DML 2010

Alan Sexton; Volker Sorge; Masakazu Suzuki
Designing a Semantic Ground Truth for Mathematical Formulae

In: Petr Sojka (ed.): Towards a Digital Mathematics Library. Paris, France, July 7-8th, 2010.
Masaryk University Press, Brno, Czech Republic, 2010. pp. 37--42.

Persistent URL: `http://dml.cz/dmlcz/702571`

Designing a Semantic Ground Truth for Mathematical Formulae

Alan Sexton[1]*, Volker Sorge[1]*, and Masakazu Suzuki[2]*

[1] School of Computer Science, University of Birmingham, UK
A.P.Sexton@cs.bham.ac.uk, V.Sorge@cs.bham.ac.uk
http://www.cs.bham.ac.uk/~aps http://www.cs.bham.ac.uk/~vxs
[2] Faculty of Mathematics, Kyushu University, Japan
suzuki@math.kyushu-u.ac.jp, http://www.math.kyushu-u.ac.jp/~suzuki/

Abstract. We report on a new project to design a semantic ground truth set for mathematical document analysis. The ground truth set will be generated by annotating recognised mathematical symbols with respect to both their global meaning in the context of the considered documents and their local function within the particular mathematical formula they occur. The aim of our work is to have a reliable database available for semantic classification during the formula recognition process with the aim of enabling correct interpretations of mathematical formulae and generating semantic markup such as Content MathML.

1 Introduction

Ground Truth sets are manually annotated or validated sets of training data that are important tools for many recognition tasks. In document image analysis research, ground truth data is crucial for the design, training and testing of algorithms for data identification and extraction. A ground truth set for an optical character recognition (OCR) system generally consists of images of single characters together with their correct syntactic interpretation, e.g. in the form of ASCII code. Bespoke ground truth sets have to be developed to cater for types of target documents and recognition methods. In spite of the availability of automated tools for the development of ground truth sets for certain special cases [8], in the majority of cases ground truth sets can only be assembled semi-automatically and their manual correction is generally a very laborious task.

For mathematical OCR and formula recognition, assembling a ground truth set is an even more daunting task as it not only needs to contain a large number of, often very similar, symbols but also has to cope with the two dimensional layout of mathematical formulae and therefore contain spatial information. There is currently only one ground truth set for mathematical documents available [6]. It has been constructed from 30 different articles on

* The authors' work was supported by Royal Society International Joint Project (2008/R3)

Petr Sojka (editor): DML 2010, Towards a Digital Mathematics Library, pp. 37–42.
© Masaryk University, 2010 ISBN 978-80-210-5242-0

pure mathematics. It is a database of over 680,000 characters occurring both in text and mathematical formulae in the articles and can be used as a collection of statistical information about the relative occurrence of, and relationships between, neighbouring characters. It does not contain information about the structural nature of the expression as a whole that the symbols are contained in. Since most of the characters appear many times in this database, there is a large amount of information that can be mined from the database. For instance, the ground truth set has been used to compile statistics on spatial relationships between mathematical symbols [1] that are exploited to resolve sub- and superscript relationships within the mathematical formula recognition of the Infty system [5].

While [6] can greatly improve the robustness of algorithms for correct syntactic recognition, it is currently still of limited use for extracting semantic meaning of formulae. However, often mathematical formulae can only be correctly recognised if the underlying semantics of the formula is clear. Enriching a ground truth set for mathematical OCR with semantic information could therefore be desirable, and if a semantic ground truth set can be constructed successfully, it should not only enhance current formula recognition techniques, but also enable direct translation of expressions from mathematical documents into semantic markup such as Content MathML or OpenMath. This would aid accessibility tools in interpreting functions and their components correctly and also make content of mathematical documents amenable to mathematical software such as Computer Algebra systems.

Part of the problem in building high quality mathematical formula recognisers is the ambiguity caused by similar constructs used for different mathematical concepts. To take just one example, is $\binom{n}{k}$ a binomial coefficient (n choose k) or a vector? Without context we can not be confident of our answer. A semantic ground truth for mathematical expressions would help us to base our decisions on well-founded scientific data, rather than the programmer's mathematical intuition.

2 Semantic Ground Truth

The aim of our work is to compile a semantic ground truth set for mathematical formula recognition. We propose to implement this in two parts. The first part is a semantic ground truth for mathematical characters and symbols—here we are concerned with associating low level information of individual symbols and their local neighbours (e.g. font information spacing and relative baseline positions) with mathematical objects and constructs from the mathematical domain that the document in question belongs to. The second part is a semantic ground truth for mathematical expressions as a whole.

2.1 Semantic Symbol Ground Truth Set

Our approach to constructing the semantic symbol ground truth set is to annotate the mathematical symbols occurring in a syntactic ground truth set similar to [6]. Annotations will be based on the following three levels:

1. Subject area
2. Usage of a symbol
3. Definition within a given context.

The three different types of annotations enable the description of a symbol's semantics on three different levels of granularity.

Subject Area Each symbol will have an annotation attribute for its origin in some mathematical field, which will correspond to the two first digits of the AMS Mathematics Subject Classification of 2000 [7]. This refers to the general mathematical field the document belongs to from which the particular symbol was extracted. Symbols, as well as documents, can quite correctly have multiple different classifications, and classifications for individual symbols can differ from the classification of the document as a whole. We intend to record these different classifications so that we can mine the information to obtain probabilistic heuristics for identifying the classification of symbols in various contexts.

Usage of Symbol A common problem with the correct interpretation of mathematical symbols is that they often have different meanings depending on the overall mathematical area or the local context in which a formula occurs. Therefore, one of the semantic annotations will record the exact mathematical usage of each symbol, e.g., is it a function symbol, an operator, a relation etc., in the formula from which it was extracted. For example the following two formulae give the symbol g two distinct meanings:

$$g \in G \tag{1}$$

$$g \in B^{A} \tag{2}$$

In (1) g is declared as an element of a group G, whereas in (2) g represents a function with domain A and co-domain B. Consequently the former would be annotated as an ordinary symbol while the latter would be annotated as a function symbol. The usage of g can then be interpreted differently in other expression. For instance, in the expression

$$g(h\,k)$$

according to the semantic usage in (1), it would be part of a multiplication within a group, while it has to be interpreted as a function application according to the semantics given to it in (2).

Definition The most fine-grained semantic annotation will be based on the mathematical definition of a particular symbol in the context of the particular document it has been extracted from. We will use, as far as possible, the definition given in OpenMath content dictionaries [4] as annotations.

2.2 Semantic Expression Ground Truth Set

While we can attach semantics to individual symbols, and, to a certain extent, relationships between neighbouring symbols, this does not extend to whole expressions where the relationships are between neighbouring sub-expressions, rather than simply symbols. Before we can hope to attach such expression semantics, therefore, we need to identify sub-expressions for semantics to be attached to. We therefore propose to build abstract syntax trees (ASTs) for the expressions in our set and attach semantics to the nodes in these.

Since the leaves of the ASTs would consist of the single characters and symbols in the expression, they would automatically have the annotations of the symbol ground truth. The inner nodes of an AST would then inherit the subject area annotation. While an inner node would not have a usage annotation, it will be annotated with the definition corresponding to the semantics of the sub-expression rooted in this node.

For example, an expression of the form $(1\ \ 2\ \ 3)$ in group theory has as symbol annotations, open fence, three ordinaries, and closed fence, while the three ordinaries in turn have definition annotations as integers. The AST representing the entire expression then will have a separate definition annotation, which will be permutation.

3 Automated Generation

We intend to assemble the semantic ground truth set based on the machinery designed and implemented for the syntactic ground truth set presented in [6]. This facilitates the automatic recognition of mathematical symbols using the Infty system with subsequent manual correction. We are extending these tools to enable the handling and storage of semantic annotations.

While the subject classification will be entered globally for all symbols from an article, the other two semantic annotations will be entered using a semi-automated approach, via a hangman style completion mechanism. That is, for each article in the ground truth set, symbols occurring in mathematical expressions will be annotated manually. If the same symbol occurs elsewhere in the same article it will automatically be given the same annotation. Thus the number of symbols that need to be annotated should gradually decrease with the majority of work spent on manually checking and correcting annotations if the completion yields incorrect results (e.g., if one symbol is used with different semantic meanings in different contexts).

Moreover, we intend, as much as possible, to exploit automatic classification of symbols with respect to the second semantic annotation for basic mathematical usage of symbols. We have recently developed a mechanism to categorise symbols based on special relations between symbols occurring in a formula [3]. The spatial analysis is based on re-engineering the basic layout rules which are traditional in mathematical typesetting and which are also employed by the LaTeX system. The mechanism is currently employed for formula recognition from PDF documents [2], however, this will be extended in order to work for formula recognition in a more general context, such as from scanned documents.

As a simple example of the working of this algorithm consider the following two formulae:

$$x\mathcal{R}y \rightarrow y\mathcal{R}x \tag{3}$$

$$x \; \mathcal{R} \; y \rightarrow y \; \mathcal{R} \; x \tag{4}$$

Here the spacing between the symbols in (3) would not distinguish the three occurring letters and therefore classify all three as ordinal symbols. On the other hand the increased spacing between the $\mathcal{R}$ and x, y in (4) would automatically identify $\mathcal{R}$ as a relation symbol, which is indeed its intended meaning in the given formula.

While these techniques can assist in assigning semantics to symbols they are not sufficient for annotating at the level of expressions in ASTs. Normally, ASTs are the output of a parser using a set of grammar rules. Here, we want to use the resulting ground truth set to deduce the appropriate grammar rules. Therefore, we must construct the ASTs manually, to correspond to how a human mathematician understands the expression. With such ASTs, we can then attach semantic interpretations to the nodes to complete our semantic expression ground truth set.

Constructing ASTs manually is a particularly arduous task. We are implementing a tool that uses previous work on formula recognition on PDF documents [2,3] to construct an initial proposed AST, and provide a convenient user interface for manipulating this tree into one closer to what the human mathematician understands. At this point, semantic annotations for nodes can be added so that, in the binomial coefficient/vector problem mentioned in the introduction, the AST would be the same in both cases but the semantic annotation would be different. Since our previous recognition tool can already produce some Content MathML we hope to exploit this to aid the manual annotation.

4 Conclusion

We have outlined our current project of creating a semantic ground truth set that should help to extend mathematical formula recognition techniques to produce semantically marked-up results. We are currently preparing the basic machinery to assemble the semantic ground truth set. One obstacle is that we can not simply base it on the existing syntactic ground truth set [6] due to copyright issues. While this implies that we will have to start from scratch it will also give us the opportunity for a new, careful selection of documents to include that will obviate any future copyright or non-open access issues.

We currently envisage that the annotation with respect to subject area and general usage of symbols is fairly straightforward and can be mostly automated, while entering the exact mathematical definitions will need to be done manually and might be more difficult to complete for the entire set.

Other potential problems with the construction of a semantic ground truth data set that could occur are:

- Ambiguities in the meaning of mathematical notation can not be resolved by considering a single article of the ground truth set, but will need a background knowledge of the mathematical literature in the field.
- The current semantic formalisation given in the OpenMath content dictionaries are not sufficient for annotating the given data.
- The OpenMath formalisations are not at the right level to give semantic meaning to "human oriented" mathematics.

Since, to the best of our knowledge, no work has been done to create semantic ground truth in the context of document analysis, we currently have no comparison to assess these factors. However, we strongly believe that if our attempt is successful the work could potentially be of great impact for mathematical formula recognition.

References

1. W. Aly, S. Uchida, A. Fujiyoshi, and M. Suzuki. Statistical classification of spatial relationships among mathematical symbols. In: *Proceedings of ICDAR 2009*, pages 1350–1354. IEEE Society Press, 2009.
2. J. Baker, A. Sexton, and V. Sorge. A linear grammar approach to mathematical formula recognition from PDF. In: *Proceedings of Intelligent Computer Mathematics*, LNAI. Springer Verlag, Germany, 2009.
3. J. Baker, A. Sexton, and V. Sorge. Faithful mathematical formula recognition from PDF documents. In: *Proceedings of DAS 2010*, 2010. Forthcoming.
4. S. Buswell, O. Caprotti, D. P. Carlisle, M. C. Dewar, M. Gaëtano, and M. Kohlhase. *The OpenMath Standard*. The OpenMath Society, June 2004.
5. M. Suzuki, F. Tamari, R. Fukuda, S. Uchida, and T. Kanahori. Infty—an integrated OCR system for mathematical documents. In: *Proceedings of ACM Symposium on Document Engineering*, pages 95–104. ACM Press, 2003.
6. M. Suzuki, S. Uchida, and A. Nomura. A ground-truthed mathematical character and symbol image database. In: *Proceedings of ICDAR 2005*, pages 675–679. IEEE Society Press, 2005.
7. The American Mathematical Society. 2000 Mathematics Subject Classification, 2000. `http://www.ams.org/msc/`.
8. J. van Beusekom, F. Shafait, and T. M. Breuel. Automated OCR ground truth generation. In: *Proceedings of DAS 2008*, Sep 2008.