ARCHIVES OF ACOUSTICS
32, 1, 111–117 (2007)

GENERALIZATION IN CONTEXT SENSITIVE GRAMMARS

Andrzej PLUCIŃSKI

Adam Mickiewicz University
Institute of Linguistics
Międzychodzka 5, 60-371 Poznań, Poland
e-mail: apl@amu.edu.pl

(*received October 16, 2006; accepted January 11, 2007*)

An alternative to the recognition systems used so far can be systems based on context sensitive grammars. A precondition for success will be an implementation with great capacity for generalization. We propose three methods of generalization based on the assumption that similar stimuli invoke similar or identical reactions.

Keywords: decision trees, supervised learning, generalization methods.

1. Introduction

Generalization consists in the transfer of the knowledge about some events to others belonging to the same class. Physically this means assigning some points of the feature space values of some function on the basis of its known values at other points. In practice, the values of an interesting function are specified on the basis of some size-limited set of observations called a training sample. The training sample usually does not provide us with all possible events, so the need for the generalization ability arises, i.e. the ability to respond correctly with high probability to events not encountered in the training sample. Low generalization ability means overtraining or overfitting [1] while too large generalization ability increases the error risk.

In commonly applied hidden Markov models, the source of generalization are assumptions pertaining to probability distributions of reactions to encountered events (cf. e.g. [2]). The source of generalization in neural nets consists in assigning the same value to sections of the feature space (cf. e.g. [3, 4]).

Here the solutions proposed are similar in effect to the generalization reached in neural nets. They rely on the assumption that similar stimuli invoke similar or identical reactions. In the following sections we discuss three generalization methods and show experimental results confirming their effectiveness when applied to phonemic transcription rules for Polish texts built on the basis of a training sample. Our final goal is to build a speech recognition system based on context sensitive grammars.

2. Generalization methods

As said, we consider and justify three generalization methods: through minimization of the context lengths, through indeterminacy absorption and through probability-based guessing.

2.1. Context minimization

The first demand on a context grammar is to ensure the distinguishing of every transcription case of every event. This requirement can be met by taking into account enough large context environments for every event. The requirement does not impose any upper limit on the contexts' extent. However, too long contexts cause overfitting and consequently decrease the generalization ability. Therefore, the need arises to constrain them to a maximal degree while still meeting the above-mentioned requirement. In this task, left and right contexts should be treated separately. Moreover, distant correlations also should be taken into account. According to these postulates, we put into practice procedures for constraining left and right contexts and cutting them from the inside.

2.1.1. Constraining context definitions

The task is performed with the following algorithm:

1. We start out from a context environment wide enough to distinguish all of the transcription cases.
2. We shorten the right context at 1 and check whether it causes transcription ambiguity. If it does, then everywhere where it happens we restore the previous context definitions and do not change them further.
3. We do the same with left contexts.
4. We repeat steps 2 and 3 for every rewriting rule until we reach zero-length left and right contexts.

The process of generating the context sensitive grammar based on some training samples was carried out as follows:

Two equinumerous sequences of input and output symbols were given. The first sequence represented events and the second their transcriptions. The input symbols sequence was swept with a window of a priori given width. The symbol from the window center was assigned to its counterpart from the output sequence and the actual context environment seen through the window was recorded. As the result, every input symbol was assigned a context-conditioned reaction histogram (Fig. 1a). A one-column histogram means unambiguous transcription, whereas a multicolumn histogram means ambiguous transcription.

If the window is wide enough, then we obtain a one-column histograms. After constriction of the window, it may happen that for a certain rewriting rule we obtain multicolumn histograms (Fig. 1b). In such cases the previous context definition should be restored (Fig. 1c). We want to build deterministic grammars, because an acceptor is

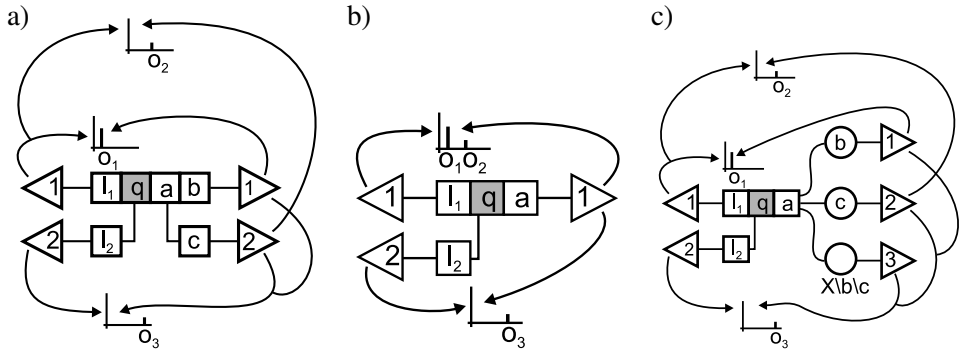

Fig. 1. Processing of rewriting rules: a) before restriction, b) after window restriction from the right, c) after graph path restoration. q – transcribed symbol, l_1, l_2 – left contexts, ab, ac – right contexts. Rectangles are elements seen through the window used for analyzing a training sample, the round elements are for the extensions.

then simpler and much faster ([5]). For this reason we added to the path, which did not need to be expanded, a single node representing the set complementary to the input alphabet for all alternative, restored continuations (Fig. 1c).

The effectiveness of the procedure was tested by applying it to phonemic transcription rules for Polish texts. For this purpose we prepare the teaching sample consisting of the orthographic text of circa 350 000 letters, and its phonemic transcription (see [5] for the aligning method). From this sample a 40 000 letter text training sample was taken at random. The rest of the teaching sample was used as a verifying sample. Experiments showed that all the transcription cases in the training sample were distinguished when 4-letter left and right contexts were taken into account. To ensure an unambiguous transcription in all cases, we began the analysis from a 10-character window and repeated

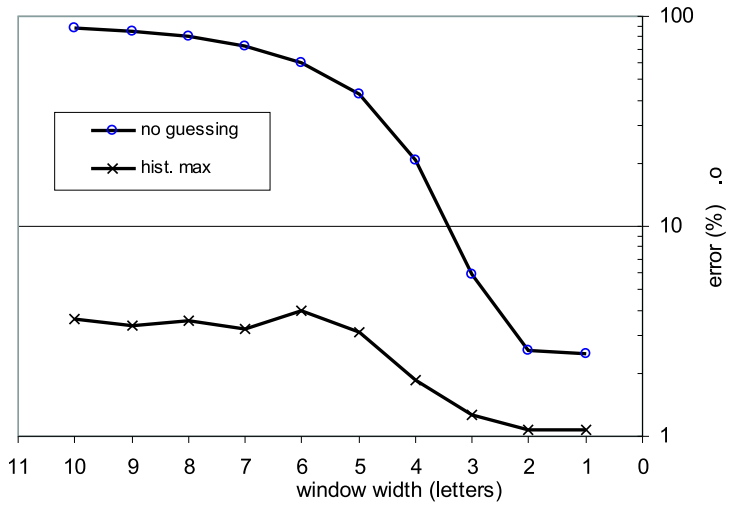

Fig. 2. Effectiveness of context minimization and guessing at different stages of data analysis.

it until a 1-character window was reached. After every stage, a verification was carried out and transcription errors were counted. The results are shown in Fig. 2 as a plot of the "no guessing" case.

As one can see, the transcription error rapidly falls as the window width decreases. This is a manifestation of the generalization ability. The algorithm in this task generated a tree heap consisting of about 5000 nodes (see [5] or [6] for more details).

2.1.2. Cutting out the contexts from the inside

Maintaining the rule that "changes in the context definition should not cause transcription ambiguities", one can remove some of their inside parts. From the graph of the grammars, one can notice that common parts of the path contribute nothing to distinguishing transcription cases. These parts can be removed providing the opportunity for a correct generalization. The removal of the inside context-definition sections can be accomplished by replacing characters with a symbol of the input alphabet. This principle is illustrated in Fig. 3.

Fig. 3. Principle of cutting out the context definition from the inside.

Effectiveness. This procedure contributed nothing to the phonemic transcription rules for a Polish text because distant correlations do not play any role here.

2.2. Indeterminacy absorption

Rewriting rules of some context sensitive grammar can be shown in a table in which row headings are definitions of left contexts and columns headings are definitions of right contexts. The table body, which we call the reaction matrix, describes context-conditioned transcriptions. Such a representation allows us to notice some possibilities for the removal of transcribing function indeterminacies. If the transcription function values are not determined for some context pairs, then we can proceed as follow:

1. For the reaction matrix column from which we want remove the indeterminacies, we look for the most determined column which has the same values in the corresponding components.
2. We replace the undetermined values with determined ones taken from the second column.
3. We repeat this procedure for the rows.

The procedure is illustrated in Fig. 4.

q	r_1	r_2
l_1	$\mathbf{o}_1$	–
l_2	$\mathbf{o}_2$	$\mathbf{o}_2$

=

q	r_1	r_2
l_1	$\mathbf{o}_1$	$\mathbf{o}_1$
l_2	$\mathbf{o}_2$	$\mathbf{o}_2$

Fig. 4. Indeterminacy absorption.

Effectiveness. The procedure was applied to the minimum context transcription rules built as described above. In order to check the effectiveness of the procedure, we draw training samples of different length at random from the teaching sample. The remainder was always used as the verifying sample. We performed the analyses described in Sec. 2.1 and applied the indeterminacy absorption. Results are shown in Fig. 5, on the "after indeterm. absorption" plot. As one can see, the indeterminacy absorption is the most effective generalization method.

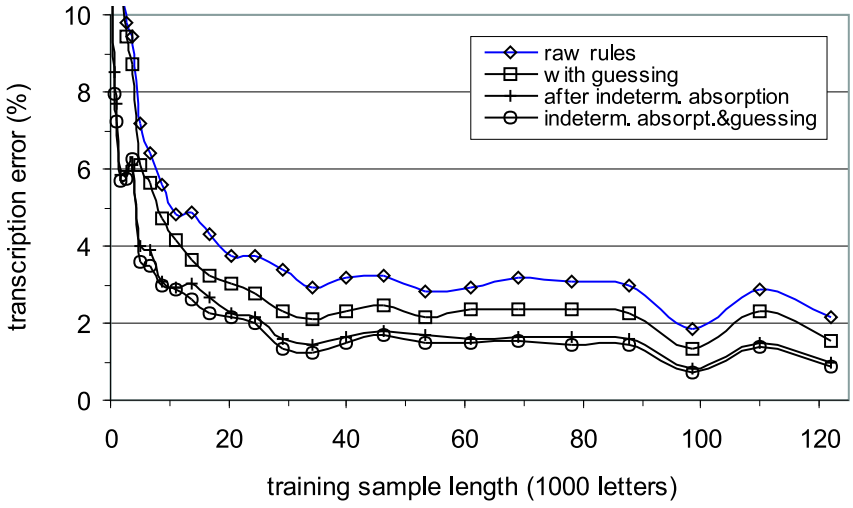

Fig. 5. Letter transcription error as a function of the training sample length.

2.3. Reaction guessing

This method can be applied when in a transcribed text there appear events in contexts not encountered in the training sample. We make most of the constrictions resulting from partial acceptance of the actual context by one of the worked-out rules. There are two possible approaches:

1. Choose the most probable value from the values contained in the reaction matrix assuming that all of them are equiprobable.
2. Proceed as above, but taking into account their probability determined on the basis of the training sample.

The first approach gives an independence from the training sample statistics. To do this we have to compute a proper histogram adequate to the present situation. The procedure boils down to summing the histograms assigned to continuation path clusters as shown in Fig. 6 when applying the second approach. If we want to apply the first approach, we simply assume that every given histogram count is equal to 1.

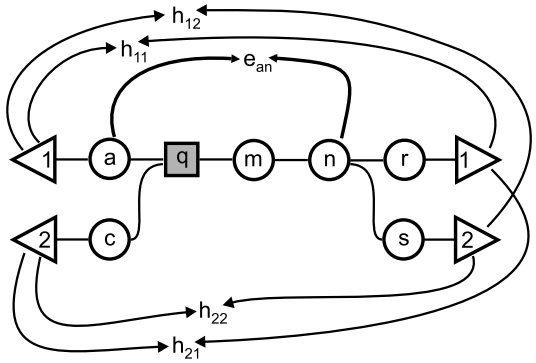

Fig. 6. Calculation of the transcription histogram for the letter q in the context aqmn; e_{ij}, h_{kl} – histograms of transcriptions, e. g. $e_{\mathrm{an}} = h_{11} + h_{12}$.

Effectiveness. We obtained slightly better results for the first approach. This case is shown in Fig. 2 as the "hist max" case plot (note logarithmic error scale) and in Fig. 5 (see [6] for more details).

3. Conclusions

The proposed generalizing methods are straightforward and very effective. We hope that they can be successfully applied in speech recognition systems. To provide input symbols it suffices to apply vector quantization to events from the acoustic features space.

Acknowledgment

This work was supported by Grant No. 2H01D 010 25 from the State Committee for Scientific Research, Poland.

References

[1] MANNING C. D., SHÜTZE H., *Foundations of statistical natural language processing*, The MIT Press, Cambridge, Massachusetts 2001.

[2] LEE K. F., *Automatic speech recognition. The development of the SPHINX system*, Kluwer Academic Publishers, Massachusetts 1989.

[3] MORGAN D. P., SCOFIELD C. L., *Neural networks and speech processing*, Kluwer Academic Publisher, Boston–Dordrecht–London 1991.

[4] PLUCIŃSKI A., *Neural nets in speech recognition* [in Polish], [in:] *Euphony and logos*, J. Pogonowski [Ed.], pp 233–256, Wydawnictwo Naukowe UAM, Poznań 1995.

[5] PLUCIŃSKI A., *Data-driven reconstruction of phonemic transcription rules* [in Polish], Wydawnictwo Naukowe UAM, Poznań 2002.

[6] PLUCIŃSKI A., *Data-driven construction of grapheme-to-phoneme transcription rules for Polish texts*, Studia Phonetica Posnaniensia, 7, 23–49 (2005).