

Evaluation of a Domain Independent Approach to Natural Language Processing for Game-like User Interfaces

Manish Mehta and Andrea Corradini

Abstract—Many researchers that develop full software applications in the broad field of natural language processing (NLP) typically implement their single system's components from scratch. While there is nothing wrong with such a methodology from an operational perspective, it typically results in a waste of time. Furthermore, it leads to a substantial diversion of the researchers' efforts from more conceptual and theoretical aspects that could be geared towards the advancement of the state-of-the-art in the field. These main drawbacks call for an implementation approach allowing components' reusability across domains and applications. In that respect, this paper presents an evaluation of a domain independent approach to natural language understanding (NLU) that we have been implementing over the last several years. We have successfully tested and used our approach in three different natural language interfaces to game-like applications, each with its own conversational domains.

I. INTRODUCTION

Several architectures have been proposed for NLP and dialogue systems. Usually they consist of a set of common components, each taking up a very specific assigned task and then passing on its output to other software modules. The conversational system has a standard processing flow typically containing components like for e.g. the dialogue manager, the parser, the speech recognizer, the text-to-speech synthesizer. These, however, are typically designed specifically for the particular conversational application at hand and are not reused. It becomes thus clear that this should be taken into account during the design phase of language technology resources in order to avoid reinventing the wheel which ultimately boils down to a great inefficiency in terms of invested time and efforts. Driven by the significant costs originating from the design, development, testing and maintenance of NLP-based applications, both industry and academia have shown an ever increasing interest in large-scale resources and infrastructure reusability.

In the area of NLP, reusability has been mainly considered for linguistic data resources [2], [10] for benchmark purposes and for the difficulties related to the collection of very large sets of data. If seen from the angle of development of language processing tools, reuse rate is instead rather low.

In general, developing reusable resources is a key challenge for any software application. Reusability across different domains and applications serves the goal of assessing

domain independence, scalability and degree of generalization of a given approach. In a dialog system, the ability to design a concept for one application and port it onto a new one also provides a strong benchmark for the task of crafting the ontological infrastructure that it is based on. In this respect, this paper reports on the evaluation of a domain independent approach to natural language processing that uses ontologies as the underlying representation formalism. Our method has been reused and successfully tested across three different domains in the case of three distinct real time game-like user interfaces. The NLU module that we utilize relies on a rule-based approach where rules are defined to detect a set of conversation features such as domain, domain (in)dependent dialog acts and properties. The approach was initially deployed in the context of a real time conversational agent that we developed to simulate interaction between children and a graphical agent impersonating an historical character. We then extended our original architecture to include new and more conversational domains to account for the topics chosen by the children as they interacted with our system. We realized that during an average 20-minute session, children had the tendency to ask and talk about subjects that were fashionable at the time of their interaction. Hence we decided to add, among others, especially the movies and games domains. We did so through a combination of existing ontological resources with Google's directory categorization. In a third step, we have deployed our approach in the framework of an interactive fiction game. The adaptation of our system in the new domain and application occurred very smoothly and with little efforts, requiring addition of lexical entries and a few grammatical rules tailored to the new application.

The rest of the paper is organized as follows. In Section II we provide the background with a discussion of the three domains where the NLU approach has been instantiated. In Section III we present our domain independent approach towards natural language processing. In Section IV we present the evaluation of our NLU approach for the three domains. Finally we conclude with some future steps we plan to undertake.

II. BACKGROUND

A. First Scenario: Embodied Conversational Agent

Our first domain consists of an interactive game-like interface where a player can interact with an embodied historical character in a 3D world, using a multimodal combination of spoken language and 2D pen gesture as input.

Manish Mehta is with the College of Computing, Georgia Institute of Technology, Atlanta, GA 30332-0760, USA, email: mehtamal@cc.gatech.edu and Andrea Corradini is with the Institute of Business Communication and Information Science, University of Southern Denmark, Engstien 1, DK-6000 Kolding, Denmark, email: andrea@sitkom.sdu.dk

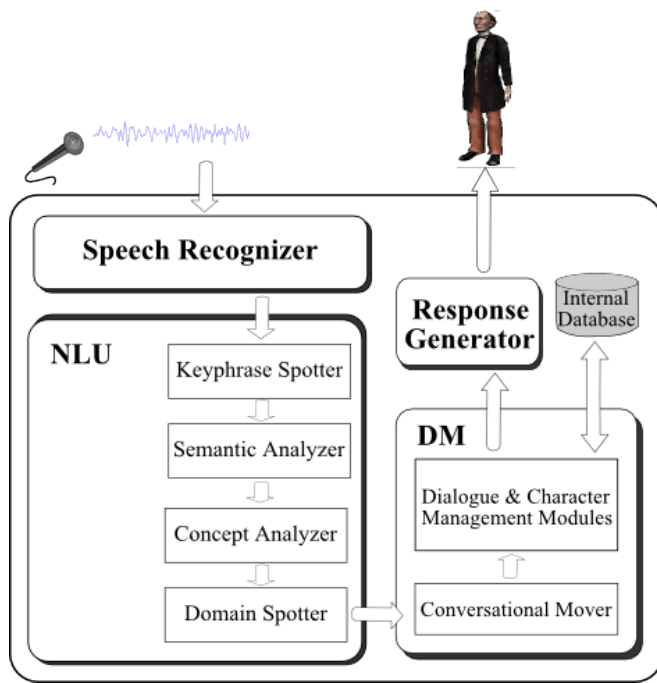


Fig. 1. Overview of the overall embodied conversational agent system architecture.

Edutainment is the domain of our choice since the objective was to merge e-learning and entertainment to allow users to learn about the character's life, historical period and work while simultaneously having fun, hence making them eager to play again [4]. In this setting, we created six domains of discourse: the work of the historical character, his life, his physical presence in the graphical computer generated setting, the user, the character's role as gate-keeper for access to a fantasy world, and eventually a meta-domain to handle problems arising from meta-communication during speech/gesture conversation.

A sketch overview of the architecture this application is built on is shown in Figure 1. The codebase of the NLU of this system architecture is the core of each of the three systems dealt with in this paper and represents the reusable resource over the different application domains.

B. Second Scenario: Conversation on Everyday Topics

Apart from engaging in conversation on few selected character's domains of expertise, we realized that users also address many out-of-domain topics. We thus developed a standalone application that is capable of dealing with these situations. Such an application has as its core the NLU component utilized in the previous setting. This module was expanded to take into account out-of-domain questions via a mechanism that resorts to an agent which searches for suitable conversational turns in the Internet (see Figure 2). In this way, it is possible for the user to discuss everyday topics like movies, games, famous personalities and others by means of the application interface. The appropriate handling of these additional topics is essential to provide the user with a richer entertaining and engaging experience with

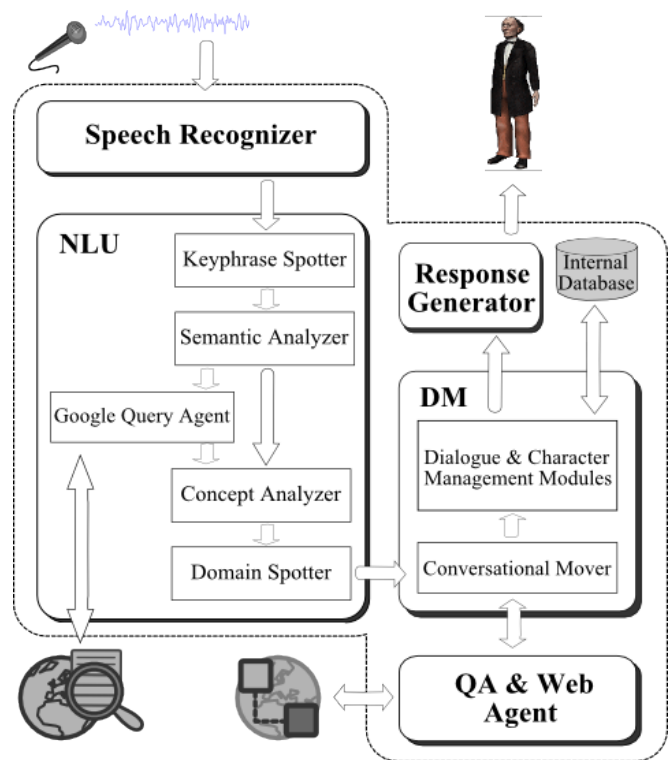


Fig. 2. Overview of the extended embodied conversational agent system architecture that allows for out-of-domain conversation with the 3D graphical character.

conversational characters, rather than ignoring it or by having the system express its inability to address them.

The system exploits a simple and freely available coarse-grained ontology, namely Google's directory' structure, in order to automatically categorize unknown words, and combines it with the existing ontological properties and dialog acts to create an automated semantic representation of a certain input sentence [12].

C. Third Scenario: Anchorhead Adventure Game

As a third scenario we considered a 2D adventure game called Anchorhead [5]. We developed a subset of the complete game that includes a text-based interface for player-game communication [17].

Players can enter commands via keyboard to interact with the game characters and environment. Text-based descriptions of the current scenario are instead presented to the player either below his/her text input or as text bubbles as part of the graphical game. These commands are essentially phrases that encode the players intentions, e.g. "enter the bedroom" to express the intention to move away from the current location into a new one.

The interactive fiction game takes place in the village of Anchorhead, a name that bears a resemblance to those of other fictional towns created by the science fiction, horror and fantasy American author H. P. Lovecraft¹. It is a first-person game i.e. the player impersonates the main character

¹[http://en.wikipedia.org/wiki/Anchorhead_\(game\)](http://en.wikipedia.org/wiki/Anchorhead_(game))

featured. The player explores the village for clues about a mansion that was inherited and has to investigate about the odd history of the previous occupants, the Verlac family. During the game, the player is free to navigate to different locations and interact with several natives, like the bum, the owner of the magic shop or the bartender. Conversational capabilities are enhanced by the inclusion of freely available chatterbots from the Internet.

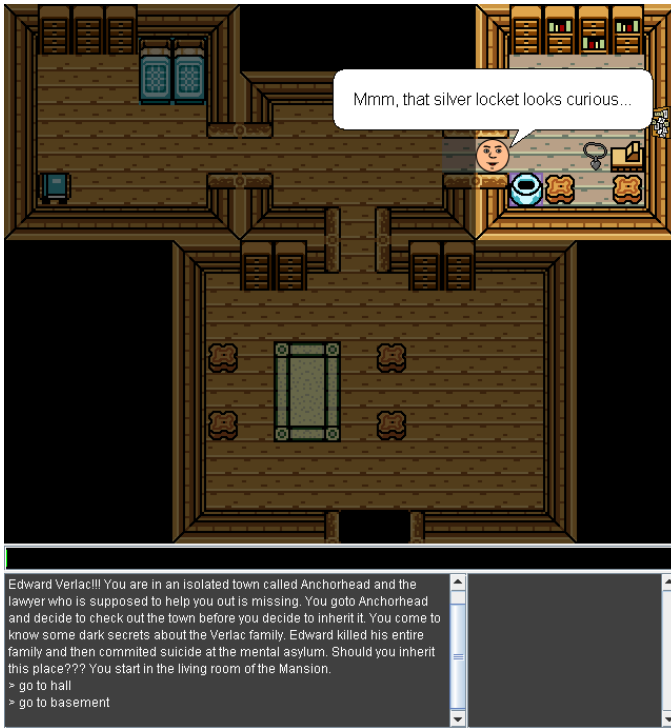


Fig. 3. A screenshot of our graphical implementation of Anchorhead; the NLU underlying the game is the same as used in the first scenario.

III. DOMAIN INDEPENDENT NATURAL LANGUAGE UNDERSTANDING

The natural language understanding module broadly consists of four main components: a key phrase spotter, a semantic analyzer, a concept finder, and a domain spotter (see Figure 4). At the first stage, inside the NLU a key phrase spotter detects multi-word expressions from a stored set of words labeled with semantic and syntactic tags. This first stage of processing usually is helpful to adjust minor errors due to mis-recognized utterances by the speech recognizer or typing errors. Key phrases that are domain-related are extracted, and a wider acceptance of utterances is achieved. The processed utterance is sent on to the semantic analyzer. Here, dates, age, and numerals in the user utterance are detected while both the syntactic and semantic categories for single words are retrieved from a lexicon. Relying upon these semantic and syntactic categories, grammar rules are then applied to the utterance to help in performing word sense disambiguation and to create a sequence of semantic and syntactic categories. The rule engine rewrites certain

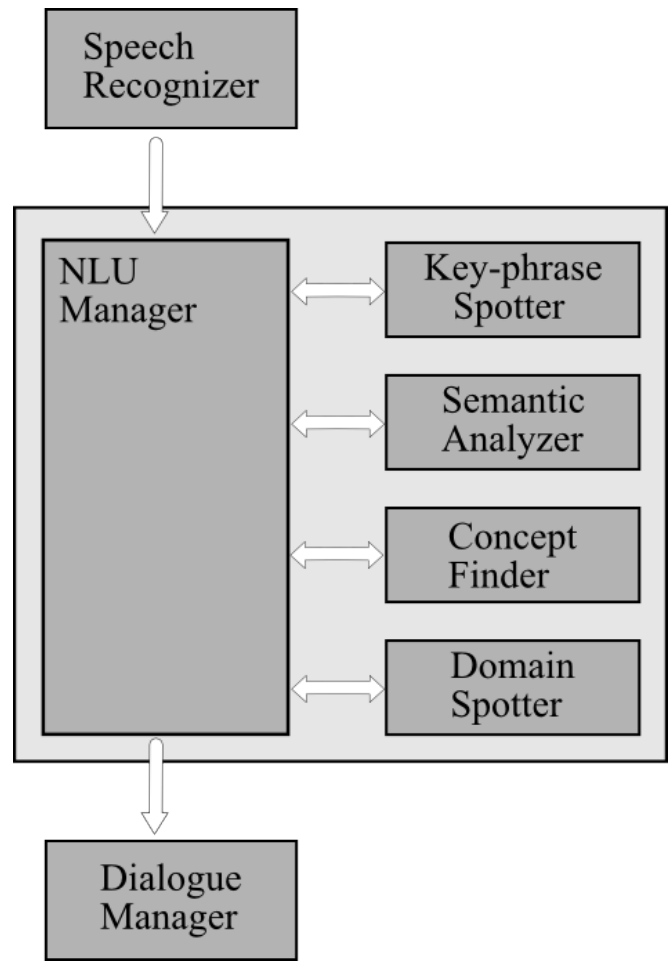


Fig. 4. Overview of the NLU module with its main components.

semantic and/or syntactic categories (or sequences thereof) in terms of other semantic and/or syntactic categories (or category thereof) in order to achieve a simplified encoding of the input. Once this new higher-level representation of the input is created, it is then fed into a set of finite-state automata.

Each automaton is trained using a large amount of linguistic data collected from several user studies and corresponding to a certain specific semantics. In this way, each automaton can be associated to a predefined semantic equivalent. As a direct consequence, anytime a sequence is able to traverse an automaton, the semantic equivalent of this latter is selected to represent the input utterance. At the same time, the NLU determines a representation of the user utterance in terms of dialog acts. At the next stage, the concept finder relates the representation of the original user input in terms of the semantic categories and the domain level ontological representation. Once semantic categories are mapped onto domain level concepts and properties, the relevant domain of the user utterance is extracted. Concepts and properties are grouped according the domain of conversation. The final output in form of concept(s)/subconcept(s) pairs, property, dialog act and domain is sent to the dialog module.

Here, inside the dialog module, the output representation from the NLU is used to reason about the next conversational move of the character. The conversational mover is module that implements a many-to-one mapping between encoded input representations and actions to accomplish that are defined through labels. For each conversational move of the character, rules are defined using the concept(s)/sub concept(s), property(s)/property type and dialog act/dialog act type pairs delivered by the NLU. This provides a systematic way to connect the user intention to the character's output to generate. More details about the system can be found here [11]

In designing the grammar used by the semantic analyzer, we have followed linguistic principles that make it possible for the grammar to scale up as the application grows and permit reuse of grammar parts in the other different application scenarios. Similarly, we have reused large parts of it anytime we have expanded an existing domain or we have added a new one.

Generic rules are defined inside the semantic analyzer. They are used mainly for detecting dialog acts. The dialog acts provide a representation of user intent like types of question asked (e.g., asking about a particular place or a particular reason), opinion statements (like positive, negative or generic comments), greetings (opening, closing) and repairs (clarification, corrections, repeats). Rules for detecting the pool of dialog acts that we are interested in, are defined based on domain independent syntactic knowledge thereby ensuring that once these rules are hand crafted they can be reused across the different domains of conversation.

Altogether, we defined several hundreds rules (currently around 350) out of which approximately one third of them are domain independent. As mentioned before, a subset of our domain independent rules is used for detecting the dialog acts and can be easily reused across different domains of conversations. The rest of the domain independent rules serves the purpose of detecting domain independent properties i.e. features of the content of the input sentence such as dislike, like, praise, certain actions such as read, write, etc. For instance, general wh-questions about a domain are handled by using the domain independent rules to detect dialog acts and properties while simultaneously domain dependent rules are deployed for detecting the concept present in the user utterance.

As explanatory example, given next is one of the rules for detecting a negative user opinion in the input sentence:

```
<user><aux:all> :- <user_opinion:negative>
                    <negative:all>
                    <subject:user>
apply at position :- [beginning]
Number of conditions :- 0
```

This rule is applied to sentences that start with strings like "I was not ..." or "I have not ...". When such a text string is encountered its lexical entries are retrieved from the lexicon and immediately converted into the sequence

'<user><aux:be><negative:not>'. The rule is domain independent and is applied to rewrite that category sequence into '<user opinion:negative><subject:user>' that specifies, among others, the dialog act 'user opinion' and its associated dialog act type 'negative'.

In the first domain, our application has proven very helpful, efficient and robust. When moving on the second scenario, it has also helped simplifying the addition of new domains to increase the range of discussion topics one could have with the animated agent. To the original domains of expertise, the addition of domains like movies and games has provided the conversational agent with the ability to address more general purpose topics. In order to properly capture and understand topics within these new domains during conversation, we utilize Google's directory structure that contains, among other things, updated information and classification of movies and games. For example, if the user asks about a certain computer game that was recently released to the public (and for which we do not have any information in our hand-crafted knowledge base), the NLU uses its internal set of rules to classify user turns into a dialog act, a property using domain independent rules and an unknown concept which is likely to be the name of the game as it was uttered/typed in by the user. This unknown concept is further resolved using Google's directory categorization. Since Google's classification is kept up to date we have the guarantee that fashionable short-lasting concepts or those that we, as designer of the system, did not know at the time of development can still be retrieved. The result of Google's categorization coupled with the domain independent properties and the dialog acts results in an automated representation of the user intent consistent with the current NLU ontological representation formalism.

We have also ported our NLP approach to a modification of the interactive fiction game Anchorhead [16]. In this new scenario, we had to add merely about 150 new lexical entries and 15 new rules. The resources that were already created for the embodied conversational domain with closed domain topics and with open domain topics have been reused in full. In order to understand more in details the reusability of our approach, let's closely analyze a couple of situations occurred from the various evaluation studies described in Section IV.

A. Explanatory Examples

To illustrate the reusability of our NLU approach, we consider a few explanatory use cases, one from each of the three applications.

Scenario 1: Here we have an utterance which was issued by a user to ask about the character's shoes within the embodied conversational agent scenario with six domains of expertise.

Spoken Input: *I don't like your shoes*
 NLU: <dialog_act:user_opinion>
 <dialog_act_type:negative>
 <concept:object>
 <sub_concept:shoes>
 <property:like>

Scenario 2: A similar sentence was input to the conversational character application with open ended domains. In this case the NLP produces the following output:

Typed Input: *I do not think quake was that good*
 NLU: <dialog_act:user_opinion>
 <dialog_act_type:negative>
 <concept:game>
 <sub_concept:quake>
 <property:good>

Scenario 3: Also in the third Anchorhead scenario a user typed in a similar sentence. After NLU processing the representation produced is very similar to those of the preceding use cases. Indeed, this is:

Typed Input: *I am not enjoying the magic shop*
 NLU: <dialog_act:user_opinion>
 <dialog_act_type:negative>
 <concept:location>
 <sub_concept:magicshop>
 <property:like>

In the above three examples, the rule for detecting the dialog act described earlier has been reused across the three domains. Lexical entries for auxiliaries and domain independent properties like 'good' and 'like' are shared as well. More details about the domain independent properties and dialog acts of the system can be found here [11], [3].

In the next section, we discuss the evaluation of the NLU approach across the different scenarios.

IV. EVALUATION

The evaluation of a dialogue system is still a poorly defined and understood task. Due to lack of both a consistent framework and sound theoretical foundations, evaluation is mainly performed through intuitive measures tailored to the application under investigation. In the past, various approaches have been put forward. Some of them deal with the performance of the system as a whole [7], [15] and others are based on the success of single components. The former approaches usually employ a variety of metrics such as task success rate, turn correction ratio, inappropriate utterance ratio, number of turns, concept accuracy, elapsed time and many others [6], [14] in an attempt to evaluate the degree to which the system is accepted by the user. Some other approaches use language input/answer pairs as an evaluation criterion [8], where the correct understanding is defined in terms of the number of right replies to the input sentences. There have also been evaluation methodologies based on the assumption that the performance of the global dialogue system depends on the quality of its single components

| Subject | Wrong Label | Right Label | No Label | Total |
|---------|-------------|-------------|----------|-------|
| 1 | 0 | 13 (86.7%) | 2 | 15 |
| 2 | 1 | 14 (93.3%) | 2 | 17 |
| 3 | 1 | 17 (77.3%) | 4 | 22 |
| 4 | 1 | 15 (83.3%) | 2 | 18 |
| 5 | 1 | 21 (87.5%) | 2 | 24 |
| 6 | 1 | 16 (72.7%) | 5 | 22 |
| 7 | 0 | 6 (100%) | 0 | 6 |
| Average | 0.7 | 14.6 | 2.43 | 17.4 |

TABLE I

Input sentences labeling outcome as determined by the conversational mover in the first application scenario categorized by subject.

and their interactions and cooperation with one another [9], [13]. These approaches have employed the subsystem evaluation techniques that consists of breaking down the entire dialogue system into its components and evaluating them independently from each other.

In our investigation, we distinguish between a qualitative and a quantitative evaluation. For the qualitative analysis, we use the classical criterion of precision i.e. the extent to which the selected conversational move is the correct one, given the input utterance at hand. We measure the precision as the percentage of correctly selected conversational moves by restricting our analysis to the utterances that are successfully processed by the speech recognizer. When humans interact with others, they tend to ignore the fragmented and disfluent qualities of the utterances and focus on extracting meaning in order to make sense of it. The robustness of language interpretation is thus defined as a measure of the ability of human speakers to communicate despite incomplete information and/or ambiguity. Using that as a benchmark for the success of the NLU and the conversational mover components, two human judges (not the authors of this paper) were given the task to independently evaluate and decide whether a third human being would be able to retrieve the meaning of the input sentence given the speech recognizer output. It is thus clear that it was the two judges who decided what were the correct moves and whether there were any shades of grey between "correct" and "incorrect". Interrater reliability was however 98% and as such rather high. A third human judge opinion was used to resolve the problem cases.

A. Evaluation for the First Application Scenario

In order to measure the effectiveness of the domain independent approach, we conducted an evaluation study with 7 subjects (4 boys, 3 girls) from 10 to 18 years old. The users were asked to interact with the conversational agent in a complete system setup. Each user test session had a duration of 45 minutes. A session included about 20 minutes interaction with HCA followed by a post-test interview to evaluate the overall interaction experience. A detailed evaluation based on a structured questionnaire approach was carried out. In this section, we concentrate on the evaluation of the understanding components.

For the quantitative analysis we analyzed the degree of correctness of the NLU output representation and the selection of the conversational move, which is the quantitative measurement on the semantic and pragmatic level of our natural language processor that expresses the quality of the understanding system. The evaluators judged whether and to what extent the NLU and conversational mover were successful in categorizing the input.

Table I shows the classification results obtained from each subject involved in the evaluation experiment. A summarization of this this analysis is presented in Table II that indicates that the system achieved a 82.26% precision measure. The user study also raised some issues as we see next in the following use cases.

| | Wrong Label | Right Label | No Label | Total |
|----------------|-------------|-------------|----------|-------|
| # of sentences | 5 | 102 | 17 | 124 |
| Percentage | 4.03% | 82.26% | 13.71% | 100% |

TABLE II

Summary of the input sentences labeling outcome as determined by the conversational mover in the first application scenario.

Use case 1: In this case the input sentence contains a filled pause as well as a repeat. The NLU is nonetheless able to correctly extract that the user wants to know about the graphical character's childhood.

User: *I would like to know about
<filled pause>like to know about
your childhood*
 SR output: *I would like to know about
like to know about your childhood*
 NLU Output : <dialogue_act:request>
<dialogue_act_type:listen>
<concept:lifetime>
<sub_concept:childhood>
 Conv. Mover : childhood_story

Use case 2: In this case the input sentence is not correctly recognized by the speech processor. The wrong translation from the acoustic signal onto a string representation that is semantically not equivalent to the one corresponding to the actually uttered sentence, makes the NLU produce a response that is not correct.

User: *why*
 SR output: *bye*
 NLU Output : <dialogue_act:greeting>
<dialogue_act_type:ending>

If the speech recognizer had produced the correct string representation of the acoustic signal uttered by the user, the NLU would have found the correct conversational move.

Use case 3: This case is similar to the previous one with the difference that now only part of the input utterance is not properly detected. Now however, errors from the speech

| | Wrong Label | Right Label | No Label | Total |
|----------------|-------------|-------------|----------|-------|
| # of sentences | 25 | 186 | 21 | 232 |
| Percentage | 10.78% | 80.17% | 9.05% | 100% |

TABLE III

Summary of the input sentences labeling outcome as determined by the conversational mover in the second application scenario.

recognizer still make it impossible to get the semantic out of the encoded speech signal. The correct rules within the rule engine could not be triggered to detect the right concept.

User: *i am a student*
 SR output: *and a student twelve*
 NLU Output : <concept:profession>
<concept_type:student>
<property:number>
<property_type:12>
 Conv. Mover : low_confidence

However, if the input were not mis-recognized, the NLU would have been able to extract the right concepts as <concept:user_info><sub_concept:general> <concept:student><concept_type:general> and Conversation Mover output as *user_student*.

B. Evaluation for the Second Application Scenario

In order to measure the effectiveness of the second case study, two independent evaluators conducted an evaluation study. They analyzed the set of out-of-domain inputs from two earlier studies [1], [11] that we ran to assess the whole system on both in-domain topics and various out-of-domain topics.

The input data utilized for the evaluation consisted of 232 sentences that were fed into the system. The analysis revealed that out of the 232 input sentences, the system was able to assign a correct label to 186 of them which amounts to 80.17% of the entire input data set. The system didn't produce any conversational move for some 21 sentences, i.e. 9.05% of the input data. The remaining 25 sentences, amounting to 10.78% of the whole data set, were instead given wrong labels and thus ultimately counts as incorrectly classified. When the system is not able to assign any label, the embodied character expresses its inability to address the input by giving an answer. Table III) shows the result of the label assignment task by the system as produced by the cascaded processing of NLU and conversational mover. There was full agreement between the evaluators on the categorization outcome.

The evaluation also highlighted one of the issue with the approach. The classification approach faces problems when the group of words overlaps with the words in the lexicon. For example, this phenomenon occurs in the following situation:

Question: *Do you like the movie
the Pirates of the Caribbean*
Answer: *I am sorry I don't have an answer*

In this case, the preposition "of" and the article "the" of the movie title "the Pirates of the Caribbean" have a lexical entry in our lexicon. When the system retrieves the categories of these lexical entries the only words that remain uncategorized are "Pirates" and "Caribbean". Because of that, when our system access Google's directory, it is not capable of finding the correct category for the movie. In order to fix this kind of errors, we are developing a module that reads in all keywords of a specific recognized domain and tries to locate them within the string to be processed before processing the string semantically. In such a situation, it would have been possible to locate the key-phrase "Pirates of the Caribbean" and assign it a semantic class before our current semantic processing.

C. Evaluation for the Third Application Scenario

For the third experiment, we invited 20 people. Each player was introduced to the game Anchorhead. The players filled a background questionnaire to obtain personal information such as previous gaming experience or types of games they like to play. At the end of both gaming episodes, participants were interviewed about their experience.

In this section, we focus on the results from the language understanding module. Two independent evaluators looked at the user input and corresponding NLU output and classified it according to the labeling scheme used earlier. The analysis revealed that out of the 1781 input sentences, the system was able to assign a correct label to 1654 of them which amounts to 92.9% of the entire input data set. The system didn't produce any label for some 30 sentences, i.e. 1.7% of the input data. The remaining 97 sentences, amounting to 5.4% of the whole data set, were instead given wrong labels and thus ultimately count as incorrectly classified.

The analysis also revealed some issues that happened during the user study. In few of the cases (around 43 sentences out of the wrongly labelled 97), the NLU could not recover enough information out of the typing errors in the user input. In the case given below, there is a typing error (e.g. the user typed 'steet' instead of 'street') and the system does not take appropriate action that it would have been able to take if the input had been correctly typed as 'could you go to the street'.

Input: *could you go to steet*
NLU Output : <dialogue.act:request>
<dialogue.act.type:general>

We plan to deal with these issues by incorporating a spell checker as part of the NLU preprocessing stage in the future.

V. CONCLUSIONS

In language research a lot of effort is being spent in developing reusable data resources. Similar efforts should focus on reusability of software components for language technology applications as well.

| | Wrong Label | Right Label | No Label | Total |
|----------------|-------------|-------------|----------|-------|
| # of sentences | 97 | 1654 | 30 | 1781 |
| Percentage | 5.4% | 92.9% | 1.7% | 100% |

TABLE IV

Summary of the input sentences labeling outcome as determined by the conversational mover in the third application scenario.

In that respect, we have presented a successful evaluation of a domain independent approach to natural language understanding that we have successfully deployed in three different scenarios, as e.g. in a game context. The core delivery is on reusability. The NLP engine has proved easy to transfer to other domains, rendering our tool quite versatile. Careful evaluation and reusable resources across different domains have allowed us to create a domain independent approach and obtain similar classification percentages (the correct classification rate ranging from 92.9% to 80.17%) over the applications with a minimum adaptation effort to port the NLU components from one system to another.

The experimental results we carried out indicate that the goal of transferability is achieved. We plan to compare our approach with other existing methods. We are aware of the fact that just presenting the percentage of success makes it difficult to judge if the same could be achieved with any other existing approach even if this latter is non-transferrable. At the same time, it is clear that such comparison is very difficult as the experimental setting will probably have to be very different for another method as much of the used rules is hand-crafted. If nothing else is possible we wish to try to compare with the same method with some features disabled to show that it is really due to some specific properties that it works well.

Concerning the experimental setting, on one hand it is clear that human judges are needed if one wants to prove that something is near to what humans would do. On the other hand, it is arguable to state that two judges suffice for this task. While the huge correlation we obtained seems to confirm that two judges are enough, we would like to investigate other possibilities like e.g. whether it is better to design the experiments in a way that the test persons talking to the NLP tool themselves give a response about the correctness of the answers.

In order to allow the linguistic research community to reuse our approach in their domains, as a future step, we aim to release our modules as releasable software.

REFERENCES

- [1] N. O. Bernsen, and L. Dybkjær, Structured interview-based evaluation of spoken multimodal conversation with H.C. Andersen, In *Proceedings of the International Conference for Spoken Language Processing*, October 4–8, Jeju Island (South Korea), 2004, Vol. 1, pp. 277–280.
- [2] C. Cieri, Multiple annotations of reusable data resources: Corpora for topic detection and tracking. *JADT 2000 5 es Journes Internationales d'Analyse Statistique des Donnees Textuelles*, 2000.
- [3] A. Corradini, M. Mehta, and M. Charfuelan, Interacting with an Animated Conversational Agent in a 3D Graphical Setting, In *Proceedings of the Workshop on Multimodal Interaction for the Visualization*

and Exploration of Scientific Data held in conjunction with the 7th International Conference on Multimodal Interfaces (ICMI'05), October 3-6, Trento (Italy), 2005, pp. 63-70.

- [4] A. Corradini, M. Mehta, and K. Robering, Conversational Characters that Support Interactive Play and Learning for Children, *Multiagent Systems*, chapter 18, I-Tech Education and Publishing, Vienna (Austria), 2009, pp. 349-374.
- [5] M. S. Gentry. Anchorhead, 1998, available online at <http://www.wurb.com/if/game/17.html>.
- [6] L. Hirschman, and C. Pao, The cost of errors in a spoken language system. In *Proceedings of the Third European Conference on Speech Communication and Technology*, 1993, pp. 1419-1422.
- [7] L. Hirschman, M. Bates, D. Dahl, W. Fisher, J. Garofolo, D. Pallet, K. H. Smith, P. Price, A. Rudnicky, and E. Tzoukermann, Multi-site data collection and evaluation in spoken language understanding. In *Proceedings of Human Language Technology of a ARPA Workshop*, 1993, pp. 19-24.
- [8] L. Hirschman, Human language evaluation. In *Proceedings of ARPA Human Language Technology Workshop*, 1994, pp. 99-101.
- [9] L. Hirschman. The evolution of evaluation: Lessons from the message understanding conferences. *Computer Speech and Language*, 1998, 12(4):249-262.
- [10] B. Maegaard, K. Choukri, N. Calzolari, and J. Odijk, Elra european language resources association, background, recent developments and future perspectives. *Language Resources and Evaluation*, 2005, 39(1):9-23.
- [11] Mehta M., and Corradini A., Understanding spoken language of children interacting with a embodied conversational character. In *Proceedings of the Combined Workshop on Language-Enabled Educational Technology and Development and Evaluation of Robust Spoken Dialog Systems at ECAI 2006*, 2006, pp. 51-58.
- [12] M. Mehta, and A. Corradini, Handling out of domain topics by a conversational character, In *Proceedings of the 3rd international conference on Digital Interactive Media in Entertainment and Arts*, 2008, pp. 273-280.
- [13] C. Mellish, and R. Dale, Evaluation in the context of natural language generation. *Computer Speech and Language*, 1998, 12:349-373.
- [14] J. Polifroni, L. Hirschman, S. Seneff, and V. Zue, Experiments in evaluating interactive spoken language systems. In *Proceedings of the DARPA Speech and NL Workshop*, 1992, pp. 28-33.
- [15] P. Price, L. Hirschman, E. Shriberg, and E. Wade, Subject-based evaluation measures for interactive spoken language systems. In *Proceedings of DARPA Speech and Natural Language Workshop*, 1992, pp. 34-39.
- [16] S. Ontanon, A. Jain, M. Mehta, and A. Ram, Developing a drama management architecture for interactive fiction games. In *Proceedings of the First Joint International Conference on Interactive Digital Storytelling*, November 26-29, Erfurt (Germany), 2008, pp. 186-197.
- [17] Sharma M., Ontanon S., Mehta M., and Ram A., Drama management evaluation for interactive fiction games. In: *Proceedings of the AAAI-07 Spring Symposium on Intelligent Narrative Technologies*, AAAI Press, 2007.