Calvo Lance, M.; García Granada, F.; Hurtado Oliver, LF.; Jiménez Serrano, S.; Sanchís Arnal, E. (2013). Exploiting multiple ASR outputs for a spoken language understanding task. En Speech and Computer. Springer Verlag (Germany). 8113:138-145. doi:10.1007/978-3-319-01931-4_19.

Exploiting Multiple ASR Outputs for a Spoken Language Understanding Task

Marcos Calvo, Fernando García, Lluís-F. Hurtado,
Santiago Jiménez, and 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 paper, we present an approach to Spoken Language Understanding, where the input to the semantic decoding process is a composition of multiple hypotheses provided by the Automatic Speech Recognition module. This way, the semantic constraints can be applied not only to a unique hypothesis, but also to other hypotheses that could represent a better recognition of the utterance. To do this, we have developed an algorithm to combine multiple sentences into a weighted graph of words, which is the input to the semantic decoding process. It has also been necessary to develop a specific algorithm to process these graphs of words according to the statistical models that represent the semantics of the task. This approach has been evaluated in a SLU task in Spanish. Results, considering different configurations of ASR outputs, show the better behavior of the system when a combination of hypotheses is considered.

Keywords: Combination of multiple outputs, graph of words, graph of concepts, Spoken Language Understanding

1 Introduction

Speech-driven human-computer interaction systems are becoming day after day more important in our lives. In many of these systems, an Automatic Speech Recognizer (ASR) is used as a front-end when the user speaks. However, an important drawback of this approach is that the errors that are generated in this first stage are impossible to be recovered afterwards, as the corresponding information is lost. This is the case of Spoken Language Understanding (SLU) systems [1], where the ASR output is processed by a semantic decoder, in order to obtain the meaning of the utterance. One possible solution for this problem is to integrate in a whole model all the knowledge sources that take part in the process (acoustic, lexical, syntactic and semantic) and apply all the linguistic constraints at once. Unfortunately, this solution generates an excessively large search space, increasing this way the difficulty of the task. Also, in this unified model, the different individual models should be properly weighted during its combination, in order to improve the performance of the system. The computation of these weights makes the application of the unified model even harder. For this reason, a more realistic option is to use a modular sequential architecture, in which the information that

is transmitted from one module to the following one is a set of hypotheses, instead of just one hypothesis.

It is possible that the ASR provides a set of hypotheses to the following modules by means of a variety of mechanisms. One of these mechanisms is that the output of the ASR is a word lattice, which can be weighted with acoustic or language model probabilities, or even with some kind of confidence measure. Another option is to combine multiple sentences provided either by a single ASR (n-best list) or by several ASRs working in parallel, in order to exploit the benefits of each of them. For this approach, there are also several ways to carry out this combination. One of them is to use a voting algorithm, like ROVER [2], to obtain a new output that is made of segments corresponding to the original sentences. Another option is to build a graph of words, which can represent as well a reasonable generalization of the input sentences, if a Grammatical Inference algorithm is used. For this work, we have developed a Grammatical Inference algorithm based on the ClustalW [3] Multiple Sequence Alignment (MSA) algorithm. The ClustalW MSA algorithm was originally used for aligning biosequences, but has also been successfully used in other fields like Machine Translation [4, 5].

Once the graphs of words are generated, it is necessary to develop a semantic decoder that is able to process these structures. This semantic decoder can be based on a statistical modelization of the semantics of the task [6]. Nevertheless, there is not many work done in SLU using graphs of words as input [7], as many of the SLU models assume that the input is a single sentence which is completely known when applying the semantic model.

Our goal in this work is to present an approach to SLU that takes advantage of multiple hypotheses provided by the ASR module, in any of the forms mentioned above, and exploit them by using graphs of words as the input to the semantic decoding module. For the SLU module, we have developed a specific Dynamic Programming (DP) algorithm for semantic decoding, which combines the weighted input graph with a set of Stochastic Finite State Automata (SFSA) that modelize the semantics of the task. To evaluate this approach, we have performed a set of experiments with the DIHANA task [8]. This is a SLU task in Spanish designed for being integrated in a telephonic Spoken Dialog System where the goal of the user is to request information about train fares and timetables. The statistical semantic model was automatically learned from the training set of the DIHANA corpus, which is segmented and labeled in terms of concepts.

2 Architecture of the SLU system

In this work, we have addressed the problem of exploiting the combination of multiple ASR outputs for SLU by means of a decoupled modular architecture (Figure 1), which is composed of the following modules:

1. A first module dedicated to Automatic Speech Recognition. We will consider three kinds of different ASR outputs: a word lattice weighted with acoustic probabilities, a n-best list provided by a single ASR, and a set of 1-best decodifications provided by several ASR working in parallel.
2. The second module is based on the idea of Grammatical Inference of generating a language that generalizes a set of positive samples provided as its input. Thus, this

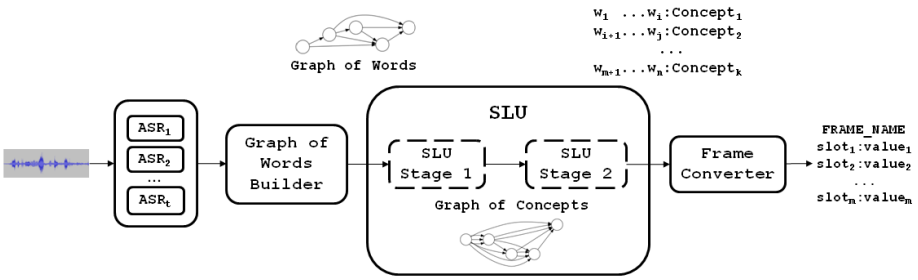

Fig. 1. Scheme of the architecture of our system.

module takes a set of sentences and outputs a graph that represents a generalization of the language represented by the individual sentences. The input sentences are either the n-best decodifications provided by an ASR, or the set of 1-best provided by several ASRs. As a lattice is itself a graph, if the output of the ASR is a word lattice this step can be skipped.

3. Then, the semantic decoding is carried out by means of a SLU module that is able to deal with graphs of words. For this system, we have developed a semantic decoding methodology that works in two stages. The first one takes as input a graph of words and a set of SFSA that modelize the lexical structures attached to each concept, and outputs a graph of concepts, which represents matchings between sequences of words and concepts. This graph of concepts is processed in a second stage, which also takes as input another SFSA that represents how the concepts are concatenated. The output of this module is the best sequence of concepts, as well as the underlying sequence of words and its segmentation in terms of the concepts.
4. Finally, the relevant semantic information is extracted and converted into a frame representation.

3 The graph of words builder module

The goal of this module is to generate a weighted graph of words from the outputs supplied by one or more ASRs. This graph represents a set of recognition alternatives that are a generalization of the individual transcriptions of the utterance. This way, the following modules can search among these alternatives for the most accurate sentence according to their specific constraints. It is also convenient that the words that appear in the graphs have associated some kind of weight, which is usually the normalized acoustic or language model probability, or a confidence measure. In our case, we have considered two ways for obtaining these graphs:

1. The output of the ASR is the lattice generated by the Viterbi algorithm, and the weights are the normalized acoustic probabilities associated to the words. In this case, it is not necessary any algorithm to generate the graph of words because it is supplied by the ASR.

2. The output of the ASR module is a set of sentences (i.e., a n-best list or a set of 1-best from different ASRs). In this case, a graph of words is estimated from these alternative hypotheses. One of the advantages of building a graph of words is that it can represent an extra-language of structures similar to the original sentences, which is a Grammatical Inference process.

This way, we have an homogeneous mechanism of communication between modules, and the same algorithms can tackle with the lattices supplied in the case 1, and with the graphs generated in the case 2.

The algorithm proposed in this work for generating the graphs of words consists of two phases. In the first phase a multi-sentence alignment is performed. To do this, an adaptation of the ClustalW [3] MSA is used. This algorithm finds the best multiple alignment that minimizes the total number of edit errors (substitution, insertion and deletion of words) among all the sentences. Then, the second phase consists in finding the synchronization points in the alignment, generating nodes in these points, and creating the arcs (labeled with the words and weighted with the normalized counters) that represent alternative paths [6]. Figure 2 shows an example of a graph generated using this method. As it is shown, this graph represents not only the input sentences, but also an extra-language of sentences of similar characteristics. For example, the correct sentence *me puede decir horarios de trenes a Alicante* (could you tell me train timetables to Alicante) was not among the candidates provided, but can be recovered using this mechanism.

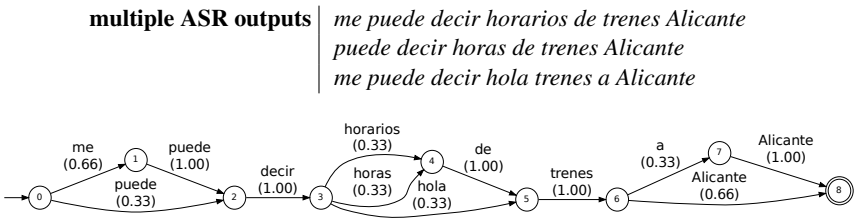

Fig. 2. Graph of words built from multiple ASR outputs.

4 The SLU module

We propose an understanding system that is able to deal with graph of words, where each arc is labeled with a word and a probability. This SLU system works in two stages. The first stage converts a graph of words into a graph of concepts using the information represented in a set of Stochastic Finite State Automata. Each of these automata modelizes a bigram Language Model that represents the lexical structures associated with a concept, as well as their probabilities. The second stage searches for the best path in the graph of concepts, according to a bigram Language Model of sequences of concepts

represented as another SFSA. The final result is the best sequence of concepts, as well as the underlying sequence of words and its segmentation in terms of the concepts. Both stages are based on Dynamic Programming algorithms.

The graph of concepts obtained as the output of the first stage (see Figure 3) is the result of finding, for each concept c and each pair of nodes i, j, the sequence of words W induced by a path from i to j in the graph of words that maximizes the product of the probability of the path and the probability of W according to the SFSA of c. Consequently, each arc in the graph of concepts is labeled with a sequence of words and a concept associated to it, and is weighted with the product of the probabilities provided by both the graph and the SFSA. Thus, a graph of concepts is a compact representation of the semantics of the segments of words contained in the graph of words. Once the graph of concepts is built, it is easy to find the path of maximum probability that goes from the start to the ending node, taking into account a Language Model of sequences of concepts.

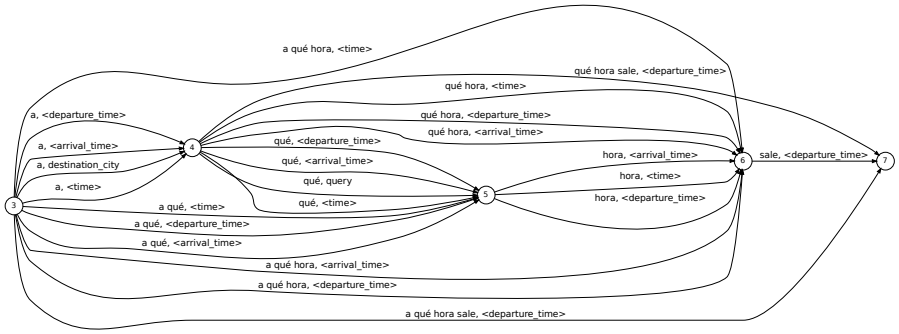

Fig. 3. Graph of concepts corresponding to the segment "a qué hora sale" (*which is the departure time*), built from a graph of words.

5 The frame converter module

This last module converts the segmentation provided by the SLU module into a frame representation of the semantics. The operations performed in this module are: 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 an ideal situation where the input utterance is correctly recognized, then the SLU module provides the correct semantic segmentation, and finally the frame converter outputs the correct frame representation.

Table 1. Example of the outputs of the SLU (semantic segmentation) and Frame Converter modules.

Input utterance	*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

6 Experiments and results

To evaluate the proposed architecture, we have performed a set of experiments using the DIHANA task [8]. The goal of this task is to use a Spoken Dialog System by phone to request information about railway timetables and fares. This task has a corpus of 900 dialogs of spontaneous telephonic speech in Spanish, which was acquired by 225 speakers using the Wizard of Oz technique, amounting to a total of $6,229$ user turns. This set of turns was split into a subset of $4,889$ utterances for training and $1,109$ for test. The orthographic transcriptions of all the user turns are available in this corpus, and are semi-automatically segmented and labeled using a set of 30 concepts.

For this experimentation, we used the HTK, Loquendo and Google ASRs. For HTK, both the Language and Acoustic Models of the ASR were trained with the training set. For Loquendo, only the Language Model was trained this way. No information of the task was provided to the Google ASR. The WERs obtained for the test set considering the 1-best output of each of the ASRs individually are: 17.85 for HTK, 17.90 for Loquendo and 29.45 for Google. The WER obtained for the Google ASR is higher than the rest because it is a general purpose ASR, without any knowledge of the task, while the others have some information about it.

We performed four types of SLU experiments, depending on the way that the ASR hypotheses are supplied:

1. Three experiments, one for each ASR, using the 1-best of each ASR separately. These experiments constitute the baselines.
2. Using a word lattice generated by HTK.
3. Taking the 3, 5 and 20-best hypotheses provided by the Google ASR, and combining them in a graph of words.
4. Taking the 1-best decodifications provided by the three ASRs, and combining them in a graph of words.

Two measures were used to evaluate each of the configurations:

- The Concept Error Rate (CER), which corresponds to errors in the output of the SLU module.
- The Frame-Slot Error Rate (FSER), which corresponds to errors in the slots of the frames in the final output of the system.

The results obtained are shown in Table 2. It must be noted that FSER is lower than CER because some concepts are not relevant for the final semantic representation. For example errors in the concept *courtesy* are not transmitted to the corresponding frame. Also, the results for the Google ASR are worse because it is a general purpose ASR, without any knowledge of the task.

These results show that all the experiments performed with multiple ASR outputs outperform the corresponding baseline. Also, they confirm the hypothesis that the combination of several sentences in a graph of words by means of a Grammatical Inference algorithm generates new sentences, belonging to the inferred extra-language, that can lead to an improvement of the semantic output.

Table 2. Results obtained using the different compositions of ASR outputs, as well as the individual 1-bests.

Input graphs of words	CER	FSER
HTK 1-best	17.72	13.02
Loquendo 1-best	18.29	11.94
Google 1-best	25.80	23.38
HTK word lattice	14.23	11.19
Google 3-best	20.04	18.65
Google 5-best	18.92	17.74
Google 20-best	18.37	17.27
HTK + Google + Loquendo 1-bests	12.85	8.87

7 Conclusions

In this work, we have presented an approach to SLU that takes advantage of multiple hypotheses generated by the ASR phase. We have developed a Grammatical Inference algorithm to generate a language representing different recognition alternatives of the uttered sentence (graph of words), and also a methodology to analyze this graph of words according to the statistical semantic model. Results, considering a task of an information system about train timetables and fares, show that adequately combining these hypotheses the behavior of the system can be improved. This means that some recognition alternatives generated by the Grammatical Inference algorithm are more adequate for the semantic model, and lead to a best semantic decodification. As future work, we want to perform some experiments with other corpora, and also research other algorithms to generate the graphs of words.

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

References

1. Tür, G., Mori, R.D.: Spoken Language Understanding: Systems for Extracting Semantic Information from Speech. 1 edn. Wiley (2011)
2. Fiscus, J.G.: A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER). In: Automatic Speech Recognition and Understanding, 1997. Proceedings., 1997 IEEE Workshop on, IEEE (1997) 347–354
3. Larkin, M.A., Blackshields, G., Brown, N.P., Chenna, R., McGettigan, P.A., McWilliam, H., Valentin, F., Wallace, I.M., Wilm, A., Lopez, R., Thompson, J.D., Gibson, T.J., Higgins, D.G.: ClustalW and ClustalX version 2.0. Bioinformatics **23**(21) (November 2007) 2947–2948
4. Sim, K.C., Byrne, W.J., Gales, M.J.F., Sahbi, H., Woodland, P.C.: Consensus network decoding for statistical machine translation system combination. In: IEEE Int. Conference on Acoustics, Speech, and Signal Processing. (2007)
5. Bangalore, S., Bordel, G., Riccardi, G.: Computing Consensus Translation from Multiple Machine Translation Systems. In: In Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU-2001. (2001) 351–354
6. Calvo, M., Hurtado, L.F., García, F., Sanchis, E.: A Multilingual SLU System Based on Semantic Decoding of Graphs of Words. In: Advances in Speech and Language Technologies for Iberian Languages. Springer (2012) 158–167
7. Hakkani-Tür, D., Béchet, F., Riccardi, G., Tür, G.: Beyond ASR 1-best: Using word confusion networks in spoken language understanding. Computer Speech & Language **20**(4) (2006) 495–514
8. Benedí, J.M., Lleida, E., Varona, A., Castro, M.J., Galiano, I., Justo, R., López de Letona, I., Miguel, A.: Design and acquisition of a telephone spontaneous speech dialogue corpus in Spanish: DIHANA. In: Proceedings of LREC 2006, Genoa (Italy) (May 2006) 1636–1639