Chatting to Personalize and Plan Cultural Itineraries

Antonio Sorgente[1], Nadia Brancati[1], Cristina Giannone[2], Fabio Massimo Zanzotto[2], Francesco Mele[1], and Roberto Basili[2]

[1] Institute of Cybernetics "Eduardo Caianiello" of the National Research Council Via Campi Flegrei 34, 80078 Pozzuoli, Naples (Italy)
{a.sorgente,n.brancati,f.mele}@cib.na.cnr.it
[2] University of Rome Tor Vergata Department of Computer Science, Systems and Production, 00133 Roma (Italy)
{giannone,zanzotto,basili}@info.uniroma2.it

Abstract. In this paper, we present a system for the generation of cultural itineraries that exploits conversational agents to implicitly build formal user profiles. The key idea is that the preferences for user profiling are not obtained in a direct way, but acquired during a natural language conversation of the tourists with the system. When the user profile is ready, it becomes the input for the generation of the customized cultural itinerary. The proposed system, called DiEM System[3], is designed for dialogues in the domain of cultural heritage, but its flexible architecture allows to customize the dialogues in different application domains (cinema, finance, medicine, etc.).

1 Introduction

Offering personalized access to cultural artifacts is one of the most interesting challenges for promoting cultural heritage. Many artworks and cultural artifacts hardly reach the general public, even if someone can find such artifacts very interesting. Finding a way to help these people to reach the cultural treasures hidden in the sea of cultural offerings is an important mission (as in [1, 2]). Recommendation Systems (RS) are designed for this purpose: personalizing the experience of users (or tourists) by helping them to select cultural objects from a large set of possible alternatives. Thus, these systems are largely applied to suggest ways to visit cultural artifacts such as interesting places or artifacts in a town or in a museum [3–5].

Recommendation Systems need two basic capabilities: first, a way to *capture* user interests; second, a way to exploit this information to compute the best selection of items to suggest to the user. Generally, in tourism-oriented systems [3–5], the first activity is done by exploiting the history of the movements of tourists or asking to the users to fill out a form. Most systems allow only to

[3] The proposed system has been implemented within the DiEM project (Dialog-based Emotional Marketing) - Italian PON (Programma Operativo Pazionale) project.

suggest POIs (Points of interest) without generating customized itinerary for the tourist. Instead, the systems that suggest itineraries, propose only POIs near current location of the tourist. Generally, in the definition of the user profile, the time availability is not taken into account. This leads to plan itineraries that will probably be discarded by the user because they will not respect his time availability. As a rule, the systems that use the user profile for the generation of suggestions, allow the acquisition of user preferences by filling out forms. Instead, an interesting way to capture user interests is to talk with the users, see the Conversational Recommendation Systems (CRS) [6]. In this way, user can express preferences in a more natural way, i.e., chatting with a system able to acquire the user interests from the dialogue. But, these CRSs determine user interests by direct questions, e.g., *What type of food do you like?*.

In this paper, we then propose a novel approach to build Conversational Recommendation Systems for accessing cultural sites. The key ideas of our approach are: the preferences of the tourists are not asked directly but, rather, acquired during an information seeking phase performed in natural language with the system; the final output are cultural itineraries by connecting POIs taking into account time availability. Generally, during the activity of trip planning tourists search information about cultural attractions in the city and then identify the places to visit. Users implicitly declare their preferences by asking information about the cultural items and the touristic attractions of specific places. We exploit this natural interaction to gather these implicit preferences in user profiles. These profiles are then used by the planner that generates the cultural itineraries. Also, our idea, implemented in the proposed system, is to take into account the temporal availability of the user, and plan an itinerary based on the time that the tourist wants to dedicate to the visit.

The proposed system is designed for dialogues in the domain of cultural heritage, but the overall architecture of the system allows to propose dialogues in different application domains (cinema, finance, medicine, etc.). The customization of the dialogue is done by defining the ontology of the domain of interest.

The rest of the paper is organized as follows. Section 2 presents the related works in the area of recommendation systems. Section 3 introduces the DiEM system that aims to merge conversational recommendation systems with itinerary generators. Section 4 presents the application of the system to some running examples. Section 5 presents the planning for the evaluation. And, finally, Section 6 draws some conclusions.

2 Related Works

Recommendation systems have been largely used for applications in the tourism domain.

The first class of systems suggest points of interest (POIs) by only using the location of the user. Cyberguide [7] and GUIDE [8] are two examples of such systems. Cyberguide [8] was developed to provide maps and information about POIs in closed (e.g., a museum) and open environment. The GUIDE

system, instead, provides information about the city of Lancaster. The system uses a WLAN access point to determine the position of the tourist and provides information through a web interface.

TIP (Tourist Information Provider) [9] introduces a personalization for the suggestions. It selects information about cultural sites to propose, based on user preferences, and suggests the ones that are close to the current position of the tourist. User preferences are acquired by asking to fill out forms.

Then, Varga & Groza and Chiu et al. [3, 10] introduced the use of knowledge bases for semantic searching, to enhance the way of detecting and selecting relevant information. For example, the recommendation system for travel planning presented in [3, 11], based on user preferences, uses DBpedia knowledge base for constructing a holiday trip, choosing a destination city.

Providing touristic itineraries is not new. Some systems that, in addition to make suggestions about POIs, provide itineraries were defined in [12–14, 5, 3]. Deep Map [12, 13] is a mobile system that generates customized itineraries for the city of Heidelberg. During the visit, the system analyzes the position of the tourist in the city and suggests information respect to his position, according to the objectives that the tourist wishes to achieve. The INTRIGUE system [14], instead, generates an itinerary to visit POIs selected by the user. The order of POIs to visit and how to move from a POIs to another one are specified in the generated plan. The planning takes into account the arrival and departure dates of the tourist. In the system presented in [5], the user gives his preferences about restaurants, accommodations, etc., and the system suggests nearby structures (spatially) to the place where the user is located, and suggests POIs that may be of interest for the user. DailyTRIP [4] is a web-based system which generates planning for several days, and for each day offers a different route. DailyTRIP takes into account the user position and preferences (time to visit the sites, the sites of interest for tourists, etc.). The system acquires the user preferences by filling out a form. Also, there is a mobile version of the system, called Mobile mTrip[4].

To the best of our knowledge, all these techniques have never been used in combination with a dialogue agent that implicitly extract user preferences by chatting about POIs for planning cultural itineraries in a city.

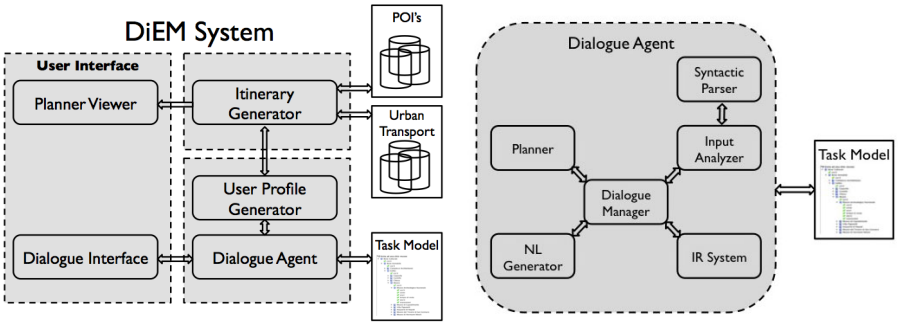
3 The DiEM System

This section describes our approach to generate cultural itineraries by exploiting conversational agents. Section 3.1 describes the general architecture of the system. Section 3.2 introduces our information seeking conversational agent that is used to derive user preferences. Section 3.3 describes the user profile generator. Finally, Section 3.4 describes the itinerary planner.

[4] http://www.mtrip.com/

3.1 The Architecture of DiEM System

The Fig. 1a shows the architecture of the DiEM system. The system is composed of three main modules: *Dialogue Agent*, *User Profile Generator* and *Itinerary Generator*. The Dialogue Agent manages the information seeking dialogue with the user. The User Profile Generator communicates, in a direct way, with the Dialogue Agent and acquires user preferences from the dialogue in order to generate the profile. The Itinerary Generator is the recommendation module: it generates cultural and personalized itineraries based on the profiles of the users.



(a) Architecture (b) Dialogue Agent

Fig. 1: DiEM system

The system has a graphical interface that allows the user to converse with the system (Dialogue Interface) and to visualize the itinerary produced by Itinerary Generator (Planner Viewer). The proposed system has been designed for application in the cultural heritage, but its architecture can be adapted to different domains (Tasks), such as cinema, finance, medicine, and so on. In particular, it can be adapted by: (1) the personalization of dialogues, by defining the domain ontology, called Task Model (see section 3.2); (2) the description of user preferences for the generation of user profile, by defining the rules depending from Task Model; (3) the identification of recommendation services to use for the suggestion of items of domain.

3.2 Dialogue Agent

A Dialogue Agent is a system that communicates with users in spoken natural language in order to perform a specific task ranging from make travel arrangement to question answering in general telephone assistant.

The DiEM Dialogue Agent is designed to assist users to find information about specific cultural objects. The by-product of this interaction is the collection of their preferences. These are represented in a user profile as described in Section 3.3.

Here the processing of utterances and the management of the tourist have to be dynamically combined with the domain knowledge base (Task Model) that contains, in a taxonomy representation, the touristic information used by the dialogue agent during the dialogue session. A user can ask to the system question about cultural related objects (i.e., the ticket price of a museum) and tourist information in general (i.e., certified restaurants). The DiEM Dialogue Agent makes use of a frame-based model for information seeking in support of interactive Question Answering (QA) tasks. The aim is to limit the complexity for the user against cases where multiple and diverging answers are possible according to the retrieved information. The conversation helps the user to disambiguate the ambiguous interactions, to converge to the most useful answer(s) with the minimum number of interactions (turns). The dialogue agent here presented follows the architecture discussed in [15].

The internal modules are almost fully independent from the specific application and constitute a general framework for dialogue-based QA systems. Figure 1b shows the overall architecture of the Dialogue Agent. The modules do not interact directly with each other, but are governed by the Dialogue Manager module in a hub architecture fashion. The hub acts as a router which sends the requests to each specific module. A brief introduction of single module is hereafter discussed, for a more complete discussion see [15].

Dialogue Manager The central component in DiEM dialogue agent is the *Dialogue Manager*. It controls the activation and the flow of information across the individual dialogue components. During the coordination activities, the *Dialogue Manager* controls the dialogue evolution by storing the intermediate information (e.g. the focus of current turns) and activating the suitable state transitions according a finite state *dialogue model*.

The dialogue management task can thus be mapped to a form of navigation through a finite set of dialogue states. These states are directly mapped into a finite set of the speech acts, i.e., states that describe the user intention in the dialogue, such as *question, negation, clarification,..* originate from a limited set of dialogue states. They can be thus assimilated to a finite state (i.e. a Mealy) machine: every dialogue act is performed by the system according to a specific state and a user focus.

In a generic state $s \in S$, the automaton (1) detects the actual speech act characterizing the last user input, (2) changes its state according to the δ function and (3) outputs a symbol according to the output function λ. Output here includes the current topics and a dialogue act (e.g. clarification) used to trigger the Natural Language generation component. In the Dialog Agent, the transition function is also influenced by the current plan: given a state and the speech act characterizing the user reply, it produces the suitable next state. For example, clarifications may follow other clarifications in general, according to the next topics foreseen in the plan.

Natural Language Interaction In order to manage the user input in form of natural language utterances, the dialogue agent supports two levels of analysis: *Speech Act Recognition* and *Focus Detection*. Speech act recognition (SAR) supports the classification of the user utterances into the major Speech Act classes discussed in [16]. This reduces the huge variety of rules required to recognize acts from the user input. The SAR is accomplished by a data-driven supervised model called *Speech Act Classifier*. The analysis of input utterance, moreover, aims to detect the topic expressed by a user and it is implemented in the *Focus Extractor*. This component recognizes the *focus* suggested by a user, by analyzing the grammatical structure of his input utterance produced by the syntactic parser *CHAOS* [17, 18].

Task and Domain Model The interpretation of the user input depends on the available knowledge of the domain of application. An ontology infrastructure is available to the system where the domain and the task are represented. The *Task Model* represents the answers and their corresponding concepts organized into a hierarchy in which general concepts are on the top of the hierarchy, while the more specific ones are leaves. When a concept is selected as result of a user question it is marked as *interesting concept* for the user. This knowledge is accessed by an *Information Retrieval System*, that initially retrieves all the candidate answers to the user question. No constraint is posed to the IR engine except the requirement of deriving a relevance score associated to each candidate answer in the ranked list.

Planner Module The mechanisms used to find the correct response in the Task model, with respect to the user input, requires a plan of the interactions. This plan is based on the initial user question, if such question is ambiguous, may produce multiple answers. The *Planner* module, according to a probabilistic model, computes a set of interactions with the user in order to disambiguate the initial question.

Natural Language Generation The generation of the natural language output is under the responsibility of the *NL Generator* module. The NL generation is based on textual grammars, i.e. *NL Templates* [19]. Given the dialogue status, the proper system responses are detected and templates allow to efficiently compile the output according to the context (e.g. the user focus).

3.3 User Profile Generator

The DiEM User Profile Generator produces the user profile from the dialogue. It is important for the step of the cultural itinerary generation. In fact, the itinerary will be personalized depending on the profile of tourist who wants to visit the city. The system extracts the tourist features from the *interesting concepts* of the Task Model marked during the dialogue between the user and the

system, i.e., preferences about what he wants to visit and behavioural aspects. This module transforms these features in constraints to be respected during the generation of the itinerary. In order to personalize the itinerary, the system takes into account: (1) tourist desires, the type of sites that the tourist wants to visit (church, museum, monument, etc.); (2) tourist interests (nature, food, exhibitions and special events that take place in the city during the period of the visit); and (3) tourist availability for visiting the city, total time that the tourist wants to spend in the visit and his consuetudes (for example the pleasure to walk). This profile is not selected within a pre-established list (for example a class of tourists interested in the church), but it is dynamically built depending on user preferences. The user profile adopted here is based on the user profile defined in [20].

As already mentioned, all the user information are acquired during the dialogue between the user and the conversational agent. The system extracts the information for generating the user profile, based on the questions that the tourist asks to the system. The user preferences are extracted with the application of a set of rules. For example, if the user asks the question "*Is there a botanical garden in Naples?*", it is possible to infer that the user is interested to visit sites like gardens, square, and in general, outdoor locations. Also, it infers that the tourist prefers to walk. In this way, the user preferences are enriched (or updated) for each turn of the dialogue.

The profile generation structure The Profile Generator Structure (PGS) is defined by the quadruple

$$\mathbf{PGS} =< F, P, I, R >,$$

where F is the set of features that characterizes the user profile. Each feature describes a property of the profile and it is expressed through a qualitative value. In the domain of cultural heritage presented, they are *pleasure_to_walk*, *pleasure_to_visit_castles*, etc. The value assigned to the features depends on the preferences P, which is the set of user preferences that the designer, who customizes the system in a particular domain, wants to evaluate. In according to the customization of the proposed system, the set of user preferences to evaluate are *church_interest*, *museum_interest*, *pleasure_to_walk*, etc. Also, the preferences are represented using quantitative values. The preferences have been introduced in order to analyze the user interests, which are fundamental for the generation of user profile. The correspondence between features F and preferences P is realized with the application of a set of rules R. For each feature f_i, a rule is defined as follows:

$$r_i =< f_i, c_i, v_i > with\ r_i \in R$$

where c_i is a condition and v_i is a value. Each condition is defined on one or more preferences and if it is true, the value v_i is assigned to the feature f_i. Finally, I is the set of influences. Each influence enables to define how an element extracted from the dialogue (such as focus, polarity, etc.) affects one or more preferences.

The influences are defined as follows:

$$i_i =< d_i, o_i > with\ i_i \in I$$

where d_i is a condition defined on the task model and o_i is a set of operations. During a dialogue when a sentence talks about a topic covered by the task model that matches with the condition d_i of the influence i_i, the set of the operations defined in o_i are executed. Each operation changes the value of a single preference.

An example. If we consider the previous sentence "*Is there a botanical garden in Naples?*" for retrieving information about user preferences and build his profile, we must define a PGS as follow:

$P = \{pleasure\_to\_visit\_garden, pleasure\_to\_visit\_square, pleasure\_to\_walk\}$
$F = \{garden\_interest, square\_interest\}$
$R = \{r_1, r_2\}$
$r_1 =< pleasure\_to\_visit\_garden, garden\_interest > 5, high >$
$r_2 =< pleasure\_to\_walk, garden\_interest > 5\ and\ square\_interest > 5, high >$
$I = \{i_1\}$
$i_1 =< topic = garden, \{inc(garden\_interest), inc(square\_interest)\} >$

From this structure, the Profile Generator System takes into account the elements extracted from each turn of the dialogue (such as focus and polarity) identified by the Dialogue Manager, and applies the influences defined in I. This step updates the values of the preferences defined in P. Updated the values of the preferences P, the Profile Generator System applies the rules defined in R that allow to define the values of features defined in F. The features F define the user profile.

3.4 Itinerary Generator

An helping system for visiting a city should suggest a cultural itinerary that satisfies the desires of the tourists related to what they want to visit, their behaviour and their time available for the visit. The Itinerary Generator proposed generates cultural itineraries that meet the need of specific users. The system plans the itinerary, performing the following steps (for a more complete discussion about the algorithms see [20]):

- *identification of the cities' areas.* The system identifies areas where are located cultural sites that tourist is interested in;
- *ranking of the areas founded in the previous step.* The system sorts the areas through an index that expresses the user interests of an area, defined in according to the user interests of the sites that belong to the area;
- *evaluation of the time required to visit each sites in the area.* For this evaluation, the system considers the user preferences (for example a user that goes quickly from a site to another or a user that lingers for a long time on some sites) and the time needed by the user to reach sites in the area on foot (in this case it takes into account the pleasure of user to walk);

- *identification of areas to visit in the days when tourists dwells in the city.* In this case, the system takes into account areas of greatest interest and with a visit time at most equal to the time that the tourist wants to dedicate in a day to visit. It provides the list of areas to visit for every single day (the visit of the areas in a day is organized to minimize the distance among areas);
- *identification of public transport to go from one area to another.* The system provides information about transportation to move among areas. The information indicates: the type of means of urban transport (bus, tram, metro), the line to use, the point of departure and arrival.

In addition, for each cultural site included in the itinerary displayed, the system shows (if available) descriptions and photos (they can be stored in multiple sources).

What happens if the system during the dialogue has not acquired enough information about the tourist preferences (*cold start*)? In these cases, the system for the generation of an itinerary uses the indices of notoriety regarding each cultural sites. These indices are used for the selection of the cultural sites to insert in the itinerary of a "generic" tourist. The indices of notoriety adopted are: (1) iconographic importance of a site (e.g. in Naples city, Piazza del Plebiscito and Palazzo Reale have an iconographic index higher than Cappella S. Severo); (2) number of annual visit for the site. The Itinerary Generator implemented is independent from the city to visit. As shown in Fig. 1a, the Itinerary Generator uses existent (external) resources to extract information about POIs and urban transport of the cities, that are different for each city. So, such resources will continue to work and to be updated by habitual users, without any change.

4 Running Examples

In this section, we present an example of dialogue (dialogue excerpt) between a tourist and the dialogue agent, and the related itinerary generated for the city of Naples. The Fig. 2 shows an interaction excerpt between user (U) and dialogue agent (A), and the type of user preferences extracted from it. This dialogue excerpt is a demo dialogue. In the boxes we explained the features that are activated. The first question of dialogue excerpt 1 in Fig. 2 suggests that the tourist is interested in the visit of museums, so the museums are inserted in the list of tourist preferences. Instead, in the second question, the tourist asserts that he doesn't have much time to visit museums: in this case the system acquires the information "little time to spend in the visit". This information will allow to plan the itinerary within the limits of temporal availability of the user. Continuing with the dialogue, in the dialogue excerpt 2 in Fig. 2, the first question suggests that the user is very interested in the visit of square, and using influences as presented in section 3.3, gardens and, in general, outdoor locations. Then, in the second question, the user expresses his pleasure to walk, and so the system, for planning the visit, takes into account this user preference. After the dialogue, the system processes the user preferences for the generation of profile and then

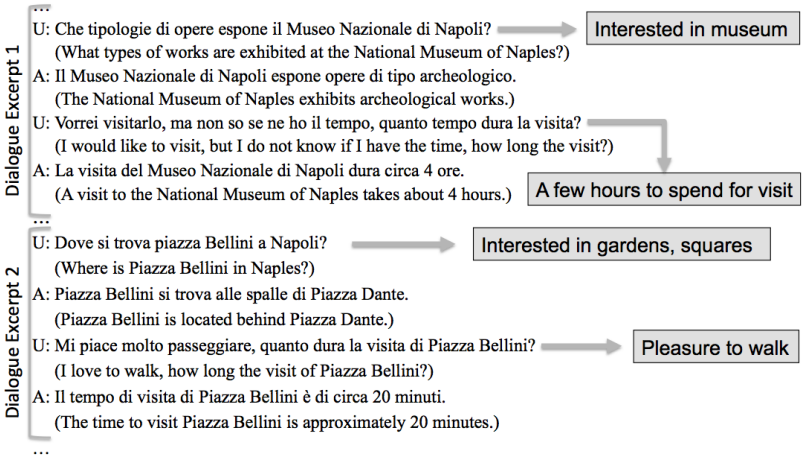


Fig. 2: Dialogue Excerpt

produces the customized itinerary that shows to the tourist through a layout similar to that shown in Fig. 3. The layout reports the map of Naples city. The

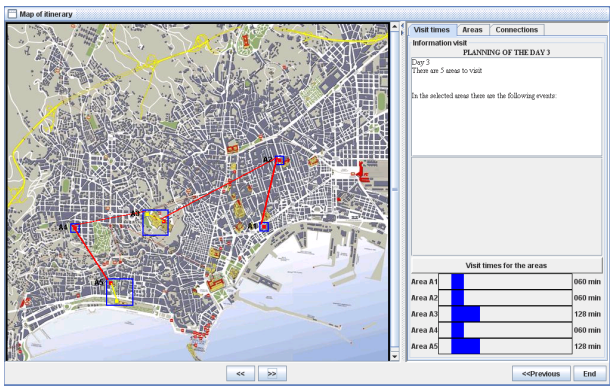


Fig. 3: Personalized Itinerary

map highlights the areas of interest for the tourist, the sites of interest for each area, and the path that connects the sites and areas (itinerary). Also, the layout shows information about urban transport to use for moving between the areas (point of departure, arrival point, and the number of the means of transport). Because the information about itineraries are different for each day planned, the system interface provides a layout for the tourist for each day planned. Also, if the system has information about cultural sites inserted in the planning, it allows tourist to visualize and to consult this information.

5 Evaluation Setup

The evaluation of dialogue systems, as well as for recommending systems, is problematic as it depends on a large set of factors, all of them highly depending on specific applications. Metrics that have been proposed for evaluate these type of systems aim to evaluate the *user satisfaction* in term of task success and to evaluate the implicit user profile acquisition versus the explicit one.

We plan to collect user satisfaction metrics via a questionnaire compiled by the users: they asked to fill out the questionnaire immediately after the completion of a system session. In order to make this evaluation more realistic, we plan to select users representative of a large variety of attitudes and criteria. The adopted criteria for evaluation suggested by the questionnaire were the following:

- *Topic understanding*: the ability of the system in recognizing the main focus of the originating question;
- *Meaningful interaction*: the quality of the system behavior according to the utility of each generated turn;
- *Topic coverage*: the user perception of how good is the system knowledge about the target topic;
- *Contextual Appropriateness*: the ability to produce clear turns consistent with the dialogue progress [21];
- *Interaction/Dialogue quality*: the overall quality of the system to generate sentences. It captures mainly the grammatical correctness of the NL generation;
- *Ease of use*: the usability of the system, that is the system friendliness perceived by the user;
- *User profiling*: The user preferences acquired by dialogue reflect the user profile;
- *Overall Effectiveness*: the user comprehensive judgment about the system usefulness for the planning of a tourist trip.

6 Conclusions

In this paper we have presented the DiEM System, a system for generating personalized cultural itineraries that can be customizable in different application domains such as cinema, finance, medicine, and so on. The main objective of this work has been to define a system for acquiring the user preferences and for generating of cultural itineraries in a natural way, through exploiting conversational agents. The DiEM System generates the cultural itineraries based on user profile that is acquired in implicit way by chatting with the tourist. The proposed approach takes into account user preferences such as tourist desires (type of cultural sites: church, museum, monument, etc.), tourist interests (nature, food, exhibitions and special events that take place in the city during the period of the visit), and tourist availability (temporal constraints, depending on total time for visiting and days of visit). Also, the system suggests information about urban transport to help the tourist to move from one area to another.

References

1. Beccaceci, R., Fallucchi, F., Giannone, C., Spagnoulo, F., Zanzotto, F.M.: Education with living artworks in museums. In: Proceedings of the CSEDU 2009 - ICCSE, Lisboa, Portugal (2009)
2. Zanzotto, F.M., Tudorache, A.G.: Travel with words: An innovative vision on travelling. IEEE/WIC/ACM Int. Conf. on WI and IAT **3** (2012) 107–111
3. Varga, B., Groza, A.: Integrating dbpedia and sentiwordnet for a tourism recommender system. In: ICCP, 2011 IEEE International Conference on. (2011) 133–136
4. Gavalas, D., et al, M.K.: Web application for recommending personalised mobile tourist routes. IET Software **6**(4) (2012) 313–322
5. Sharma, J., Singh, S.P., Singh, P.: Location based information delivery in tourism. IJCSCT **4(2)** (2012)
6. Thompson, C.A., Gker, M.H., Langley, P.: A personalized system for conversational recommendations. Journal of AI Research **21** (2004) 393–428
7. Abowd, G.D., Atkeson, C.G.e.a.: Cyberguide: a mobile context-aware tour guide. Wirel. Netw. **3**(5) (October 1997) 421–433
8. Cheverst, K., Davies, N., Mitchell, K.e.a.: Developing a context-aware electronic tourist guide: some issues and experiences. In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems. CHI '00, New York, NY, USA, ACM (2000) 17–24
9. Hinze, A., Buchanan, G.: Context-awareness in mobile tourist information systems: Challenges for user interaction. In: International Workshop on Context in mobile HCI at the 7th Int. Conf. on HCI with Mobile Devices and Services. (2005)
10. Chiu, D.K., Yueh, Y.T.e.a.: Towards ubiquitous tourist service coordination and process integration: A collaborative travel agent system architecture with semantic web services. Information Systems Frontiers **11**(3) (July 2009) 241–256
11. Esuli, A., Sebastiani, F.: Sentiwordnet: A publicly available lexical resource for opinion mining. In: In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC06. (2006) 417–422
12. Kray, C.: Situated interaction on spatial topics. PhD thesis (2003)
13. Malaka, R., Zipf, A.: Deep map: Challenging it research in the framework of a tourist information system. In Fesenmaier, D.e.a., ed.: ICT in Tourism 2000. Springer Vienna (2000) 15–27
14. Ardissono, L., et al., A.G.: Intrigue: Personalized recommendation of tourist attractions for desktop and handset devices. In: Applied AI, Special Issue on AI for Cultural Heritage and Digital Libraries. Volume 17. (2003) 687–714
15. Basili, R., et al.: Data-driven dialogue for interactive question answering. AIIA 2007 Artificial Intelligence and Human Oriented Computing (2007) 326–338
16. Bunt, H.C.: Context and dialogue control. THINK Quarterly **3** (1994) 19–31
17. Basili, R., Pazienza, M.T., Zanzotto, F.M.: Efficient parsing for information extraction. In: Proc. of the ECAI98, Brighton, UK (1998)
18. Basili, R., Zanzotto, F.M.: Parsing engineering and empirical robusteness. Journal of Language Engineering **8 (2/3)** (2002)
19. Stent, A., et al, J.D.: The commandtalk spoken dialogue system. In: Proceedings of ACL'99. (1999)
20. Gloria, M., Mele, F., Minei, G.: Generazione di itinerari culturali adattabili al turista. AI*IA 2003, Workshop AI for Cultural Heritage (2003)
21. Danieli, M., Gerbino, E.: Metrics for evaluating dialogue strategies in a spoken language system. In: In Working Notes of the AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation. (1995)