# Multimedia Responses in Natural Language Dialogues

Antonio Sorgente
Inst. of Applied Sciences and Intelligent Systems of CNR
Naples, Italy
a.sorgente@isasi.cnr.it

Paolo Vanacore
Inst. of Applied Sciences and Intelligent Systems of CNR
Naples, Italy
p.vanacore@isasi.cnr.it

Antonio Origlia
Dept. of Electrical Engineering and Information Technology, University of Naples "Federico II" &
Inst. of Applied Science and Intelligent Systems of CNR
Naples, Italy
antonio.origlia@unina.it

Enrico Leone
Dept. of Electrical Engineering and Information Technology, University of Naples "Federico II"
Naples, Italy
erik.leone82@gmail.com

Francesco Cutugno
Dept. of Electrical Engineering and Information Technology, University of Naples "Federico II" &
Inst. of Applied Science and Intelligent Systems of CNR
Naples, Italy
cutugno@unina.it

Francesco Mele
Inst. of Applied Sciences and Intelligent Systems of CNR
Naples, Italy
f.mele@isasi.cnr.it

## ABSTRACT

Offering contents to a visitor in a natural and attractive way is one of the most interesting challenges in promoting cultural heritage. In this paper, we present an ongoing research about the design and development of interactive systems based on dialogues in natural language to assist a user during a visit to a cultural space. The responses of system contain multimedia elements and are generated by users' queries or following contextual updates associated to their position, so the system can take initiative in absence of explicit stimuli. The response of system results from a composition process that coherently synchronises media elements with a synthetic voice delivering the textual content. This way, the visitor receives an audio explanation *commented* by images. To implement this approach, a semantic archive containing the annotation of stories has been built. The formalism used for the annotation is CSWL (Cultural Stories Web Language), used to represent cultural stories through events. A case study on the '800 exhibit at the Capodimonte museum is presented to describe how the system was designed and deployed.

## CCS Concepts

•**Human-centered computing** → **Human computer interaction (HCI); Natural language interfaces;** *Interaction paradigms;*



## Keywords

multimedia composition, spoken dialogue system, cultural heritage

## 1. INTRODUCTION

The amount of information related to the domain of Cultural Heritage built by experts, and published on the web, is growing day by day. With this significant load of information, offering contents to a visitor in a natural and attractive way is one of the most interesting challenges in promoting cultural heritage. In the last years, applications of dialogue systems have been presented in the Cultural Heritage domain to study how to understand users' requests, and how to provide adequate responses [**?**, **?**, **?**]. Interaction is typically based on textual chats so both input and output are only textual. With the diffusion of mobile and wearable devices, the possibility to have systems based on spoken language interaction has increased, and the advantage of having devices that support high quality video material introduces an advantage for content presentation. The basic idea is, therefore, to use portable and/or wearable devices to engage interaction in natural language and provide information about a cultural asset. The challenge is not to create a new knowledge base each time a new task is designed, but to exploit the same great deal of information about the cultural heritage domain that is already available. The main feature of this approach is to select the appropriate contents related to a cultural item and aggregate them in an unique multimedia response. In this paper, we will present a description of the multimedia dialogue system architecture and we will briefly describe the formalism used for the annotation of stories, which is based on the syncretic model introduced in [**?**]. We will also describe how the system assembles the multimedia response. To provide an example of the overall interaction, we will present a dialogue excerpt together with the relative multimedia answers.

## 2. DIALOGUE SYSTEM

The core of the dialogue system is centred on the Opendial framework [**?**], which provides a flexible environment to design dialogue systems using an XML-based language and can also be extended with customised plugins using Java. Opendial represents the dialogue state as a set of variables and it lets the user define a series of internal models. These are triggered by variable updates that automatically produce reactions in accordance with the observed state.

Although not mandatory, three main models are, typically, used in an Opendial application: the *Natural Language Understanding* (NLU) model analyses the user input and maps it on a set of possible user actions; the *Action Selection Model* (ASM) associates the user action to the correspondent machine action; and the *Natural Language Generation* (NLG) model produces a spoken content in accordance with the selected machine action. In the proposed framework, we have three separate NLU models to handle different phases of the interaction: the first separates commands to the device (volume control, taking pictures or videos, etc.) from user queries concerning cultural heritage items; the second detects the requests for device-related functions; the third one detects incomplete commands and summarises the possible outcomes, so that clarification strategies can be applied to recover the interaction. In this work, we concentrate on the management of responses to user queries detected by the first NLU model. As the system is focused on Italian, a set of plugins to process this language have been included:

1) a plugin to receive the audio stream from the client and transcribe it using Google Speech; 2) a plugin to obtain Part of Speech tags from the Treetagger[**?**] tool; 3) a plugin to normalize the utterance substituting the synonyms of target terms with the target term itself and to perform lemmatisation (synsets are obtained from the MultiWordNet[1] database); 4) a plugin to extract the dependency-based parse tree of the normalised utterance using the Turin University Linguistic Environment (TULE)[2] converting from the Turin University Treebank (TUT)[3] format to the Maltparser[4] format. This occurs as Opendial natively supports Maltparser to represent dependency trees so, with this method, it is possible to fully program the system's behaviour using the Opendial XML-based language. This also makes it easier to extend the system to English; 5) a plugin to connect Opendial to the higher-level system handling user queries.

Concerning the last plugin, a communication protocol based on JavaScript Object Notation (JSON) has been adopted. The JSON string contains the multimedia response (see section 4) for the user and defines the synchronisation of synthesised text with media. The syntax of JSON is similar to Synchronised Multimedia Integration Language (SMIL)[5]. For the implementation of the question answering process to generate the response a RESTful architecture has been implemented. The main modules used for the interpretation of user requests are: a parser to identify its grammatical structure; a set of semantic services for the detection of semantic concepts based on events; and services for accessing to semantic repository (as MultiWordnet and Wiktionary[6]).

Opendial is a probabilistic framework so, while in the present version we only make use of deterministic rules to manage the dialogue. It is possible to fine-tune the system with a combination of probability estimates and utility functions to plan machine actions and even to estimate the probability of some events in the next interactions. These estimates can be computed on the basis of previous users' interactions with the system, which can be collected either with Wizard-of-Oz approaches or using a first prototype version of the system.

## 3. CSWL

In this section we briefly present CSWL (Cultural Story Web Language) [**?**], a formalism to represent cultural stories published on the Web. These stories concern the life of historical characters, the history of artworks, and architectural structures and their transformation over time. CSWL is an event-based formalism and it defines three types of entities changing over time: *simple events*, *complex events* and *fluents*. The CSWL formalism constitutes the semantic reference to build the annotation of texts and media that are an important part of dialogue system in natural language capable to provide multimedia responses.

**Simple Events** are represented by four components: *When* - the time interval in which the event happens; *What* - the action happening in the event; *Where* - the location where an event takes place; and *Who* - the participants to the event. In CSWL, the event component *why* is represented by a causal relation between two events. For this reason, such a relation is defined as a complex event.

In CSWL, stories are represented through complex events. A **Complex Event** is composed of a set of events, a set of causal and temporal relationships between them, and of all properties holding over the time in which story unfolds. A complex event has the same type of components of a simple event, but its components are calculated starting from the elements composing the story.

Each participant to a story (character, archaeological structure, art object, etc.) can be represented through properties, spatial relations, and meronomic relations changing over time. In the same way, mental states of historical characters can be depicted through their desires, intentions, and beliefs. In CWSL these relations are represented as **Fluents** (fluent is the same concept as defined in Event Calculus [**?**]).

For our experimentation, we have annotated through CSWL a collection of texts and media related to the '800 exhibit provided by museum experts. It contains textual information describing 4 museum rooms and 7 artworks and it also contains 123 media objects linked to the relevant parts of the reference texts. For the annotation process a graphical tool has been developed.

## 4. RESPONSE THROUGH MULTIMEDIA SEMANTIC MASHUP

An issue we have dealt with in this work concerns the creation of a multimedia response temporally synchronising texts and media according to semantic annotations. This is called **syncretic text**[**?**], a text capable of organising heterogeneous languages within a unitary communications

[1] http://multiwordnet.fbk.eu/english/home.php
[2] http://www.tule.di.unito.it
[3] http://www.di.unito.it/∼tutreeb/
[4] http://www.maltparser.org/
[5] https://www.w3.org/TR/smil/
[6] https://www.wiktionary.org/

model[**?**], with features of cohesion and coherence that refer to the same enunciation instance. This way, the visitor receives an audio explanation *commented* by images.

The starting point of this process is the selection of text that will be given in response to the user request. Then, on this text, we associate multimedia resources to it. The system developed to achieve this goal is defined according to the guidelines described in [**?**]. A summary of these are: 1) a multimedia element concerning a participant can only be displayed after being uttered in the multimedia composition; 2) for each response, it's preferable to display a multimedia element that represents the current topic; 3) if for some significant textual element there are no associated multimedia elements, then the visualisation of previous multimedia elements persists; 4) an expression in the text relating to a totality can be associated with a multimedia element that represents a part (a part for the totality) and viceversa (a totality for the part); 5) the duration of all the selected multimedia elements should not, in principle, exceed the enunciation time of the text; 6) each selected multimedia element has to be shown for at least two seconds.

The above guidelines describe some good practice for the selection of media and composition of multimedia response. These are used to implement constraints on the selection and on the composition process of the response.

For the system implementation, we have used the techniques presented in [**?**]. In particular, in this work we have implemented the module to compose the multimedia response. The structure of the software modules is depicted in Figure 1. *Textual Response Generator* starting from user query extracts the answer from the text. The recognition process identifies the components of the query using the same semantics adopted to annotate the story of an artwork. For this purpose a set of rules based on the relations contained in the dependency tree of the sentence and contextual information of user's position has been defined. Analysing the dependency tree we discover the events from their components: *what*, *where*, *who* and *when*. The list of answers (events) obtained from query results are ranked, then the best answer is selected and the corresponding text associated to it is chosen as textual answer.

The *Multimedia Selector* module selects and ranks available media that can be associated to the response sentence received by the *Textual Response Generator* that returns the sentences annotated by CSWL so that media selection is based on annotated entities. The ranking is based on an index calculated by comparing the CSWL annotation of media with respect to the annotations of the text. It checks if they (media and text) have a common annotation of some entities, that is if on media are annotated entities that are cited in the text. The media coming out from this phase may be too much to be displayed, so for selection *Multimedia Selector* also takes into account: the story of the previous multimedia responses; the duration of the voice audio; and minimum visualisation duration for each media.

The *Multimedia Synchroniser* module synchronises media with synthesised text through a Text To Speech tool, so that media items are coherently visualised with the relevant time intervals in which a synthetic voice talks about the content represented in the media. The final composition is made by *Multimedia Response Streamer* or *Multimedia Response Script Builder*. *Multimedia Response Streamer* merges the element in a single media and presents it using the FFmpeg[7] tool. Instead, *Multimedia Response Script Builder* produces a composition script that reports the synchronisation times between audio texts and media. This module generates a JSON code similar to SMIL representation.

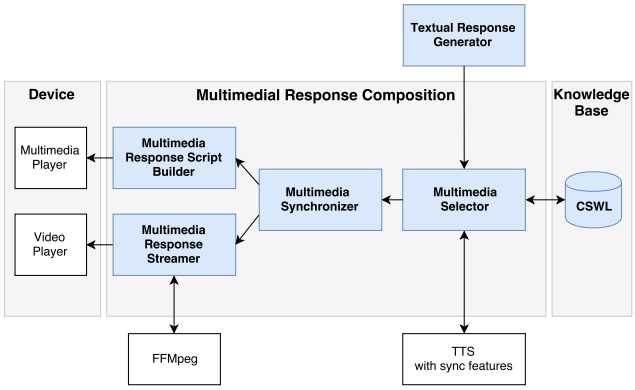


**Figure 1: multimedia composition schema**

## 5. AN EXAMPLE: '800 EXHIBIT AT THE CAPODIMONTE MUSEUM

In this section we present an example of interaction where the system, in addition to textual answer, provides a multimedia response. Texts and media for this test have been provided by experts. They describe in detail the contents of the '800 exhibit at the Capodimonte Museum.

The interaction model is a Question&Answering type, and it also able to take the initiative in the absence of user's queries and it takes into account the user's movement to suggest new information coherent to what the user expects. So, the interaction design has a model related to the user's movement that describes where the user is located: in a museum room, near a cultural asset, or in transit areas. For the localisation system a beacon system has been used. In each location, the system can interact with the user and the interaction phase is modelled on the base of information that the system can provide: general information or deepening after user's query. The system initiative is activated when the user changes an area or by timer that tracks the user inactivity.

In the listing below, we show an excerpt of a dialogue about the '800 exhibit. The topic is the picture *Caesar's Death*. We use the notation *#near(x)* to indicate that the user is near the cultural object $x$ and *#elapsed_time* to express that the user did not provide stimuli for a given time interval. In this case, the system takes the initiative and provides new information. The underlined words in the text are concepts for which multimedia materials are associated and so can be visualised.

### *Example of Multimedia Dialogue: '800 exhibit.*

[1.1] **User**: #near(la_morte_di_cesare)
[1.2] **System**: "*Il dipinto del pittore romano Vincenzo Camuccini è stato realizzato ad olio su carta poi incollata su tavoletta.*"

[7]https://www.ffmpeg.org/

(English) "*The painting, made by the Roman artist Vincenzo Camuccini, was produced using oil on paper then pasted on a tablet.*"

[2.1] **System**: #elapsed_time
[2.2] **System**: "*Questa versione è stata realizzata probabilmente tra il 1804 e il 1806 durante la lunga fase di gestazione e di vendita della tela grande.*"
(English) "*This version was probably made between 1804 and 1806 during the long period of gestation and selling of the large painting.*"

[3.1] **User**: "*Cos'è la tela grande?*"
(English) "*What is the large painting?*"
[3.2] **System**: "*La Morte di Giulio Cesare e' un dipinto a olio su tela realizzato dal pittore Vincenzo Camuccini nel 1798 e conservato nel Museo nazionale di Capodimonte a Napoli*".
(English) "*The "La Morte di Giulio Cesare" is an oil painting on canvas by the artist Vincenzo Camuccini, it has been produced in the 1798 and it's hosted in the National Museum of Capodimonte in Naples.*"

[4.1] **User**: "*Cosa rappresenta il quadro?*"
(English) "What does the painting show?"
[4.2] **System**: "*Il momento raffigurato è quello in cui Cesare, attaccato dai congiurati, cade sotto i colpi dei pugnali, durante la riunione del Senato delle Idi di Marzo il 15 del 44 a.C.*"
(English) "*The depicted moment is the one when Caesar, attacked by the conspirators, falls under the blows of the daggers, during the meeting of the Senate at the Ides of March in the 15 of the 44 B.C.*"

From the example, we can see that, when the user is near the painting named "Caesar's Death" [1.1], the system provides generic information about the author and the painting technique [1.2]. Then, the user does not make any request and stays near the picture, so the system takes the initiative providing more information [2.2]. The last turns of dialogue are stimulated by user's queries. If we consider the answer [4.2], the system produces the response by associating the relevant media as shown in Figure 2. The answers of the system are JSON scripts containing the media used and how they are temporally synchronised. The layout depends on the device (pc, smart-phone, tablet, glasses). On a smartphone we show the media like slide-show, they are shown sequentially on the display, while on glasses, the images are located in the space, they don't always appear in front of the user, but in a specific area of space.

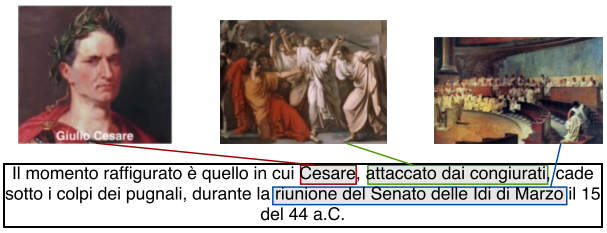


**Figure 2: example of multimedia response**

# 6. CONCLUSIONS

We have presented a multimedia spoken dialogue system for the Italian language, using texts and media to produce multimedia responses through a semantic aggregation of such resources. The reasoning engine assembles texts and media annotated with the event-based formalism CSWL. The system architecture is modular and can easily be adapted to include upcoming interaction devices. The current prototype lacks, in this phase of research, a formal evaluation of the interaction quality so future work will consist of formal on-site tests to evaluate the *user satisfaction*. Users' feedback will be collected through questionnaires filled out by visitors at end of each visit session. Also, will be defined controlled tasks to measure the usability of the system and how visitors actually use it through log analysis.

# 7. ACKNOWLEDGMENTS

Part of this work is supported by the Italian PAC project Cultural Heritage Emotional Experience See-Through Eyewear (CHEESE).