A SINGLE-STAGE TOP-DOWN PROBABILISTIC APPROACH TOWARDS UNDERSTANDING SPOKEN AND HANDWRITTEN MATHEMATICAL FORMULAS

Jörg Hunsinger and Manfred Lang

Institute for Human-Machine Communication,
Technical University of Munich, D-80290 Munich, Germany
{hunsinger, lang}@ei.tum.de

ABSTRACT

We present a novel approach towards a multimodal analysis of natural speech and handwriting input for entering mathematical expressions into a computer. It utilizes an integrated, multilevel probabilistic architecture with a joint semantic and two distinct syntactic models describing speech and script properties, respectively. Compared to classical multistage solutions our single-stage strategy benefits from an implicit transfer of higher level contextual information into the lower level segmentation and pattern recognition processes involved. For visualization and postprocessing purposes, a transformation into Adobe® FrameMaker® documents is performed.

Fully spoken or handwritten realistic formulas were examined, yielding a structural recognition accuracy of 61.1 % for speech (speaker independent) and 83.3 % for handwriting (writer dependent).

1. INTRODUCTION

Electronic acquisition of mathematical formulas via conventional tools is a time consuming and complicated task. Therefore it is essential to exploit the capabilities of natural, especially speech and handwriting interaction, both being the fastest and most intuitive channels for registering mathematical expressions [1]. In order to facilitate future data fusion techniques and for uniformity reasons, it makes sense to use common semantic and syntactic representation formalisms for both modalities which may be integrated into a generalized input parsing mechanism. Our context free grammar implementation via probabilistic network structures in conjunction with an extended Earley-type top-down chart parser fulfills these requirements. Fig. 1 shows an overview of the current system components on the different abstraction levels. The following section gives an outline of the applied system architecture.

2. SYSTEM OUTLINE

2.1 Grammar

The syntactic-semantic attributes of spoken and handwritten mathematical formulas are represented by the parameters of a so-called **Multimodal Probabilistic Grammar**. It combines properties of context free phrase structure grammars with those of graph grammars by allowing for word-type, symbol-type, and position-type terminals. Formally, it is defined by a Chomsky set $G = \langle \Sigma, V, T, P \rangle$ including a start symbol Σ, a set of variables V, a set of terminals T, and a set of context free production rules P [2]. All the production rules are associated with statistical weights obtained from authentic spoken or handwritten training corpora, respectively.

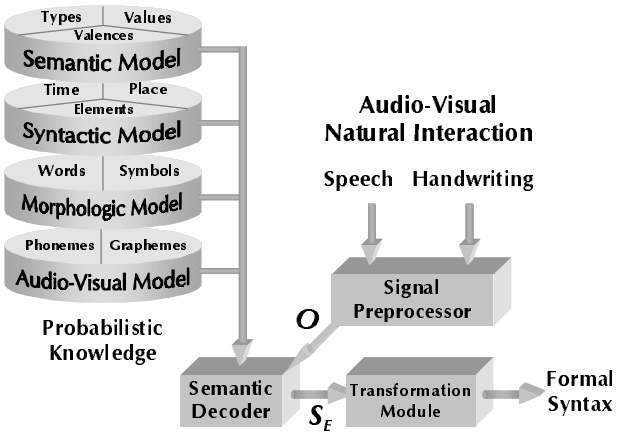


Figure 1: System overview. The top-down probabilistic semantic decoder derives a recognized semantic representation S_E from a preprocessed observation sequence O. The result is transformed to a formal mathematical description language to be fed into a conventional formula editor.

2.2 Semantic Representation

The grammar is implemented into a single stage semantic decoder by means of a compact semantic representation called **Semantic Structure** S [3]. It is given by an N-fold hierarchically structured combination out of a predefined inventory of **semuns** s (semantic units) with corresponding types t, values v, and successor attributes, every unit referring to a certain mathematical operator or operand [1]:

$$S = \{s_n\},\ 1 \le n \le N;\quad s_n = \{t, v, X(t)\},\ X \ge 1, \qquad (2)$$

where $X(t)$ denotes the type specific semantic valence, i.e. the number of successor semuns.

The probabilistic nature of this semantic representation is incorporated by three types of statistical weights:

From: Proc. ICSLP 2000, IEEE

- **root probabilities** $\eta_0 = P(s_1.t)$ (3)
  assigned to every occurring root semun type t (note: henceforth, the notation *a.b* identifies a property *b* of the entity *a*),

- **value probabilities** $\varepsilon_n = P(s_n.v \mid s_n.t)$ (4)
  assigned to every existing semantic value v of a given semun type *t*, and

- **successor probabilities** $\eta_n = P(r(s_n) \mid s_n.t)$ (5)
  assigned to every allowed combination *r* of successor semun types.

Further details of the semantic formalism may be found in [4].

2.3 Syntactic Representation

On the syntactic level every semun of a given semantic hypothesis is assigned to a so-called **Syntactic Module** *SM*. It consists of an advanced transition network which enables two distinct stochastic processes: 1) transitions from one node to another and 2) emissions of **elements** (i.e. spoken words or handwritten symbols) or local **offsets** between associated symbols or symbol groups. Transitions are responsible for modeling speaking and writing order, whereas emissions account for varying word, symbol, or position choice, respectively. All the *SM*s belonging to a complete Semantic Structure form an interconnected so-called **Syntactic Network** *SN* which is constituted as follows:

$$\begin{aligned} SN &= \{SM_n\},\ 1 \le n \le N; \\ SM_n &= \{St_n, E_n, A_{n1}, \ldots, A_{nX}, B_{n1}, \ldots, B_{nY}, C_n\} \end{aligned} \quad (6)$$

Every SM is opened via its start node *St* and closed via its end node *E*. Successor SMs are connected to their parent SM via *X* individual *A* nodes (also called successor nodes), so that the hierarchy of the corresponding Semantic Structure is mapped to the SN. These *A* nodes are also responsible for emitting pairwise positional offset vectors $\vec{O}$ between all the handwritten symbols or symbol groups belonging to the SN subbranches connected to them (see below). Further, there is a type specific number $Y(t) \ge 0$ of *B* nodes, each emitting one significant element e^+ of the total input. In case of speech these elements represent spoken words, in case of script they correspond to handwritten symbols. Optionally, a single *C* node is entered which emits an insignificant element e^-. This feature is especially used to model expletive spoken phrases, whereas usually no insignificant elements are found on the syntactic level of handwritten input.

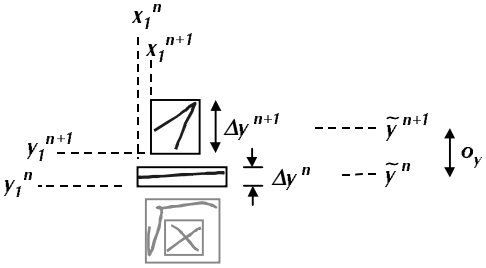
The following three types of probabilistic parameters are needed for a statistical rating of the syntactic contribution to an overall hypothesis score:

- Different paths through a given SM_n are statistically weighted by means of matrices of **transition probabilities**
  $$\Delta_n = \left[\delta_n^{ij} = P(i \to j \mid s_n.t, i)\right], \quad i, j \in SM_n, \quad (7)$$
  where $i \to j$ denotes any allowed transition from node *i* to *j*. Additionally, every single *SM* node must be passed exactly once – except for the optional *C* node – before the end node is reached.

- The type specific **offset emission probabilities**
  $$\alpha_{nk} = P(A_{nk} \to \{\vec{o}\} \mid s_n.t), \quad 1 \le k \le X \quad (8)$$
  model the statistical weight for the emission of a set of offset vectors covering the positional relations between all the elements emitted inside the *SN* subbranch connected to node A_{nk}. Since this procedure is performed recursively every time a successor *SM* is closed by returning to the next higher *SM*'s *A* node, a complete pairwise rating of every symbol's position relative to all its syntactic-semantic predecessor symbols is guaranteed. The pairwise offset definition for two consecutive elements is illustrated in Fig.2.

- The so-called **element emission probabilities**
  $$\beta_{nl} = P(B_{nl} \to e^+ \mid s_n.t, s_n.v), \quad 1 \le l \le Y \quad (9)$$
  $$\gamma_n = P(C_n \to e^- \mid s_n.t) \quad (10)$$
  account for a statistical rating of significant (type and value specific) or insignificant (type specific) element emissions, respectively. For consistency reasons, γ_n is set to unity if the corresponding *C* node is not passed.

All transition and emission probabilities as well as semantic root, value, and successor probabilities were estimated from training corpora obtained from separate speech and handwriting usability tests (cf. section 2.8). As a summary, a schematic view of a general Syntactic Module with all its attributes is displayed in Fig. 3.



$$\vec{o} = \begin{bmatrix} o_x = \Delta(x_{1+} g_x \cdot \Delta x) = \Delta\tilde{x} \\ o_y = \Delta(y_{1+} g_y \cdot \Delta y) = \Delta\tilde{y} \end{bmatrix}$$

offset weighting factors $0 \le g_{x,y} \le 1$

Figure 2: Offset vector calculation based on surrounding rectangles. In this example the position of a single handwritten symbol belonging to SM_{n+1} is charged against that of another single symbol belonging to the predecessor SM_n. The type specific weighting factors $g_{x,y}$ account for special constraints due to handwriting conventions.

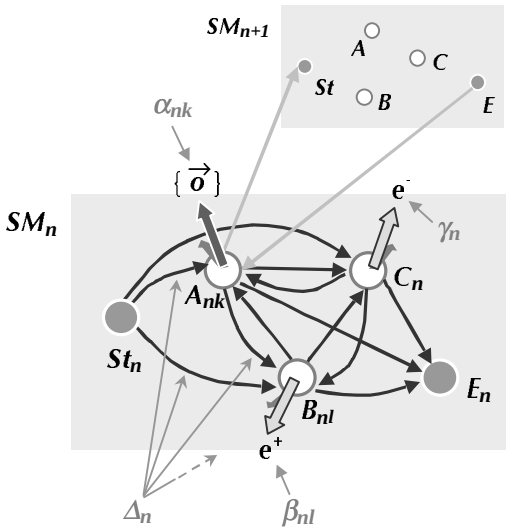


Figure 3: Generalized Syntactic Module. The different types of nodes, transitions, emissions, and the corresponding probabilities are annotated according to section 2.3.

2.4 Morphologic Representation

A phonetic lexicon holds phonetic transcriptions for every spoken word that was found in the acoustic training corpus; accordingly, every word is represented by a chain of its respective phoneme models (cf. section 2.5).

Likewise, every handwritten mathematical symbol included in the graphic training corpus is made up of a varying number of graphemes which are naturally equivalent to continuous handwriting strokes, i.e. sequences of pen movements between a pen-down and a pen-up event. The according graphemic representation is an implicit part of the applied pattern recognition technique and the attached handwriting sample database (cf. section 2.5).

2.5 Signal Representation

On the signal near levels the preprocessed speech or handwriting input sequences are rated using 30-dimensional phoneme based semi-continuous HMMs (Hidden Markov Models) [5] or 7-dimensional DTW (Dynamic Time Warping) matching [6], respectively.

Handwritten input is acquired via a WACOM PL-400 LCD digitizing tablet. Pen trajectories are sampled at 60 Hz and preprocessed via length equidistant resampling, smoothing due to low pass filtering, and size normalization. Euclidian distance minimization against all the samples inside a reference symbol database is then performed on the basis of feature vectors including horizontal and vertical positions, their first and second derivates, and binary pen-up/down values. The applied preprocessing and feature extraction methods were adopted from [7] and [8].

2.6 Classification

An extended Earley-type top-down chart parser performs a MAP (maximum a-posteriori) classification across all abstraction levels:

$$S_E = \underset{S}{\operatorname{argmax}} \max_{\Xi} \max_{\Phi} \left[P(O|\Phi)\, P(\Phi|\Xi)\, P(\Xi|S)\, P(S) \right] \quad (11)$$

The above equation indicates how an audio-visual observation sequence O is classified to a certain Semantic Structure S_E by maximizing over all possible speech/handwriting expressions Ξ and phoneme/viseme sequences Φ. In a one pass search algorithm all possible semantic hypotheses are successively tested until the best overall semantic representation of a given input is found. While $P(O|\Phi)$ and $P(\Phi|\Xi)$ are calculated successively every time the chart parser sends a request to the HMM/DTW classification layer, $P(\Xi|S)$ and $P(S)$ are obtained by incrementally multiplying semantic or syntactic probabilities as listed in sections 2.2 or 2.3, respectively, over all the semuns s_n belonging to a given Semantic Structure S.

One of the most prominent advantages of our single-stage top-down parsing mechanism lies in the fact that – especially for handwriting – no more isolated symbol segmentation rules or mechanisms are needed: In consideration of the local syntactic-semantic context per examined hypothesis, the optimum combination of handwriting strokes for matching a particular expected symbol is determined simultaneously with the pattern recognition process itself.

Due to a breadth-first search strategy, inline first-last processing is enabled.

2.7 Transformation

In order to achieve a uniform semantic representation for post-processing and future data fusion purposes, our concept relies on a single, modality independent semantic model. In the course of generalizing our syntactic-semantic formalism as well as the applied chart parsing mechanism from speech to handwriting needs we worked out such a revised semantic model.

Therefore our transformation module *MTrans* which transforms the hierarchically organized data of a recognized semantic structure into a formal mathematical description syntax [1] had to be updated. In addition, we implemented a back transformation module which will enable the user to switch freely between natural and classical input modes (e.g. during corrections or modifications) in the future.

2.8 Training

The current system implementation is based on two detached parameter sets obtained from separate speech and handwriting interaction usability studies. A specification of the speech corpus is given in [1].

As a next step we acquired a limited handwriting training corpus consisting of 85 realistic full formula samples, each containing about 50 symbols on average. A writer dependent set of probabilistic syntactic-semantic parameters (cf. sections 2.2 and 2.3) as well as a compressed database of DTW reference

patterns were derived from this corpus by means of our incidence based iterative training algorithms. We substantially reduced preparative efforts such as manual symbol segmentation and syntactic-semantic annotation by implementing a novel graphical analyzing environment called *StrokeTool* with a universal pen gesture based interface.

The positional parameters (eq. (8)) were estimated by calculating type specific two-dimensional Gaussians over all occurring pairwise symbol (or symbol group) offset vectors. Since only first order dependencies are considered in our approach, any successor semun subbranch was handled as an entity with a unified surrounding rectangle. However, every individual symbol position is included in the resulting parameter set due to recursive processing.

After refining our present position analysis technique, the handwriting knowledge bases will be enlarged in order to improve their statistical significance and to achieve writer independence.

3. RESULTS & CONCLUSIONS

For evaluation purposes we performed independent test classifications in either modality. The results for fully spoken or handwritten realistic formulas are summarized in Table 1. Since the positional part of our syntactic model (cf. section 2.8) has not yet been extended to the full range of supported mathematical functions [1], its contribution was neglected in this study. We will present the final recognition results in a subsequent paper.

	Recognition Accuracy	
	Speech	Handwriting
Training Corpus Reclassification	**76.2 %**	**87.5 %**
Independent Test Classification	**61.1 %**	**83.3 %**

Table 1: Recognition results. The numbers refer to full formula structural correctness under toleration of mere character confusions.

For the future we wish to support freely interfering speech and handwriting interactions including mutual coreferencing due to deictic wording and pen gesturing. To this end, the use of speech will presumably be focussed to subterm input and error corrections so that we anticipate a robust and approximately realtime forthcoming system performance.

4. REFERENCES

1. Hunsinger J. and Lang M., *A Speech Understanding Module for a Multimodal Mathematical Formula Editor*, Proc. ICASSP 2000, Vol. IV, pp. 2413-2416, Istanbul, Turkey, June 2000.

2. Gazdar G., Klein E., Pullum G.K., and Sag I.A., *Generalized Phrase Structure Grammar*, Basil Blackwell, Oxford, England, 1985.

3. Stahl H., Müller J., and Lang M., *Controlling Limited-Domain Applications by Probabilistic Semantic Decoding of Natural Speech*, Proc. ICASSP 97, pp. 1163-1166, Munich, Germany, April 1997.

4. Müller J. and Stahl H., *Speech Understanding and Speech Translation by Maximum a-posteriori Semantic Decoding*, Artificial Intelligence in Engineering, Vol. 13, No. 4, pp. 373-384, Elsevier Science, 1999.

5. Ruske G., Faltlhauser R., and Pfau T., *Extended linear discriminant analysis (ELDA) for speech recognition*, Proc. ICSLP 98, Vol. 3, pp. 1095-1098, Sydney, Australia, November/December 1998.

6. *Intel Recognition Primitives Library Reference Manual*, Intel Corporation, Order number 637785-007, 1998.

7. Winkler H.-J., Lang M., *Symbol Segmentation and Recognition for Understanding Handwritten Mathematical Expressions*, in A.C. Downton, S. Impedovo, editors, *Progress in Handwritimg Recognition*, Proc. 5th International Workshop on Frontiers in Handwriting Recognition IWFHR-5, Essex, England, September 1996, pp. 407-412, World Scientific, 1997.

8. Manke S., Finke M., and Waibel A., *NPen++: A Writer Independent, Large Vocabulary On-Line Cursive Handwriting Recognition System*, Proc. ICDAR 95, Montreal, Canada, August 1995.