Department of
Information Engineering
and Computer Science
DISI

UNIVERSITY
OF TRENTO - Italy

DISI - Via Sommarive, 14 - 38123 POVO, Trento - Italy
http://disi.unitn.it

# Guidelines for annotating the LUNA corpus with frame information

Sara Tonelli, Giuseppe Riccardi

February 2010

Technical Report # DISI-10-017

# Contents

# 1 Abstract

This document defines the annotation workflow aimed at adding frame information to the LUNA corpus of conversational speech. In particular, it details both the corpus pre-processing steps and the proper annotation process, giving hints about how to choose the frame and the frame element labels. Besides, the description of 20 new domain-specific and language-specific frames is reported. To our knowledge, this is the first attempt to adapt the frame paradigm to dialogs and at the same time to define new frames and frame elements for the specific domain of software/hardware assistance.

The technical report is structured as follows: in Section 2 an overview of the FrameNet project is given, while Section 3 introduces the LUNA project and the annotation framework involving the Italian dialogs. Section 4 details the annotation workflow, including the format preparation of the dialog files and the annotation strategy. In Section 5 we discuss the main issues of the annotation of frame information in dialogs and we describe how the standard annotation procedure was changed in order to face such issues. Then, the 20 newly introduced frames are reported in Section 6.

# 2 The FrameNet project

FrameNet [2] is a lexical resource for English based on frame semantics and supported by corpus evidence, whose aim is to collect the range of semantic and syntactic combinatory possibilities of each word in each of its senses through annotation of example sentences. The conceptual model is based on three main elements:

- **Semantic frame**: the conceptual structure that describes a particular type of situation, object or event and the participants involved in it. Ex. REQUEST
- **Lexical unit** (LU): a word, a multiword or an idiomatic expression that evokes a frame. Ex. for REQUEST: *ask.v*, *beg.v*, *command.v*, *demand.v*, *implore.v*, *order.v*, *petition.n*, *request.v*, *urge.v*.
- **Frame element** (FE): the semantic roles expressed by the syntactic dependents of the LU. Ex. for REQUEST : *Speaker*, *Addressee*, *Topic*, *Message*, *Medium*

The ongoing FrameNet project for English contains 825 frames covering 6,100 fully annotated lexical units. The data are continuously updated and can be consulted on the project website (`http://framenet.icsi.berkeley.edu/`).

Whereas frame description and frame element definition are generally language-independent, lexical units and frame element instantiations are usually annotated in a set of example sentences extracted from a given corpus. An example sentence for the REQUEST frame extracted from the Berkeley FrameNet corpus is reported below. Note that the lexical unit is underlined and that the strings bearing a FE label are between square brackets, followed by the specific frame element:

[Tong]$_{\text{Speaker}}$ ordered [the pilot]$_{\text{Addressee}}$ [to circle Ho Chi Minh City]$_{\text{Message}}$.

Given the two-fold structure of the FrameNet database, we assume we can apply the frame ontology to Italian texts as well, while we have to populate the frames with Italian lexical units based on corpus evidence. In particular, frame instances are annotated in the LUNA Italian corpus (http://www.ist-luna.eu) of human-human and human-machine conversation collected by CSI Piemonte. The annotation process is aimed at creating the first spoken conversational corpus in Italian applying the FrameNet model.

# 3 The LUNA project

The LUNA project (Language UNderstanding in multilinguAl communication systems)[1] has been a three-year project funded under the Sixth Research Framework Programme of the European Union, whose main goal was to enhance real-time understanding of spontaneous speech in advanced telecom services. The project, which ended in August 2009 and operated over Italian, French and Polish, focused on different objectives, namely the language and semantic modeling of speech, the automatic learning and the multilingual portability of spoken language understanding components.

In this framework, a considerable part of the work about semantic modeling of dialogs consisted in the multi-layered annotation of a corpus of Italian spontaneous speech recorded in the help-desk facility of the Consortium for Information Systems of Piemonte Region. The corpus contains

[1] `http://www.ist-luna.eu/`

1000 equally partitioned Human-Human (HH) and Human-Machine (HM) dialogs. The former are real conversations about software/hardware troubleshooting, while the latter are dialogs where an operator acting as Wizard of Oz reacts to the caller's requests following one of ten possible scenarios.

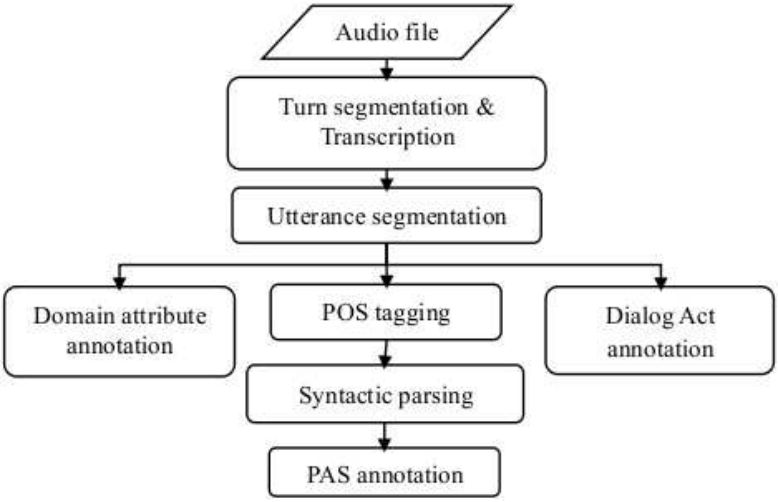


Figure 1: The annotation process

The annotation workflow, described in [6], is displayed in Fig. 1: the dialogs are first recorded as audio files and then segmented at turn level and semi-automatically transcribed. Then, they are further segmented by hand at utterance level[2] and are annotated at three parallel semantic levels:

- The domain attribute annotation is based on a pre-definite domain ontology and specifies concepts and their relations.
- Dialog acts (DAs), which describe the meaning of an utterance at the level of its illocutionary force [1], are annotated following the ADAMACH taxonomy for domain-independent, task-oriented dialogs [10].

[2]The interval when the speaker is active is defined as a *turn*, which is included between two pauses in the speech flow. *Utterances* are complex semantic entities that usually represent the annotation unit for dialog acts. Their relation to speaker turns is not one-to-one, because in most cases a single turn contains multiple utterances, and sometimes utterances can span more than one turn.

- Predicate-argument structure is annotated following the FrameNet paradigm. As shown in Fig. 1, this step requires POS-tagging and syntactic parsing (via Bikel's parser trained for Italian [5]). Then, a shallow manual correction is carried out to make sure that the tree nodes that may carry semantic information have correct constituent boundaries. The details of the annotation workflow are described in Section 4.1 and 4.2.

The multi-level annotation protocol was specifically studied within the project in order to investigate statistical relations between the layers, in particular between semantic and discourse features used in spontaneous conversations. At the time of writing, a further annotation level is being added, namely discourse relations (see [13]), so that the study of the LUNA dialog structure will involve also relationships across different turns, going beyond turn boundaries.

While this study is part of a broad investigation involving several researchers, we focus here on the annotation of frame information.

# 4 Annotation workflow: from tagged files to Tiger / Salsa XML format

The annotation workflow comprises two main phases: the first one is the pre-processing aimed at converting the tokenized and PoS-tagged dialogs into files in Tiger/XML format. The second is the proper annotation step, whose goal is the development of a dialog corpus with manually annotated frame information at utterance level.

The overall process is composed by five steps, as shown in Fig. 2. The steps from 1 to 4 all belong to the pre-processing phase, while the last one is the final annotation carried out by hand.

A thorough description of all steps is presented in the following subsections.

## 4.1 Corpus preparation

This subsection details the four steps in the pre-processing phase aimed at converting the LUNA dialogs from raw text into Tiger XML format.

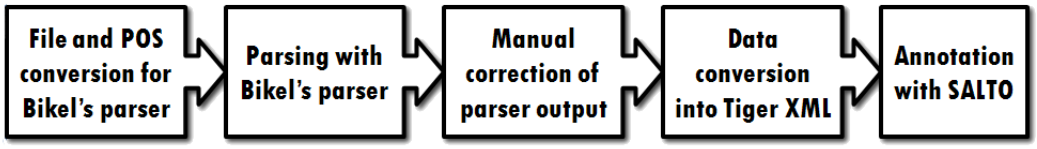


Figure 2: Steps for annotating the LUNA corpus with frame information

#### 4.1.1 Format and PoS conversion

In the first step, we have to convert the transcribed, tokenized and PoS-tagged dialogues stored in .xml or .pro format into the formalism required by Bikel's parser. Since the dialogs to be annotated have been automatically PoS-tagged using the Chaos tagset [3], we have to convert the original PoS into the PoS tagset suitable for Bikel's parser trained on Italian [5].

We follow two different procedures for Human-Machine (HM) and Human-Human (HH) dialogues because they present different formats. For the first, we convert the data from the files ending with *_words.xml* using a Perl script. For HH dialogues, we extract the sentences from the files ending with *.pos*, reconstruct overlaps and list in a separate file which original sentences belong now to each turn. In both cases, we discard *FILLER* words. Every dialog is stored in a file with one turn per line.

The conversion table is displayed in Table 1. Note that VIN labels are converted into VMA only in case they do not correspond to a modal verb. Otherwise, the label becomes VMO. We introduced a regular expression for the identification of modals.

As an example, we report in Figure 3 an utterance extracted from a HM dialog in the original xml format and after the conversion. Figure 4, instead, shows the original tabular format of HH dialogs and the utterance after conversion.

| Original tagset (CHAOS PoS tags) | Parser Label | Description |
|---|---|---|
| PSG, PPL, PRR | PRO | Pronoun |
| ADS, ADP, AGS, AGV, AGP | ADJ | Adjective |
| NCS, NCP, NC, NPR, TAG | NOU | Noun |
| AVV | ADVB | Adverb |
| ARS, ARP | ART | Article |
| COP | CONJ | Conjunction |
| PSE, PAP, PAS | PREP | Preposition |
| NUM | NUMR | Number |
| VFT, VFI, VNT, VNI, VNP, VIP, VTR, VIN | VMA | Main verb |
| VX | VAU | Auxiliary verb |

Table 1: Conversion table CHAOS PoS - Bikel's parser

```
<words>
<w id="word_1" word="Per" lemma="per" cat="PSE" morph="invariante" />
<w id="word_2" word="quale" lemma="quale" cat="PRR" morph="mas.sing" />
<w id="word_3" word="ente" lemma="ente" cat="NCS" morph="mas.sing" />
<w id="word_4" word="lavori" lemma="lavoro" cat="NCP" morph="mas.plur" />
<w id="word_5" word="?" lemma="?" cat="CO" morph="invariante" />
</words>
➔
( (Per (PREP))(quale (PRO))(ente (NOU))(lavori (NOU))(? (?)) )
```

Figure 3: Format conversion of HM utterances

```
perche' COP
avevate VFT
mandato NCS
la ARS
mail NC
➔
( (perche' (CONJ)) (avevate (VMA)) (mandato (NOU)) (la (ART)) (mail (NOU))
(. (.)))
```

Figure 4: Format conversion of HH utterances

### 4.1.2 Parsing

The Italian sentences are parsed with Bikel's phrase-based statistical parser trained for Italian [5], which obtained the best score in the evaluation cam-

paign Evalita 2007 for Italian NLP tools with 70.79 f-measure. The parser output is in Penn-Treebank format, as follows (note that this is the same utterance of Fig. 4):

(S (CONJ perche') (S (VP (VMA avevate) (VP (VP (VMA mandato)) (NP (ART la) (NOU mail))))) (. .))

#### 4.1.3 Manual correction of parse trees

In order to annotate constituents with the correct label and the right span, we carry out a shallow correction of parse trees. The correction is manual and is realized with the help of an online visualization tool called *phpSyntaxTree* (`http://ironcreek.net/phpsyntaxtree/`), that takes the parenthesized format of the sentence as input and displays a parse tree in output.

To our knowledge, no visualization tool that allows annotators to directly correct the parse trees in the graphical environment is available. For this reason, the correction step cannot be carried out directly on the displayed tree, but has to be manually accomplished in the output file of the parser. We do not correct all trees because it would be too time-consuming. We only correct those nodes that we assume to be good candidates for bearing frame information. For example, the sentence "*Indossava occhiali dalla montatura leggera*" (*He wore thin-rimmed spectacles*) was parsed as:

(S (VP (VMA Indossava) (NP (NOU occhiali)) (PP (PREP dalla) (NP (NOU montatura) (ADJ leggera)))) (. .)) (Fig. 5).

The parsing delivered an error in the PP-attachment. The correct version is:

(S (VP (VMA Indossava) (NP (NP (NOU occhiali)) (PP (PREP dalla) (NP (NOU montatura) (ADJ leggera))))) (. .)) (Fig. 6).

#### 4.1.4 Format conversion

In order to annotate the dialogs with the SALTO tool [4], data in Penn-Treebank format must be first converted into Tiger-XML format [8] using the freely available TIGERRegistry tool (`http://www.ims.uni-stuttgart.de/projekte/TIGER/TIGERSearch/doc/html/TIGERRegistry.html`). This application supports many popular treebank and parser output formats, e.g.

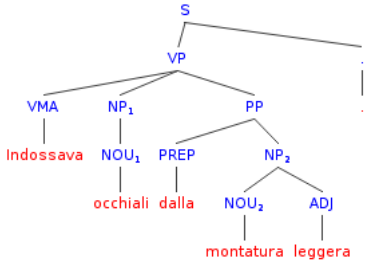


Figure 5: Wrong parse tree

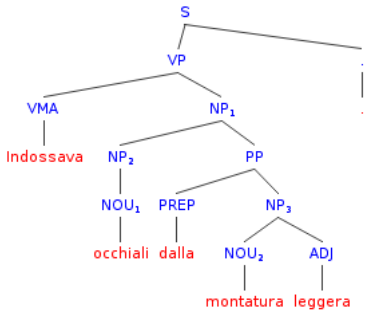


Figure 6: Correct parse tree

Penn Treebank, SWITCHBOARD, Susanne and Negra, and converts them into the Tiger-XML format required by the tool for frame annotation. In such format, which represents a standard for XML-based annotation of syntactic information, every sentence is seen as a `<graph>` consisting of `<terminals>` and `<nonterminals>`. The `<terminals>` element is a list of `<t>`erminals with PoS information reported as attribute. Instead, `<nonterminals>` include a list of syntactic nodes `<nt>`. Within each node, the `<edge>` label links the node to its direct constituents (`<t>`s or `<nt>`s). An example sentence in XML-Tiger format is reported in Fig. 7, where the sentence of Fig. 6 is displayed.

The sentence number is 3430-459771, which is repeated in every terminal id (e.g. 3430-459771_1 etc.) and in every node id. Words are numbered in increasing order and are described by the PoS feature. Non-terminal nodes are listed separately with the category label and the edges that they include. Nodes are numbered starting from 500 and follow a top-down order, from the root node down to pre-terminals.

The conversion step using TIGERRegistry is quite straightforward. The annotator has to select 'Corpus/Insert corpus' in the menu, load the parsed file (browse with the button 'Choose') and finally select the settings as shown in Fig. 8.

Note that it is important to keep the same file name as the original one preceded by *Hum_Hum* or *Hum_Mach*, depending on the dialogue type. In case the input data format is not consistent, an error message will be displayed. Otherwise, the tool produces two xml files, one is the header

```
<s id="3430-459771">
    <graph root="3430-459771_500">
        <terminals>
            <t id="3430-459771_1" word="Indossava" pos="VMA"/>
            <t id="3430-459771_2" word="occhiali" pos="NOU"/>
            <t id="3430-459771_3" word="dalla" pos="PREP"/>
            <t id="3430-459771_4" word="montatura" pos="NOU"/>
            <t id="3430-459771_5" word="leggera" pos="ADJ"/>
            <t id="3430-459771_6" word="." pos="."/>
        </terminals>
        <nonterminals>
            <nt id="3430-459771_503" cat="NP">
                <edge label="--" idref="3430-459771_2"/>
            </nt>
            <nt id="3430-459771_505" cat="NP">
                <edge label="--" idref="3430-459771_4"/>
                <edge label="--" idref="3430-459771_5"/>
            </nt>
            <nt id="3430-459771_504" cat="PP">
                <edge label="--" idref="3430-459771_3"/>
                <edge label="--" idref="3430-459771_505"/>
            </nt>
            <nt id="3430-459771_502" cat="NP">
                <edge label="--" idref="3430-459771_503"/>
                <edge label="--" idref="3430-459771_504"/>
            </nt>
            <nt id="3430-459771_501" cat="VP">
                <edge label="--" idref="3430-459771_1"/>
                <edge label="--" idref="3430-459771_502"/>
            </nt>
            <nt id="3430-459771_500" cat="S">
                <edge label="--" idref="3430-459771_501"/>
                <edge label="--" idref="3430-459771_6"/>
            </nt>
        </nonterminals>
    </graph>
</s>
```

Figure 7: Example of Tiger-XML format

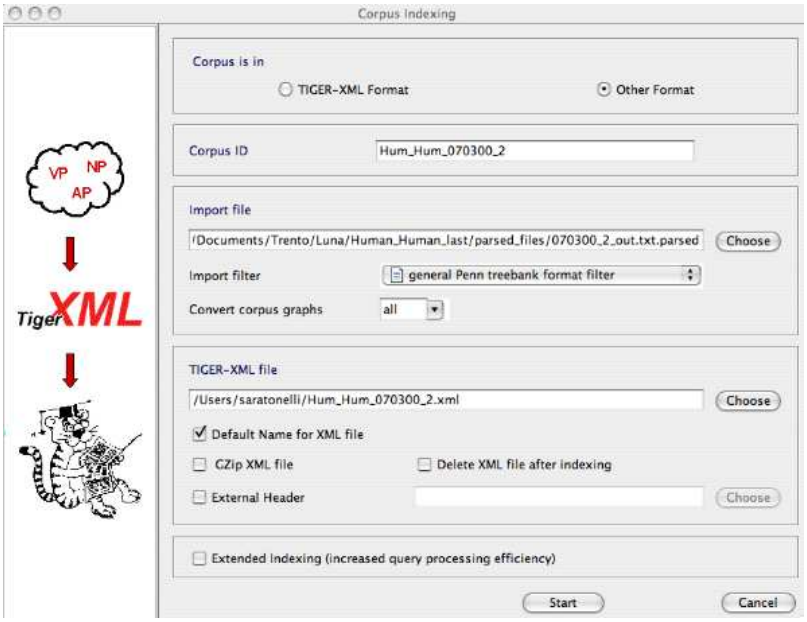


Figure 8: Screenshot of TIGERRegistry conversion module

(FILE_NAME_generated_header.xml) and the other one the main xml file with the dialog data (FILE_NAME.xml).

## 4.2 Annotation of frame information

Manual annotation of frame information is carried out using SALTO [4], a freely available Java application that can be downloaded at `http://www.coli.uni-saarland.de/projects/salsa/page.php?id=software`.

The tool can load parsed sentences in Tiger-XML format, displays them as parse trees and gives the possibility to add frame and FE labels pointing to the tree nodes. An example sentence displayed with SALTO is reported in Fig. 9.

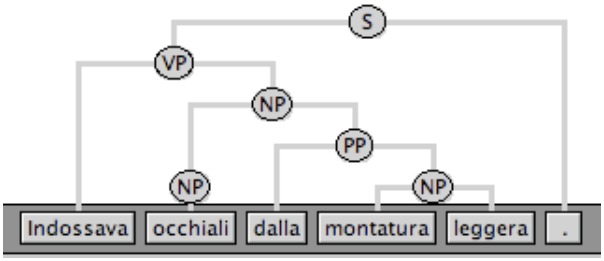


Figure 9: Parse tree displayed with SALTO

After opening the application, every annotator has to log in with his account name (the box 'Administrator' in the login screenshot should not be selected). A folder will be created under 'Salto/repository/user/YourName/work'. Then, all files to be annotated have to be copied in the 'work' subfolder.

Annotation proceeds file-wise for each dialog. It is possible to open a file to be annotated by clicking on the file symbol in the 'work' folder of the 'user' window (lower left on the screen). During loading, the application displays a warning message that can be ignored.

Annotation of the LUNA corpus comprises three main steps, that we describe in the following:

### 4.2.1 Selection of target words

In the Berkeley FrameNet project two annotation strategies have been adopted: in the first project phase, target annotation was carried out lemma-by-lemma,

i.e. for every lexical unit all its corpus attestations were collected and then assigned to a given frame, so that in every sentence only one LU was annotated. In the following phase, annotation of continuous text was started, meaning that in a given text, annotators had to identify and annotate all possible target words following a sentence-by-sentence fashion. In the LUNA project, a third task-oriented annotation strategy was adopted: annotation should involve all words that are *semantically relevant* to the domain and the assistance task. In general, we consider relevant:

- Verbs/nouns classified as actions at attribute-value level of annotation ex. *accendersi*, *funzionare*, *verifica*, *check*
- Verbs of statement and of opinion that introduce the speaker's attitude, ex. *credere*, *sapere*
- Verbs and nouns describing the office environment, the general situation where a dialog takes place, or the problem that is being described

In general, we consider *relevant* all the verbs that convey semantic information that is important to describe the setting, the situation, how the speakers behave, etc. We don't annotate sentences without verbs, sentences composed only by numbers (ex. Rws number), sentences which are semantically not relevant/not clear. For instance, the following sentences have no relevant targets:

(1) allora centossessantaquattro diciassette okay a nome di evangelista
angela per ca
ecco io ho provato poi
si puo' fare un po' di
(so one hundred sixty-four seventeen okay with the name of evangelista
angela for ins
so I tried then
we can do some)

### 4.2.2 Annotation of frame label

After selecting the target, the annotator has to identify the conceptual situation evoked by it and to assign a frame label. With SALTO it is possible to import the frame list from the FrameNet database and to assign a frame

label to a target just by clicking on one item in the list. Otherwise, it is possible to add newly created frame descriptions.

Technically, annotators using SALTO have first to load the frame definition by selecting 'Edit frames' in the 'Corpus' menu. By clicking on 'Add frames', the list of all available frames in FrameNet 1.3 will be displayed in the left column. After selecting a frame and choosing 'Ok', the selected frame will be available for annotation. Then, annotators have to double click (or right-click) on the target word being a leaf in the displayed syntax tree and select one of the frames listed as shown in the picture below:

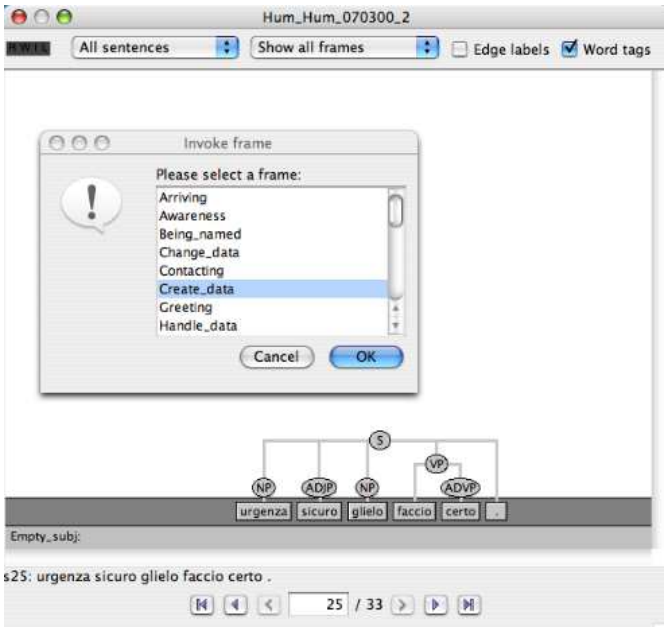


Figure 10: Frame assignment with SALTO

The **frame assignment** task can be problematic in case of ambiguous targets or when no frame seems to be suitable for the assignment. In general, annotators should first look at the textual definition of the candidate frames and at the list of available frame elements on the FrameNet website. They should check if this definition can be applied also to Italian sentences, and see if the candidate frame(s) contain an English translation equivalent of the Italian target. In case of doubt, it is recommended to look at the example sentences available for the English target in order to compare the usage of the

English and corresponding Italian LUs. Annotators should also check if some annotated sentences in Italian containing the given LU are already available. If the Italian example sentence to annotate contains an ambiguous target, it is also recommended to try and paraphrase it using a possibly unambiguous LU, so as to clarify which the evoked frame is.

Some ambiguities that are present in English can occur also in Italian, for example the polysemous verb *ask.v* evokes both QUESTIONING and REQUEST, as does the Italian translation equivalent *chiedere.v*. In other cases, frame assignment in Italian is more straightforward than in English because the alternation between reflexive and non-reflexive forms is captured by different frame types, as in *svegliarsi.v* (*to wake up*, intrans.), belonging to WAKING_UP, and *svegliare.v* (*to wake up*, trans.) in CAUSE_TO_WAKE_UP. In English, instead, the verbs *wake.v*, *wake_up.v* and *got_up.v* appear in both frames and there is no distinction between the reflexive and the causative form.

If frame assignment is still problematic after looking at the frame definitions and at the English examples, annotators should try and match the FEs provided for every candidate frame to the subcategorisation pattern of the current Italian target. If the target in the example sentence has some (realized or unrealized) arguments that do not correspond to any FE of the candidate frame, then the frame has to be discarded. For example, the target *collegare.v* (*to connect*, trans.) in the sentence "*Il tecnico collega la stampante alla rete*" (*The technician connects the printer to the net*) could in principle be assigned to the following candidate frames:

(2) INCHOATIVE_ATTACHING: An Item comes to be attached to a Goal, with a Connector forming a bond between the Handle of the Item and the Goal.
ATTACHING: An Agent attaches an Item to a Goal by manipulating a Connector, creating an asymmetric relationship between the Item and the Goal.

The INCHOATIVE_ATTACHING frame is clearly not appropriate to the sentence because its definition does not include the *Agent* frame element, which is the role of "*Il tecnico*" (*The technician*) in the Italian sentence. On the contrary, ATTACHING is a suitable frame because it includes the roles *Agent*, *Item* and *Goal* needed to annotate all the constituents in the example sentences.

Despite these suggestions, there are still cases in which it is very difficult to make a decision between two frames, because they express related meaning components. For example, the sentence "*Credo che X*" (*I believe that X*) without further context could belong both to the AWARENESS frame (to have a fact in his/her mental representation, as a belief or knowledge) and to CERTAINTY (to be certain of a fact). The assignment decision should depend on which of the meaning components is dominant in the example at hand. If it focuses on expressing the content of the belief/knowledge, like in "*So che X*" (*I know that X*), AWARENESS is more appropriate; if the main information is about the degree of certainty of the belief, like in "*Sono sicuro che X*" (*I am sure that X*), it is a case of CERTAINTY. Again, frame assignment could benefit from paraphrasing the example in order to stress the role of its meaning components. Anyhow, such examples involve case-to-case decisions which can be influenced by the annotator's interpretation.

### 4.2.3 Annotation of frame elements

After a frame has been assigned to the target in a sentence, annotators have to **identify the frame elements**. If a frame label has been chosen for the target, SALTO displays automatically the core FEs available for the given frame, as shown in Fig. 11 for the ADDICTION frame pointing to *dipendenza.n* (*dependency*) as a target. A FE label is assigned by dragging it onto a syntactic constituent. Other labels for peripheral and extra-thematic FEs can be added by hand.

The annotated information is internally recorded in Tiger/SALSA XML format [7], a modular extension of Tiger-XML where syntax and semantics are stored independently. The semantic information layer is contained in the additional element `<sem>`, while the syntactic representation is still contained in `<graph>`, as represented in Fig. 12. `<sem>` contains the semantic information for the current sentence, with unique identifiers for all semantic nodes and edges. In our annotation, the only semantic information encoded is about frames, and is contained in the `<frames>` element. `<frames>` can include the annotation of different frames in the same sentence (between `<frame>` tags). For each `<frame>`, `<target>` and `<fe>` can be encoded and connected to the tree nodes. Even if all information about a sentence is included within one `<s>` element, the different annotation levels, namely `<graph>` and `<sem>`, are kept in separate blocks. However, they can be straightforwardly related through pointers from semantic labels to syntactic nodes.

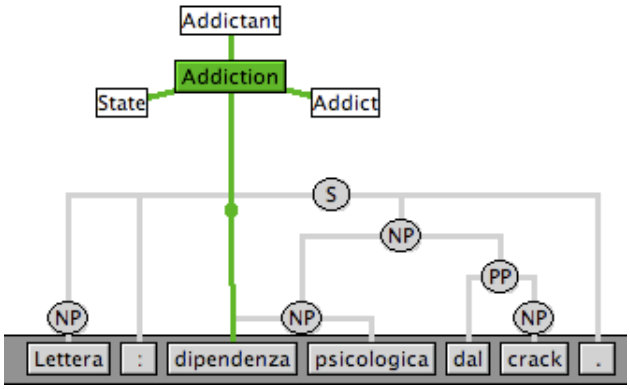


Figure 11: Core FEs available with SALTO. Transl.: "*Letter: psychological dependency on crack*"

Generally speaking, no frame element occurs necessarily. There may be frames which don't have a single realised frame element and are attested only by the target. On the other hand, multiple occurrences of the same frame element are allowed in specific cases. The most frequent case involves repetitions, redundancies, discontinuous constituents, etc. and are typical of the conversational style of the texts. As an example, consider Figure 13. We assign the *Theme* role to "*tempistiche*" (*deadline*) and to "*le*" (*it*) because both express the same frame element.

Other cases of **FE duplication** involve the annotation of wrong parse nodes. Since the syntactic structure was created automatically and only partially corrected, the parse trees can contain some errors, for instance a constituent could be wrongly split in two phrases. Even if we cannot modify the constituent structure, we can duplicate a FE label so that it covers all terminals expressing the corresponding role, even if the nodes are wrong. For example, in Figure 14 the conjunction "*che*" ( *that*) is not included in the following S node "*ha dei problemi sull'ascolto dei file audio*" (*you have some problems with the audio files*). Since we cannot modify the constituent structure, we duplicate the *Message* FE and repeat the assignment for "*che*" and for the S node.

```
<s id="3430-459771">
    <graph root="3430-459771_500">
        <terminals>
            <t id="3430-459771_1" word="Indossava" pos="VMA"/>
            <t id="3430-459771_2" word="occhiali" pos="NOU"/>
            <t id="3430-459771_3" word="dalla" pos="PREP"/>
            <t id="3430-459771_4" word="montatura" pos="NOU"/>
            <t id="3430-459771_5" word="leggera" pos="ADJ"/>
            <t id="3430-459771_6" word="." pos="."/>
        </terminals>
        <nonterminals>
            <nt id="3430-459771_503" cat="NP">
                <edge label="--" idref="3430-459771_2"/>
            </nt>
            <nt id="3430-459771_505" cat="NP">
                <edge label="--" idref="3430-459771_4"/>
                <edge label="--" idref="3430-459771_5"/>
            </nt>
            <nt id="3430-459771_504" cat="PP">
                <edge label="--" idref="3430-459771_3"/>
                <edge label="--" idref="3430-459771_505"/>
            </nt>
            <nt id="3430-459771_502" cat="NP">
                <edge label="--" idref="3430-459771_503"/>
                <edge label="--" idref="3430-459771_504"/>
            </nt>
            <nt id="3430-459771_501" cat="VP">
                <edge label="--" idref="3430-459771_1"/>
                <edge label="--" idref="3430-459771_502"/>
            </nt>
            <nt id="3430-459771_500" cat="S">
                <edge label="--" idref="3430-459771_501"/>
                <edge label="--" idref="3430-459771_6"/>
            </nt>
        </nonterminals>
    </graph>
    <matches>
    </matches>
    <sem>
        <globals>
        </globals>
        <frames>
            <frame name="Accoutrements" id="3430-459771_f1">
                <target>
                    <fenode idref="3430-459771_2"/>
                </target>
                <fe name="Descriptor" id="3430-459771_f1_e1">
                    <fenode idref="3430-459771_504"/>
                </fe>
            </frame>
        </frames>
        <usp>
            <uspframes>
            </uspframes>
            <uspfes>
            </uspfes>
        </usp>
        <wordtags>
        </wordtags>
    </sem>
</s>
```

Figure 12: Example of Tiger/SALSA-XML format

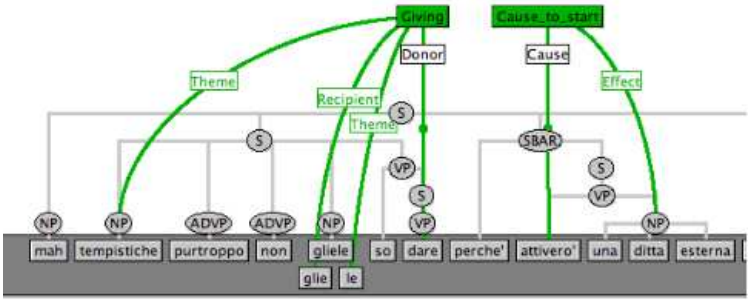


Figure 13: Example of FE repetition. Transl.: "*Well, a* deadline *unfortunately I cannot give you* (it) *because I will activate an external company*"

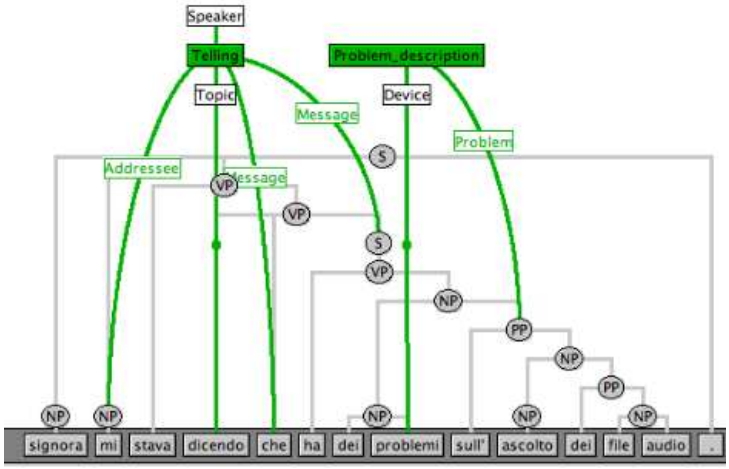


Figure 14: Example of FE repetition due to parsing error. Transl.: "*Madam you were telling me that you have some problems with the audio files*"

On a technical level, FE duplication can be achieved by clicking on the frame box and choosing the entry called "Add Element/FE name" in the context menu. In this way, a new instance of the FE has been added and can now be assigned to the corresponding constituent.

Annotators are instructed to annotate all frame elements which can be recognised with certainty. Sometimes, a FE can be annotated considering different extensions and it can be difficult to choose one over the other. In such cases, we adopted the *maximality principle* described in [9, pp. 188-189], which we summarize as:

(3) If possible, the complete lexical material describing a frame element should be annotated. Ideally, this material is located below one single node, the so-called maximal constituent. If the lexical material of a FE is distributed over several syntactic constituents, it is allowed to annotate *discontinuous* frame elements.

SALTO gives also the possibility to assign a FE label to part of a word. This is useful in case of verbal targets with role-bearing clitics, as shown in Fig. 15: the clitic "*li*" was split from the target word "*inzuppare.v*" (*to soak*), so that it can bear the *Theme* FE label. To this purpose, annotators just have to right-click on the word in the syntax tree they want to split and select the option "Split...", then enter the components in the field separated by |.

With the SALTO interface, it is possible to associate attributes to words. We exploit this mechanism to annotate the fact that one of the FEs could be assigned to an empty subject whose presence is revealed by the verb morphology. In practice, we associate the attribute *Empty_subj: FE label* to the verb word. In this way, we can cope with the problem of unexpressed roles, since Italian verbs can have an empty subject whose person and number is conveyed by the verb conjugation, whereas English and German verbs all require a mandatory explicit subject. An example annotation is reported in Fig. 16.

In this sentence, the verbal target "*rassicurò*" (*reassured*), which evokes the Reassuring frame, implies the presence of an implicit subject in the third person singular. This subject would bear the *Speaker* role.

In general, we add the empty subject label only when the target is a finite verb bearing explicit subject agreement information. This excludes for instance the *passato prossimo* tense (present perfect). In this case, the target

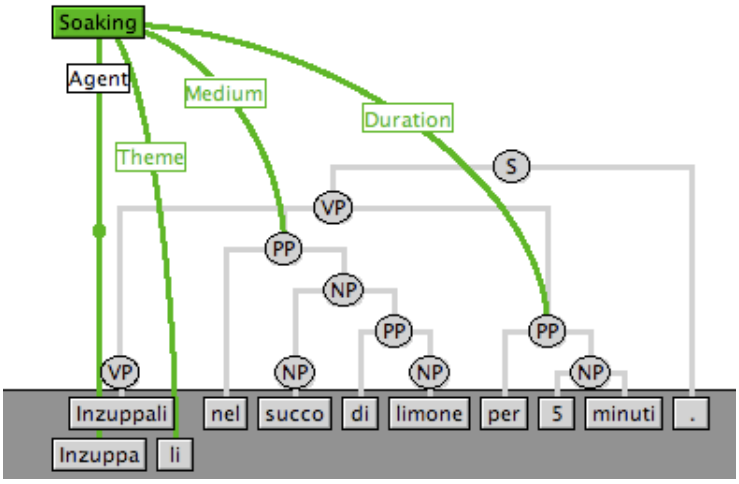


Figure 15: Splitting words with SALTO. Transl.: "*Soak them in lemon juice for 5 minutes*"

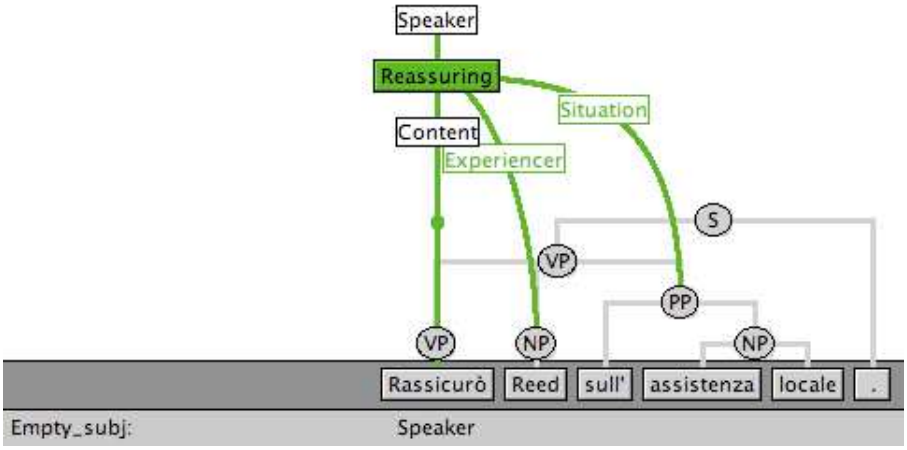


Figure 16: Empty subject annotated with SALTO. Transl.: "∅ *reassured Reed on local assistance.*"

is the past participle which in some cases bears subject gender and number agreement but in no case person agreement. Instead, most subject agreement information is conveyed by the auxiliary. For this reason, if the target is a past participle, we do not add the *Empty_subj* attribute. The same applies to cases of *gerundio* (gerund).

It is very important to distinguish the cases labeled as *Empty_subj* from the *null instantiations* encoded in the English FrameNet [11, pp. 33-36]. In the Berkeley database, FEs that do not appear in a given sentence as lexical or phrasal material are annotated to convey lexicographically relevant information about *omissibility conditions*. In particular, null instantiations defined as Constructional (CNI) are usually motivated by particular grammatical constructions that license such omissions such as imperative sentences with missing subject or passive sentences with missing agent. We report in example (4) a CNI annotation from the FrameNet database, where the core FE *Communicator* is omitted in the passive construction:

(4) But [anyone]$_{\text{Evaluee}}$ could be arrested and accused$_{\text{Judgment_communication}}$ [of communist sympathies]$_{\text{Reason}}$ (CNI=Communicator).

While we do not annotate cases of null instantiations in our gold standard, we identify empty subjects because they are morphologically expressed through verb agreement.

# 5 Annotation issues

While the described workflow, and in particular the annotation procedure, can be generally applied to every kind of text, there are some specific issues concerning dialogs. For this reason, we introduce few new practical suggestions that cope with some genre-specific issues:

- It was not always possible to annotate every utterance, particularly in case of disfluencies and semantically empty expressions. The general approach was to annotate utterances, if at least part of their meaning was expressed. For example, if a speaker could utter part of the turn and then was interrupted, annotation involved the tokens that were understandable, as shown in Fig. 17 (translation: "No, well, in the sense that if you put the plug in the yes"). According to the guidelines,

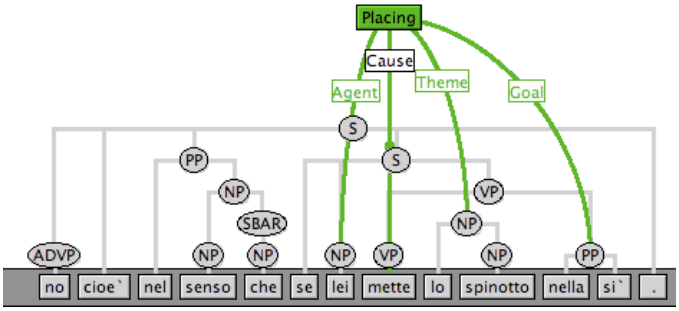


Figure 17: Assignment of a FE label to an incomplete constituent. Transl.: "*No well in the sense that if you put the plug in the yes*"

the *Goal* FE is assigned to the last PP in the sentence, even if it is not complete.

This approach represented a good solution to identify and annotate as much semantic information as possible, but we are aware that it cannot be easily generalized because the idea of "understandability" strongly depends on the annotator's choice and intuition.

- According to the same "understandability" principle, we introduced the *Corrected* flag for words which were clearly misspelled, due both to transcription errors and to speaker's mistakes. The flag allows to introduce the corrected version of misspelled word, which is displayed below the corresponding token. We report an example in Fig. 18 (the sentence means "No, well, but I am sorry to make you go there"). From the context, it is clear that *mandare* is the misspelling of *andare* (*go*), which is manually added by the annotator. In this case, the correction was particularly relevant because it involved the target word of the ARRIVING frame.

- Annotation with SALTO is carried out sentence-by-sentence, while in dialogs semantic elements bearing frame information (LU and FEs) can span different turns because of interruptions and overlaps. For the moment, we limited annotation to the utterance level and to frame

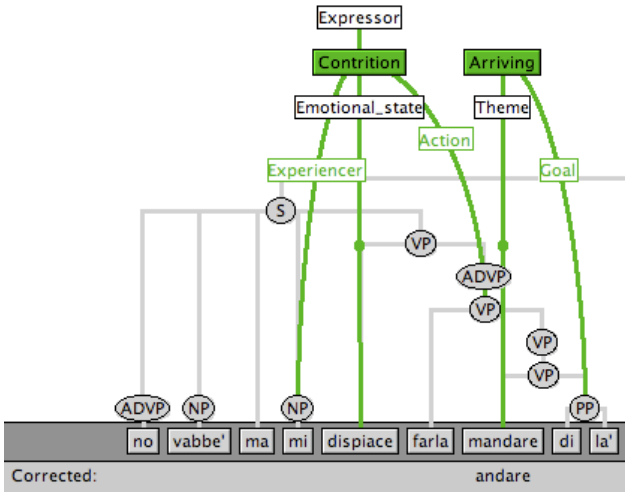


Figure 18: Correction of misspelled *mandare*

elements that are explicitly expressed by some lexical or phrasal material. In order to take into account inter-sentential relationships and the discourse context, however, it would be important to annotate also null instantiated FEs [11]. In particular, the so-called Definite Null Instantiations (DNI), which characterize lexically null instantiations of FEs that are already known from the context, could contribute to the identification of anaphoric relations between utterances. For example, the annotated sentence reported in Fig. 19 ("Ok, just a minute and I connect myself") would allow the introduction of a DNI label for the *Goal* FE, which is not expressed but can be understood from the previous turns as being the computer. A further annotation step would then involve the identification of the connection between DNI label and the referent [12].

- For all target words whose literal meaning does not correspond to the figurative one, we assign a frame label according to the figurative reading. This does not involve only idiomatic expressions, which are very frequent in dialogs, but also verbs with a generic meaning such as *"fare"* (*do/make*) and *"mettere"* (*put*), that in spontaneous conversations are

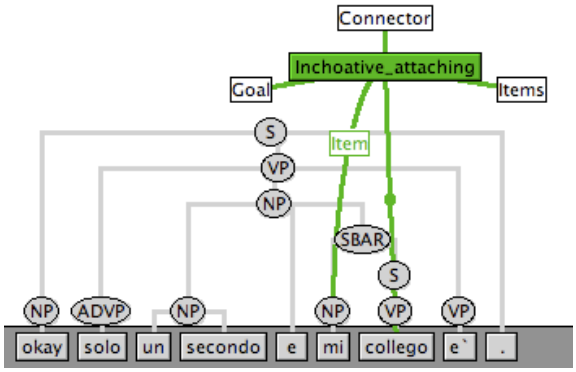


Figure 19: Case of Definite Null Instantiation

used very often instead of more specific verbs. We prefer to annotate them with the *intended* meaning, if it can be unambiguously understood from the context.

As for the domain-specificity of the language and the influence of conversational style, several new frames were introduced, which will be described in the following Section.

# 6 Newly introduced frames

We introduced 20 new frames out of the 174 identified in the subcorpus used for pilot annotation. The newly introduced frames can be grouped into three main classes:

1. Some frames were created because there was a gap in the English FrameNet hierarchy, for example RENDER_NONFUNCTIONAL is in the FrameNet database, but BECOMING_NONFUNCTIONAL is missing. The newly introduced frames of this kind have the same level of specificity of the existing ones.

2. Some frames were created to cover the domain-specific topics discussed, since the original definition of frames related to hardware/software,

data-handling and customer assistance was sometimes too coarse-grained. Some domain-oriented adaptations of existing frames were needed to describe specific situations, for example to distinguish between "real" and "virtual" movement (ARRIVING vs. NAVIGATION).

3. The GREETING frame was introduced because the FrameNet database so far does not take into account frames related to oral communication, apart from ATTENTION_GETTING. Few new frame elements were introduced as well, mostly expressing syntactic realizations that are typical of spoken Italian.

The list of new frames with the corresponding definition is reported in Table 2. This list has been developed for internal use during the annotation of the LUNA corpus and is not meant to be definitive.

| **Frame** | **Definition** |
|---|---|
| ACQUIRE_DATA | A *User* moves some *Data* from a *Source* into an *Application* or a *Goal*.<br>Ex. Hai importato [la password]$_{\text{Data}}$ [dalla vecchia versione]$_{\text{Source}}$? (Have you imported the password from the older version?). |
| ASSIGN | An *Item* is assigned to a *Receiver* so that he carries out or is in charge of a particular job or *Task*.<br>Ex. [La richiesta]$_{\text{Item}}$ è in carico [al gruppo fonia]$_{\text{Receiver}}$. (Transl: "The telephonic group is in charge of the request". In English the order of the frame elements is inverted). |
| BECOMING_NONFUNCTIONAL | An *Artifact* becomes no longer capable of performing its inherent function.<br>Ex. [La stampante]$_{\text{Artifact}}$ si rompe facilmente. (The printer breaks easily).<br>Notice that "La stampante è rotta" (The printer is broken) would be BEING_OPERATIONAL.<br>Precedes BEING_OPERATIONAL. |
| CHANGE_DATA | A *User* changes the content of a document so that *New_data* replace *Old_data*.<br>Ex. Devi aggiornare [la password]$_{\text{Old\_data}}$. (You must update your password).<br>This is the domain-specific version of REPLACING. |
| COME_TO_SIGHT | A *Graphical_element* becomes visible to a *Perceiver*. A *Duration* and *Manner* may also be specified |

| Frame | Definition |
|---|---|
| | Ex. Appare [per qualche istante]$_{\text{Duration}}$ [una schermata nera]$_{\text{Graphical\_element}}$. (A black screen appears for some seconds). |
| CREATE_DATA | A *User* newly creates *Data*, information or a document containing them. He can assign a name or a *Label* to the data.<br>Ex. [La richiesta]$_{\text{Data}}$ è stata aperta [come reset password]$_{\text{Label}}$. (The request was opened as password reset). |
| DISPLAY_DATA | A *Device* shows to a *Perceiver* some *Data* or a *Message* through a *Support* so that the data become visible.<br>Ex. [Il PC]$_{\text{Device}}$ [ti]$_{\text{Perceiver}}$ ripropone [delle altre lettere]$_{\text{Message}}$ [sullo schermo]$_{\text{Support}}$. (The PC shows you some other letters on the screen). |
| CREATE_SPACE | This frame was introduced for the lexical units *liberare.v* (to free) and *libero.a* (free.a) when they are used to refer to *Empty_space* available on a *Device*<br>Ex. Cerca di liberare [un po' di spazio]$_{\text{Empty\_space}}$ [sul PC]$_{\text{Device}}$. (Try to create some space on the PC). |
| GREETING | This frame includes all words and expressions used to give a sign of welcome or recognition to an *Addressee*.<br>Ex. Buongiorno [a lei]$_{\text{Addressee}}$. (Good morning to you). |
| HANDLE_DATA | An *Operator* handles some *Data* or documents he is in charge of in order to carry out a task or a particular job. The *Application* used is optional<br>Adesso gestisco [io]$_{\text{Operator}}$ [questa richiesta]$_{\text{Data}}$. (Now I handle this request). |
| INSERT_DATA | A *User* inserts some *Data* in a *Document* or a *Device*. The *Category* of the inserted data, the *Purpose* as well as the *Manner* in which the data are inserted may also be specified.<br>Ex. [Il suo responsabile]$_{\text{User}}$ deve compilare [il documento]$_{\text{Document}}$ [per la richiesta]$_{\text{Purpose}}$. (Your boss has to fill out the document for the request). |
| LEND | A *Lender* grants to a *Borrower* the use of an *Object* on the understanding that it shall be returned. The *Duration* may also be specified. |

| Frame | Definition |
|---|---|
| | [Mi]Borrower puoi prestare [il tuo PC]Object [solo un secondo]Duration? (Can you lend me your PC for a second?). |
| Lose_data | Some *Data* or documents are unwillingly lost at the expenses of an *Affected_user*<br>Ex. Se non lo risolvi potrei perder[mi]Affected_user [qualche lavoro]Data. (If you don't solve this, I could lose some work). |
| Navigation | A *User* enters or leaves an *Application* or moves in a virtual space (ex. to a generic *Goal*). *Reason*, *Time* and *Means* may also be specified.<br>The valence pattern of this frame is similar to the Arriving frame but it refers to the software environment.<br>Ex. Devi andare [alla web-mail]Application. (You have to go to the web-mail). |
| Open_data | A *User* opens an already existing *Document* with an *Application*.<br>The frame is different from Create_data because in this case the document already exists. It is also different from Change_operational_state, which involves the opening of an application.<br>Ex. Devi aprire [il file]Document. (You must open the file). |
| Problem_description | This frame was created to annotate all the possible elements involved in the description of a problem in the technical domain. The lexical units in the frame are usually synonyms of *problema.n* (problem). It is a more specific version of Predicament.<br>The problem can have a certain *Degree* and can involve a *Device* and an *Affected_person*. Sometimes it can hinder the execution of an *Affected_activity*.<br>Ex. [I tuoi colleghi]Affected_person hanno lo stesso problema [a collegare la stampante]Affected_activity. (Your colleagues have the same problem while connecting the printer). |
| Read_data | A *User* reads some *Data* or a *Device_with_data* using some *Reading_device*. Alternatively, a *Reading_device* reads some *Data* for a *User*. |

| Frame | Definition |
|---|---|
| | It is a domain-specific version of the READING frame because it involves a device and some electronic data. Ex. Faccio fatica a leggere [il CD]$_{\text{Device\_with\_data}}$ [dal lettore]$_{\text{Reading\_device}}$. (I can hardly read the CD from the CD-player). |
| RUN_OPERATION | This frame refers specifically to the technical domain and describes a situation where an *Operator* runs or executes an *Operation* in a given *Environment*<br>Ex. Hai provato ad eseguire [un ipconfig]$_{\text{Operation}}$ [da una sessione DOS]$_{\text{Environment}}$? (Have you tried to run an ipconfig from a DOS session?). |
| SELECT_DATA | An *Operator* selects some *Data*, optionally with a *Device*.<br>Ex. Clicca [sulla password]$_{\text{Data}}$ [con il tasto destro del mouse]$_{\text{Device}}$. (Click on the password with the right mouse button). |
| UNDERGO_CHANGE_OF_OP._STATE | A *Device* or application goes in (or out of) service. The *Time* and the *Place* where the *Device* goes in or out of use may be specified.<br>Precedes BEING_IN_OPERATION. It is similar to PROCESS_START but in that case an *Event* is involved, while here it is a *Device*.<br>Ex. [Il computer]$_{\text{Device}}$ non si accende. (The computer does not start). |

Table 2: The 20 newly introduced frames

# 7 Summary

This report describes the workflow devised for annotating frame information in the LUNA corpus of Italian spontaneous speech. HM and HH dialogs are first converted in Tiger/XML format and then manually annotated with the SALTO tool.

The guidelines include some general instructions that are valid for frame annotation in every language and for every kind of text. Besides, some language-, domain- and genre-specific hints are added. In particular, suggestions for the annotation of Italian null subject pronouns are given, as

well as instructions for dealing with disfluencies, interruptions, etc. which are typical of conversational speech. The last section details the 20 newly introduced frames. Such definitions are still preliminary and it would be important to integrate the frames in the FrameNet ontology, defining the missing frame-to-frame relations and the relevant semantic types for frames and FEs.

# References


[1] J. L. Austin. *How to do Things with Words*. Clarendon Press, Oxford, 1962.

[2] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The Berkeley FrameNet Project. In *Proceedings of the 36th ACL Meeting and 17th ICCL Conference*. Morgan Kaufmann, 1998.

[3] Roberto Basili and Fabio Massimo Zanzotto. Parsing Engineering and Empirical Robustness. *Journal of Natural Language Engineering*, June 2002.

[4] Aljoscha Burchardt, Katrin Erk, Anette Frank, Andrea Kowalski, Sebastian Padó, and Manfred Pinkal. SALTO - A Versatile Multi-Level Annotation Tool. In *Proceedings of the 5th Language Resources and Evaluation Conference (LREC 06)*, pages 517–520, Genoa Italy, 2006.

[5] Anna Corazza, Alberto Lavelli, and Giorgio Satta. Analisi Sintattica-Statistica basata su Costituenti. *Intelligenza Artificiale - Numero speciale su Strumenti per l'elaborazione del linguaggio naturale per l'Italiano*, 4(2):38–39, 2007.

[6] Marco Dinarelli, Silvia Quarteroni, Sara Tonelli, Alessandro Moschitti, and Giuseppe Riccardi. Annotating Spoken Dialogs: from Speech Segments to Dialog Acts and Frame Semantics. In *Proceedings of the EACL Workshop on Semantic Representation of Spoken Language (SRSL)*, Athens, Greece, 2009.

[7] Katrin Erk and Sebastian Padó. A powerful and versatile XML Format for representing role-semantic annotation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 04)*, Lisbon, Portugal, 2004.

[8] Andreas Mengel and Wolfgang Lezius. An XML-based encoding format for syntactically annotated corpora. In *Proceedings of LREC-2000, Second International Conference on Language Resources and Evaluation*, Athens, Greece, 2000.

[9] Sebastian Padó. *Cross-Lingual Annotation Projection Models for Role-Semantic Information*. PhD thesis, Universität des Saarlandes, 2007.

[10] Silvia Quarteroni, Giuseppe Riccardi, Sebastian Varges, and Arianna Bisazza. An Open-Domain Dialog Act Taxonomy. Technical Report 08-032, DISI, University of Trento, 2008.

[11] Josef Ruppenhofer, Michael Ellsworth, Miriam R.L. Petruck, Christopher R. Johnson, and Jan Scheffczyk. *FrameNet II: Extended Theory and Practice*. Available at http://framenet.icsi.berkeley.edu/book/book.html, 2006.

[12] Josef Ruppenhofer, Caroline Sporleder, Roser Morante, Collin F. Baker, and Martha Palmer. SemEval-2010 Task 10: Linking Events and Their Participants in Discourse. In *Proceedings of the NAACL-HLT Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 106–111, Boulder, CO, USA, June 2009.

[13] Sara Tonelli, Giuseppe Riccardi, Rashmi Prasad, and Aravind Joshi. Annotation of Discourse Relations for Conversational Spoken Dialogs. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC) (to appear)*, Malta, 2010.