AUTOMATIC FRAMENET-BASED ANNOTATION OF CONVERSATIONAL SPEECH

Bonaventura Coppola, Alessandro Moschitti, Sara Tonelli, Giuseppe Riccardi

Department of Engineering and Computer Science
University of Trento
38050 Povo - Trento, Italy
{coppola, moschitti, riccardi}@disi.unitn.it, satonelli@fbk.eu

ABSTRACT

Current Spoken Language Understanding technology is based on a simple concept annotation of word sequences, where the interdependencies between concepts and their compositional semantics are neglected. This prevents an effective handling of language phenomena, with a consequential limitation on the design of more complex dialog systems.
In this paper, we argue that shallow semantic representation as formulated in the Berkeley FrameNet Project may be useful to improve the capability of managing more complex dialogs. To prove this, the first step is to show that a FrameNet parser of sufficient accuracy can be designed for conversational speech. We show that exploiting a small set of FrameNet-based manual annotations, it is possible to design an effective semantic parser. Our experiments on an Italian spoken dialog corpus, created within the LUNA project, show that our approach is able to automatically annotate unseen dialog turns with a high accuracy.

Index Terms— Spoken Dialog Systems, Computational Semantics, Learning Models

1. INTRODUCTION

In recent years, commercial services based on spoken dialog systems have consistently increased both in number and application scenarios. Their main limitation relates to a low capability of handling language variability and of performing conceptual analysis over speech transcriptions. Indeed, the current Spoken Language Understanding (SLU) technology is based on a simple concept annotation of word sequences, where the interdependencies between concepts and their compositional semantics are not even attempted.

Although Natural Language Understanding approaches are hardly suitable for real applications, shallow semantic methods devised in computational linguistics research appear promising to tackle the above mentioned tasks. The Berkeley FrameNet Project [1] proposes semantic models and resources for open domain semantic analysis which can be adapted to specific dialog domains. According to this paradigm, prototypical situations (*frames*) and predicates evoking these situations (*lexical units*) are annotated in the text along with their involved participants (*frame elements*). For example, the sentence *"I would like to buy an insurance policy"* will evoke the COMMERCE_SCENARIO frame where *buy* is the lexical unit (or predicate) while *I* and *insurance policy* are the BUYER and the GOODS frame elements, i.e. the arguments of the predicate.

The semantic model proposed in FrameNet is well founded at formal level [2]. In addition, nearly 800 frames and more than 4,000 frame-dependent concepts have been already identified and described. A wide English corpus of manually annotated examples is available as well, such that supervised machine learning can be applied to automatize the frame annotation process. Since this technology is very recent, it has not been used yet in any spoken dialog system. The first step to make it possible is to design an automatic FrameNet-based labeler able to work on conversational speech.

In this paper, we face the problem of automatically performing the above analysis over speech transcriptions from real-world dialogs. In particular, we present a novel approach based on Support Vector Machines (SVMs), Tree Kernels and Frame Semantics. Our technique is language independent and achieves state-of-the-art results in dealing with several thousands of concepts defined within hundreds of different semantic contexts (frames). Our system can be trained on any corpus which just includes plain text and frame-based semantic annotation. Although we deeply exploit syntax, it is not required in principle, since we can robustly rely on automatic syntactic analysis made by an off-the-shelf analyzer like the Charniak's parser [3]. Since the proposed approach is inherently supervised, we are particularly interested in testing its portability on languages with minor availability of resources than English. Actually, several efforts to develop annotated FrameNet-like databases in other languages are currently in progress, for example in German [4] and Italian [5]. In this work, we report on successful experiments performed on Italian, on the basis of a reasonably small amount of annotated data, which is drawn from the spoken dialog corpus being de-

This work has been partially funded by the European Commission - LUNA project (contract no. 33549), and by the Marie Curie Excellence Grant for the ADAMACH project (contract no. 022593).

veloped within the LUNA EU Project[1].

The rest of the paper is organized as follows. Section 2 introduces Frame Semantics and our automatic analysis technique, Section 3 presents the dataset, and Section 4 describes the experiment setting, the achieved results, and draws the final conclusions.

2. AUTOMATIC ANNOTATION OF FRAME SEMANTICS

Frame Semantics [2] allows real-world knowledge to be captured by semantic frames, script-like conceptual structures that describe particular types of situations, objects, or events along with their participating elements. For example, here is a short definition of a sample frame:

COMMERCE_SCENARIO
Core Elements: BUYER, GOODS, MONEY, SELLER
non-Core Elements: MANNER, MEANS, PURPOSE, RATE
Subframes: COMMERCIAL_TRANSACTION

where the core frame elements are participant entities which are supposed to be always present, whereas non-core are just optional, more generic participants. Frame-to-frame relations are also defined, like the *Subframe* relation which states here a hierarchical dependency of the COMMERCIAL_TRANSACTION frame. The Berkeley FrameNet Project currently includes the definitions of nearly 800 frames, 4,000 frame elements, and 135,000 annotated English sentences. An example of sentence annotation for the COMMERCE_SCENARIO is reported hereafter:

Ralemberg said [*he*]$_{\text{SELLER}}$ *already had a* [*buyer*]$_{\text{BUYER}}$ [*for the wine*]$_{\text{GOODS}}$

where the underlined word *buyer* is the target word (or *lexical unit*, or predicate) which plays the role of *evoker* for this particular frame. To automatically parse this information from plain text, we need to (a) represent the relation between the target word and the words compounding an argument in terms of feature vectors, and (b) learn classification models able to process such vectors.

2.1. Classification Steps

To implement a FrameNet-based parsing system we adopt a multi-stage classification scheme over natural language. Previous studies in this direction apply Semantic Role Labeling (SRL) approaches [6]. We extended the same strategy developed in [7, 8], which now includes: (1) *Target Word Detection*, i.e. the semantically relevant words bringing predicative information are detected; (2) *Frame Disambiguation*, i.e. the correct frame for any target word is chosen; (3) *Boundary Detection (BD)*, i.e. the sequences of words constituting the frame elements (arguments) are detected; and (4) *Role Classification (RC)*, which assigns semantic labels to the frame elements detected in the previous stage.

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

The first two stages can be carried out in several ways (depending on the application), which include heuristics based on FrameNet lexical units found in the text, or traditional supervised multi-classification approaches. BD is typically carried out as a binary classification problem, where the classification instances are the nodes of the syntactic parse tree of the considered sentence (or dialog turn). Indeed, predicate arguments, according to some linguistic theories, are univocally associated with syntactic constituents, i.e. internal the parse tree nodes. At training time, the positive examples are the nodes corresponding to arguments, whereas all the remaining nodes are negative examples. RC is a multi-classification problem over the set of the possible labels for an argument (with respect to the chosen frame). Even in this case, role labels are strictly associated with internal tree nodes as selected in the previous stage.

In this work we focus on the two last steps of the system since they are the most interesting. The representation of the nodes in a learning algorithm is traditionally carried out by exploiting syntactic information, since syntax is strongly linked to semantics. Many features for representing the nodes have been provided [6], which form the vectors to train SVMs. We further exploit the potential of SVMs by using kernel methods: we use Tree Kernels to encode the subtree which includes a target word and one of its arguments into the learning algorithm, as shown in [7]. The next sections briefly summarize SVMs, kernel methods and Tree Kernels.

2.2. SVMs and the Kernel Trick

Kernel Methods refer to a large class of learning algorithms based on inner product vector spaces, among which Support Vector Machines (SVMs) are one of the most well-known. SVMs learn a hyperplane $H(\vec{x}) = \vec{w} \cdot \vec{x} + b = 0$, where $\vec{x}$ is the feature vector representation of a classifying object o, $\vec{w} \in \Re^n$ (a vector space) and $b \in \Re$ are parameters [9]. The classifying object o is mapped in $\vec{x}$ by a feature function ϕ.

The kernel trick allows us to rewrite the decision hyperplane as $\sum_{i=1..l} y_i \alpha_i \phi(o_i) \cdot \phi(o) + b$, where y_i is equal to 1 for positive and -1 for negative examples, $\alpha_i \in \Re^+$, $o_i \; \forall i \in \{1, .., l\}$ are the training instances, and the product $K(o_i, o) = \langle \phi(o_i) \cdot \phi(o) \rangle$ is the kernel function associated with the mapping ϕ. Note that we do not need to explicitly apply the mapping ϕ, since we can directly use the kernel function $K(o_i, o)$.

A traditional example is given by the polynomial kernel: $PK(o_1, o_2) = (c + \vec{x}_1 \cdot \vec{x}_2)^d$, where c is a constant and d is the degree of the polynomial. Given the features used to map objects in $\Re^n$, this kernel generates the space of all conjunctions of feature groups, up to d elements.

2.3. Tree Kernels

Tree kernels are scalar products that evaluate the number of common subtrees. For example, Figure 1 shows a tree along with some of its tree fragments. These are matched against

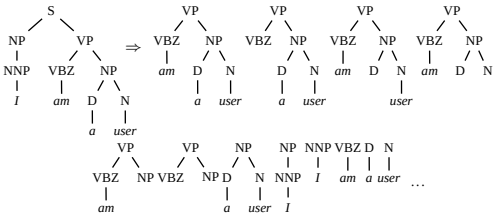

Fig. 1. A tree for the sentence *"I am a user"* along with some of its tree fragments.

those from another tree. More formally, given two trees T_1 and T_2, let $\{f_1, f_2, ..\} = \mathcal{F}$ be the set of substructures (fragments) and $I_i(n)$ be equal to 1 if f_i is rooted at node n, 0 otherwise. The Collins and Duffy's kernel is defined as

$$TK(T_1, T_2) = \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} \Delta(n_1, n_2), \quad (1)$$

where N_{T_1} and N_{T_2} are the sets of nodes in T_1 and T_2 respectively, and $\Delta(n_1, n_2) = \sum_{i=1}^{|\mathcal{F}|} I_i(n_1) I_i(n_2)$. The latter is equal to the number of common fragments rooted in nodes n_1 and n_2. Δ can be computed as follows:
(1) if the productions (i.e. the nodes with their direct children) at n_1 and n_2 are different then $\Delta(n_1, n_2) = 0$;
(2) if the productions at n_1 and n_2 are the same, and n_1 and n_2 only have leaf children (i.e. they are pre-terminal symbols) then $\Delta(n_1, n_2) = 1$;
(3) if the productions at n_1 and n_2 are the same, and n_1 and n_2 are not pre-terminals, then $\Delta(n_1, n_2) = \prod_{j=1}^{nc(n_1)} (1 + \Delta(c_{n_1}^j, c_{n_2}^j))$, where $nc(n_1)$ is the number of children of n_1 and c_n^j is the j-th child of n.

Such tree kernel can be normalized, and a λ factor can be added to reduce the weight of large structures (refer to [10] for a complete description). Most important, we can take advantage of the joint space between the tree and the polynomial kernel by simply summing them, i.e. $K_{sum} = TK + PK$.

3. THE LUNA SPOKEN DIALOG CORPUS

The LUNA European Project addresses the problem of real-time understanding of spontaneous speech in the context of advanced telecom services, and it applies to Italian, French and Polish. As a first step, the project has made available a benchmark collection of Italian dialogs. The corpus currently includes 50 human-human (HH) and 50 human-machine (HM) dialogs, recorded in the call center of the help-desk facility of the Italian Consortium for Information Systems. The HH dialogs are spontaneous conversations between a caller and an operator about software and hardware problems. The HM dialogs are a set of "wizard of oz" dialogs where the user explains a problem and the wizard reacts according to one of ten possible predefined scenarios.

The corpus was first annotated with part of speech and morphosyntactic features at word level using an automatic tagger, and then syntactically parsed with Bikel's constituency-based parser trained for Italian [11]. Next, a manual correction was carried out to make sure that the nodes potentially carrying semantic information have correct constituent boundaries. Frame information was then annotated on top of the parse trees, attaching target labels to their related words, and frame element labels to internal tree nodes. Where possible, we applied the frame and frame element definitions as in the English FrameNet. Nonetheless, in case of gaps in the original model (with respect to our *very* specific domain), we introduced new frames and related frame elements. In particular, we identified 154 already existing frames and introduced 20 new frames, mainly concerning data processing such as NAVIGATION, DISPLAY_DATA, LOSE_DATA, CREATE_DATA. The most frequent frames are related to the information exchange that is typical of a help-desk facility, for example TELLING, GREETING, CONTACTING, STATEMENT, RECORDING, COMMUNICATION. Another important group includes frames describing software/hardware functionality such as BEING_IN_OPERATION, BEING_OPERATIONAL, CHANGE_OPERATIONAL_STATE, OPERATIONAL_TESTING. TELLING and GREETING are the most frequent frames, with 277 and 270 frame instances respectively (also see [12] for a complete analysis). Overall, we annotated 662 turns of HM dialogs with 923 frame instances, and 1,997 turns of HH dialogs with 1,951 frame instances. In general, HH dialogs show a higher frame variability than HM dialogs because spontaneous conversations can concern minor less related topics as well, whereas HM dialogs are more task-oriented. Every HM turn has 1.39 annotated instances on average, whereas the HH turns show a lower semantic density with 0.98 annotated instances per turn. This can be explained by the fact that in human turns there are speech disfluencies such as interruptions and ungrammatical sentences.

4. EXPERIMENTS

We carried out several experiments on the spoken dialog corpus described above to test the effectiveness of our FrameNet parser. We present the results of the second and third stage of the system described in Section 2.1, that is BD and BD+RC. Therefore, we assume the target word (i.e. the predicate for which the arguments must be identified) along with its correct frame as given. We only used the HH corpus portion, since HM dialogs are less interesting with respect to language variability. For each dialog, the set of its turns was considered, creating a dataset of 1,677 target words over 162 different frames. Such dataset was further split in a 90% for training (1,521 target words) and a 10% for testing (156 target words).

Given the above dataset, different learning strategies were carried out. For both BD and RC, we can split the data related to all the frames in several ways. For BD, five models are trained across all the frames according to the part of speech of the target words[2] (*POSwise splitting*). For RC, the

[2]Frame Semantics allows *verbal*, *nominal*, *adjectival*, *adverbial* and *prepositional* predicates.

multi-classification models are naturally split according to the different frames. In addition, POSwise splitting can either be applied or not. This leads to two different RC settings: "byPOSandFrame" and "byFrame".

Eval Setting	P	R	F_1	P	R	F_1
byPOSandFrame RC learning configuration						
				PK		
BD	-	-	-	.900	.869	.884
BD+RC	-	-	-	.679	.655	.667
	TK			**PK+TK**		
BD	.887	.856	.871	.905	.873	.889
BD+RC	.674	.651	.662	.688	.664	.676
byFrame RC learning configuration						
				PK		
BD	-	-	-	.900	.869	.884
BD+RC	-	-	-	.769	.742	.756
	TK			**PK+TK**		
BD	.887	.856	.871	.905	.873	**.889**
BD+RC	.765	.738	.751	.774	.747	**.760**

Table 1. Results for different learning schemes and kernels.

We tested several learning models over the standard features described in [6] and the structured features [7], described in Section 2.3. In particular, we experimented with the Polynomial Kernel (PK), the Tree Kernel (TK) and their combination (PK+TK).

4.1. Results and Discussion

The results are reported in Table 1. Each table block shows Precision, Recall and F_1 for either PK, TK, or PK+TK. Also, the table distinguishes between the byPOSandFrame and byFrame splitting schemes. The rows marked as BD show the results for the task of marking the exact constituent boundaries of every frame element (argument) found. The rows marked as BD+RC show the results for the two-stage pipeline of *both* marking the exact constituent boundaries and *also* assigning the correct semantic label. Based on results, several observations hold.

First, in both splitting configurations, the highest F_1 has been achieved using PK+TK. In particular, "byFrame" PK+TK performs best and reaches P=0.905, R=0.873, F_1=0.889 for BD, and P=0.774, R=0.747, F_1=0.760 for the whole task (BD+RC). The lower F_1 reached by the semantic labeler using byPOSandFrame method is due to the higher data sparseness caused by such split. In fact, while just 167 multi-classifiers are learned in byFrame configuration, they increase to 221 in byPOSandFrame split.

Second, the F_1 of PK is surprisingly high, since it exploits a set of standard SRL standard features [6] developed for English and left unmodified for Italian. Nonetheless, PKs are comparable to TKs, and when combined produce an improvement. Concerning the structured features exploited by TKs, they work as well without any language-specific tuning.

Third, the best F_1 achieved is extremely good. Our corresponding result on the FrameNet corpus is P=0.784, R=0.571, F_1=0.661 (with byPOSandFrame setting), where the corpus contains much more data, its sentences come from a standard written text (no disfluencies are present), and it is in English language which is morphologically simpler than Italian. On the other hand, the LUNA corpus includes optimal syntactic annotation which exactly fits Frame Semantics, and the number of frames is far lower than in FrameNet.

Finally, the good performance achieved for Italian shows that this FrameNet parsing approach can be used to label conversational speech in any language using small training data. Moreover, the approach works well for specific domains (ours is a *very* specific one). Nonetheless, additional tests on automatic transcriptions are needed since at the moment our experiments have been only carried out on manual transcriptions. However, our findings are important since they show that future research on complex spoken dialog systems can successfully exploit automatically generated Frame Semantics.

5. REFERENCES

[1] Collin F. Baker, Charles J. Fillmore, and John B. Lowe, "The Berkeley FrameNet project," in *Proceedings of COLING-ACL '98*, 1998, pp. 86–90.

[2] Charles J. Fillmore, "The Case for Case," in *Universals in Linguistic Theory*, Emmon Bach and Robert T. Harms, Eds., pp. 1–210. Holt, Rinehart, and Winston, New York, 1968.

[3] Eugene Charniak, "A Maximum-Entropy-Inspired Parser," in *Proceedings of NAACL 2000*, San Francisco, CA, USA, 2000.

[4] *The SALSA Corpus: a German Corpus Resource for Lexical Semantics*, Proceedings of LREC 2006, Genoa, Italy, 2006.

[5] Sara Tonelli and Emanuele Pianta, "Frame Information Transfer from English to Italian," in *Proceedings of LREC-2008*, ELRA, Ed., Marrakech, Morocco, 2008.

[6] Daniel Gildea and Daniel Jurafsky, "Automatic Labeling of Semantic Roles," *Computational Linguistics*, vol. 28, no. 3, pp. 245–288, 2002.

[7] Alessandro Moschitti, Daniele Pighin, and Roberto Basili, "Tree kernels for semantic role labeling," *Computational Linguistics*, vol. 34, no. 2, pp. 193–224, 2008.

[8] A. Moschitti, B. Coppola, A. Giuglea, and R. Basili, "Hierarchical semantic role labeling," in *CoNLL 2005 Shared Task*.

[9] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1995.

[10] Michael Collins and Nigel Duffy, "New Ranking Algorithms for Parsing and Tagging: Kernels over Discrete structures, and the voted perceptron," in *ACL02*, 2002, pp. 263–270.

[11] Anna Corazza, Alberto Lavelli, and Giorgio Satta, "Analisi sintattica-statistica basata su costituenti," *Intelligenza Artificiale*, vol. 2, pp. 38–39, 2007.

[12] A. Bisazza, M. Dinarelli, S. Quarteroni, S. Tonelli, A. Moschitti, and G. Riccardi, "Semantic annotations for conversational speech: from speech transcriptions to predicate argument structures," in *Proceedings of IEEE-SLT'08)*, Goa, India, December 2008.