Hypotheses Selection Criteria in a Reranking Framework for Spoken Language Understanding

Marco Dinarelli
LIMSI-CNRS
B.P. 133, 91403 Orsay Cedex
France
marcod@limsi.fr

Sophie Rosset
LIMSI-CNRS
B.P. 133, 91403 Orsay Cedex
France
rosset@limsi.fr

Abstract

Reranking models have been successfully applied to many tasks of Natural Language Processing. However, there are two aspects of this approach that need a deeper investigation: (i) Assessment of hypotheses generated for reranking at classification phase: baseline models generate a list of hypotheses and these are used for reranking without any assessment; (ii) Detection of cases where reranking models provide a worst result: the best hypothesis provided by the reranking model is assumed to be always the best result. In some cases the reranking model provides an incorrect hypothesis while the baseline best hypothesis is correct, especially when baseline models are accurate. In this paper we propose solutions for these two aspects: (i) a semantic inconsistency metric to select possibly more correct n-best hypotheses, from a large set generated by an SLU basiline model. The selected hypotheses are reranked applying a state-of-the-art model based on Partial Tree Kernels, which encode SLU hypotheses in Support Vector Machines with complex structured features; (ii) finally, we apply a decision strategy, based on confidence values, to select the final hypothesis between the first ranked hypothesis provided by the baseline SLU model and the first ranked hypothesis provided by the re-ranker. We show the effectiveness of these solutions presenting comparative results obtained reranking hypotheses generated by a very accurate Conditional Random Field model. We evaluate our approach on the French MEDIA corpus. The results show significant improvements with respect to current state-of-the-art and previous re-ranking models.

1 Introduction

Discriminative reranking is a widely used approach for several Natural Language Processing (NLP) tasks: Syntactic Parsing (Collins, 2000), Named Entity Recognition (Collins, 2000; Collins and Duffy, 2001), Semantic Role Labelling (Moschitti et al., 2008), Machine Translation (Shen et al., 2004), Question Answering (Moschitti et al., 2007). Recently reranking approaches have been successfully applied also to Spoken Language Understanding (SLU) (Dinarelli et al., 2009b).

Discriminative Reranking combines two models: a first SLU model is used to generate a ranked list of *n*-best hypotheses; a reranking model sorts the list based on a different score and the final result is the new top ranked hypothesis. The advantage of reranking approaches is in the possibility to learn directly complex dependencies in the output domain, as this is provided in the hypotheses generated by the baseline model.

In previous approaches complex features are extracted from the hypotheses for both training and classification phase, but there are very few studies on approaches that can be applied to search in the hypotheses space generated by the baseline SLU model. Moreover, to keep overall computational cost reasonable, the size of the n-best list is typically small (few tens). This is a limitation since the larger is the hypotheses space generated, the more likely is to find a better hypothesis. On the other hand, reranking a large set of hypotheses is computationally

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 1104–1115,
Edinburgh, Scotland, UK, July 27–31, 2011.

expensive, thus a strategy to select the best hypotheses to be re-ranked would overcome this problem.

Another aspect of reranking that deserves to be deeper studied is its applicability. Although a reranking model improves the baseline model in the overall performance, in some cases the reranked best hypotheses can contain more mistakes than the baseline best hypothesis. A strategy to decide when the reranking model should be applied and when the first hypothesis of the baseline model is more accurate would improve reranking performances.

In this paper, we propose two new models for improving discriminative reranking: (a) a semantic inconsistency metric that can be applied to SLU hypotheses to select those that are more likely to be correct; (b) a model selection strategy based on the confidence scores provided by the baseline SLU model and the reranker. This provides a decision function that detects if the original top ranked hypothesis is more accurate than the reranked best hypothesis.

Our re-ranking strategies turn out to be effective on very accurate baseline models based on state-of-the-art Conditinal Random Fields (CRF) implementation (Lavergne et al., 2010). We evaluate our approach on the well-known French MEDIA corpus for SLU (Bonneau-Maynard et al., 2006). The results show that our approach significantly improves both "traditional" reranking approaches and state-of-the-art SLU models.

The remainder of the paper is organized as follows: in Section 2 we introduce the SLU task. Section 3 describes our discriminative reranking framework for SLU, in particular the baseline model adopted, in sub-section 3.1, and the reranking model, in sub-section 3.2. Section 4 describes the two strategies proposed in this paper for SLU reranking, whereas the experiments to evaluate our approaches are described in Section 5. Finally, after a discussion in Section 6, in Section 7 we draw some conclusions.

2 Spoken Language Understanding

Spoken Language Understanding is the task of representing and extracting the meaning of natural language sentences. Designing a general meaning representation which can capture the semantics of a spoken language is very complex. Therefore, in practice, the meaning representations depend on the specific application domain being modeled.

For the corpus used in this work, the semantic representation is defined in an ontology described in (Bonneau-Maynard et al., 2006). As an example, given the following natural language sentence translated from the MEDIA corpus:

"*Good morning I would like to book an hotel room in London*"

The semantic representation extraction for the SLU task is performed in two steps:

1. **Automatic Concept Labeling**

   Null{*Good morning*} **command-task**{*I would like to book*} **object-bd**{*an hotel room*} **localization-city**{*in London*}

2. **Attribute-Value Extraction**

   command-task[reservation] **object-bd**[hotel] **localization-city**[London]

command-task, **object-bd** and **localization-city** are three domain concepts, called also "attributes", defined in the ontology and **Null** is the concept for words not associated to any concept. As shown in the example, **Null** concepts are removed from the final output since they don't bring any semantic content with respect to the application domain. **reservation**, **hotel** and **London** are the normalized attribute values, defined also in the application ontology. This representation is usually called attribute-value representation.

In the last decade several probabilistic models have been proposed for the Automatic Concept Labeling step: in (Raymond et al., 2006) a conceptual language model encoded in Stochastic Finite State Transducers (SFST) is proposed. In (Raymond and Riccardi, 2007), the SFST-based model is compared with Support Vector Machines (SVM) (Vapnik, 1998) and Conditional Random Fields (CRF) (Lafferty et al., 2001). Moreover, in (Hahn et al., 2008a) two more models are applied to SLU: a Maximum Entropy (EM) model and a model coming from the Statistical Machine Translation (SMT) community (it is actually a log-linear combination of SMT models). Among these models, CRF has shown in general superior performances on sequence labeling tasks like Named Entity Recognition (NER) (Tjong Kim Sang and De Meulder, 2003), Grapheme-to-Phoneme transcription (Sejnowski and Rosenberg,

1987) and also Spoken Language Understanding (Hahn et al., 2008a).

In addition to individual systems, more recently also some system combination approaches have been tried on SLU. In (Hahn et al., 2010), two such approaches are compared, one based on weighted ROVER (Fiscus, 1997) while the other is the reranking approach proposed in (Dinarelli et al., 2009b). Both system combination approaches are applied on the MEDIA corpus, thus we will refer to (Hahn et al., 2010) for a comparison with our approach.

Like the other tasks mentioned above, SLU is usually a supervised learning task, this means that models are learned from annotated data. This is an important aspect to take into account when designing SLU systems. In this respect accurate SLU models can in part alleviate the problem of manually annotating data.

The second step of SLU, that is Attribute Value Extraction (from now on *AVE*) is performed with two approaches: a) Rule-based approaches apply Regular Expressions (RE) to map the words realizing a concept into a normalized value. Regular expressions are defined for each attribute-value pair. Given a concept and its realizing surface form, if a RE for that concept matches the surface, the corresponding value is returned.

An example of surfaces that can be mapped into the value **hotel** given the concept **object-bd** is:

1. *an hotel room*
2. *a hotel room*
3. *the hotel*

   ...

Note that these surfaces share the same keyword for the concept **object-bd**, which is "*hotel*". Thus, a possible rule extracted from data, for the concept **object-bd** can be:

$R_{object-bd}(S)$ =
if S = "an hotel room" **or**
S = "a hotel room" **or**
S = "the hotel" **then**
return "hotel"
end

This kind of rules can be easily refined using regular expressions, so that they can capture all possible linguistic patterns containing the triggering keyword ("hotel" in the example).

b) The other approach used for attribute value extraction is based on probabilistic models. In this case the model learns from data the conditional probability of values V, given the concept C and the corresponding sequence of words W realizing the concept: $P(V|W, C)$.

The most meaningful work about AVE approaches in SLU tasks is (Hahn et al., 2010).

The model used in this work for *Automatic Concept Labeling* is based on CRF. For the *Attribute-Value Extraction* phase we use a combination of rule based and probabilistic approaches. The first is made of regular expressions, as explained above. The probabilistic approach is based again on CRF.

3 Reranking Framework

This section describes the different models involved in the pipeline realising our reranking framework:

- Conditional Random Fields
- Semantic Inconsistency Metric for hypotheses selection, which is optional and is applied only at the classification phase
- Support Vector Machines with Partial Tree Kernel
- Decision Strategy to detect when the top ranked hypothesis of the baseline model is more accurate than the reranked best hypothesis

It is important to underline that the phases involved in the reranking framewrok are distinguished for a matter of clarity. In principle, the phases from the hypotheses selection to the last, the decision strategy, can be thought of as a whole reranking model.

In the next two subsection we describe the two models used for hypotheses generation and for reranking: CRF and SVM with kernel methods. The two improvements proposed in this paper and listed above are presented in a dedicated section (4).

3.1 Conditional Random Fields

CRFs have been proposed for the first time for sequence segmentation and labeling tasks in (Lafferty et al., 2001). This model belongs to the family of exponential or log-linear models. Its main characteristics are the possibility to include a huge number

of features, like the Maximum Entropy (ME) model, but computing global conditional probabilities normalized at sentence level, instead of position level like in ME. In particular this last point results very effective since it solves the label bias problem, as pointed out in (Lafferty et al., 2001).

Given a sequence of N words $W_1^N = w_1, ..., w_N$ and its corresponding sequence of concepts $C_1^N = c_1, ..., c_N$, CRF trains the conditional probabilities

$$P(C_1^N|W_1^N) = \frac{1}{Z}\prod_{n=1}^{N}\exp\left(\sum_{m=1}^{M}\lambda_m \cdot h_m(c_{n-1}, c_n, w_{n-2}^{n+2})\right) \quad (1)$$

where λ_m are the training parameters. $h_m(c_{n-1}, c_n, w_{n-2}^{n+2})$ are the feature functions capturing conditional dependencies of concepts and words. Z is a probability normalization factor in order to model well defined probability distribution:

$$Z = \sum_{\tilde{c}_1^N}\prod_{n=1}^{N} H(\tilde{c}_{n-1}, \tilde{c}_n, w_{n-2}^{n+2}) \quad (2)$$

here $\tilde{c}_{n-1}$ and $\tilde{c}_n$ are the concepts hypothesized for the previous and current words, $H(\tilde{c}_{n-1}, \tilde{c}_n, w_{n-2}^{n+2})$ is an abbreviation for $\sum_{m=1}^{M}\lambda_m \cdot h_m(c_{n-1}, c_n, w_{n-2}^{n+2})$.

The CRF model used for the *Attribute-Value Extraction* phase learns in the same way the conditional probability $P(V_1^N|C_1^N, W_1^N)$. In particular we use attributes-words concatenations on the source side and attribute values on the target side.

Two particular effective implementations of CRFs have been recently proposed. One is described in (Hahn et al., 2009) and uses a margin based criterion for probabilities estimation. The other is described in (Lavergne et al., 2010) and has been implemented in the software *wapiti*[1]. The latter solution in particular trains the model using two different regularization factors at the same time:

Gaussian prior, used as *l2* regularization and used in many softwares to avoid overfitting;

Laplacian prior, used as *l1* regularization (Riezler and Vasserman, 2010), which has the effect to filter out features with very low scores.

[1]available at `http://wapiti.limsi.fr`

The two regularization parameters are used together in the model implementing the so-called *elastic net* regularization (Zou and Hastie, 2005):

$$l(\lambda) + \rho_1\|\lambda\|_1 + \frac{\rho_2}{2}\|\lambda\|_2^2 \quad (3)$$

λ is the set of parameters of the model introduced in equation 1, $l(\lambda)$ is the minus-logarithm of equation 1, used as loss function for training CRF. $\|\lambda\|_1$ and $\|\lambda\|_2$ are the *l1* and *l2* regularization, respectively, while ρ_1 and ρ_2 are two parameters that can be optimized as usual on development data or with cross validation.

As explained in (Lavergne et al., 2010), using *l1* regularization is an effective way for feature selection in CRF at training time. Note that other approaches have been proposed for feature selection, e.g. in (McCallum, 2003). This type of features selection, performed directly at training time, yields very accurate models, since only the most meaningful features are kept in the final model, which guarantee a strong robustness on unseen data.

In this work we refer in particular to the CRF implementation described in (Lavergne et al., 2010).

3.2 SVM and Kernel Methods

Our reranking model is based on SVM (Vapnik, 1998) with the use of the Partial Tree Kernel defined in (Moschitti, 2006).

SVMs are well-known machine learning algorithms belonging to the class of maximal-margin linear classifiers (Vapnik, 1998). The model represents a hyperplane which separates the training examples with a maximum margin. The hyperplane is learned using optimization theory and is represented in the dual form as a linear combination of training examples:

$\sum_{i=1..l} y_i\alpha_i\vec{x_i}\vec{x} + b = 0$,

where $\vec{x_i}, i \in [1, .., l]$ are training examples representing objects o_i and o in any feature space, y_i is the label associated with $\vec{x_i}$ and α_i are the lagrange multipliers. The dual form of the hyperplane shows that SVM training depends on the inner product between instances. Kernel methods theory (Shawe-Taylor and Cristianini, 2004), allows us to substitute the inner product with a so-called kernel function, computing the same result: $K(o_i, o) = \vec{x_i} \cdot \vec{x}$.

The interesting aspect of using such formulation is the possibility to compare objects in arbitrarily complex feature spaces implicitly, i.e. without knowing the features to be used. Since in real world scenarios data cannot be classified using a simple linear classifier, kernel methods can be used to carry out learning in complex feature spaces. In this work we use the Partial Tree Kernel (PTK) (Moschitti, 2006).

3.3 Reranking Model

In order to give an effective representation to SLU hypotheses in SVM, since we are using PTK, we need to represent as trees SLU hypotheses like the one described in section 2.

This problem is easily solved by transforming the hypotheses into trees like the one depicted in figure 1. Although there may be more formal solutions to represent semantic information of SLU hypotheses as trees, we would like to remark that the tree structure shown in figure 1 contains all the key information needed for our purposes: the first level of the tree represents the concept sequence annotated on surface form. The second level of the tree allow to implicitly represent the segmentation of each concept, while the third level, i.e. the leaves, are the input words. Moreover, from figure 1 we removed word categories associated to words in order to keep the figure readable. Word categories are provided together with the corpus as an application knowledge base. They comprise domain categories like city names, hotel names, street names etc., and some domain independent categories like numbers, dates, months etc. The categories are used at the same level of words, they provide a generalization over words and alleviate the effect of Out-of-Vocabulary (OOV) words.

The CRF model used as baseline generates the *n* most likely conceptual annotations for each input sentence. These are ranked by the global conditional probability of the concept sequence, given the input word sequence of CRF. The *n*-best list produced by the baseline model is the list of candidate hypotheses $H_1, H_2, .., H_n$ used in the reranking step.

The candidate hypotheses are organized into pairs, e.g. (H_1, H_2) or (H_1, H_3). We build training pairs such that a reranker can learn to select the best one between the two hypotheses in a pair, i.e. the more correct hypothesis with respect to a reference annotation and a given metric. In particular, we compute the edit distance of each hypothesis in the list, with respect to the manual annotation taken from the corpus. The best hypothesis H_b is used to build positive instances for the reranker as pairs (H_b, H_i) for $i \in [1..n]$ and $i \neq b$, negative instances are built as (H_i, H_b), with same constraints on index i. This means that, if n hypotheses are generated for a sentence, $2 \cdot n$ instances are generated from them. Note that by construction of pairs the model is symmetric, this provides a property that will be exploited at classification phase, as described in (Shen et al., 2003b).

Hypotheses are then converted into trees like the one shown in figure 1. Pairs of trees $e_k = (t_{i,k}, t_{j,k})$, for k varying along all the training or classification instances, are given as input to the SVM model to train the reranker using the following reranking kernel:

$$\begin{aligned} K_R(e_1, e_2) &= PTK(t_{1,1}, t_{1,2}) + PTK(t_{2,1}, t_{2,2}) \\ &- PTK(t_{1,1}, t_{2,2}) - PTK(t_{2,1}, t_{1,2}), \end{aligned} \quad (4)$$

where e_1 and e_2 are two pairs of trees to be compared.

The reranking kernel in equation 4, consisting in summing four different kernels, has been proposed in (Shen et al., 2003b) for syntactic parsing reranking, where the basic kernel was a Tree Kernel, and the idea was taken in turn from (Heibrich et al., 2000), where pairs where used to learn preference ranking. The same idea appears also, in a slightly different form, in early work about reranking, e.g. (Collins and Duffy, 2002). The same reranking schema has been used also in (Shen et al., 2004) for reranking different candidate hypotheses for machine translation.

For classification, observing that the model is symmetric and exploiting kernel properties, we can use, as classification instances, simple hypotheses instead of pairs. More precisely we use pairs where the second hypothesis is empty, i.e. $(H_i, 0)$, for $i \in [1..n]$. This simplification allow a relatively fast classification phase, since only n instances are generated for each sentence, instead of n^2. This simplification has been proposed in (Shen et al., 2003b).

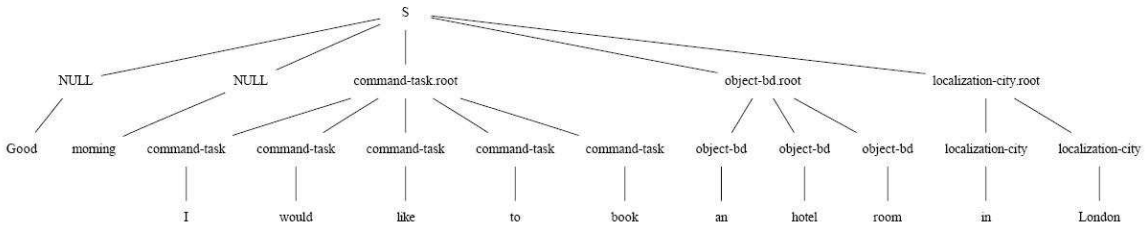


Figure 1: An example of semantic tree constructed from an SLU hypothesis from the MEDIA corpus and used in PTK

4 Hypotheses Selection Criteria

This section describes the main contribution of our work: first, a semantic inconsistency metric based on the AVE phase of SLU and allowing to select hypotheses generated by the baseline model; second, a strategy to decide, after the reranking phase, if it is more likely that the baseline best hypothesis is more accurate than the best reranked hypothesis and allowing to recover the mistake. Similar ideas have been proposed in (Dinarelli et al., 2010), here we propose a significant evolution and we give a much wider description and evaluation.

4.1 Hypotheses Selection via Attribute Value Extraction (AVE)

In previous reranking approaches (Collins, 2000; Collins and Duffy, 2002; Shen et al., 2003a; Shen et al., 2003b; Shen et al., 2004; Collins and Koo, 2005; Kudo et al., 2005; Dinarelli et al., 2009b), few hypotheses are generated with the baseline model, ranked by the model probability. These are then used for the reranking model. An interesting strategy to improve reranking performance is the selection of the best set of hypotheses to be reranked.

In this work we propose a semantic inconsistency metric (SIM) based on the attribute-value extraction phase that allows to select better n-best hypotheses. We combine the scores provided by the rule based approach and the CRF approach for AVE, computing a confidence measure.

The rule-based approach for AVE is defined by a set of rules that map concepts and their realizing words into the corresponding value. The rules are extracted from the training data, thus they are defined to extract correct values from well formed phrases annotated with correct concepts. This means that when the corresponding words are annotated with a wrong concept, the extracted value will probably be wrong. We use this property to compute a semantic inconsistency value for hypotheses, which in turn allows to select hypotheses with higher probabilities to be correct.

We show the application of SIM using the same example of Section 2. For space issues we abbreviate **command-task** with **com-task**, **object-bd** with **obj-bd** and **localization-city** with **loc-city**. We also suppose to have already removed **Null** concepts. From the same sentence, the three first hypotheses that may be generated by the baseline model are:

1. **obj-bd**{*I would like to book*} **obj-bd**{*an hotel room*} **loc-city**{*in London*}
2. **com-task**{*I would like to book*} **obj-bd**{*an hotel room*} **loc-city**{*in London*}
3. **com-task**{*I would like to book*} **obj-bd**{*an hotel*} **obj-bd**{*room*} **loc-city**{*in London*}

Two of these annotations show typical errors of an SLU model:
(i) wrong concepts annotation: in the first hypothesis the phrase "I would like to book" is erroneously annotated as **obj-bd**;
(ii) wrong concept segmentation: in the third hypothesis the phrase "an hotel room" is split in two concepts.

If we apply the AVE module to these hypotheses the result is:

1. **obj-bd**[] **obj-bd**[hotel] **loc-city**[london]
2. **cmd-task**[reservation] **obj-bd**[hotel] **loc-city**[london]
3. **cmd-task**[reservation] **obj-bd**[hotel] **obj-bd**[] **loc-city**[london]

As we can see the first concept **obj-bd** in the first hypothesis has an empty value since it was incorrectly annotated and, therefore, it is not supported

MEDIA	training		dev		test	
# sentences	12,908		1,259		3,005	
	words	concepts	words	concepts	words	concepts
# tokens	94,466	43,078	10,849	4,705	25,606	11,383
# vocabulary	2,210	99	838	66	1,276	78
# singletons	798	16	338	4	494	10
# OOV rate [%]	–	–	1.33	0.02	1.39	0.04

Table 1: Statistics of the MEDIA training and evaluation sets used for all experiments.

by words from which the AVE module can extract a correct value. In this case, the output of AVE is empty. In the same way, in the third hypothesis, the AVE module cannot extract a correct value from the phrase "room" since it doesn't contain any keyword for a **obj-bd** concept.

For each hypothesis, our SIM simply counts the number of wrong (or empty) values. In the example above, we have 1, 0 and 1 for the three hypothesis, respectively. Accordingly, the most accurate hypothesis under SIM is the second, which is also the correct one.

In order to combine the SIM score computed by the rule-based AVE module with the score provided by the CRF AVE model, we consider per-concept scores from both approaches. In particular, we formalize the definition of the SIM metric above on a concept c_i as $SIM(c_i, w_i^{1,...,m})$. The value of SIM is simply 0 if the rule-based AVE module can extract a value from the surface form $w_i^{1,...,m}$ realizing the concept c_i. 1 otherwise. For each concept in a hypothesis, we compute its semantic consistency $s(c_i)$ as

$$s(c_i) = \frac{P(v_i|c_i, w_i^{1,...,m})}{SIM(c_i, w_i^{1,...,m}) + 1} \quad (5)$$

where $P(v_i|c_i, w_i^{1,...,m})$ is the conditional probability output by the CRF model for the value v_i, given the concept c_i and its realizing surface $w_i^{1,...,m}$. Equation 5 means that the CRF score provided for a given value is halved if SIM returns 1, i.e. if the AVE module cannot extract any value. Otherwise the score output by the CRF AVE model is kept unchanged. The semantic inconsistency metric of an hypothesis H_k containing the concept sequence $C_1^N = c_1, ..., c_N$ is then defined as

$$S(H_k) = \sum_{i=1}^{N} s(c_i) \quad (6)$$

Using $S(H_k)$ as semantic inconsistency metric, we generate a huge number of hypotheses with the baseline model and we select only the top *n*-best. We use these hypotheses in the discriminative reranking model, instead of the original *n*-best generated by the CRF model. For simplicity, in general context we denote $S(H_k)$ as `SIM`.

4.2 Wrong Rerank Rejection

After the reranking model is applied, the first hypothesis is selected as final result. This choice assumes that the new hypothesis is more accurate than the one provided by the baseline model. In general this assumption is not true. Indeed, a reranking model must be carefully tuned in order to correctly rerank wrong first best hypotheses but keeping the original baseline best for correct hypotheses. When the baseline model is relatively accurate, the latter case occurs in most of the cases. In this situation it becomes hard to train an accurate reranking model.

Our idea to overcome this problem is to apply the reranking model and then post-process results to detect when the original best hypothesis is actually better than the reranked best.

For this purpose we propose a simple strategy based on the scores computed by the two models involved in reranking: CRF for the baseline and SVM with PTK for reranking.

Let H_{crf} and H_{RR} be the best hypothesis of the CRF and reranking (RR) models, respectively. Let $S_{crf}(H_{crf})$ and $S_{crf}(H_{RR})$ be the scores of the CRF model for H_{crf} and H_{RR}. In the same way, let $S_{RR}(H_{crf})$ and $S_{RR}(H_{RR})$ be the scores of the reranking model on the same hypotheses. We define the *confidence margin* of the CRF model the quantity: $M_{crf} = S_{crf}(H_{crf}) - S_{crf}(H_{RR})$.

In the same way we define the *confidence margin* of the RR model: $M_{RR} = S_{RR}(H_{crf}) - S_{RR}(H_{RR})$.

We compute two thresholds T_{crf} and T_{RR} for the

Average score	Feature type
0.0528186	Pref2
0.044189	CATEGORY-2
0.0355579	CATEGORY
0.0354006	Pref3-2
0.0338949	Pref4-2
0.0332647	Suff3-2
0.0314831	Suff2
0.030613	Suff4-2
...	...
0.0165602	Suff1
0.000579602	Pref1

Table 2: Ranks of average score given by the CRF model to feature types

two margins with respect to error rate minimization (with a *"line search"* algorithm).

We select the final best interpretation hypothesis for a given sentence with the decision function:

$$BestHypothesis = \begin{cases} H_{RR} & \text{if } M_{crf} \leq T_{crf} \text{ and } M_{RR} \geq T_{RR} \\ H_{crf} & \text{otherwise.} \end{cases}$$

Since this strategy allows to recover from reranking mistakes, we call it Wrong Rerank Rejection (WRR).

5 Experiments

The data used in our experiments are taken from the French MEDIA corpus (Bonneau-Maynard et al., 2006). The corpus is made of 1.250 Human-Machine dialogs acquired with a Wizard-of-Oz approach in the domain of informtation and reservation of French hotels. The data are split into training, development and test set. Statistics of the corpus are presented in table 1.

For our CRF models, both *Automatic Concept Annotation* and *Attribute Value Extraction* SLU phases, we used wapiti[2] (Lavergne et al., 2010). The CRF model for the first SLU phase integrates a traditional set of features like word prefixes and suffixes (of length up to 5), plus some *Yes/No* features like *"Does the word start with capital letter ?"*, *"Does the word contain non alphanumeric characters ?"*, *"Is the word preceded by non alphanumeric characteris ?"* etc. The CRF model for AVE integrates only words, prefixes and suffixes (length 3 and 4) concatenated with concepts. Since in this case labels are attribute values, which are a huge set with respect to concepts ($\tilde{7}$00 VS 99), using a lot of features would make model training problematic. Despite the reduced set of features, training error rate at both token and sentence level is under 1%. We didn't carry out optimization for parameters ρ_1 and ρ_2 of the elastic net (see section 3.1), default values lead in most cases to very accurate models.

MEDIA Text Input	DEV		TEST	
Model	Attr	Attr+Val	Attr	Attr+Val
CRF	12.1%	14.8%	11.5%	13.8%
CRF+RR	12.0%	14.6%	11.5%	13.7%
CRF+RR_{SIM}	11.7%	13.9%	11.3%	13.4%
CRF+RR_{WRR}	11.2%	13.4%	11.3%	13.0%

Table 3: Results of baseline CRF model and reranking models on MEDIA text input

Reranking models based on SVM and PTK have been trained with *"SVM-Light-TK"*[3]. Kernel parameters *M* and SVM parameter *C* have been optimized on the development set, as well as thresholds for the WRR (see section 4.2).

Concerning hypotheses generation, for training we generate 100 hypotheses, we select the best with respect to the edit distance and the reference annotation and we keep a total of 10 hypotheses to build pairs. For classification, with the "standard" reranking approach we generate and we keep the 10 best hypotheses. While using SIM for hypotheses selection, we generate 1.000 hypotheses and we keep the 10 best with respect to SIM. 1.000 is the best threshold between oracle accuracy and computational cost for evaluating the hypotheses.

Experiments have been performed on both manual and automatic transcriptions of dialog turns. For automatic transcriptions the WER of the ASR is 30.3% on development set and 31.4% on test set.

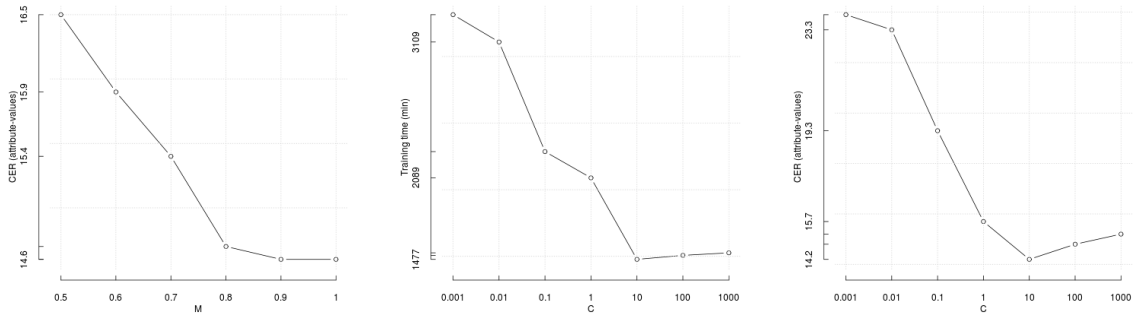
All results are reported in terms of Concept Error Rate (CER), which is the same as WER, but it is computed on concept sequences. In all cases we give results for both attributes only and attributes and values extraction

5.1 Results

In order to understand feature relevance, in table 2 we report feature types ranked by the average score given by the CRF model. Each type correspond to features at any position with respect to the target

[2]available at http://wapiti.limsi.fr

[3]available at http://disi.unitn.it/moschitti/Tree-Kernel.htm



(a) M kernel parameter VS CER (on attribute-value extraction)

(b) C SVM parameter VS training time

(c) C SVM parameter VS CER (on attribute-value extraction)

Figure 2: Optimization of the PTK M parameter and C parameter of SVM

MEDIA Speech Input	DEV		TEST	
Model	Attr	Attr+Val	Attr	Attr+Val
CRF	24.1%	29.1%	23.7%	27.6%
CRF+RR	23.9%	29.1%	23.5%	27.6%
CRF+RR_{SIM}	23.9%	28.3%	23.2%	26.8%
CRF+RR_{WRR}	23.3%	27.5%	22.7%	26.1%

Table 4: Results of baseline CRF model and reranking models on MEDIA speech input

word, with label unigrams. In contrast observation unigrams are distinguished from bigrams using suffixes *-1* and *-2* respectively. Feature types *wrd* are words converted to lower case, *Wrd* are words kept with original capitalization. Feature types *Pre*n are word prefixes of length n, *Suf*n are word suffixes of length n. *CATEGORY* features are word categories (see section 3.3). As we can see from the table, although feature relevance depends of course from the task, surprisingly word prefixes of length 2 are the most meaningful features. As expected, *CATEGORY* features are also very relevant features, since they provide a strong generalization over words. Another expected outcome is the fact that prefixes and suffixes of length 1 are the least relevant features.

In figure 2(a), 2(b) and 2(c) we show the curves resulting from optimization of parameters of reranking models. In particular we optimized the *M* kernel parameter (μ decay factor, see (Moschitti, 2006) for details), and the *C* SVM parameter, i.e. the scale factor for the soft margin (please refer to (Vapnik, 1998) for SVM details). Figure 2(b) shows the learning time as a function of the *C* SVM parameter. This gives an idea of how long takes training our reranking models.

In table 3 and 4 we report comparative results over the baseline CRF model, the baseline reranking model (*CRF+RR*) and the reranking models obtained applying the two improvements proposed in this work (*CRF+*RR_{SIM} and *CRF+*RR_{WRR}). As we can see, the baseline reranking model does not improve significantly the baseline CRF model. This outcome is expected since we don't use any other information in the reranking model than the semantic tree shown in figure 1. Previous approaches like for example (Collins and Duffy, 2002), use the baseline model score as feature, as that the reranking model cannot do worst than the baseline model. As we pointed out in section 4.2, this solution require a fine tuning of the reranking model, especially when the baseline model is relatively accurate. In our case, the CRF model has a Sentence Error Rate of 25.0% on the MEDIA test set. This means that 75% of the times the best hypothesis of CRF is correct. In turn this implies that the reranking model must not rerank 75% of times and rerank the other 25% of times, someway contrasting the evidence provided by the baseline model score. In contrast, using our *WRR* strategy, we can tune the reranking model to maximize reranking effect and recover from reranking errors applying *WRR*. As shown in tables 3 and 4, we consistently improve CRF baseline as well as reranking baseline *CRF+RR*, especially applying both SIM and WRR (*CRF+*RR_{WRR}). Comparing our results with those reported in (Hahn et al., 2010), we can see that our model reaches, and even im-

MEDIA Test set Model	OER[%]	correct found/present
CRF	9.5	2359/2657
CRF+RR	9.5	2375/2657
CRF+RR_{SIM}	7.5	2381/2758
CRF+RR_{WRR}	7.5	2444/2758

Table 5: Analysis over 10-best hypotheses for CRF baseline and the reranking models showing the effect of hypotheses selection

MEDIA Text Input **Model Pair**	**DEV** **Attr+Val**	**TEST** **Attr+Val**
CRF vs. CRF+RR	0.2235	0.4075
CRF vs. CRF+RR_{SIM}	0.0299	0.065
CRF vs. CRF+RR_{WRR}	0.0044	1.9998E-4
CRF+RR vs. CRF+RR_{SIM}	0.002	5.9994E-4
CRF+RR vs. CRF+RR_{WRR}	4.9995E-4	9.999E-5
CRF+RR_{SIM} vs. CRF+RR_{WRR}	0.1355	0.0031

Table 6: Significance tests on results of models described in this work. The significance test is based on computationally-intensive randomizations as described in (Yeh and Church, 2000).

proves in some cases, state-of-the-art performance. This is particularly meaningful since best results reported in (Hahn et al., 2010) are obtained combining 6 different SLU models.

In table 5 we report some statistics to show the effect of SIM on the 10-best hypotheses list. It is particularly interesting to see that when hypotheses selection is applied, oracle error rate (OER) drops of 2% points from an already accurate OER of 9.5%. This is reflected also by the number of oracles present in the 10-best list without applying and applying SIM. We pass from 2657 without SIM to 2758 applying our hypotheses selection metric.

Finally, in table 6 we report statistical significance tests over the models described in this work. We used the significance test described in (Yeh and Church, 2000), it is based on computationally-intensive randomizations of data and tests the null hypothesis, i.e. the lower the score, the higher the statistical significance of results difference. Scores in table 5 reflect results given in terms of CER. We can see that when the difference between results is small, this is not statistically significant, when the score is above 0.05, the difference between the two corresponding models is not significant. We can thus conclude that the reranking model we propose, using hypotheses selection and reranking errors recover, significantly improves baseline CRF model and "traditional" reranking models.

6 Discussion

Although the new ideas proposed in this paper are effective and interesting, an important issue is their applicability to other tasks and domains. In this respect, it is sufficient to note that our ideas comes from the multi-stage nature of the task and of the proposed reranking framework. SLU is performed in two intertwined steps, since attribute values are extracted from syntactic chunks annotated with concept in the first step. This allows to use the model for the second step to validate the output of the first step, and vice versa, which is the principle of our hypotheses selection metric. There are many other tasks, in NLP and in other domains, that can be modeled with multiple steps and thus the same idea of *"validation"* of the output of one step with the other's model output can be applied. An example is syntactic parsing, where in most cases parsing is performed upon POS tagging output.

7 Conclusions

In this paper we propose two improvements for reranking models to be integrated in a reranking framework for Spoken Language Understanding. The reranking model is based on a CRF baseline model and Support Vector Machines with the Partial Tree Kernel for the reraning model. The two improvements we propose are: i) hypotheses selection criteria, used before applying reranking to select better hypotheses amongst those generated by CRF. ii) a strategy to recover from reranking errors called Wrong Rerank Rejection.

We presented a full set of comparative results showing the viability of our approach. We can reach performances of state-of-the-art models, improving them in some cases, especially on automatic transcriptions coming from ASR (speech input).

In particular, the effectiveness of hypotheses selection is shown reporting the improvement of the Oracle Error Rate on the 10-best hypotheses list.

Acknowledgments

This work has been funded by OSEO under the Quaero program.

References

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4*, pages 142–147, Morristown, NJ, USA. Association for Computational Linguistics.

Ellen M. Voorhees. 2001. The trec question answering track. *Nat. Lang. Eng.*, 7:361–378, December.

X. Carreras and Lluis Marquez. 2005. Introduction to the conll-2005 shared task: Semantic role labeling.

R. De Mori, F. Bechet, D. Hakkani-Tur, M. McTear, G. Riccardi, and G. Tur. 2008. Spoken language understanding: A survey. *IEEE Signal Processing Magazine*, 25:50–58.

Sylvain Galliano, Guillaume Gravier, and Maura Chaubard. 2009. The ester 2 evaluation campaign for the rich transcription of french radio broadcasts. In *Proceedings of the International Conference of the Speech Communication Assosiation (Interspeech)*, Brighton, U.K.

E. Charniak and M. Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of the Conference of the Association for Computational Linguistics (ACL)*, page 363370, Ann Arbor, MI.

Michael Collins and Terry Koo. 2005. Discriminative reranking for natural language parsing. *Computational Linguistic (CL)*, 31(1):25–70.

J. Lafferty, A. McCallum, and F. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*, pages 282–289, Williamstown, MA, USA, June.

Brigitte Krenn and Christer Samuelsson. 1997. The linguist's guide to statistics - don't panic.

Thomas Lavergne, Olivier Cappé, and François Yvon. 2010. Practical very large scale CRFs. In *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 504–513. Association for Computational Linguistics, July.

Stefan Hahn, Patrick Lehnen, Georg Heigold, and Hermann Ney. 2009. Optimizing crfs for slu tasks in various languages using modified training criteria. In *Proceedings of the International Conference of the Speech Communication Assosiation (Interspeech)*, Brighton, U.K.

Stefan Riezler and Alexander Vasserman. 2010. Incremental feature selection and l1 regularization for relaxed maximum-entropy modeling.

Hui Zou and Trevor Hastie. 2005. Regularization and variable selection via the Elastic Net. *Journal of the Royal Statistical Society B*, 67:301–320.

Andrew McCallum. 2003. Efficiently inducing features of conditional random fields. In *19th Conference on Uncertainty in Artificial Intelligence*.

Lawrence R. Rabiner. 1989. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Mark Johnson. 1998. Pcfg models of linguistic tree representations. *Computational Linguistics*, 24:613–632.

Olivier Galibert, Ludovic Quintard, Sophie Rosset, Pierre Zweigenbaum, Claire Ndellec, Sophie Aubin, Laurent Gillard, Jean-Pierre Raysz, Delphine Pois, Xavier Tannier, Louise Delger, and Dominique Laurent. 2010. Named and specific entity detection in varied data: The quro named entity baseline evaluation. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Olivier Galibert. 2009. *Approches et méthodologies pour la réponse automatique à des questions adaptées un cadre interactif en domaine ouvert*. Ph.D. thesis, Universit Paris Sud, Orsay.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. 2004. The Automatic Content Extraction (ACE) Program–Tasks, Data, and Evaluation. *Proceedings of LREC 2004*, pages 837–840.

Hélène Bonneau-Maynard, Christelle Ayache, F. Bechet, A Denis, A Kuhn, Fabrice Lefèvre, D. Mostefa, M. Qugnard, S. Rosset, and J. Servan, S. Vilaneau. 2006. Results of the french evalda-media evaluation campaign for literal understanding. In *LREC*, pages 2054–2059, Genoa, Italy, May.

Christian Raymond, Frdric Bchet, Renato De Mori, and Graldine Damnati. 2006. On the use of finite state transducers for semantic interpretation. *Speech Communication*, 48(3-4):288–304, March-April.

Christian Raymond and Giuseppe Riccardi. 2007. Generative and discriminative algorithms for spoken language understanding. In *Proceedings of the International Conference of the Speech Communication Assosiation (Interspeech)*, pages 1605–1608, Antwerp, Belgium, August.

Vladimir N. Vapnik. 1998. *Statistical Learning Theory*. John Wiley and Sons.

T. J. Sejnowski and C. S. Rosenberg. 1987. Parallel networks that learn to pronounce English text. *Complex Systems*, 1:145–168.

Stefan Hahn, Marco Dinarelli, Christian Raymond, Fabrice Lefèvre, Patrick Lehen, Renato De Mori, Alessandro Moschitti, Hermann Ney, and Giuseppe Riccardi. 2010. Comparing stochastic approaches to spoken language understanding in multiple languages. *IEEE Transactions on Audio, Speech and Language Processing (TASLP)*, 99.

Marco Dinarelli, Alessandro Moschitti, and Giuseppe Riccardi. 2009b. Re-ranking models based on small training data for spoken language understanding. In *Conference of Empirical Methods for Natural Language Processing*, pages 11–18, Singapore, August.

Marco Dinarelli, Alessandro Moschitti, and Giuseppe Riccardi. 2010. Hypotheses Selection for Reranking Semantic Annotation. In *IEEE Workshop of Spoken Language Technology (SLT)*, Berkeley, USA.

Alessandro Moschitti. 2006. Efficient Convolution Kernels for Dependency and Constituent Syntactic Trees. In *Proceedings of ECML 2006*, pages 318–329, Berlin, Germany.

M. Collins and N. Duffy. 2002. New Ranking Algorithms for Parsing and Tagging: Kernels over Discrete structures, and the voted perceptron. In *Proceedings of the Association for Computational Linguistics*, pages 263–270.

Libin Shen, Anoop Sarkar, and Aravind K. Joshi. 2003. Using LTAG Based Features in Parse Reranking. In *Proceedings of EMNLP'06*.

Herbrich, Ralf and Graepel, Thore and Obermayer, Klaus. 2000. Large Margin Rank Boundaries for Ordinal Regression. In *Advances in Large Margin Classifiers*.

Libin Shen, and Aravind K. Joshi. 2003. An SVM Based Voting Algorithm with Application to Parse Reranking. In *Proceedings of CoNLL 2003*.

Libin Shen, Anoop Sarkar, and Franz J. Och. 2004. Discriminative reranking for machine translation. In *HLT-NAACL*, pages 177–184.

Taku Kudo, Jun Suzuki, and Hideki Isozaki. 2005. Boosting-based parse reranking with subtree features. In *Proceedings of ACL'05*.

Stefan Hahn, Patrick Lehnen, and Hermann Ney. 2008a. System combination for spoken language understanding. In *Proceedings of the International Conference of the Speech Communication Assosiation (Interspeech)*, pages 236–239, Brisbane, Australia.

J. G. Fiscus. 1997. A post-processing system to yield reduced word error rates: Recogniser output voting error reduction (ROVER). In *Proceedings 1997 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 347–352, Santa Barbara, CA, December.

John Shawe-Taylor and Nello Cristianini. 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press.

Michael Collins. 2000. Discriminative reranking for natural language parsing. In *ICML*, pages 175–182.

Michael Collins and Nigel Duffy. 2001. Convolution kernels for natural language. In *Advances in Neural Information Processing Systems 14*, pages 625–632. MIT Press.

Rush, Alexander M. and Sontag, David and Collins, Michael and Jaakkola, Tommi. 2010. On dual decomposition and linear programming relaxations for natural language processing. In *Empirical Methods for Natural Language Processing (EMNLP)*. Cambridge, Massachusetts, USA.

Alessandro Moschitti, Daniele Pighin, and Roberto Basili. 2008. Tree kernels for semantic role labeling. *Computational Linguistics*, 34(2):193–224.

Alessandro Moschitti, Silvia Quarteroni, Roberto Basili, and Suresh Manandhar. 2007. Exploiting syntactic and shallow semantic kernels for question/answer classification. In *Proceedings of ACL'07*, Prague, Czech Republic.

Alexander Yeh and Kelmeth Church. 2000. More accurate tests for the statistical significance of result differences.