algorithm

score = **CombinedScore**($w$){
sNet = SenticNet score for $w$;
n = number of $w$ senses;
T = {};
**for** $i \leftarrow 1$ ***to*** $n$ **do**
    $s_i$ = extract next sense of $w$ from WordNet in decreasing order of tag_count;
    **if** $tag\_count[s_{i-1}] > 10 \times tag\_count[s_i]$ **then**
        **break**;
    T = T ∪ {s};
**end**
T' = SentiWordNet score values for each element in T;
sWN = AVERAGE(T'{..});
**return** AVERAGE(sWN,sNet);
}

# Sentiment propagation algorithm

Given an entity, identified as a topic of an opinion (either a main or subtopic), we compute its sentiment score by combining the scores of all its associated opinion features (i.e., values of dul:hasQuality relations), which are extracted from the RDF graph representing the opinionated sentence.

If a topic participates in an event or a situation occurrence, we propagate their sentiment scores to it, according to the semantics expressed by the frame-based thematic role (e.g., vn.role:Agent) that it plays, its sensitiveness and factual impact attribute values. The sentiment score $sc_{sentiment}$ of a topic *t* can be defined as a function *f* taking the following arguments:

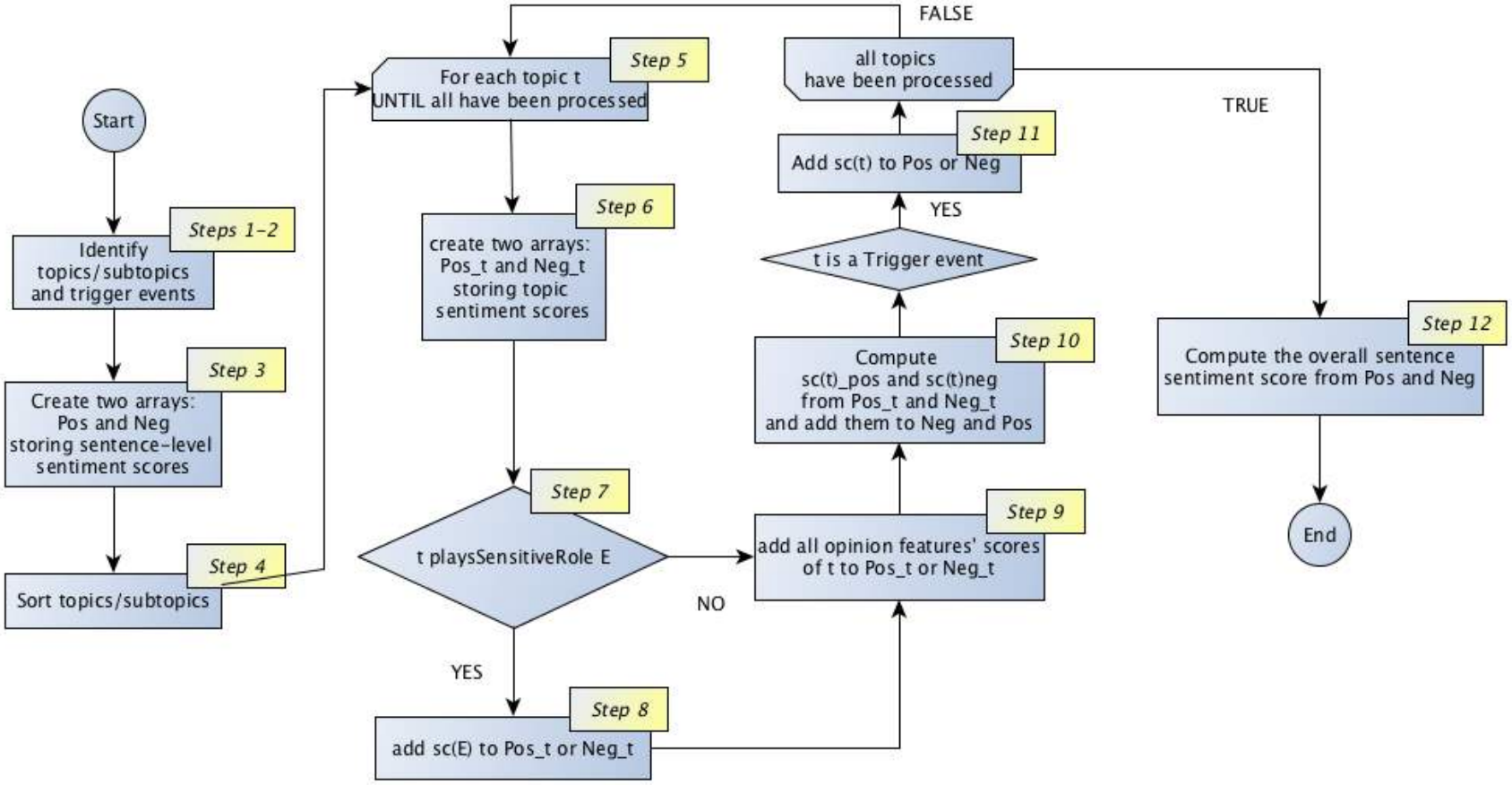
$$SC_{sentiment} = f\left(\sum_{i=0}^{n} sc(q_i\,(t), \sum_{j=0}^{m} type_j\,(t), \sum_{i=0}^{n}\ cxt_k\,(t), truth(t), mod(t), sc(trig(sent))\right)$$

Where

- sc(x) is the score of an entity x as provided by the CombinedScore algorithm;
- $q_i$(t) is an object value of a triple t dul : hasQuality $q_i$. Such triples represent direct opinion features, i.e. adjectives and adverbs, associated with entities composing the opinion sentence;
- $type_j$(t) is a type of t expressed in the RDF graph by means of rdf:type triples;
- $cxt_k$(t) is a context of t, if any. It can be either a situation or an event, which t participates in;
- truth(t) is a truth value associated with t, where t is typically an event or situation occurrence, or a quality. If its value is false, it means that the entity is negated. For example, in a sentence such as ‘‘John is not a good guy’’, a RDF triple situation_1 boxing:hasTruthValue boxing:False would be included in the graph, and its effect would be to change the sign of the sentiment score assigned to the feature good;
- mod(t) is a marked modality of a topic t, if any. For example, in a sentence such as I would like a dog, an RDF relationship *fred:like_1 boxing:hasModality boxing:Necessary* would be included. At this time, Sentilo propagation algorithm does not yet use this information, but its abstract model, the *f* function, includes it;
- Trig(sent) is an opinion trigger expression in the sentence containing t.

Function f is an abstract model referring to the Sentiment Propagation Algorithm (SP) implemented by Sentilo for computing scSentilo of a topic t.

# Sentiment Propagation algorithm flowchart



# Nest steps

- Integrating sentiment analysis module with Memory and Play music app
- Testing apps with patients (as part of the trials)
- Refinement of sentiment analysis services
- Evaluation of the sentiment analysis component:
  - We plan to evaluate our components both by using news and review (from open rating review site) corpora as gold standard datasets, as well as to inspect and evaluate the data collected during the field trials with patients.

# References

[1] E. Cambria, C. Havasi, and A. Hussain, “SenticNet 2: A semantic and affective resource for opinion mining and sentiment analysis,” in Proc. FLAIRS Conf., 2012.

[2] A. Esuli, S. Baccianella, and F. Sebastiani, “SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining,” in Proc. 7th conf. Int. Language Resources Evaluation, Valletta, Malta, May 2010.

[3] DepecheMood. Available at https://github.com/marcoguerini/DepecheMood/releases

[4] Levin opinion. Available at http://www.stlab.istc.cnr.it/documents/sentilo/levin-opinion.zip

[5] Sentilo-Net. Available at http://www.stlab.istc.cnr.it/documents/sentilo/sentilonet.zip

[6] Stanfod CoreNLP Sentiment. Available at: http://nlp.stanford.edu/sentiment/code.html