SIMPLIFYING LANGUAGE THROUGH ERROR-CORRECTING DECODING*

†*Juan-Carlos Amengual*, ‡*Enrique Vidal, and* ‡*José-Miguel Benedí*

†Unidad Predepartamental de Informática
Universidad Jaume I de Castellón
Campus de Penyeta Roja, 12071 Castellón (Spain)
‡Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
Camino de Vera s/n., 46071 Valencia (Spain)

ABSTRACT

In many speech processing tasks, most of the sentences generally convey rather simple meanings. In these tasks, the "word-recognition" problem is much more difficult than the underlying "speech understanding" problem would be. Accordingly we try to develop an adequate framework to focus on a properly defined "understanding" of the sentences rather than "recognizing" the (possibly) superfluous words. This can be seen as closely related with Spontaneous Language Understanding and Disfluence Modeling. In our approach, these problems are placed under the framework of Error-Correcting Decoding (ECD). A complex task is modeled in terms of a basic stochastic grammar, G, and an Error Model, E (taking insertions, substitutions and deletions into account). G should account for the basic (syntactic) structures underlying this task which would convey the semantics. E should account for general vocabulary variations, speech disfluencies, word disappearance, superfluous words, and so on. Each "complex" user sentence, x, will thus be considered as a corrupted version (according to E) of some "simple" sentence y of $L(G)$. Recognition can then be seen as an ECD process: given x, find a sentence y^* of $L(G)$ with maximum posterior probability. We introduce fast ECD techniques and adequate procedures for simultaneously training G and E and apply these ideas to a simple task with results showing the potential of the proposed approach.

1. INTRODUCTION

Many applications of interest to industry and business have *limited domains*; that is, only relatively simple lexicon and syntax are needed to express natural-sounding sentences to drive the actions involved in the application of the task. In many cases, this lexicon and grammar can be easily determined, leading to very high performance for input utterances that comply with the stated linguistic restrictions. However, in real operation casual users are by no means expected to strictly comply with the imposed linguistic restrictions, and performance often degrades dramatically.

A natural way of dealing with this typical situation is through *Error Correcting* (EC). Let G be a given (stochastic) grammar for the given task and let V be its lexicon. Each spontaneous user sentence, x, is thus considered as a corrupted version of some sentence y of L(G) (The corruption process is assumed to be driven by an adequate *Error Model*, E). Under such a setting, recognition can be seen as an Error-Correcting Decoding (ECD) process (see Figure 1): given x, find a sentence y^* of $L(G)$ with maximum posterior probability; that is,

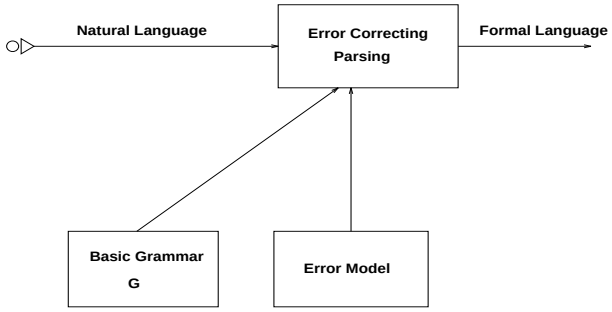

Figure 1: General Overview of the Recognition Process.

$$y^* = \underset{y \in L(G)}{argmax} \; P(y) \cdot P(x|y)$$

where $P(y) = P_G(y)$ is the probability of y in $L(G)$ and $P(x|y) = P_E(x|y)$ is the probability of x being a corrupted version of y according to E. If a Finite-State Model of G can be provided, the error–correcting parser can be implemented as a fairly simple extension of the Viterbi decoding algorithm (See [2] for an efficient implementation of this extension).

2. MODEL PROBABILITY ESTIMATION

While the structure of both G and E are fixed beforehand, the corresponding probabilistic parameters needed to compute $P_G(\cdot)$ and $P_E(\cdot|\cdot)$ have to be estimated from training data. A possible approach for a conventional error model is as follows:

First, initial probabilities of *substituting* words in V by other (spontaneous) words are roughly estimated, e.g., from co-occurrence (contextual) statistics over a corpus, S, of spontaneous (text) sentences of the task and/or a larger text corpus using techniques like

* This work has been partially supported by the Spanish CICYT under contract TIC95-0984-CO2-01/2.

those proposed in [8] [3] [4]. As an alternative, a (probabilistic) synonym dictionary could be used directly. Second, initial estimates for probabilities of *inserting* and *deleting* words (of V or S) are established from observed differences between the number of occurrences of each word in each sentence and the measured average number of occurrences of this word in the whole training set. Third, a reasonable amount of sentences of S are stochastic-error-correcting parsed through (G, E). If a resulting sentence of $L(G)$ is considered an (semantically) adequate digest of the corresponding sentence of S, then the parsing is accepted and the statistics of insertion, deletion and substitution errors and those of the rules of G are updated accordingly; otherwise, the parsing is discarded. Alternatively, the updating procedure can be supervised using N-Best alternatives which are automatically proposed by the error correcting parser itself, so that:

i) the meaning of the original sentence is preserved and

ii) the output belongs to $L(G)$ (which is *always guaranteed* by the ECD process itself).

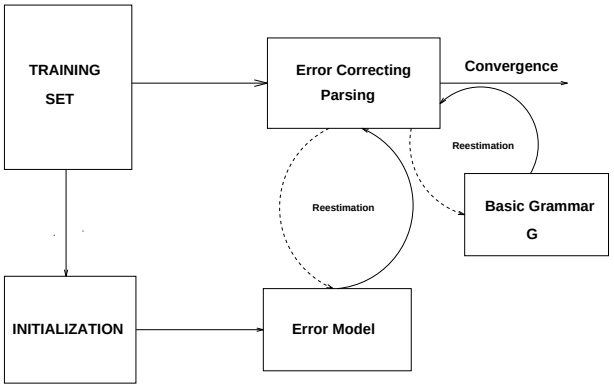

Figure 2: General Overview of the Training Process.

After this supervised run, better probability estimates should be available and a larger set of sentences in S can be submitted to stochastic-error-correcting parsing with, perhaps, less supervision. Eventually, the whole corpus could be used for a standard, unsupervised (Baum-Welch) reestimation of all the error and rule probabilities (see Figure 2).

3. EXPERIMENTS AND RESULTS

We have applied the above approach to a small application task recently proposed by Feldman et al [5]. It consists of describing simple two-dimensional visual scenes which involve a few geometric objects with (a small, fixed number of) different shapes, shades and sizes, which are located in different relative positions. Two (regular) *non-stochastic* grammars which model this simple task in both Spanish and English were constructed by hand. For the results, which are presented below, the required "spontaneous" training and test data were generated *automatically* as follows: First, basic sentences of the task were randomly generated using the given grammars. Then each of these sentences was submitted to a process of "spontaneization" in which many vocabulary variations, deletion of function words, repetitions of words and phrases and, in general, "disfluencies" of a variety of types were randomly introduced. This procedure was guided by empirical data about disfluencies provided by Shriberg [9]. This study suggests that disfluencies in spontaneous speech exhibit certain *regularities* and encourages researchers in Speech Processing areas to *explicitly* model them. Therefore, we wanted to check whether some of these regularities could be "learned" using the proposed approach.

Thanks to this generation of data, an *ideal* output is always known for each generated spontaneous training sentence and experiments can be easily performed in a *fully automatic* manner in order to explore the capabilities and detect the (possible) pitfalls of our approach. Moreover, large training data sets can be easily produced in order to empirically determine the "training needs" of our system. Some examples of pairs of "spontaneous"/"ideal" sentences are shown in Table 1. The automatic training process consisted in performing a Viterbi reestimation procedure guided by the following convergence criterion: *exact match* between system outputs and the corresponding expected ideal sentences for the training data, and *training-set perplexity difference* between two consecutive iterations less than a given threshold. In any case, the reestimation procedure was *halted* if the system reached a previously defined maximum number of iterations (25) in these experiments. Figure 3 shows an example of the evolution of this automatic training process for a given training set size.

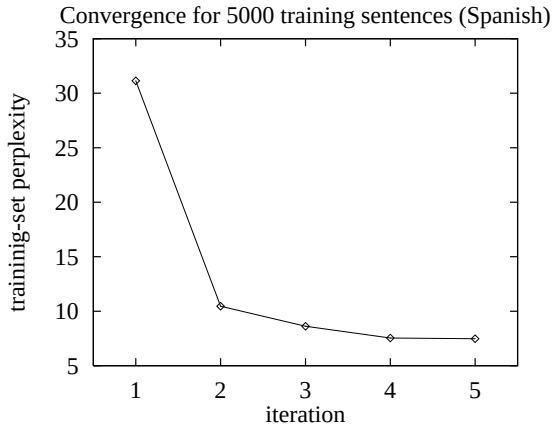

Figure 3: Training-set perplexity evolution for $5,000$ training sentences in $Exp1$ (convergence criterion fulfilled at iteration 5).

The Spanish version of Feldman's task has 29 words in the basic alphabet and our "spontaneous" Spanish sentences contained 41 out-of-vocabulary words, giving a total of 70 words ($2.4\times$ the size of the basic lexicon). This experiment is labeled $Exp1$. The English version has 25 words in the basic alphabet with 64 out-of-vocabulary words, giving a total of 89 words ($3.5\times$ the size of the basic lexicon). This experiment is labeled $Exp2$. In both cases, increasing size training-sets of pairs of *ideal/spontaneous* sentences were generated as explained above (250–$20,000$ pairs in $Exp1$, and 500–$20,000$ pairs in $Exp2$). Similarly, an *independent* test set of $1,000$ "spontaneous" sentences was generated for each language. See [1] for further details about these experiments.

SPONT	por favor se se se quita este un círculo negro que que está muy encima triángulo oscuro
IDEAL	se elimina el círculo oscuro que está muy por encima del triángulo oscuro
SPONT	that box and miniscule clear sphere are far to the to the left of of that dark box and that dark coloured sphere
IDEAL	a square and a small light circle are far to the left of a dark square and a dark circle
SPONT	un <unfillp> este <unfillp> <fillp> ese <unfillp> <fillp> una circunferencia está debajo de un <fillp> <unfillp> este <unfillp> este <fillp> una circunferencia mediana y un <fillp> este triángulo
IDEAL	un círculo está debajo de un círculo mediano y un triángulo

Table 1: Examples of pairs of sentences used in $Exp1$, $Exp2$ and $Exp3$, respectively (<fillp> and <unfillp> account for a filled and an unfilled pause, respectively).

Another important issue we wanted to check was the performance that could be achieved by using an automatically-learned grammar instead of using a manually-constructed one. It is easy to manually construct a grammar for this simple task, but this is not the case for most of real speech processing tasks. To this end, a third experiment, labeled $Exp3$, was performed. In this case, a 3-gram Finite-State grammar was trained from a set of $5,000$ training "basic" (clean) sentences of the task. This grammar was embedded into a Finite-State translation model in order to translate Spanish ("spontaneous") sentences of the task into English [10]. In this experiment ($Exp3$) the process of "spontaneization" focused mainly on two types of disfluencies: false starts and hesitations (simulating filled and unfilled pauses in particular, see third example of Table 1). The overall vocabulary size was 65 words ($2.25\times$ the size of the basic lexicon). Increasing size training-sets of pairs of *ideal/spontaneous* sentences were generated ($1,000$–$16,000$ pairs in this experiment). The test-set size was $10,000$ independent "spontaneous" sentences. See [10] for further details about this experiment.

The *initial* test-set word-error rate which was measured for each experiment was 43.5% ($Exp1$), 45.1% ($Exp2$) and 53.9% ($Exp3$). Figure 4 shows the resulting test-set word-error rate which was achieved using the models learned by the proposed system (after convergence).

The corresponding test-set perplexities and whole-sentence error rates are shown in Table 2 for the largest training set of each experiment.

	$Exp1$	$Exp2$	$Exp3$
Perplexity	7.5	12.2	11.0
Sentence-Error rate	0.0%	9.0%	0.6%

Table 2: Test-set Perplexity and Sentence-Error rate for the largest training set used in each experiment.

The results are perfect or almost perfect for the two experiments with the Spanish task. For the English task a residual word-error rate of 0.62% is obtained, which leads to a 9% sentence-error rate. Table 3 shows some examples of *wrongly* decoded sentences in $Exp2$. It can be observed that, for the "spontaneous" construction

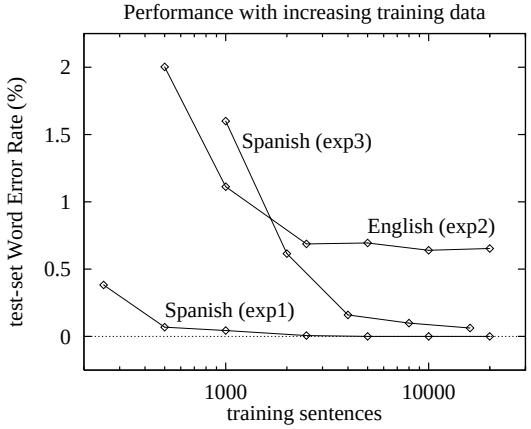

Figure 4: Observed final test-set word-error rate for the experiments performed.

on the right-hand side, the best alternative provided by the system consists in *deleting* **right-hand** and *substituting* **side** by **left**. These problems are due to poor training of error-model probabilities: typically, the reestimation loop arrives to a "bad" local maximum. This should be solved by better initialization techniques and/or by more powerful supervisation, perhaps based on N-Best techniques.

4. DISCUSSION AND FUTURE DEVELOPMENTS

The results reported in the above section clearly show the potential of the proposed approach to simplify spontaneous language by *explicitly* modeling typical disfluencies, vocabulary variations, and so on. The measured word-error rate for the "spontaneous" input sentences (see Section 3) was large enough to dramatically degrade the performance of any speech recognition system (one out of two words was erroneous). Assuming that a variety of spontaneous speech disfluencies show regularities [9], the results achieved indicate that these regularities can be learned under the framework of Error-Correcting Decoding. Nevertheless, many issues remain to be considered to successfully deal with real tasks:

1. In real speech processing tasks, it is not always possible to have the required amount of training data to be able to learn language and error models which are adequate enough.
2. The assumption of knowing the expected ideal output *beforehand* is clearly not realistic for most speech processing tasks.
3. Poor estimation of error-model probabilities is likely with complex tasks involving contrived disfluencies and/or vocabulary variations (see examples in Table 3).

Some short-term developments to be carried out in order to solve these important problems are described below:

1. To improve the initialization process with regard to the initialization values provided for insertions and deletions. Some

INPUT	*the* square is added to the *on the* **right-hand side** of the small light square and the small square
OUTPUT	*a* square is added to the \ \ \ **left** of the small light square and the small square
EXPECTED	a square is added to the **right** of the small light square and the small square
INPUT	**little** dark *rectangle with four equal sides* is far above a small *bright* circle *plus this great* light *three sided figure*
OUTPUT	**a** dark \ \ *square* \ \ is far above a small *light* circle *and a large* light *triangle* \ \
EXPECTED	a **small** dark square is far above a small light circle and a large light triangle

Table 3: Examples of sentences which are *wrongly* "simplified".

techniques based on using resemblance coefficients to measure the similarity between qualitative attributes could be helpful to obtain this (necessary) improvement [7].

2. To train the error model and the basic grammar not only when the system output *entirely matches* the expected ideal output, but also with *partially-correct* decoded sentences. This can be achieved by computing the Longest Common Subsequence between the expected ideal output and the output provided by the system.
3. To supervise the updating procedure in the training phase by computing the N-Best alternatives which can be automatically provided by the system, using efficient techniques as those proposed in [6]. It is expected that this technique will help solving problem 3 above mentioned.

While these solutions can actually solve some of the problems mentioned above, an important problem remains open: the necessity of knowing the expected ideal output beforehand for a large quantity of spontaneous sentences. Clearly, this is not always possible in many real speech processing tasks. Nonetheless, this problem can be overcome through *bootstrapping*: First, an *initial* system is trained with a few *manually* "simplified" sentences; then, this initial system is improved by using a supervised training phase. The whole process can be iterated until sufficient training material has been processed and the training can be considered completed.

Finally, an important issue to be achieved in the future is *to fully integrate* the error models learned using the proposed approach with acoustic models in order to develop a spontaneous speech recognition system.

5. REFERENCES

1. J.C. Amengual, E. Vidal. *Canonización del Lenguaje mediante Técnicas de Corrección de Errores*(in Spanish). Technical Report, DSIC-II/17/95. Depto. de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia. Spain. September, 1995.
2. J.C. Amengual, E. Vidal. *Two Different Approaches for Cost-efficient Viterbi Parsing with Error Correction*. SSPR'96, IAPR International Workshop on Structural and Syntactical Pattern Recognition, August 20–23, 1996, Leipzig. To be published in the Proceedings.
3. H. Chen, P. Hsu, R. Orwig, I. Hoopes and J.F. Nunamaker. *Automatic Concept Classification of Text from Electronic Meetings*. Communications of the ACM, Vol.37, No.10, pp. 56–73. October 1994.
4. I. Dagan, K. Church and W. Gale. *Robust bilingual word alignment for machine aided translation*. In Proceedings of the Workshop on Very Large Corpora: Academic and Industrial Perspectives, pp. 1–8, 1993.
5. J.A. Feldman, G. Lakoff, A. Stolcke, S.H. Weber. *Miniature Language Acquisition: A touchstone for cognitive science*. Technical Report, TR-90-009. International Computer Science Institute, Berkeley, California. April, 1990.
6. V.M. Jiménez, A. Marzal. *An Algorithm for Efficient Computation of K Shortest Paths*. Technical Report, DSIC-II/38/94. Depto. de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia. Spain. 1994.
7. H.C. Romesburg. *Cluster Analysis for Researchers*. Robert E. Krieger Publishing Company, Inc., 1990.
8. G. Salton. *Automatic Text Processing*. Addison-Wesley Publishing, Reading, Mass., 1989.
9. Elizabeth E. Shriberg. *Preliminaries to a Theory of Speech Disfluencies*. PhD Dissertation. University of California at Berkeley, 1994.
10. J.M. Vilar, E. Vidal and J.C. Amengual. *Learning Finite State Models for Language Translation Applications*. Technical Report, DSIC, 1996. Depto. de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia. Spain. 1996.