A SPEECH UNDERSTANDING MODULE FOR A MULTIMODAL MATHEMATICAL FORMULA EDITOR

Jörg Hunsinger and Manfred Lang

Institute for Human-Machine Communication
Technical University of Munich
D-80290 Munich, Germany
{huj, lg}@mmk.e-technik.tu-muenchen.de

ABSTRACT

As part of a framework for a multimodal mathematical formula editor which will support natural speech and handwriting interaction, a single stage speech understanding module is presented. It is based on a multilevel statistical, expectation driven approach.

Completely spoken realistic formulas containing basic arithmetic operations, roots, indexed sums, integrals, trigonometric functions, logarithms, convolutions, fourier transforms, exponentiations, and indexing (among others) were examined. The speaker specific or formula specific structural recognition accuracies reach up to 90 % or 100 %, respectively. For visualization and postprocessing purposes, a transformation into Adobe® FrameMaker® documents is performed.

An advanced variant of this architecture will further be utilized as the basis for a multimodal semantic decoder incorporating combined script and speech analysis. It will enclose a so-called *Multimodal Probabilistic Grammar* which will be trained via multimodal usability tests.

1. INTRODUCTION

Mathematical formulas of various types are present in scientific publications, schoolbooks, technical manuals, and other writings. Considering the increasing amount of mathematical typesetting, there is an urgent need for intuitive and efficient input techniques.

For typesetting and calculation purposes, either descriptive tag syntaxes like those embedded in LaTeX or Mathematica® are applied (and therefore need to be known in detail by the user), or conventional formula editors based on graphical user interfaces (e.g. integrated in Microsoft® Word® or Adobe® FrameMaker®) are utilized. Meanwhile, the latters have become the predominant means for mathematical expression input, as much as DTP (Desktop Publishing) products have been supplied with extensive WYSIWYG ("what you see is what you get") characteristics.

However, some major disadvantages regarding three different aspects arise from the use of these classical formula editors:

- **Time requirement.** As a direct implication of the step-by-step graphical construction of mathematical expressions via mouse and keyboard interaction, the time needed to produce a typical mathematical formula is substantial.
- **Intuitivity.** Even quite simple expressions require a set of relatively complicated mouse-keyboard interactions to be performed by the user. Apart from preparatory training needs it is in most cases necessary to have the full intended formula in mind before selecting the first operation symbol from a graphical palette.
- **Input order.** Dependent on the implicit formal description language, the user is usually forced to build up an expression hierarchically in correspondence with its mathematical content. This means that an approximately sequential or natural input order – as familiar to the user from writing or speaking – is normally not supported.

In order to overcome these problems by deploying more natural and at the same time efficient interaction modes, numerous attempts have been reported which make use of handwriting conveniences; a comprehensive overview can be found in [1]. From a slightly more extended viewpoint, we are working on a multimodal concept for a novel mathematical formula editor including natural speech and handwriting input as well as pen gesture and speech coreferences to create, manipulate, and modify mathematical expressions. As a first step, a speech understanding module has been realized which is capable of interpreting and visualizing completely spoken mathematical formulas.

The overall benefit to be expected from the integration of the modalities mentioned above may be illustrated with the following preliminary usability study:

By means of a quantitative survey it was examined to which extent the average input time varies across the different modalities taken into account. 7 probands – throughout familiar with basic GUI (Graphical User Interface) functionalities – were asked to enter 10 notedly different mathematical formulas by the use of 1) natural speech, 2) natural handwriting, and 3) a conventional graphical formula editor[1], respectively. Training effects were ruled out by varying modality as well as formula orders among the participants. A subset of 3 probands already were familiar with the applied formula editor so that experience aspects could also be evaluated. The results are summarized in Table 1.

While it is obvious that handwritten input yields a significantly lower time consumption than conventional interfaces – a reduction factor of at least 5 is obtained – the additional savings of another 50 percent to be gained from spoken dialogue are remarkable.

[1] Microsoft® Equation® 3.0 (German version)

From: Proc. ICASSP 2000, IEEE

	Speech	Script	Mouse/ keyboard (expert)	Mouse/ keyboard (amateur)
Average input time	**10 s**	**20 s**	**100 s**	**140 s**
Rel. temporal efficiency	**14**	**7**	**1.4**	**1**

Table 1: Acquisition of mathematical formulas. Average input times required and relative temporal efficiencies referring to different user interfaces.

It must be noted, however, that any system response time due to speech or handwriting recognition components was neglected in this study – having in mind approximately real-time operable modules, the anticipated gain of time is still considerable. As a conclusion, granted that robustness and recognition rates be sufficient, it appears to be worthwhile to exploit the capabilities of both the fastest and most intuitive modalities for acquiring mathematics, particularly when their use may be combined in a reasonable way like it is found in interpersonal communication.

2. THE SPEECH MODULE

2.1 Mathematical Domain

Most standard and several special mathematical operations applied to any Latin or Greek alphabetic characters and Arabic numerals are included in our approach; essentially, basic arithmetic operations, roots, indexed sums, bracketing, integrals, equations, convolutions, fourier transforms, trigonometric functions, logarithms, exponentiations, and indexing are covered.

2.2 Architecture

Following [2], an integrated semantic decoding algorithm is applied which evaluates probabilistic knowledge on the acoustic, phonetic, syntactic, and semantic levels in a single stage architecture. A compact semantic representation called *semantic structure* S is used; it is a hierarchical combination of *semuns* (semantic units) s_n which – in our domain – represent mathematical operators or operands. In an expectation driven approach, a probabilistic chart parser [3] scans the search space across all representation levels and performs a maximum a-posteriori (MAP) classification to find the best overall semantic hypothesis S_E referring to a given observation sequence O (i.e. the preprocessed speech signal):

$$S_E = \underset{S}{\operatorname{argmax}} \max_{W} \max_{Ph} \left[P(O|Ph)\, P(Ph|W)\, P(W|S)\, P(S) \right]$$

Statistical independence of the above probabilities is assumed; phoneme sequence Ph and word chain W were introduced as intermediate representation levels. Semantic and syntactic probabilities correspond to the parameters of an extended context-free grammar, while Hidden Markov Models are used to calculate acoustic probabilities.

For visualization and postprocessing purposes, we implemented a transformation module called *MTrans*. It consists of a preprocessor and a modular compiler unit (to be easily extendable to

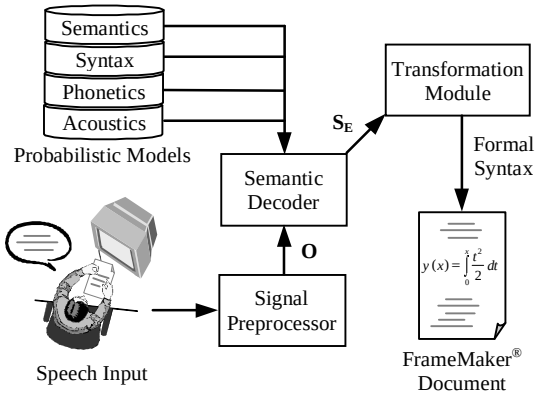

Figure 1: Schematic view of the speech understanding module's architecture. The recognized semantic structure S_E (from the observation sequence O) is transformed via a formal syntax expression into a FrameMaker® document for further processing.

further mathematical operations) which transforms the hierarchically organized data of a recognized semantic structure into a formal mathematical description syntax compatible with Adobe® FrameMaker®'s equation editor. Apart from being a widespread commercial word processing system – particularly in technical and scientific sectors – this application offers mathematical typesetting as well as calculation features. In contrast to most other approaches known so far the integration of FrameMaker® will enable the user to switch freely between natural and classical input modes (e.g. during corrections or modifications), provided that a back transformation module will be added.

The overall system architecture is sketched in Figure 1. A complete implementation exists for Windows NT® 4.0 or Solaris® 2.6, respectively.

2.3 Training

For the estimation of our statistical model parameters, i.e. the probabilities that are used to determine the ranking of concurrent hypotheses during the recognition procedure, an adequate set of training utterances had to be collected. To this end, a broad collection of typical formulas from the supported domain was adopted from [4] in order to guarantee high comparability of our recognition results. A printed version of this reference material was presented to the probands, who then recited it into a microphone. After all, 323 acoustic samples were obtained from 8 male and 2 female speakers, using a sample rate of 16 kHz. Feature vectors for HMM recognition were calculated on the basis of loudness spectra in 10 ms intervals.

After assigning the corresponding word chain to every utterance, the stochastic grammar's iterative training algorithms require all the samples to be manually classified on the syntactic-semantic level. To facilitate and standardize this laborious procedure, a graphical tool called *SemTree* was designed which allows the operator to successively construct semantic structures (with their word chain correlations included) in an intuitional drag & drop interface.

Considering the high percentage of similar monosyllables (e.g. lowercase variable names) in the investigated domain, a new set of Hidden Markov phoneme models was generated on the basis of the acoustic training corpus, thereby capturing some domain-specific pronunciation features. Via an LDA transformation the feature space was reduced to 30 from originally 66 dimensions [5].

2.4 Results

When realistic mathematical formulas are spoken as a whole – as performed in this preliminary approach – certain problems arise regarding the extraction of the appropriate mathematical meaning from an utterance. The most prominent ones and their implications are listed below:

- **Ambiguity.** Particularly on a subterm scale, natural speech is often ambiguous with respect to functional precedence. For example, the phrase *"a plus b divided by c"* may represent the terms (a+b)/c or a+(b/c), respectively. Since most typical formulas are structurally preserved despite such small-scale errors, subsequent corrections via speech, handwriting or conventional modes should be practicable in a forthcoming multimodal environment.
- **Character recognition errors.** Spoken alphabetic characters – usually representing variable names or numerical constants – are prevalent in most formulas. Due to their phonetic shortness, characters are often mistaken or even ommitted or erroneously inserted. Our semantically driven approach automatically limits the latter two cases (only semantically consistent hypotheses are tolerated), while character confusion persists as the most frequent source of error especially in completely spoken formulas.
- **Compound verbalism.** Certain phrases include multiple semantic contents in a single word, e.g. *"sixth"* or *"square"* describing a division by 6 or a potentiation by 2, respectively. In such cases the corresponding words were artificially split into two distinct pseudo-words, each assigned to the corresponding semun. This method provides a uniform semantic representation of mathematically equivalent expressions but also implies the risk of accidentally inserting "half-words" in the recognition result. Actually this turned out to be a minor problem, since the German pseudo-words that occurred bear little resemblance with the remaining vocabulary.
- **Redundance.** Although most people use a "semiformal" description style for mathematical expressions, some probands applied more or less diffuse circumscriptions by saying, e.g., *"fraction line, nominator a, denominator b"*. To model such phrases, semantic meta-types were introduced which "absorb" superfluous words; the transformation module (Figure 1) eliminates these meta-types when constructing the corresponding FrameMaker®-compatible expression.

A quantitative evaluation of the speech understanding accuracy was achieved by (a) reclassifying all the 323 training utterances and (b) decoding a set of 36 independent test utterances obtained from 2 additional speakers and 9 different mathematical formulas which were distinct from the training material. The results are summarized in Table 2.

	Recognition Accuracy		
	Microscopic	Mathematical	Structural[2]
Training Corpus Reclassification	**95.2 %**	**42.3 %**	**76.2 %**
Independent Test Classification	**93.6 %**	**22.2 %**	**61.1 %**

Table 2: Recognition results. The first column refers to the semantic unit level (all semuns correctly recognized), the second column denotes completely recognized mathematical formulas, and the third column includes a toleration of mere character confusions.

Although the recognition rates listed above are not very impressive at a first glance, they are of course due to the fact that only completely spoken large formulas – with an average of 40 words per utterance – were taken into account. In a multimodal environment, the use of speech will presumably be focussed to subterm input or completion and correction of handwritten expressions, respectively. Furthermore, the sample quality was strongly speaker dependent, so that the speaker specific structural reclassification accuracy reached up to 90 %.

Another interesting point is the recognition rate variance with formula type (Figure 2): It turned out that a higher degree of nested functions (predominantly roots and fractions) as well as character and numeral prevalence cause recognition losses, while distinctive operator names (e.g., trigonometric functions, logarithms, sums, integrals) stabilize the recognition process.

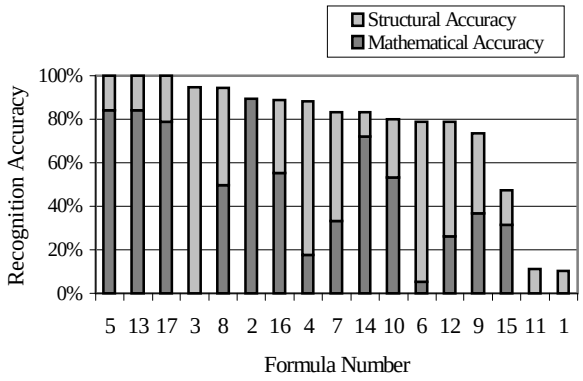

Figure 2: Formula specific recognition accuracy (reclassification). The formula numbers refer to [4] and are ordered by the overall recognition accuracy (character misinterpretations tolerated). On the whole, the quota of characters and numerals as well as function nesting increase to the right hand side, while the quota of function names decreases.

The example in Figure 3 illustrates the correlation between a typical recognized utterance and the corresponding semantic structure.

[2] (accumulated)

$$y = \int_0^{\infty} \sqrt{\frac{x}{6}}\, dx$$

„ Y equals - root x six th integrated over d x from zero to infinity “

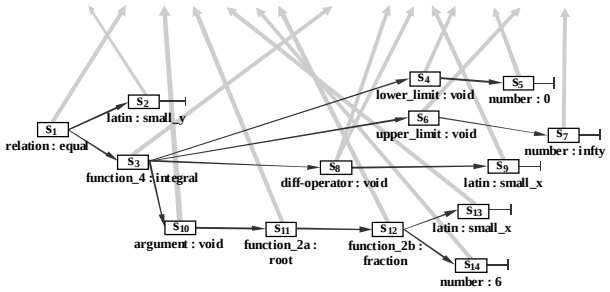

Figure 3: Example of a recognized utterance and the corresponding semantic structure. For easy understanding, a literal translation of the original German sentence into an English word chain is given. The semuns (semantic units) are labelled with their types and values, separated by colons. The light grey arrows mark the different semun - word correlations.

3. FUTURE WORK

The most significant advantage of a single stage semantic decoding architecture as it is used in this work results from the simultaneous evaluation of knowledge belonging to all the involved abstraction levels: Apart from self-focussing effects achieved by restricting the search process to locally consistent subhypotheses only, semantically corrupt recognition results are prohibited due to the applied top-down chart parsing algorithm.

To preserve this convenience and at the same time to allow an utmost degree of modality integration, our multimodal concept is based on a generalized and extended version of the context-free stochastic grammar described in [2]. Therefore, a formal *Multimodal Probabilistic Grammar* combining properties of extended phrase structure grammars [6] and graph grammars [7] will be defined. The corresponding set of rewriting rules will incorporate handwriting as well as speech characteristics, spanning a multimodal terminal space. This includes instances of spoken words, handwritten symbols, and so-called *offsets* representing relative spatial symbol or symbol group alignments. The statistical weights associated with the different production rules will again be extracted from authentic training material to constitute a joint semantic model (modality independent) and distinct syntactic models for handwriting and natural speech interaction. On the signal level, DTW or HMM based representations for the supported handwritten mathematical symbols will be used. A detailed specification of the applied formalism will be given elsewhere.

Extensive usability studies are necessary to determine users' needs regarding the combined use of the different modalities; especially, handwriting and speech coreferences including pen gestures, referencing vocabulary and coincidence aspects have to be evaluated. According to the classification given in [8], we thereby aim at the synergistic type of multimodality so that concurrent use of modalities and subsequent data fusion are suggested.

In parallel, the syntactic-semantic and acoustic knowledge bases of our speech understanding module will be optimized and refined to yield reliable recognition results particularly in a multimodal context.

4. SUMMARY AND CONCLUSIONS

The capability of natural speech input for mathematical typesetting was examined by means of a prototypic speech understanding module which accepts completely spoken complex formulas. The major sources of error were found to be ambiguities at a subterm level and character confusions. Despite these obstacles the resulting structural recognition accuracy indicates that natural speech will serve well as a supportive medium in a forthcoming multimodal environment.

Modality integration will be achieved by merging syntactic-semantic knowledge obtained from multimodal usability tests into a generalized probabilistic grammar's parameters. Via a back and forth transformation to FrameMaker®'s formula editor natural speech and handwriting interaction may also be complemented by conventional input modes in the future.

5. REFERENCES

[1] Blostein D. and Grbavec A.: *Recognition of Mathematical Notation*. In P.S.P. Wang and H. Bunke, editors, *Handbook on Optical Character Recognition and Document Image Analysis*, chapter 21, pp. 557-582, World Scientific Publishing, 1997.

[2] Stahl H., Müller J., and Lang M.: *Controlling Limited-Domain Applications by Probabilistic Semantic Decoding of Natural Speech*. Proc. ICASSP 97 (Munich, Germany), pp. 1163-1166, 1997.

[3] Stahl H., Müller J., and Lang M.: *An Efficient top-down Parsing Algorithm for Understanding Speech by Using Stochastic Syntactic and Semantic Models*. Proc. ICASSP 96 (Atlanta, USA), Vol. 1, pp. 397-400, 1996.

[4] Winkler H.-J. and Lang M.: *On-line Symbol Segmentation and Recognition in Handwritten Mathematical Expressions*. Proc. ICASSP 97 (Munich, Germany), pp. 3377-3380, 1997.

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

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

[7] Grbavec A. and Blostein D.: *Mathematics Recognition Using Graph Rewriting*. Proc. ICDAR 95 (Montreal, Canada), pp. 417-421, 1995.

[8] Nigay L. and Coutaz J.: *A design space for multimodal interfaces: concurrent processing and data fusion.* Proc. INTERCHI 93 (Amsterdam, Netherlands), pp. 172-178, ACM Press, 1993.