Exploiting multiple hypotheses for Multilingual Spoken Language Understanding

Marcos Calvo, Fernando García, Lluís-F. Hurtado, Santiago Jiménez, Emilio Sanchis
Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València, València, Spain
{mcalvo,fgarcia,lhurtado,sjimenez,esanchis}@dsic.upv.es

Abstract

In this work, we present an approach for multilingual portability of Spoken Language Understanding systems. The goal of this approach is to avoid the effort of acquiring and labeling new corpora to learn models when changing the language. The work presented in this paper is focused on the learning of a specific translator for the task and the mechanism of transmitting the information among the modules by means of graphs. These graphs represent a set of hypotheses (a language) that is the input to the statistical semantic decoder that provides the meaning of the sentence. Some experiments in a Spanish task evaluated with input French utterances and text are presented. They show the good behavior of the system, mainly when speech input is considered.

1 Introduction

Spoken Language Understanding (SLU) is one of the key modules in many voice-driven human-computer interaction systems. Many successful SLU systems that have been developed in the last few years are based on statistical models automatically learned from semantically labeled corpora (Maynard and Lefèvre, 2001; Segarra et al., 2002; He and Young, 2006; Lefèvre, 2007; De Mori et al., 2008). One of the advantages of statistical models is the capability of representing the variability of lexical realizations of concepts (meanings). On the other hand, they are usually plain models, that is, they can not represent a hierarchical semantic dependency, although there are some works in this area (He and Young, 2003). However, this is not a problem in most Spoken Dialog Systems since the semantic information to be extracted is not very hierarchically structured.

Another important aspect of these models is that they can be learned from corpora. The corpora used for training must be large enough to allow an accurate estimation of the probabilities, and it must represent the lexical and syntactic variability that is used in the language to express the semantics as much as possible. Although there are some approaches based on semi-supervised or unsupervised learning (Tür et al., 2005; Riccardi and Hakkani-Tür, 2005; Ortega et al., 2010), the most common approaches need to have a segmented and labeled training corpus. This is the case of discriminative models (like Conditional Random Fields (Hahn et al., 2010)), and generative models (such as Hidden Markov Models and Stochastic Finite State Automata (Segarra et al., 2002; Hahn et al., 2010)). In the case of supervised learning, it is necessary to define a set of concepts that represent the semantic domain of the task and to associate these concepts to the corresponding sequences of words in the sentences. This is the case of the French MEDIA corpus (Bonneau-Maynard et al., 2005), and the Spanish DIHANA corpus (Benedí et al., 2006). Since the corpus acquisition and labeling require a great manual effort, being able to reuse the corpus generated for a task to easily develop SLU systems for other tasks, or languages, is an important issue.

This work focuses on the problem of SLU portability between languages (García et al., 2012; He et al., 2013; Jabaian et al., 2013). We propose a semi-supervised approach for adapting the system to tackle sentences that are uttered in a new language. In order to learn a domain-specific translation model, a parallel corpus is automatically generated from the training set by using web translators. Due to the fact that the speech recognition and the translation phases can generate many errors, a mechanism to obtain the correct meaning despite these errors is needed. This can be performed by supplying many hypotheses between

Proceedings of the Seventeenth Conference on Computational Natural Language Learning, pages 193–201,
Sofia, Bulgaria, August 8-9 2013.

the different stages, either as a set of n sentences or as a graph that represents not only the original sentences but also an adequate generalization of them. This graph can be obtained from a Grammatical Inference process. We have also developed a specific algorithm to perform the semantic decoding by taking graphs of words as the input and considering statistical semantic models. We have applied these techniques for the DIHANA corpus, which is a task to access the information of train timetables and fares in Spanish by phone. This corpus was originally generated in Spanish, and we have evaluated our system by using input sentences in French.

2 Description of the system

One way of solving the SLU problem is to find the sequence of concepts $\hat{C}$ that best fits the semantics contained in an utterance A. Considering a stochastic modelization, it can be stated as:

$$\hat{C} = \underset{C}{\operatorname{argmax}}\, p(C|A) \tag{1}$$

In the case of Multilingual SLU, the user utters a sentence in a source language s, which is different to the language t of the original data of the SLU task. Thus, either the uttered sentence or the training data (or maybe both) should be translated into a common language in order to be able to apply the semantic decoding process to the input utterance. In our case, we recognize the input utterance by using an Automatic Speech Recognizer (ASR) in the source language, and we then translate the hypotheses provided by the ASR into the target language t by means of a statistical Machine Translation system (see Figure 1). Consequently, by considering both the input sentence W_s uttered by the user and its translation into the target language W_t, Equation (1) can be rewritten as:

$$\hat{C} = \underset{C}{\operatorname{argmax}} \max_{W_s,W_t} p(C, W_s, W_t|A) \tag{2}$$

Equation (2) can be decomposed into several factors, as shown in Equation (3). This is achieved by applying the Bayes' Rule and making some reasonable assumptions about the independence of the variables.

$$\hat{C} = \underset{C}{\operatorname{argmax}} \max_{W_s,W_t} \frac{p(A|W_s) \cdot p(W_s|W_t) \cdot p(W_t|C) \cdot p(C)}{p(A)} \tag{3}$$

To perform this maximization, we propose a decoupled architecture, which sequentially applies all the knowledge sources. One of the most important drawbacks of decoupled architectures is that the errors generated in one stage can not be recovered in following phases. To overcome this problem, we propose an architecture in which the communication between the modules is done by means of structures that provide more than one hypothesis, like n-best and graphs of words. A scheme of this architecture is shown in Figure 1. Its modules are the following:

1. First, the input utterance is processed by an ASR in the source language, providing as its output either the 1-best or a set of n-best transcriptions. We have used a general purpose, freely available web ASR, which means that the ASR has no specific information about the task.

2. These transcriptions are translated into the target language by means of a state-of-the-art Machine Translation system: MOSES (Koehn et al., 2007). The translation models have been trained without using any manually generated data. Instead, a set of freely available web translators was used to translate the training sentences of the corpus from the target language (the original language of the corpus sentences) into the source language (the language of the speaker), thereby building a parallel training corpus. MOSES provides as its output a set of candidate translations (n-best) of the transcriptions supplied by the ASR.

3. A graph of words is built from the n-best provided by the translator. This graph is built through a Grammatical Inference process. This way the graph not only represents the translations, but also a reasonable generalization of them. This makes it possible for the semantic decoder to consider some sentences that were not in the initial set but that are made of pieces of those sentences.

4. This graph of words is processed by a SLU module that is able to tackle graphs. The semantic model for this stage has been learned using only the training data in the target language. As an intermediate result, this process builds a graph of concepts, which is a compact representation of all the possible semantics contained in the language represented by

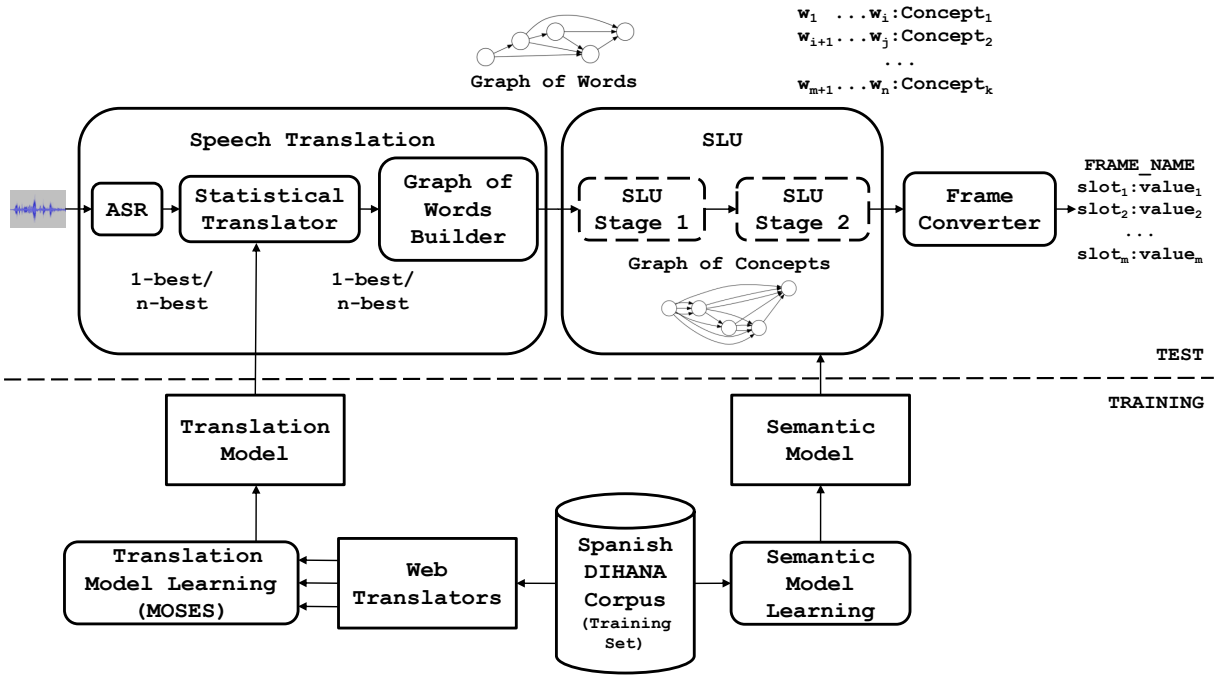

Figure 1: Scheme of the decoupled architecture.

the graph of words. The output of this module is the best sequence of concepts $\hat{C}$ and also its underlying sequence of words $\tilde{W}_t$ in the target language and a segmentation of $\tilde{W}_t$ according to $\hat{C}$.

5. Finally, the segmentation obtained in the previous step is processed in order to convert it into a frame representation. This involves extracting the relevant information from the segmentation and representing it in a canonical way.

Assuming that all the translations W_t that belong to the language represented by the graph of words are a priori equiprobable in the target language, we can rewrite Equation (3) as follows:

$$\hat{C} = \underset{C}{\operatorname{argmax}} \max_{W_s, W_t} \frac{p(A|W_s) \cdot p(W_s) \cdot p(W_t|W_s) \cdot p(W_t|C) \cdot p(C)}{p(A)} \quad (4)$$

The first three modules of the architecture can be viewed as a speech translation process, where the input is an utterance and the output is a set of possible translations of this utterance, represented as a graph of words. Each one of these translations is weighted with the probability $p(W_t|A)$. Considering that

$$p(W_t|A) \approx \max_{W_s} \frac{p(A|W_s) \cdot p(W_s) \cdot p(W_t|W_s)}{p(A)}$$

it stands that Equation (4) can be rewritten as:

$$\hat{C} = \underset{C}{\operatorname{argmax}} \max_{W_t} p(W_t|A) \cdot p(W_t|C) \cdot p(C) \quad (5)$$

The fact that the communication between the different modules is a set of hypotheses makes it possible to apply the different constraints (acoustic, lexical, syntactic, and semantic) in a global way, while the modular architecture allows local pruning taking into account only a subset of the knowledge sources. This way each of the modules contributes to the computation of the global maximization, but it is not completely performed until the end of the process.

3 Learning of the translation model

It has been shown that statistical models achieve good performance in speech translation tasks (Mathias and Byrne, 2006). Also, they have the advantage that they can be adapted to a specific task, as long as a large enough amount of parallel training data is available in order to adequately train the parameters of the Machine Translation system. However, obtaining this task-specific training data by translating the original data by hand is very expensive and time-consuming. A solution to this problem is to use several general-purpose web translators (which are available on

the Internet) to automatically translate the task-specific training sentences into another language. Although these translators can generate many errors, they are an interesting way to obtain several hypotheses for a translation without much effort. However, the use of these translators at testing time is not very convenient due to the fact that the system would depend on the Internet connection and the reaction time of the corresponding web pages. Another drawback is that it is impossible to adapt them to a specific task, which could generate many errors that are important to the task.

The approach that we propose attempts to take advantage of these resources, but for training purposes. In other words, given the training sentences in Spanish, they are translated into a new language (French in this case) by using several web translators. This way we build a parallel corpus where each sentence has different translations associated to it. From this parallel corpus, we train a statistical translator that is specific for the task. It should be noted that by means of this process, the learned translator can represent and modelize the variability generated by the different translators. However, due to the difficulty of the problem, this modelization may not be enough. Therefore we can not guarantee that the best translation obtained by the model is consistent with the meaning of the original sentence. This is why it is convenient to supply more than one hypothesis to the semantic decoding module in order to have the possibility of finding the correct semantic meaning even when some errors were generated in the recognition and translation processes. We think that separately processing the n-best translated sentences (for each input sentence) generated by the translator is not the best solution. In contrast, it would be better to adequately combine segments of different sentences. Thus, we have developed a Grammatical Inference mechanism to build a graph of words from a set of hypotheses as described in the following section.

4 Generating the graphs of words

In this section, the process of obtaining the graphs of words in the target language from multiple translation hypotheses is explained. This process is divided into two steps:

1. The translation hypotheses are aligned using a Multiple Sequence Alignment (MSA) algorithm. The result of the MSA process is an alignment matrix.
2. The aligned sentences, represented by the alignment matrix, are used to obtain a weighted graph of words that will be the input to the graph-based SLU module.

A Multiple Sequence Alignment is a process of sequence alignment that involves more than two sequences. It takes a set of sequences of symbols (in our case, sequences of words) and provides the alignment of the elements of the set that minimizes the number of edit operations (substitutions, insertions, and deletions) among all the symbols of the sequences. In this work, a modification of the ClustalW (Larkin et al., 2007) Multiple Sequence Alignment software has been used.

The result of the MSA process is an alignment matrix. Each row in this matrix represents a different aligned sentence, and each column represents the alignment of each symbol. The total number of columns is usually greater than the length of the longest sequence, since not all the symbols can be aligned. The special symbol '-' is used to represent the positions of non-alignment points in a sentence.

A weighted directed acyclic graph of words is created from the MSA alignment matrix, The graph construction consists of creating as many nodes as columns in the alignment matrix plus one for the final state and as many arcs as cells in the matrix that contain a symbol different to '-'. The arcs with the same source, destination, and symbol are joined, and the weights are obtained by normalizing these counters (Calvo et al., 2012).

Figure 2 shows a real example (extracted from the test set) of the full process of obtaining the graph of words. As the figure shows, the obtained graph of words (where the arcs are labeled with words and weighted with the normalized counters) represents a language which is a generalization of the individual translations of the original utterance. That is, this process is a Grammatical Inference mechanism that represents sentences with characteristics that are similar to those used to build the graph. A full path from the initial node to the final node in the graph may be seen as an alternative translation of the original utterance. For example, the correct translation of the utterance "*el precio del billete del tren de las seis treinta y cinco*" was not among the candidates provided, but it can be recovered using this algorithm.

This graph builder module completes the sequence of modules that perform the speech trans-

source utterance

le prix du billet de train de six heures trente-cinq
(the price of the train ticket for six thirty-five)
(el precio del billete del tren de las seis treinta y cinco)

multiple ASR outputs

le prix du billet train de sezer trente-cinq
le prix du billet train de six vers trente-cinq
le prix du billet train de six onze trente-cinq
le prix du billet train des six heures trente cinq

multiple translations

el precio del billete de tren de sezer treinta y cinco
el precio del billete de tren alrededor de las seis treinta y cinco
el precio del billete del tren de las seis once y treinta y cinco
el precio del billete de tren de las seis treinta de las cinco de la tarde

alignment matrix

el	precio	del	billete	de	tren	-	de	sezer	treinta	-	-	-	y	cinco	-	-	-
el	precio	del	billete	de	tren	alrededor	de	las	seis	treinta	-	-	y	cinco	-	-	-
el	precio	del	billete	del	tren	-	de	las	seis	once	y	treinta	y	cinco	-	-	-
el	precio	del	billete	de	tren	-	de	las	seis	treinta	-	de	las	cinco	de	la	tarde

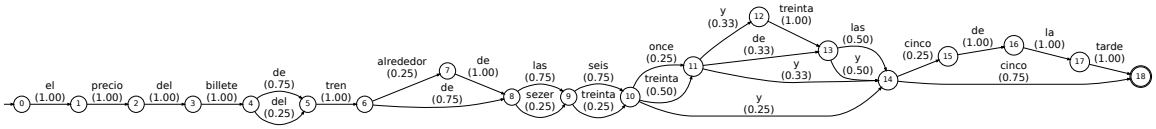

Figure 2: Steps for obtaining the graph of words from the original utterance *le prix du billet de train de six heures trente-cinq*, (*the price of the train ticket for six thirty-five*).

lation process. This process takes as its input an utterance and outputs a weighted graph of words, which represents the probability distribution $p(W_t|A)$. In other words, each full path in the graph of words (from the initial to the ending node) is a candidate translation of the input utterance, and is weighted with the probability of the translation given the utterance.

5 Performing the semantic decoding

Our semantic decoding process is based on the idea of finding segments of words contained in the graph of words that are relevant to each of the concepts of the task. In order to compactly represent this set of segments and the concepts they are relevant to, a second graph is created, which we have called a graph of concepts. This graph has the same set of nodes as the graph of words, but each arc represents that there is a path in the graph of words between the initial and ending node of the arc, which induces a sequence of words that is relevant to some of the concepts of the task. Thus, each of these arcs is labeled with the corresponding sequence of words and the concept they represent. To assign a proper weight to the arcs, both the weights represented in the graph of words and the semantic model are considered. As the set of nodes is the same as in the graph of words, we will say that for every two nodes i, j, it stands that $i < j$ if i comes before j in the topological order of the nodes in the graph of words (there is a topological order because the graph of words is directed and acyclic).

As stated in Equation (5), one of the important factors in this approach is the probability of the sequence of words in the target language given the sequence of concepts $p(W_t|C)$. This probability can be decomposed as the product of the probabilities assigned by each concept of the sequence of concepts C to the segment of words that is attached to it; that is, $\prod_{\forall c_k \in C} p(W_{t_k}|c_k)$, where W_{t_k} is the sequence of words corresponding to the concept c_k in the segmentation. To compute these probabilities, our semantic model includes a set of bigram Language Models (LMs), one for each concept in the task, which provide the probability of any sequence of words given the concept. To train these LMs, the training sentences of the corpus in the target language must be segmented and labeled in terms of the concepts of the task. The consequence of defining the semantic model this way is that every arc from node i to node j in the graph of concepts represents the probability $p(W_t^{i,j}|A) \cdot p(W_t^{i,j}|c)$, where $W_t^{i,j}$ and c are the sequence of words and the concept attached to the arc, respectively. Furthermore, each full path (from the initial to the ending node) in the graph of concepts represents the probability

$p(W_t|A) \cdot p(W_t|C)$, for the sequence of concepts C and the sentence W_t induced by the path.

The set of arcs of the graph of concepts can be built by means of a Dynamic Programming (DP) algorithm that finds the sequence of words that maximizes the combined probability stated above, for each pair of nodes i, j and each concept c. Only the arc that represents the sequence of words of maximum probability is needed because this information will afterwards be combined with the probability of the sequence of concepts to find the path of maximum probability (see Equation (5)), and if there are many arcs between nodes i and j corresponding to the concept c only the one with maximum probability will be considered. This allows us to prune the arcs of the graph of concepts without any loss of information. For the DP algorithm, we will consider a representation of the LM corresponding to each concept as a Stochastic Finite State Automaton (SFSA). Then, in the DP process, for each concept c we will obtain the best path from node i to node j in the graph of words such that its underlying sequence of words arrives to the state q_c in the SFSA LM_c (the LM of the concept c). This can be achieved by means of the following algorithm:

$$M(i,j,q_c) =$$

$$\begin{cases} 1 & \text{if } i = j \wedge q_c \text{ is the initial state of } LM_c \\ 0 & \text{if } i = j \wedge q_c \text{ is not the initial state of } LM_c \\ 0 & \text{if } j < i \\ \max\limits_{\substack{\forall a \in E_{GW} : \text{dest}(a) = j \\ \forall (q'_c, \text{wd}(a), q_c) \in LM_c}} M(i, \text{src}(a), q'_c) \cdot p(q'_c, \text{wd}(a), q_c) \cdot \text{wt}(a) & \\ & \text{otherwise} \end{cases} \quad (6)$$

where dest(a) stands for the destination node of the arc a in the graph of words, src(a) refers to its source node, and wd(a) and wt(a) refer to the word and the weight attached to the arc, respectively. Also, $(q'_c, \text{wd}(a), q_c)$ represents a transition from the state q'_c to the state q_c labeled with wd(a) in the SFSA that represents LM_c.

It is worth noting that this process must be performed for each concept in the task. Also, it is important for the algorithm to keep track of the words that constitute the paths that maximize the expression for each cell. When this matrix has been filled for a specific concept c, the cell that maximizes $M(i,j,q_c)$ for each pair i and j becomes an arc in the graph of concepts between nodes i and j. This arc is labeled with the sequence underlying the winning path and the concept c and is weighted with the score (probability) contained in $M(i,j,q_c)$.

This process shapes the first stage of the SLU process, which provides the graph of concepts as a result. Then, this graph of concepts is processed by a second stage. This second stage finds the path in the graph that maximizes the combination of its probability and the probability that a LM of bigrams of concepts gives to the sequence of concepts underlying the path. The LM of bigrams of concepts is also part of the semantic model, and to train it we take advantage of the segmentation and labeling in term of concepts provided by the training corpus. Finding the best path this way completely fulfills what is stated in Equation (5). Also, this best path in the graph of concepts provides the best sequence of concepts $\hat{C}$, the underlying sequence of words $\tilde{W}_t$, and a segmentation of $\tilde{W}_t$ according to $\hat{C}$.

6 The DIHANA task and the semantic representation

The DIHANA task consists of a telephone-based information service for trains in Spanish. A set of 900 dialogs was acquired by using the Wizard of Oz technique. The number of user turns was 6,280 and the vocabulary was 823. As in many other dialog systems (Minker, 1999), the semantic representation chosen for the task is based on a frame representation. Therefore, the final output of the understanding process is one or more frames with their corresponding attributes.

Even though the frame representation is the output of the system, we propose an intermediate semantic labeling that consists of assigning concepts to segments of the sentence in a sequential way. This is the output provided by the graph-based SLU module.

In order to represent the meaning of the utterances in terms of this intermediate semantic language, a set of 31 concepts was defined. Some of them are: *query*, *affirmation*, *origin_city*, and *courtesy*.

Each concept represents the meaning of words (or sequences of words) in the sentences. For example, the semantic unit $query$ can be associated to "can you tell me", "please tell me", "what is", etc. This way, each sentence (sequence of words) has a semantic sentence (sequence of concepts) associated to it, and there is an inherent segmentation. The advantage of this kind of representation is that statistical models of the lexical realization of concepts and the n-gram probabilities of the se-

Sentence	*hola buenos días quería saber los horarios de trenes para ir a Madrid* *(hello good morning I'd like to know the train timetables to go to Madrid)*
Semantic segments	*hola buenos días* : courtesy *quería saber* : query *los horarios de trenes para ir* : <time> *a Madrid* : destination_city
Frame	(TIME?) DEST_CITY : Madrid

Table 1: Example of the outputs of the SLU and Frame Converter modules.

quences of semantic units can be learned.

Finally, a set of rules are used to transduce this intermediate representation into a frame. Since the intermediate language is close to the frame representation, only a small set of rules are required to build the frame. This phase consists of the following: the deletion of irrelevant segments (such as courtesies), the reordering of the relevant concepts and attributes that appeared in the segmentation following an order which has been defined a priori, the automatic instantiation of certain task-dependent values, etc.

Table 1 shows an example of the semantic representation in terms of the intermediate semantic segmentation provided by the SLU module and the final frame representation.

7 Experiments and results

To evaluate this architecture, we performed a set of experiments with the DIHANA corpus. The user turns of the corpus were split into a set of 4889 turns for training and 1227 turns for test. To train the translation models, the training set was automatically translated from Spanish into French by four freely available web translators (Apertium, Bing, Google, Lucy), which provided us a parallel training corpus. The semantic model was learned from the segmentation and labeling provided in the DIHANA corpus for the training sentences in Spanish. All the Language Models in the semantic model were bigram models trained using Witten-Bell smoothing.

For evaluation purposes, all the test set was manually translated into French, and 500 turns were uttered by four native French speakers. Thus, we have carried out experiments both considering as the input to our system the correct sentences in French (which is the same than assuming a *perfect ASR*) and the utterances. To recognize the utterances the Google ASR was used, which for this test set provides a Word Error Rate of 21.9% considering only the 1-best recognized sentence.

For this experimentation we have considered three kinds of ASR outputs, namely, a *Perfect ASR* (text input), the 1-best output, and finally the n-best hypotheses (with n ranging from 1 to 20).

Also, we have configured the system in two different ways:

- Configuration 1: The output of the statistical translation system are the n-best translations for the input. Note that these n-best could contain repeated translations, which may lead to the reinforcement of some paths in the graphs of words.
- Configuration 2: The output of the statistical translation system is the set formed by the best n different (unique) translations that it can provide for the given input.

When the output of the ASR are n-best, we have only considered the Configuration 1.

We have evaluated each experiment using two measures: the Concept Error Rate (CER), which corresponds to errors in the output of the SLU module, and the Frame-Slot Error Rate (FSER), which corresponds to errors in the slots of the frames in the final output of the system.

Figures 3, 4, and 5 show the results obtained for each of the ASR outputs and configurations considered. The horizontal axis represents the number of hypotheses provided by the statistical translator.

As expected, in all the cases the FSER is lower than the CER, as some errors at level of the concept sequence are not relevant for the frame conversion (for example, courtesies). In the case of text input (Fig. 3), the best results are achieved when just one or two hypotheses are provided by the translator. This is because the translation model has also been learned using correct sentences, which makes the translation system more robust for this kind of input. However, when considering speech as input (Figs. 4 and 5), the generalization provided by the graphs obtained using a relatively large set of n-best translations leads to

a better behavior. This is due to the fact that the errors introduced by the recognition of the speech input increases the errors in the translation stage. Thus, working with different alternatives makes it possible to recover some of the errors. Table 2 shows the results obtained when optimizing the FSER, and the number of hypotheses n used to build the graphs that provide the best results.

Figures 3 and 4 also show that the parameters that optimize FSER and CER may not be the same. This behavior is due to the different nature of both measures. While CER is defined in terms of the sequence of concepts extracted by the SLU module, FSER only takes into account those segments that have relevant information.

It can be seen in Figures 3 and 4 that, for Configuration 2, when n takes the value 18, both error measures descend. However, after this, the errors continue with their ascending tendency. The reason for this is that with these parameters, the translations provided by the translator generate a graph of words that allows the semantic model to better recover the semantics of the sentence. However, this effect is spurious, as for higher values of n the error measures present higher values.

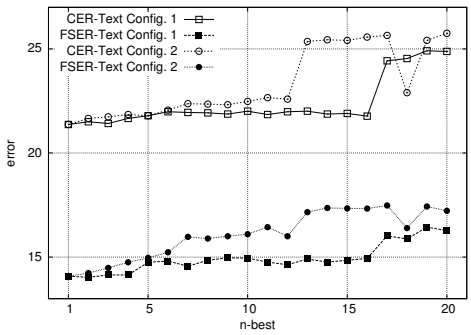

Figure 3: Results obtained with the text input.

ASR output	Config.	CER	FSER	n
Text input	Config. 1	21.50	14.03	2
	Config. 2	21.37	14.08	1
1-best	Config. 1	24.27	19.11	3
	Config. 2	24.13	19.28	3
n-best	Config. 1	22.40	19.63	7

Table 2: Results obtained optimizing the FSER.

8 Conclusions

We have presented an approach for developing multilingual SLU systems without any manual effort in the adaptation of the models. It has been shown that the use of graphs of words, as a mechanism of generalization and transmission of hypotheses, is a good approach to recover from errors generated in the different phases of the system. As future work it may be interesting to explore other Grammatical Inference techniques to combine the n-best hypotheses generated by both the ASR and the translator. It would also be interesting to study the behavior of this approach with other languages that have greater differences than Spanish and French, for example non-Latin languages like English and German.

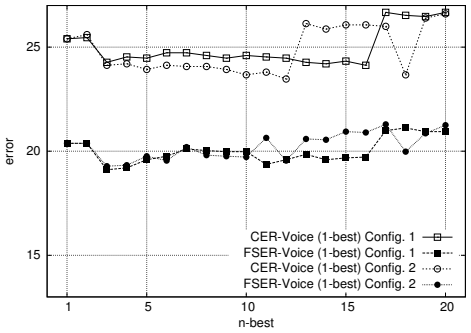

Figure 4: Results obtained with the voice input, taking the 1-best from the ASR and the n-best from MOSES.

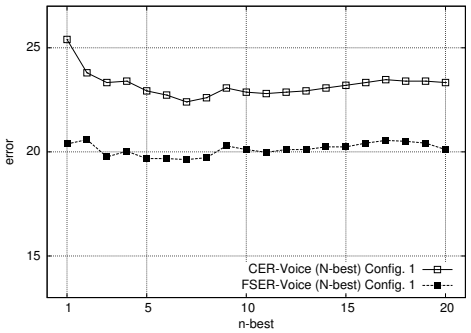

Figure 5: Results obtained with the voice input, taking the n-best from the ASR and the corresponding 1-best from MOSES.

Acknowledgements

This work is partially supported by the Spanish MICINN under contract TIN2011-28169-C05-01, and under FPU Grant AP2010-4193.

References

José-Miguel Benedí, Eduardo Lleida, Amparo Varona, María-José Castro, Isabel Galiano, Raquel Justo, Iñigo López de Letona, and Antonio Miguel. 2006. Design and acquisition of a telephone spontaneous speech dialogue corpus in Spanish: DIHANA. In *Proceedings of LREC 2006*, pages 1636–1639, Genoa (Italy).

H. Bonneau-Maynard, Sophie Rosset, C. Ayache, A. Kuhn, and Djamel Mostefa. 2005. Semantic annotation of the French MEDIA dialog corpus. In *Proc. of InterSpeech 2005*, pages 3457–3460, Portugal.

Marcos Calvo, Lluís-F Hurtado, Fernando García, and Emilio Sanchis. 2012. A Multilingual SLU System Based on Semantic Decoding of Graphs of Words. In *Advances in Speech and Language Technologies for Iberian Languages*, pages 158–167. Springer.

R. De Mori, F. Bechet, D. Hakkani-Tür, M. McTear, G. Riccardi, and G. Tür. 2008. Spoken language understanding: A survey. *IEEE Signal Processing magazine*, 25(3):50–58.

F. García, L.-F. Hurtado, E. Segarra, E. Sanchis, and G. Riccardi. 2012. Combining multiple translation systems for spoken language understanding portability. In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 194–198. IEEE.

S. Hahn, M. Dinarelli, C. Raymond, F. Lefèvre, P. Lehnen, R. De Mori, A. Moschitti, H. Ney, and G. Riccardi. 2010. Comparing stochastic approaches to spoken language understanding in multiple languages. *IEEE Transactions on Audio, Speech, and Language Processing*, 6(99):1569–1583.

Yulan He and S. Young. 2003. Hidden vector state model for hierarchical semantic parsing. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, volume 1, pages 268–271.

Yulan He and Steve Young. 2006. Spoken language understanding using the hidden vector state model. *Speech Communication*, 48:262–275.

Xiaodong He, Li Deng, Dilek Hakkani-Tur, and Gokhan Tur. 2013. Multi-style adaptive training for robust cross-lingual spoken language understanding. In *Proc. ICASSP*.

B. Jabaian, L. Besacier, and F. Lefèvre. 2013. Comparison and combination of lightly supervised approaches for language portability of a spoken language understanding system. *Audio, Speech, and Language Processing, IEEE Transactions on*, 21(3):636–648.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, Moran C., R. Zens, C. Dyer, Bojar O., A. Constantin, and E. Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of Association for Computational Linguistics (ACL'07)*, pages 177–180.

M. A. Larkin, G. Blackshields, N. P. Brown, R. Chenna, P. A. McGettigan, H. McWilliam, F. Valentin, I. M. Wallace, A. Wilm, R. Lopez, J. D. Thompson, T. J. Gibson, and D. G. Higgins. 2007. ClustalW and ClustalX version 2.0. *Bioinformatics*, 23(21):2947–2948.

F. Lefèvre. 2007. Dynamic bayesian networks and discriminative classifiers for multi-stage semantic interpretation. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'07)*, volume 4, pages 13–16. IEEE.

Lambert Mathias and William Byrne. 2006. Statistical phrase-based speech translation. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'06)*, volume 1, pages 561–564. IEEE.

H. Bonneau Maynard and F. Lefèvre. 2001. Investigating Stochastic Speech Understanding. In *Proc. of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU'01)*.

W. Minker. 1999. Stocastically-based semantic analysis. In *Kluwer Academic Publishers*, Boston, USA.

Lucía Ortega, Isabel Galiano, Lluís-F. Hurtado, Emilio Sanchis, and Encarna Segarra. 2010. A statistical segment-based approach for spoken language understanding. In *Proc. of InterSpeech 2010*, pages 1836–1839, Makuhari, Chiba, Japan.

G. Riccardi and D. Hakkani-Tür. 2005. Active learning: theory and applications to automatic speech recognition. *IEEE Transactions on Speech and Audio Processing*, 13(4):504 – 511.

E. Segarra, E. Sanchis, M. Galiano, F. García, and L. Hurtado. 2002. Extracting Semantic Information Through Automatic Learning Techniques. *IJPRAI*, 16(3):301–307.

Gokhan Tür, Dilek Hakkani-Tür, and Robert E. Schapire. 2005. Combining active and semi-supervised learning for spoken language understanding. In *Speech Communication*, volume 45, pages 171–186.