Extending MMIL Semantic Representation: Experiments in Dialogue Systems and Semantic Annotation of Corpora

Alexandre Denis, Lina Maria Rojas Barahona, Matthieu Quignard

Extending MMIL Semantic Representation: Experiments in Dialogue Systems and Semantic Annotation of Corpora.

Alexandre Denis, Lina M. Rojas-Barahona and Matthieu Quignard
TALARIS team / UMR 7503 LORIA/INRIA
Lorrain. Campus scientifique, BP 239
F-54506 Vandoeuvre-lès-Nancy cedex
alexandre.denis, lina.rojas, matthieu.quignard@loria.fr

Abstract

The MultiModal Interface Language formalism (MMIL) is a modality-independent high-level semantic representation language. It has been used in different projects, related to different domains, and with distinct tasks and interaction modes. MMIL is a metamodel that enables the definition of generic and domain specific descriptors to dialogue management, offering flexibility and high reusability. This paper presents the results of our experimentation with MMIL in diverse projects as well as the recent specifications that cover extensible thematic roles and complex linguistic phenomena.

1 Introduction

The increasing development of natural language processing (NLP) applications, many of them involving several modalities, has highlighted the importance of having an abstract representation language that facilitates the communication among the different modules within the system architecture. Intermediate representation languages, like the one presented in this paper, permit the integration of divergent resources in distributed systems as well as the representation of various levels of linguistic analysis within a single application. MMIL (MultiModal Interface Language) was created as a metamodel, a model that allows developers to define their own model, that provides elements (descriptors) to represent the form and content of linguistic resources in generic dialogue systems (Landragin et al., 2004). For instance, one can use MMIL to represent an utterance syntactically by modeling its surface form. In other cases, one might be interested in representing the semantics or in storing the referring expressions for further discourse processing. In addition, MMIL is ontology-oriented since it makes it possible to associate ontological concepts to its descriptors for the purpose of maintaining the integrity and consistence of both the dialogue and its application domain.

Therefore, MMIL is a language for representing valuable information of linguistic resources. It can be transformed, or translated into other specific formalisms e.g., symbolic formalisms, graphs or domain-specific representations such as flat semantics. Throughout this document, the process of transforming MMIL into other specialized languages is called "projection". In this paper we describe the usage of MMIL as intermediate representation for language understanding and generation within different NLP applications. However, MMIL can be also used in multimodal dialogue systems and projected into languages for emotions representation and modalities synchronization in Embodied Conversational Agents.

This paper briefly introduces the MMIL language. It describes our experience in using MMIL in different projects, such as the MEDIA campaign (Bonneau-Maynard et al., 2009) and CCCP.[1] Moreover, it presents the recent MMIL characteristics for dealing with thematic roles and complex utterances. Furthermore, it illustrates the application of MMIL in the Portmedia Project [2] for semantic annotation.

2 MMIL Intermediate Representation Language

2.1 Background

Although a variety of languages has been proposed for multimodal dialogue systems, MMIL is an ontology-oriented approach that attempt to cover the maximum number of phenomena at sev-

[1] http://recherche.telecom-bretagne.eu/labo_communicant/cccp-prosodie
[2] http://www.port-media.org/doku.php

eral linguistic levels (from lexical up to pragmatics and discourse). It has been used in three European projects – MIAMM (Kumar and Romary, 2002), AMIGO[3] and OZONE (Landragin et al., 2004) – each of them having different interaction mode and application domains (multimedia databases retrieval, train reservation and integration of heterogeneous systems). Contrary to other languages, e.g. Multimodal Markup Language (M3L), Multimodal Presentation Markup Language (MPML) (Prendinger et al., 2004) and the Universal Networking Language (UNL, 2000), MMIL is a metamodel that enables to define generic and domain specific descriptors to dialogue management, offering flexibility in the XML syntax and high reusability (Landragin et al., 2004).

2.2 MMIL Meta-model

The MMIL meta-model allows the representation of communicative actions. A communicative action is represented as a *component*, a structure that gathers the communicative event and its propositional content. It is composed of two main types of entities: *events*, which are entities anchored in the time dimension, and *participants*, which are entities not bounded by time. Entities are linked together by *relations* and are described by sets of *features* (i.e. pairs of attribute-value). Components, entities, features and relations are called MMIL *elements*.

Every component has a unique communicative event, which describes the occurrence of a communicative action and its features, namely the time when it occurs, the speaker and the addressee. The communicative event also bears the illocutionary force, represented through the **dialogueAct** feature, which describes the function applied over the propositional content[4].

The propositional content is represented as a *main event* with its arguments, which can be either events or participants, linked to the communicative event by a relation **propContent**. The main event is not always present in utterances, especially in the case of performing simple communicative actions, such as **Accept**, **Reject** or **Opening** and **Closing**. Nevertheless, in utterances with a propositional content, the main event is required, even in the case of ellipsis where an elliptical event is created. In addition, there should exist a path to the main event and its arguments (the other events and participants of the propositional content).

Suppose that Jack whispers to Bill: *"John ate the red apple"*. In this example (Figure 1, 2), there are two events, the communicative event of whispering, of which the agent is Jack (represented as the feature **speaker**) and the patient is Bill (represented as **addressee**), and the event of eating whose agent is John and whose patient is the red apple, both represented as participants of the propositional content. The adjective "red" is represented by the feature *modifier* inside the participant apple. In this case, the type of the communicative event is **Whisper**, but other communicative types are possible, for example **Show** for a gesture or **Write** for a textual communication.

As mentioned before, within the MMIL framework the agent and patient of the communicative event are not represented as participants, because participants are meant to represent the objects about which something is said and do not extend to the description of the utterance itself. Arguments of other predicates such as adjectives and adverbs are usually represented as participants that have them as *modifiers*. Nevertheless, nominalization or other linguistic representation of actions can be represented as events for the purpose of resolving a given task. In the large-scale lexical resource FrameNet (Baker et al., 1998) predicates are treated as frame-evoking elements calling different frames containing information about the roles of their arguments. However, sometimes in the context of a specific application, one can establish that some predicates are more important than others. Thus, one can consider only the frame evoking elements that are relevant in the context in question. MMIL permits the representation of the utterance's information. The distinct representation of predicates is independent of the information stored, which remains available for further processing such as evoking FrameNet frames.

The MMIL meta-model describes all the possible features that events and participant might have and all their possible values. As such, it covers morphology (*gender*, *number*, etc.), semantics (*objType*, *evtType*, *modifier*, etc.) and pragmatics (*refType*, *mmilId*, etc.). Most of the features have a *default value*, thus, they can be omitted.[5]

[3] `http://www.hitech-projects.com/euprojects/amigo/`

[4] Handling multifunctionality may be done by removing the functionality constraint on the *dialogueAct* feature.

[5] See the MMIL 1.5 specifications for further de-

```
<mmilcomponent>
    <event id="e0">
        <evtType>Whisper</evtType>
        <dialogueAct>Inform</dialogueAct>
        <speaker>Jack</speaker>
        <addressee>Bill</addressee>
    </event>
    <event id="e1">
        <evtType>Eat</evtType>
        <tense>past</tense>
        <person>3rd</person>
        <number>singular</number>
    </event>
    <participant id="p0">
        <refType>properNoun</refType>
        <name>John</name>
    </participant>
    <participant id="p1">
        <objType>Apple</objType>
        <refType>definite</refType>
        <modifier>red</modifier>
    </participant>
    <relation name="propContent"
              source="e1" target="e0"/>
    <relation name="agent"
              source="p0" target="e1" />
    <relation name="patient"
              source="p1" target="e1" />
</mmilcomponent>
```

Figure 1: MMIL representation of Jack whispers to Bill: "John ate the red apple". Here, *Jack*, *Bill*, *John* and *Apple* are ontological concepts in the domain knowledge-base.

2.3 Different instantiations

The MMIL meta-model describes elements and restricts the possible valid structures syntactically. However, it does not describe exactly how to represent a given utterance. The utterance representation depends on how designers intend to use the representation. This means that, the level of detail may vary not only from one system to another, but also from one representation level to another within the same system. Typically, in bottom-up approaches, the system parses the utterance and builds a shallow representation, close to what is expressed explicitly. Afterwards it builds a deep representation of the intention of the speaker. For example the utterances (1) and (2) convey the same intention with two different surface forms. Whereas the surface form is defined standardly in MMIL, the deep intentional form is left free for system designers.

How much does this room cost? (1)

I want to know the price of this room (2)

Shallow instantiation The shallow representation of utterances can be specified using general

tails: http://www.port-media.org/doku.php?id=general_mmil_specifications

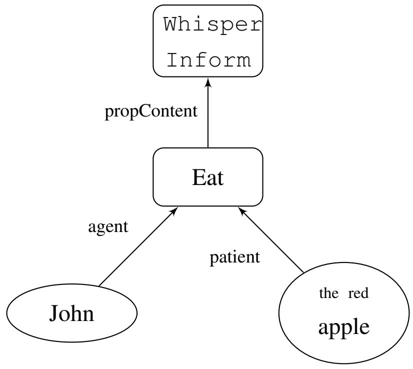

Figure 2: Graph representation of Jack whispers to Bill: "John ate the red apple". Events are depicted as squared boxes, participants as ovals and relation as arrows from the source to the target entity.

purpose principles: *in general*, noun phrases are participants, verbs are events, and modifiers are features [6]. The figures 3 and 4 show the shallow representation of the two utterances (1) and (2).

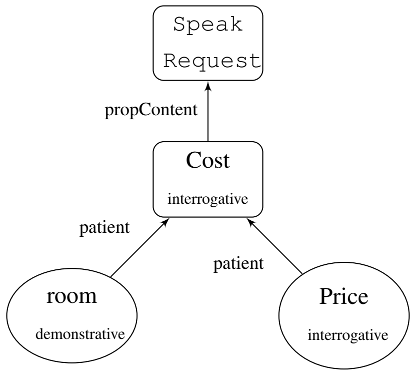

Figure 3: Shallow representation of "How much does this room cost?"

The important aspect of the shallow instantiation is that it should keep the referring expressions. It is well known that two different ways to express the same intention may have two different effects on the context. In our case, it would be weird (if not impossible) to directly refer to the price by a pronoun in the first utterance "How much does this room cost? * Is it high?" while it would be possible to do it in "I want to know the price of this room. Is it high?".

Deep instantiation In contrast to the shallow instantiation, the deep instantiation is just a matter of choice from the system designer. It is generally

[6] In some cases noun phrases can be represented as events if they refer to time variables or if they are nominalization of verbs.

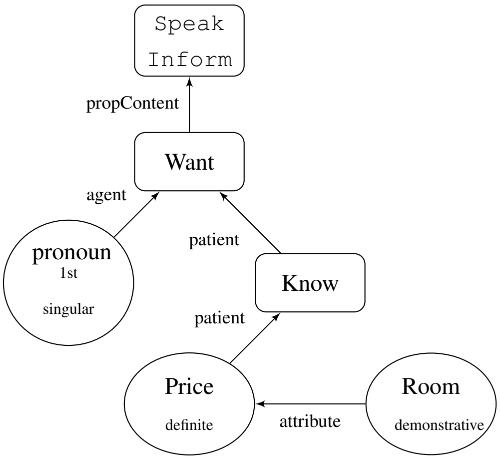

Figure 4: Shallow representation of ”I want to know the price of this room”

advisable that two utterances that bear the same intention are represented the same way, however it is not a requirement considering that the MMIL representation might be projected in other frameworks, such as a logical framework, as explained in (Denis et al., 2006). For instance, a possible deep representation of the sentences (1) and (2) after reference resolution could be having a request with the following propositional content: "*GivenAttributeOf*$(Room(room27))$", where Room is a participant and the id of the room is stored as its feature.

2.4 MMIL for semantic annotation

In order to use MMIL for semantic annotation, it is required to map each MMIL element within a given textual content. The most straightforward mapping consists in: given a textual content, linearly segmented as a list of segments $L = (S_1..S_n)$, in which segments are sequences of words, a *mapping* of a component is a function from each of its elements into continuous sublists of L, such that, the mapping of any element contains the mapping of its sub-elements[7]. Since mappings are continuous, they can be represented on the basis of their left and right boundaries over the segmentation, annotated with the XML attributes *start* and *end*. When these boundaries are omitted for an element, it means it has the same mapping as its parent. The figure 5 illustrates the mapping over a word-level segmentation defined in a TEI (Text Encoding Initiative) compliant format (TEIP5, 2009).

```
<u>
    <w n="0">John</w>
    <w n="1">ate</w>
    <w n="2">the</w>
    <w n="3">red</w>
    <w n="4">apple</w>

<mmilcomponent start="0" end="4" >
  <event id="e0" start="0" end="4">
    ...
  </event>
  <event id="e1" start="1" end="1">
    ...
  </event>
  <participant id="p0" start="0" end="0">
    ...
  </participant>
  <participant id="p1" start="2" end="4">
    <objType start="4" end="4">Apple</objType>
    ...
  </participant>
  <relation name="propContent"
       source="e1" target="e0"/>
    ...
</mmilcomponent>
</u>
```

Figure 5: MMIL for TEI-compliant annotation

3 Recent Usage and Application Domains

MMIL has been used in several NLP applications as an interface language between modules, from which here we present four employments: application queries handling in Prolog, consistency checking in Description Logics, content representation for generation and graph rewriting for interpretation.

In the OZONE dialogue system (Landragin et al., 2004), MMIL has been used as a representation of the messages between modules in a multimodal dialogue system, including the application module, which was implemented in Prolog. Thus, the MMIL components were projected back and forth in Prolog. This was especially useful for the OZONE domain (train reservation) where one can specify some parameters for the request (Prolog constants) whereas other parameters can be let unspecified (Prolog variables). For example the utterance of ”When does the train from Paris to Versailles leave ?” would be first represented in MMIL, and then would be projected into Prolog, that is $train(paris, versailles, Departure, \_)$. The projection was two-fold. First a pattern-matching on the input component retrieved the type of the query and built a Prolog query skeleton. Then, the

[7] The continuity hypothesis limits the possible mappings. It is probably more accurate, but more complex, to handle discontinuous sublists of L.

query was filled by Prolog constants when parameters were provided or by Prolog variables when they were not. Eventually, in this example, the Prolog unification provided a set of possible instantiations for variable $Departure$, which can be represented back into MMIL as a disjunction.

In the MEDIA project (Bonneau-Maynard et al., 2009), the focus was on using MMIL for annotating spoken language utterances in a hotel reservation domain. In contrast to OZONE's domain, the MEDIA's domain was more complex and needed to be defined in an ontology (around 220 concepts). MMIL was first projected into description logics. All the types of entities, *objType* and *evtType* (for example RESERVATION or HOTEL), all the domain-dependent features (such as ROOMTYPE) and relations were then associated to classes or properties in the ontology. It was then possible, from the projection of a component into an Abox, to eliminate the components that were built from syntactically valid but not semantically sound hypothesis (typically a prepositional phrase modifying the wrong entity). In addition, it was possible to specify relations that were lost during the parsing because of disfluencies. Afterwards, the MMIL components were projected into a sequence of semantic features (i.e. a flat list of attribute-value pairs) aligned with the utterance as detailed in (Denis et al., 2006) . The main difficulty of this projection was to flatten the component linearly to match the sequence of words. This was done thanks to the mapping defined in section 2.4.

In the dialogue system presented in (Denis, 2008), MMIL was also used to describe the content that has to be generated by the generation module. While in the OZONE project the generation was template-based, in this dialogue system we used the GenI surface realizer (Gardent and Kow, 2007) to do the generation. Given that MMIL is primarily a representation language, it was possible to easily extract from the components, the parts of the representation that had to be generated and translate them into the flat semantic formalism with variables expected by GenI.

In the latest project, the ongoing CCCP project, in which the task is to profile users in communities of practice, a deep MMIL representation is used to describe the utterance. This deep representation is produced thanks to graph rewriting technique. Thats is, first the components are projected into a generic graph representation, then a rule-based rewriting process occurs, and the resulting graphs are projected back into MMIL. From both utterances "How much does this room cost ?" and "I want to know the price of this room" we are able to produce the same deep representation by matching entities or sub-structures of the input components translated as graphs and by rewriting them. In this example, "How much does X cost ?" would be transformed into a request about the price of X, while the assertion "I want to know Y" would be transformed into a request about Y, resulting in the same graph representation, which in turn would be projected back into the same MMIL component.

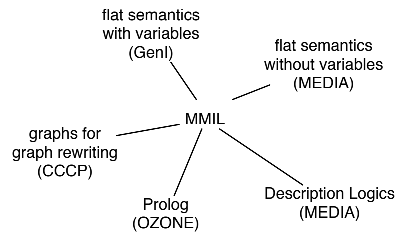

Figure 6: MMIL projections

Therefore, MMIL has been projected into different formalisms for several projects as summarized in Figure 6. This demonstrates its usability and flexibility.

4 MMIL Specification Extension

Previous versions of MMIL (Kumar and Romary, 2002) did not define thematic roles clearly. Relations among events and participants were roughly labeled as *subject* and *object*. Moreover, the representation of complex utterances such as questions, subordination and coordination, was quite limited. Recently, the specification for MMIL 1.5 extends the metamodel with new syntactic and semantic features. This section explains these features together with the strategy for domain-specific semantic roles labeling to be implemented in the Portmedia project for the purpose of annotating semantically the MEDIA corpus.

4.1 Syntactic Features

Questions

Questions are modeled by the communicative act *request* and by the *interrogative* value in the main event's feature *clause type*. Closed questions (yes-no questions) query for the truth-value of the propositional content, whilst open questions (whquestions) query for a particular value (the target) in a propositional content. To distinguish closed and open questions, the value queried in open questions is represented by a participant that bears the interrogative form in its feature *refType* (See Figure 7). Similarly, interrogative adverbs are represented as open questions, but the adverb is indicated in the *relation* between the target and the main event (e.g. manner, cause, time, quantity and location).

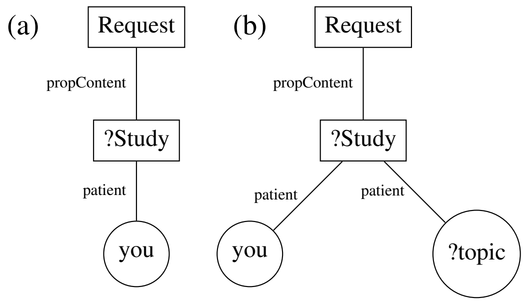

Figure 7: (a) MMIL representation of the close-question: “Do you study?”, (b) MMIL representation of the open question: “What do you study?”

Subordinate Clauses

Subordinate clauses are represented by using the feature *clauseForm* and, in some cases, by using a relation called *dependency* relation. The type of subordination, namely adverbial, relative (i.e. adjective) and noun, is defined in the feature *clauseForm* of the subordinate event. The relation “*dependency*” is usually defined among adverbial clauses and the main clause as illustrated in Figure 8. In relative and noun clauses, on the other hand, the dependency relation is not explicitly represented since the existing relations among either subjects or objects of the main and dependent clauses are preserved, as shown in Figure 9. Note that “one” is the patient of both the verb of the subordinate clause and the verb of the main clause.

Coordination

Coordination was not well defined in previous versions of MMIL. Noun phrases were coordinated together by having *sub-entities* within an entity. Sentences were coordinated by using a relation, however there was no event which gathered together the coordinated entities. Thus, it was not possible to refer to the whole coordination in a referring expression. For these reasons, coordination of noun phrases, adjectives and sentences is now represented by an entity (either an event or participant) which gathers together the coordinated entities and contains information about the type of coordination via the feature *coordType*. The possible values for this feature are *conjunctive*, *disjunctive*, *adversative*, *resultative* and *purposive*, from which conjunctive is the default value. The entities coordinated are linked to the coordination entity by the *member* relation (Figure 10). In order to keep the order of the coordination, the attribute *index* can be used.

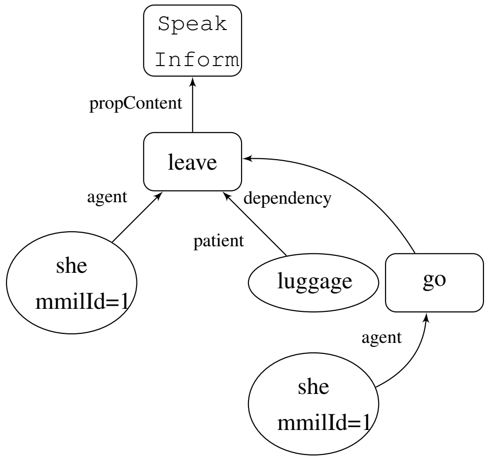

Figure 8: Shallow instantiation of the adverbial clause present in “Wherever she goes, she leaves a piece of luggage”. Note that the pronoun is represented twice bearing the same *mmilId*; this indicates that both pronouns refer to the same person.

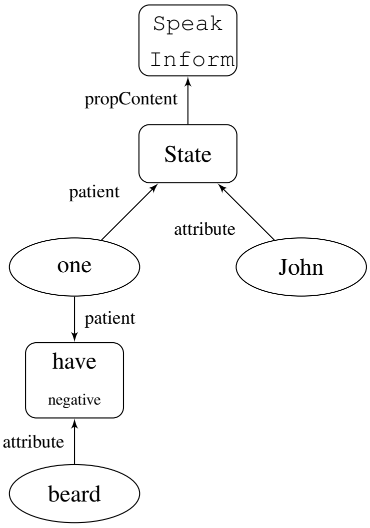

Figure 9: Shallow instantiation of the relative clause: “The one who does not have a beard is John”

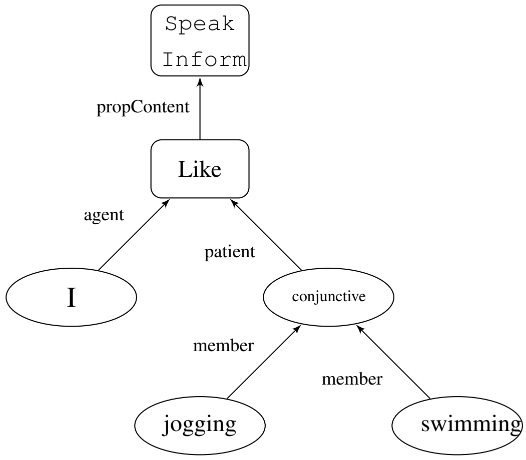

Figure 10: MMIL graph of the sentence: “I like jogging and swimming”.

Thus, coordination entities group together events (even distinct propositional contents under the same dialogue act) and/or participants. Coordination of adjectives and adverbs, on the other hand, is represented inside a special MMIL feature called “*modifGroup*”, which gathers the modifiers (adjectives and adverbs)(Figure 11) .

```
<participant id="p0" >
    <objType>Room</objType>
    <refType>indefinite</refType>
    <modifGroup>
           <modifier index="0">very</modifier>
           <modifier index="1">simple</modifier>
    </modifGroup>
</participant>
```

Figure 11: MMIL representation of “a very simple room”

4.2 Semantic Features

Thematic Roles

Thematic roles have been used to describe predicate arguments by providing them with a semantic description, which is more detailed than simply numbering the arguments. Although the set of role ranges vary greatly from very specific to very general, the research community has not established a clear criteria for semantic role labeling (Gildea and Jurafsky, 2002). Dowty proposes the agent and patient proto-roles (Dowty, 1991) as a solution to this problem. Broadly, he claimed that when the roles of agent and patient are used in arguments, they might have different degrees of membership, because they are not discrete categories. Despite the lack of consensus, sets of semantic roles have been defined in important domain independent implementations such as PropBank (Palmer et al., 2005), FrameNet (Baker et al., 1998), VerbNet (Kipper, 2005) and Lyrics (Lyrics D4.2, 2007).

In MMIL, roles are represented as a *relation* among predicates and their arguments, which can be either events or participants, as shown in Figure 2. The general roles of *agent*, *patient* and *attribute* were adopted as common roles for MMIL representations, in which agent and patient refer to the agent and patient proto-roles respectively:

- Agent corresponds to the agent proto-role, it includes Experiencer and Actor.
- Patient corresponds to the patient proto-role, it also includes Theme.
- Attribute refers to properties (attributes) of an entity, for instance “he is happy”.

MMIL allows to extend this generic roles according to the task, for instance, **Location**, **Instrument** and **Topic**. Moreover, the general roles can be re-defined on the basis of any project requirements. Furthermore, whenever the roles for indirect objects are not explicitly defined in the domain, they can be declared as undefined through unnamed relations. This allows freedom when defining thematic-roles on the basis of the specific needs of a project.

Thematic Roles in Portmedia

The thematic roles proposed in the Portmedia (PM) project are related to predicates in the domain of hotel booking reservation. Portmedia-frames (PM-frames) have been defined for the purpose of ameliorating the relations (i.e. semantic roles) labeling process in a deep MMIL instantiation. Each PM-frame defines the roles of the MMIL representation, based on verb predicates and dialogue acts. Whenever an indirect request is uttered, the deep MMIL will represent the underlying direct request. Thereby, roles are not represented according to the utterance’s surface form.

To clarify this issue, let us present the canonical representation for the *reserve* event, which will be always represented as a *request to reserve*, regardless the illocutionary force of the utterance. That is to say, it does not matter whether the speaker is informing a desire to reserve politely or is simply giving an order. In the case of the reserve event, the underlying requested action concerns the hearer helping the speaker with the reservation task and will be internally represented as Request(Reserve(X_1,...,X_7)), where each argument has a specific role. Therefore, if the *speaker* has just uttered "I would like to reserve", the deep MMIL would be: *Request(Reserve(I))*. The PM-frame states that the argument "I" is the proto-patient, because it represents the ultimate beneficiary after the hearer performs the action requested. The *hearer*, on the other hand, is the proto-agent, because he/she has the obligation to perform the action. The other arguments will have several roles, defined in the knowledge-base, namely the object to reserve, the beneficiary (i.e. the person, not necessarily the same speaker, or people who will use the object reserved), the period of time, the price and the localization of the object reserved.

Consequently, PM-frames are made up of dialogue acts (e.g. request, inform, request acknowledgment, accept, reject), domain-specific events (e.g. reserve, inform, cancel, repeat), semantic roles (either general or domain-specific roles). In addition, PM-frames contain flat semantic chunks (i.e. MEDIA annotation) and lexical units, which can be associated to either the semantic roles or the whole frame. The application of PM-frames in the deep instantiation is reflected by the representation of dialogue-acts, main events and relations among predicates and their arguments. Actually, this deep MMIL will be the new structured semantics of the MEDIA Corpus.

5 Conclusion

We presented in this paper our experience of almost eight years of working with MMIL as an intermediate representation language. Moreover, we described its application in different projects including the ongoing projects CCCP and Portmedia. Each of these projects has different application domains and architectures. Furthermore, MMIL has been applied for different purposes including question answering, dialogue systems and semantic annotation of corpora. The variety of MMIL applications and the way this formalism can be easily projected into other formalisms show the extensibility and high reusability of this representation language.

References

Ashwani. Kumar and Laurent. Romary. 2002. *A comprehensive Framework for Multimodal Meaning Representation*. In proceedings of the International Workshop on Computational Semantics (IWCS-5), Tilburg, Netherlands.

Frédéric. Landragin, Alexandre. Denis, Annalisa. Ricci and Laurent. Romary. 2007. *Multimodal Meaning Representation for Generic Dialogue Systems Architectures*. In Proceedings of the Fourth International Conference on Language Resources and Evaluation(LREC 2004):521-524.

Helmut Prendinger, Sylvain Descamps and Mitsuru Ishizuka. 2004. MPML: a markup language for controlling the behavior of life-like characters. *Journal of Visual Languages and Computing*, 15(2):183–203.

UNL Center (2000) The Universal Networking Language (UNL) Specification, UNDL Foundation

Daniel Gildea and Daniel Jurafsky. 2002. Automatic Labeling of Semantic Roles. *Computational Linguistics*, 28(3):245–288.

Martha Palmer, Paul Kingsbury and Daniel Gildea. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.

Collin Baker and Charles Fillmore and John Lowe. 1998. The Berkeley FrameNet Project. *Proceedings of the 17th international conference on Computational linguistics*, 86–90. Association for Computational Linguistics.

Karen Kipper. 2005. VerbNet: A broad-coverage, comprehensive verb lexicon. University of Pennsylvania.

David Dowty. 1991. Thematic proto-roles and argument selection. *Language*, 67(3):547–615.

A. Denis, M. Quignard, G. Pitel. 2006. A deep-parsing approach to natural language understanding in dialogue system: results of a corpus-based evaluation. *In: 5th International Conference on Language Resources and Evaluation (LREC2006).*

Hélène Bonneau-Maynard and Matthieu Quignard and Alexandre Denis. 2009. MEDIA: a semantically annotated corpus of task oriented dialogs in French. *Language Resources and Evaluation*.

TEI Consortium. 2009. TEI P5: Guidelines for Electronic Text Encoding and Interchange. *Edited by Lou Burnard and Syd Bauman1.4.1.*

Alexandre Denis. 2008. Robustesse dans les systèmes de dialogue finalisé : modélisation et évaluation du processus d'ancrage pour la gestion de l'incompréhension. *PhD Thesis Université Henri Poincaré - Nancy I.*

Claire Gardent and Erik Kow. 2007. A symbolic approach to near-deterministic surface realisation using tree adjoining grammars. *ACL07.*

Lyrics Deliverable D4.2 Defining a preliminary set of interoperable semantic descriptors. [On-line] Available: `http://lirics.loria.fr/doc_pub/D4-2_2007-04-13.pdf`. 2007.