Unsupervised structured semantic inference for spoken dialog reservation tasks

Alejandra Lorenzo
Université de Lorraine
LORIA, UMR 7503
Nancy, France
`alejandra.lorenzo@loria.fr`

Lina M. Rojas-Barahona
LORIA, UMR 7503
Nancy, France
`lina.rojas@loria.fr`

Christophe Cerisara
LORIA, UMR 7503
Nancy, France
`cerisara@loria.fr`

Abstract

This work proposes a generative model to infer latent semantic structures on top of manual speech transcriptions in a spoken dialog reservation task. The proposed model is akin to a standard semantic role labeling system, except that it is unsupervised, it does not rely on any syntactic information and it exploits concepts derived from a domain-specific ontology. The semantic structure is obtained with unsupervised Bayesian inference, using the Metropolis-Hastings sampling algorithm. It is evaluated both in terms of attachment accuracy and purity-collocation for clustering, and compared with strong baselines on the French MEDIA spoken-dialog corpus.

1 Introduction

Many concrete applications that involve human-machine spoken dialogues exploit some hand-crafted ontology that defines and relates the concepts that are useful for the application. The main challenge for the dialog manager used in the application is then to interpret the user's spoken input in order to correctly answer the user's expectations and conduct a dialogue that shall be satisfactory for the user. This whole process may be decomposed into the following stages:

- Automatic speech recognition, to transform the acoustic signal into a sequence of words (or sequences of word hypotheses);

- Spoken language understanding, to segment and map these sequences of words into concepts of the ontology;

- Semantic analysis, to relate these concepts together and interpret the semantic of the user

input at the level of the utterance, or of the speaker turn;

- Dialogue act recognition

- Dialogue planning

- Text generation

- ...

Note that the process sketched here often further involves several other important steps that are used internally within one or several of these broad stages, for instance named entity recognition, co-reference resolution, syntactic parsing, marcov decision process, reinforcement learning, etc.

This work focuses mainly on the second and third stages, since we assume that segmentation is given and we want to discover the underlying concepts and relations in the data. The third stage is very important because it exhibits the latent semantic structure hidden in the user utterance: what is the object affected by a given predicate ? What are the modifiers that may alter the meaning of a predicate ? Without such a structure, the system can hardly push understanding beyond lexical semantics and reach fine-grained semantic representations, which are thus often limited to well-formed inputs and cannot handle spontaneous speech as considered here. But still, despite its importance, most spoken dialog systems do not make use of such structure.

We propose an approach here to address this issue by directly inferring the semantic structure from the flat sequence of concepts using the unsupervised Bayesian learning framework. Hence, the proposed model does not rely on any predefined corpus annotated with semantic structure, which makes it much more robust to spoken inputs and adaptable to new domains than traditional supervised approaches.

12

2 Related work

In recent years, an increasing number of works have addressed robustness and adaptability issues in most of standard Natural Language Processing tasks with unsupervised or semi-supervised machine learning approaches. Unsupervised learning attempts to induce the annotations from large amounts of unlabeled data. Several approaches have recently been proposed in this context for the semantic role labeling task. (Swier and Stevenson, 2004) were the first to introduce an unsupervised semantic parser, followed by (Grenager and Manning, 2006), (Lang and Lapata, 2010), (Lang and Lapata, 2011b) and (Lang and Lapata, 2011a). Finally, (Titov and Klementiev, 2012), introduced two new Bayesian models that achieve the best current state-of-the-art results. However, all these works use some kind of supervision (namely a verb lexicon or a supervised syntactic system, which is the case in most of the approaches). (Abend et al., 2009) proposed an unsupervised algorithm for argument identification that uses a fully unsupervised syntactic parser and where the only supervised annotation is part-of-speech (POS) tagging.

Semi-supervised learning attempts to improve the performance of unsupervised algorithms by using both labeled and unlabeled data for training, where typically the amount of labeled data is smaller. A variety of algorithms have been proposed for semi-supervised learning[1]. In the context of semantic role labeling, (He and Gildea, 2006) and (Lee et al., 2007) hence tested self-training and co-training, while (Fürstenau and Lapata, 2009) used a graph-alignment method to semantic role labeling (SRL). Finally, in (Deschacht and Moens, 2009) the authors present a semi-supervised Latent Words Language Model, which outperforms a state-of-the-art supervised baseline. Although semi-supervised learning approaches minimize the manual effort involved, they still require some amount of annotation. This annotation is not always available, sometimes expensive to create and often domain specific. Moreover, these systems assume a specific role labeling (e.g. PropBank, FrameNet or VerbNet) and are not generally portable from one framework to another.

A number of works related to semantic inference have already been realized on the French MEDIA corpus. Hence, dynamic Bayesian networks were proposed for semantic composition in (Meurs et al., 2009), however their model relies on manual semantic annotation (i.e. concept-value pairs) and supervised training through the definition of 70 rules. In (Huet and Lefèvre, 2011; Camelin et al., 2011) unsupervised models were proposed that use stochastic alignment and Latent Dirichlet Allocation respectively, but these models infer a flat concept-value semantic representation. Compared to these works, we rather propose a purely unsupervised approach for structured semantic Metropolis-Hastings inference with a generative model specifically designed for this task.

3 Proposed model

3.1 Principle

We consider a human-machine dialog, with the objective of automatically building a semantic structure on top of the user's spoken utterances that shall help the dialog system to interpret the user inputs. This work focuses on inferring the semantic structure, and it assumes that a segmentation of users' utterances into concepts is given. More precisely, we exploit as input a manual segmentation of each utterance into word segments, where each segment represents a single concept that belongs to MEDIA ontology (Denis et al., 2006) (see Figure 1).
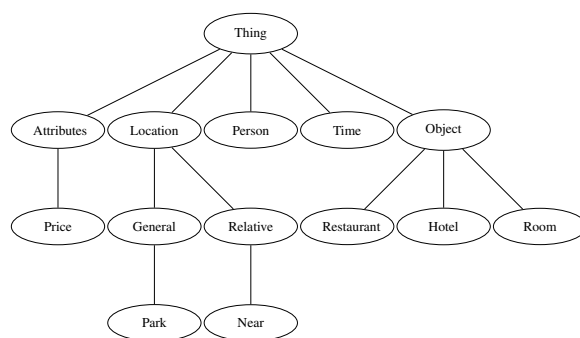

Figure 1: Excerpt of MEDIA ontology

This ontology identifies the concepts that can have arguments, and we thus use this information to further distinguish between *head* segments that can have arguments (noted W_h[2] in Figure 3) and *argument* segments that cannot govern another concept (noted W_a). From these two classes of

---

[1] We refer the reader to (Zhu, 2005) or (Pise and Kulkarni, 2008) for an overview on semi-supervised learning methods.

[2] W_h actually represents one word in a segment composed of N_h words, but by extension, we implicitly refer here to the full segment.

segments and the words' inflected forms that compose each segment we infer:

- A semantic structure composed of triplets (W_a, W_h, A) where A is the type of argument, or, in other words, the type of semantic relation between both segments;

- A semantic class C_t for the head segment

An example of the target structure we want to obtain is shown in Figure 2.

Inference of these structure and classes is realized with an unsupervised Bayesian model, i.e., without training the model on any corpus annotated with such relations. Instead, the model is trained on an unlabeled dialog corpus composed of raw manual speech transcriptions, which have also been manually segmented into utterances and words' segments as described above. Training is actually realized on this corpus using an approximate Bayesian inference algorithm that computes the posterior distribution of the model's parameters given the dataset. We have used for this purpose the Metropolis-Hastings Markov Chain Monte Carlo algorithm.

3.2 Bayesian model

Figure 3 shows the plate diagram of the proposed model. The plate N_h (respectively N_w) that surrounds a shaded node represents a single words' segment of length N_h (respectively N_w). The outer plate N_u indicates that the graphical model shall be repeated for each of the N_u utterances in the corpus.

Variable	Description
C_t	latent semantic type assigned to predicate t
W_h	observed words in each head segment. $P(W_h|C_t)$ encodes lexical preferences for the semantic inference
A_i	latent semantic type assigned to the ith argument of predicate t
Rp_i	latent relative position assigned to the ith argument of predicate t
W_a	observed words in each argument segment. $P(W_a|A_i)$ encodes lexical preferences for the semantic inference

Table 1: Variables of the model

Each *head* word segment has a latent semantic type C_t, and governs N_a arguments. Each argument is represented by an *argument* words' segment, which has a latent semantic type A. Each argument is further characterized by its relative position Rp with respect to its *head* segment. Rp
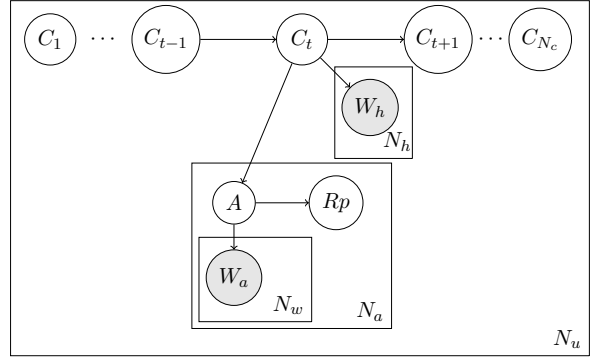

Figure 3: Plate diagram of the proposed model. N_u represents the number of utterances; N_h, the number of words in a head segment; N_w, the number of words in an argument segment; and N_a the number of arguments assigned to predicate t.

can have 4 values, depending on whether the argument is linked to the closest (1) or another (2) verbal[3] head, or the closest (3) or another (4) nominal head. Rp is derived from the argument-to-head assignment, which is latent. So, Rp is also latent. The sequence of N_c *head* segments in utterance u is captured by the HMM shown on top of the plate diagram, which models the temporal dependency between successive "semantic actions" of the user.

The variables of the model are explained in Table 1.

The most important property of this model is that the number of arguments N_a is not known beforehand. In fact, every *argument* segment can be governed by any of the N_c *head* segments in the utterance, and it is the role of the inference process to actually decide with which head it should be linked. This is why the model performs *structured* inference.

Concretely, at any time during training, every argument is governed by a single head. Then, inference explores a new possible head attachment for an argument W_a, which impacts the model as follows:

- The number of arguments N_a of the previous head is decreased by one;

- The number of arguments N_a of the new head is increased by one;

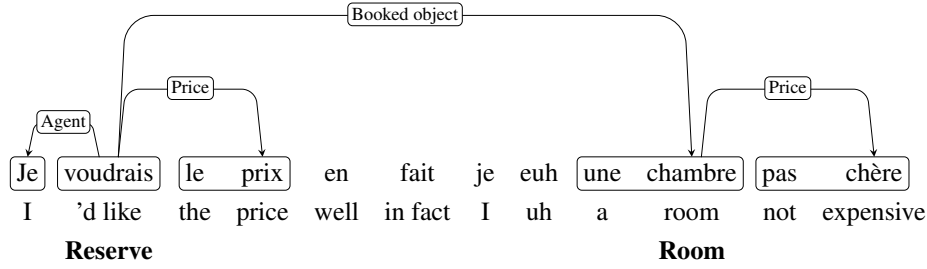[3]Morphosyntactic classes are obtained with the Treetagger

14

Figure 2: Example of inferred semantic structure for a sentence in the MEDIA corpus. Traditional dependency notations are used: the head segment points to the argument segment, where segments are shown with boxes (arrows link segments, not words !). The semantic class assigned to each head segment is shown in bold below the translated text.

- The relative position Rp of the argument is recomputed based on its new head position;

- The argument type A is also re sampled given the new head type C_t.

This reassignment process, which is at the heart of our inference algorithm, is illustrated in Figure 4.

3.3 Metropolis inference

Bayesian inference aims at computing the posterior distribution of the model's parameters, given the observed data. We assume that all distributions in our model are multinomial with uniform priors. The parameters are thus:

$P(W_h\|C_t) \sim \mathcal{M}(\theta_{C_t}^H)$	Distribution of the words for a given head semantic class
$P(C_t\|C_{t-1}) \sim \mathcal{M}(\theta_{C_{t-1}}^C)$	Transition probabilities between semantic classes
$P(W_a\|A) \sim \mathcal{M}(\theta_A^W)$	Distribution of the words for a given argument type
$P(Rp\|A) \sim \mathcal{M}(\theta_A^R)$	Distrib. of the relative position of a given argument to its head given the argument type
$P(A\|C_t) \sim \mathcal{M}(\theta_{C_t}^A)$	Distrib. of the argument types given a head semantic class

3.3.1 Inference algorithm

To perform inference, we have chosen a Markov Chain Monte Carlo algorithm. As our model is finite, parametric and identifiable, Doob's theorem guarantees the consistency of its posterior, and thus the convergence of MCMC algorithms towards the true posterior. Because changing the head of one argument affects several variables simultaneously in the model, it is problematic to use the basic Gibbs sampling algorithm. A block-Gibbs sampling would have been possible, but this would have increased the computational complexity and we also wanted to keep as much flexibility as possible in the jumps that could be realized in the search space, in order to prevent slow-mixing and avoid (nearly) non-ergodic Markov chains, which are likely to occur in such structured inference problems.

We have thus chosen a Metropolis-Hastings sampling algorithm, which allows us to design an efficient proposal distribution that is adapted to our task. The algorithm proceeds by first initializing the variables with a random assignment of arguments to one of the heads in the utterance, and a uniform sampling of the class variables. Then, it iterates through the following steps:

1. Sample uniformly one utterance u

2. Sample one jump following the proposal distribution detailed in Section 3.3.2.

3. Because the proposal is uniform, compute the acceptance ratio between the model's joint probability at the proposed (noted with a $'$) and current states:

$$r = \frac{P(C', W_h', W_a', Rp', A')}{P(C, W_h, W_a, Rp, A)}$$

4. Accept the new sample with probability $\min(1, r)$; while the sample is not accepted, iterate from step 2.
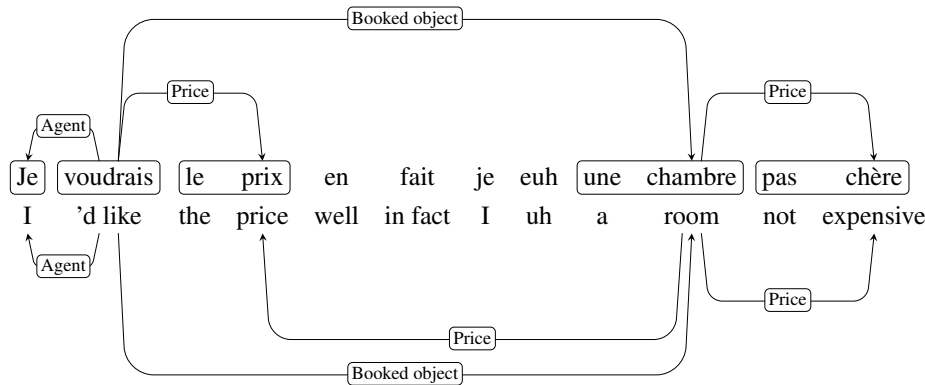
Figure 4: Illustration of the reassignment process following the expample presented in Figure 2. This example illustrates the third Metropolis proposed move, which changes the head of argument "le prix": arcs above the text represent the initial state, while arcs below the text represent the new proposed state.

5. When the sample is accepted, update the multinomials accordingly and iterate from step 1 until convergence.

This process is actually repeated for 2,000,000 iterations, and the sample that gives the largest joint probability is chosen.

3.3.2 Metropolis proposal distribution

The proposal distribution is used to explore the search space in an efficient way for the target application. Each state in the search space is uniquely defined by a value assignment to every variable in the model, for every utterance in the corpus. It corresponds to one possible *sample* of all variables, or in other words, to the choice of one possible semantic structure and class assignment to all utterances in the corpus.

Given a current state in this search space, the proposal distribution "proposes" to jump to a new state, which will then be evaluated by the Metropolis algorithm. Our proposal samples a new state in the following successive steps:

1. Sample uniformly one of the three possible moves:

Move1: Change the semantic class of a head;

Move2: Change the argument type of an argument segment;

Move3: Change the assignment of an argument to a new head;

2. If Move1 is chosen, sample uniformly one head segment and one target semantic class;

3. If Move2 is chosen, sample uniformly one argument segment and one target argument type;

4. If Move3 is chosen, sample uniformly one argument segment W_a and "detach" it from its current head. Then, sample uniformly one target head segment W_h', and reattach W_a to its new head W_h'. Because the distribution of argument types differ from one head class to another, it would be interesting at this stage to resample the argument type of W_a from the new head class distribution. But in this work, we resample the argument type from the uniform distribution.

This proposal distribution $Q(x \rightarrow x')$ is reversible, i.e., $Q(x \rightarrow x') > 0 \Rightarrow Q(x' \rightarrow x) > 0$. We can show that it is further symmetric, i.e., $Q(x \rightarrow x') = Q(x' \rightarrow x)$, because the same move is sampled to jump from x to x' than to jump from x' to x, and because the proposal distribution within each move is uniform.

4 Experimental validation

4.1 Experimental setup

The French MEDIA corpus collects about 70 hours of spontaneous speech (1258 dialogues, 46k utterances, 494.048 words and 4068 distinct words) for the task of hotel reservation and tourist information (Bonneau-Maynard et al., 2005). Calls from 250 speakers to a simulated reservation system (i.e. the Wizard-of-Oz) were recorded and transcribed. Dialogues are full of disfluencies, hesitations, false starts, truncations or fillers words (e.g., euh or ben).

16

Gold Standard Annotation	
Semantic Relation	Frequency
Agent	320
Booked object	298
Location	285
Time	209
Coordination	134
Beneficiary	117
Price	108
Reference Location	66

Table 2: Most frequent semantic relations in the gold annotation.

This corpus has been semantically annotated as part of the French ANR project PORT-MEDIA (Rojas-Barahona et al., 2011). We are using a set of 330 utterances manually annotated with gold semantic relations (i.e. High-Level Semantics). This gold corpus gathers 653 head segments and 1555 argument segments, from which around 20% are both arguments and heads, such as *une chambre* in Figure 4. Table 2 shows the semantic relations frequencies in the gold annotation. 12 head segment types and 19 different argument segment types are defined in the gold annotations. In the evaluation, we assume the number of both classes is given. A possible extension of the approach to automatically infer the number of classes would be to use a non-parametric model, but this is left for future work.

4.2 Evaluation metrics

The proposed method infers three types of semantic information:

- The semantic relation between an argument and its head;

- The argument type A

- The semantic class of the head C_t.

The three outcomes are evaluated as follows.

- The output structure is a forest of trees that is similar to a partial syntactic dependency structure. We thus use a classical unsupervised dependency parsing metric, the Unlabeled Attachment Score (UAS), which is simply the accuracy of argument attachment: an argument is correctly attached if and only if its inferred head matches the gold head.

- Both argument and head classes correspond to the outcome of a clustering process into semantic classes, akin to the semantic classes obtained in unsupervised semantic role labeling tasks. We then evaluate them with a classical metric used to evaluate these classes in unsupervised SRL (as done for instance in (Lang and Lapata, 2011a) and (Titov and Klementiev, 2012)): purity and collocation.

Purity measures the degree to which each cluster contains instances that share the same gold class, while collocation measures the degree to which instances with the same gold class are assigned to a single cluster.

More formally, the purity of argument segments' (head segment') clusters for the whole corpus is computed as follows:

$$PU = \frac{1}{N} \sum_i \max_j |G_j \cap C_i|$$

where C_i is the set of argument (head) segments in the ith cluster found, G_j is the set of argument (head) segments in the jth gold class, and N is the number of gold argument (head) segment instances. In a similar way, the collocation of argument segments' (head segment') clusters is computed as follows:

$$CO = \frac{1}{N} \sum_j \max_i |G_j \cap C_i|$$

Finally the F1 measure is the harmonic mean of the purity and collocation:

$$F1 = \frac{2 * CO * PU}{CO + PU}$$

4.3 Experimental results

We compare the proposed approach against two baselines:

- An argument-head "attachment" baseline, which attaches each argument to the closest head segment.

- A strong clustering baseline, which respectively clusters the head and argument segments using a very effective topic model: the Latent Dirichlet Allocation (LDA) approach (Blei et al., 2003).

Table 3 shows the UAS obtained for the proposed model on the MEDIA corpus, while Table 4 shows the obtained Purity, Collocation and F1-measure. In both cases, we compare the performances of the proposed model with the respective baseline. Our system outperforms both baselines by a large margin.

System	UAS
Closest attachment	68% (±2%)
Proposed - UAS	**74%** (±2%)

Table 3: Experimental results for UAS on the MEDIA database. The statistical confidence interval at 95% with Gaussian approximation is reported.

System	Purity	Col.	F-mes
LDA - Heads	51.7%	25.5%	34.2%
LDA - Args	31.7%	22.2%	26.1%
Proposed - Heads	78.7%	50.8%	**61.8%**
Proposed - Args	61.8%	53.3%	**59.3%**

Table 4: Experimental results on the MEDIA database for purity, collocation and F1-measure.

4.3.1 Qualitative Evaluation

We further carried out a qualitative evaluation, where we inspected the inferred clusters and compared them with the baseline. Figures 7 and 8 show, for every head class C_t in each stacked column, the distribution of instances from all gold clusters. Each column can also be viewed as a graphical representation of the intersection of one inferred class with all gold clusters. Figure 7 illustrates this for our model, and Figure 8 for LDA. The same comparison for the argument types is shown, respectively, in Figure 5 and Figure 6.

For head segment clusters, we can observe that most inferred clusters contain many instances of the Reservation type (in dark blue), both in the LDA baseline and in the proposed system. The main reason for that is that the corpus is very unbalanced in favor of the Reservation class, while we do not assume any prior knowledge about the data and thus use a uniform prior. Still, every other gold type that occurs with a reasonnably high enough frequency, apart from two special types that are discussed next, is well captured by one of
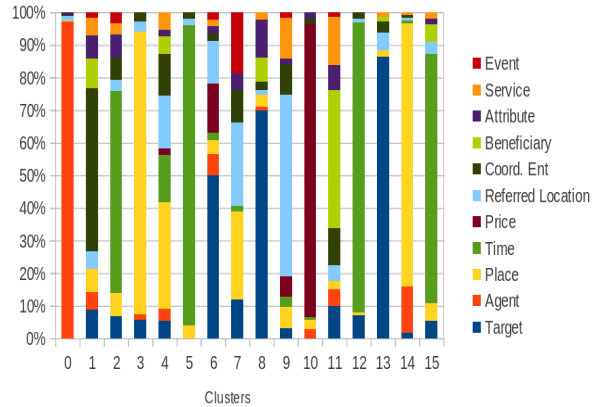

Figure 5: Distribution of the gold types (one per color) into the clusters inferred by our system (shown on the X-axis) for argument segments.

our inferred class: this is the case for "Room" that mainly intersects with our class 1, "Place" with our class 2 and "Hotel" with our class 9.

Some examples of instances for each case are:

- Reservation: *"voudrais réserver", "aimerais partir", "voudrais une *réservation une réservation", "prends", "recherche", "*désire désire", "il me faudrait", "opte", "aimerais s' il vous plaît si c' est possible avoir prendre".*

- Room: *"deux chambres pour un coup(le) avec trois enfants avec bon standing", "trois singles", "deux chambres de bon standing à peu près niveau trois étoiles", "trois doubles".*

- Place: *"Paris", "à Saintes", "à Charleville", "dans le dix huitième arrondissement de Paris".*

- Hotel: *"un hôtel deux étoiles", "dans un hôtel beau standing", "un hôtel formule un", "l' hôt(el) le l' hôtel", "un autre hôtel dans les mêmes conditions", "le Beaugency", "l' autre", "au Novotel", "le premier".*

Two "special" head segment types that are neither nicely captured by our system nor LDA are Coordination and Inform, which are instead assigned to the clusters corresponding to the gold segments that they coordinate or inform about.

For argument segments we also observed that the inferred clusters are semantically related to the gold types. We found, for instance, four clusters
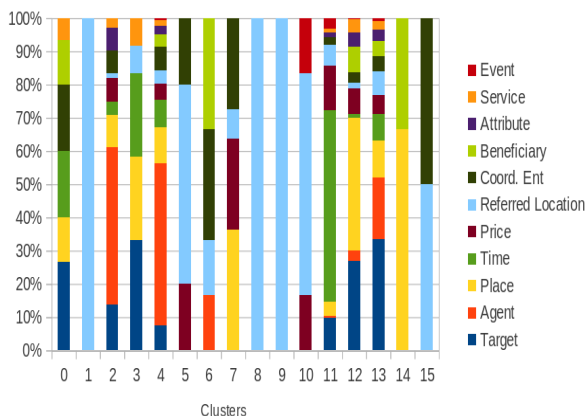
Figure 6: Distribution of the gold types (one per color) into the clusters inferred by the LDA baseline (shown on the X-axis) for argument segments.
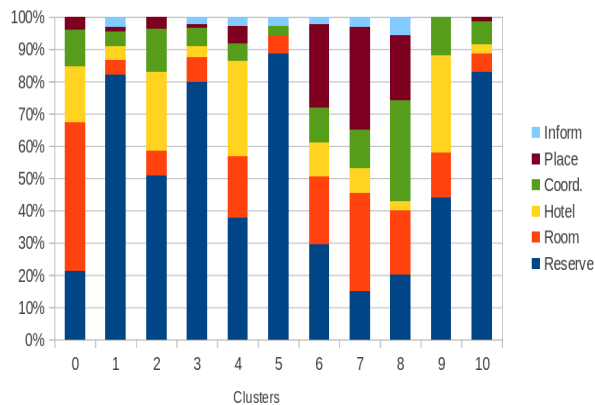

Figure 8: Distribution of the gold types (one per color) into the clusters inferred by the LDA baseline (shown on the X-axis) for head segments.
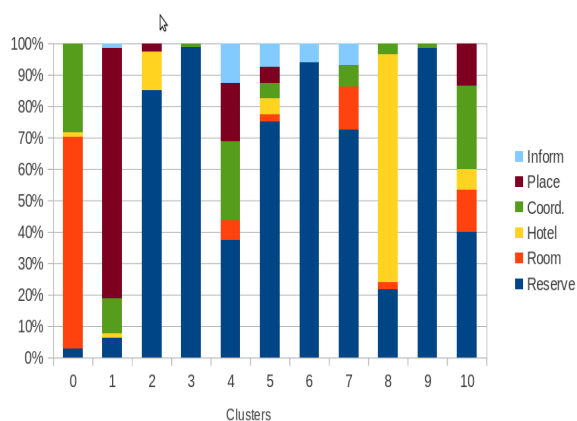

Figure 7: Distribution of the gold types (one per color) into the clusters inferred by our system (shown on the X-axis) for head segments.

(2, 5, 12 and 15) containing mainly "Time" arguments (*"du premier au trois Novembre", "dix nuit", "le festival du film", "au seize Novembre"*, etc.), two (3 and 14) dedicated to "Location" arguments (*"à Menton", "au festival lyrique de belle euh Belle Ile En mer", "bastille", "sur le ville de Paris", "parking privé"*), one (10) for "Price" arguments (*"pas plus de cent euros par personne", "un tarif inférieur à quatre vingts euros", "pas trop chère", "à cent vingt euros", "moins de cent* cent euros"*) etc.

Finally, as noted for the head segments, we can observe that the most frequent gold types largely intersect with several inferred clusters, for the same reason: data is very unbalanced and we do not assume any prior knowledge about the data

and thus use an uniform prior. Nevertheless, several other important classes such as Event, Price and Agent are well captured by our system.

5 Conclusions

This work proposes an unsupervised generative model to infer latent semantic structures on top of user spontaneous utterances. It relies on the Metropolis-Hastings sampling algorithm to jointly infer both the structure and semantic classes. It is evaluated in the context of the French MEDIA corpus for the hotel reservation task. Although the system proposed in this work is evaluated on a specific spoken dialog reservation task, it actually relies on a generic unsupervised structured inference model and can thus be applied to many other structured inference tasks, as long as observed word segments are given.

An interesting future direction of research would be to modify this model so that it jointly infers both the latent syntactic and semantic structures, which are known to be closely related but still carry complementary information. We of course also plan to evaluate the proposed model with automatic speech transcriptions and concepts decoding. Another advantage of the proposed model is the possibility to build better Metropolis-Hastings proposals, which may greatly improve the convergence rate of the algorithm. In particular, we would like to investigate the use of some non-uniform proposal distributions when reattaching an argument to a new head, which shall improve mixing.

References

Omri Abend, Roi Reichart, and Ari Rappoport. 2009. Unsupervised argument identification for semantic role labeling. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, ACL '09, pages 28–36, Stroudsburg, PA, USA. Association for Computational Linguistics.

D.M. Blei, A.Y. Ng, and M.I. Jordan. 2003. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.

Helene Bonneau-Maynard, Sophie Rosset, Christelle Ayache, Anne Kuhn, and Djamel Mostefa. 2005. Semantic annotation of the french MEDIA dialog corpus. In *INTERSPEECH-2005, 3457-3460.*

N. Camelin, B. Detienne, S. Huet, D. Quadri, and F. Lefèvre. 2011. Unsupervised concept annotation using latent dirichlet allocation and segmental methods. In *EMNLP 1st Workshop on Unsupervised Learning in NLP*, Edinburgh (UK).

Alexandre Denis, Matthieu Quignard, and Guillaume Pitel. 2006. A Deep-Parsing Approach to Natural Language Understanding in Dialogue System: Results of a Corpus-Based Evaluation. In *Proceedings of the 5th international Conference on Language Resources and Evaluation (LREC 2006) Proceedings of Language Resources and Evaluation Conference*, pages 339–344, Genoa Italie.

Koen Deschacht and Marie-Francine Moens. 2009. Semi-supervised semantic role labeling using the latent words language model. In *Proc. EMNLP*, pages 21–29.

Hagen Fürstenau and Mirella Lapata. 2009. Graph alignment for semi-supervised semantic role labeling. In *Proc. EMNLP*, pages 11–20.

Trond Grenager and Christopher D. Manning. 2006. Unsupervised discovery of a statistical verb lexicon. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.

S. He and H. Gildea. 2006. Self-training and Cotraining for Semantic Role Labeling: Primary Report. Technical report, TR 891, University of Colorado at Boulder.

Stéphane Huet and Fabrice Lefèvre. 2011. Unsupervised alignment for segmental-based language understanding. In *Proceedings of the First Workshop on Unsupervised Learning in NLP*, EMNLP '11, pages 97–104, Stroudsburg, PA, USA. Association for Computational Linguistics.

Joel Lang and Mirella Lapata. 2010. Unsupervised induction of semantic roles. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 939–947, Stroudsburg, PA, USA. Association for Computational Linguistics.

Joel Lang and Mirella Lapata. 2011a. Unsupervised semantic role induction via split-merge clustering. In *Proc. ACL*, pages 1117–1126.

Joel Lang and Mirella Lapata. 2011b. Unsupervised semantic role induction with graph partitioning. In *EMNLP*, pages 1320–1331. Association for Computer Linguistics.

Joo-Young Lee, Young-In Song, and Hae-Chang Rim. 2007. Investigation of weakly supervised learning for semantic role labeling. In *ALPIT*, pages 165–170. IEEE Computer Society.

Marie-Jean Meurs, Fabrice Lefèvre, and Renato de Mori. 2009. Spoken language interpretation: On the use of dynamic bayesian networks for semantic composition. In *Proc. ICASSP*, pages 4773–4776.

Nitin Namdeo Pise and Parag Kulkarni. 2008. A survey of semi-supervised learning methods. In *Proceedings of the 2008 International Conference on Computational Intelligence and Security - Volume 02*, CIS '08, pages 30–34, Washington, DC, USA. IEEE Computer Society.

Lina Maria Rojas-Barahona, Thierry Bazillon, Matthieu Quignard, and Fabrice Lefevre. 2011. Using MMIL for the high level semantic annotation of the french MEDIA dialogue corpus. In *Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011)*.

Robert S. Swier and Suzanne Stevenson. 2004. Unsupervised Semantic Role Labelling. In *EMNLP*, pages 95–102. Association for Computational Linguistics.

Ivan Titov and Alexandre Klementiev. 2012. A bayesian approach to unsupervised semantic role induction. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, Avignon, France, April.

Xiaojin Zhu. 2005. Semi-Supervised Learning Literature Survey. Technical report, Computer Sciences, University of Wisconsin-Madison.