Genetic Programming and Databases

W. B. Langdon, Email: W.Langdon@cs.ucl.ac.uk

Dept. of Computer Science,
Internal Note IN/96/4,
University College London,
Gower Street, London WC1E 6BT, UK

Switchboard tel:	+44 (0) 171 387 7050 xtn. 3701
Dept. Office Tel:	+44 (0) 171 380 7214
Fax:	+44 (0) 171 387 1397

11 February 1996

1 Introduction

Genetic Programming (GP) is a means of automatically creating programs using a genetic algorithm (GA). Briefly a number of candidate programs are used starting points for a search through the space of possible programs looking for an acceptable program. New programs are created by random change to existing programs (known as mutation) or by combining elements of two existing programs (known as crossover).

A mechanism to decide which programs are more promising is required. This is used to decide which programs are retained and which are used to produce new programs. In genetic algorithms this is known as the fitness function.

2 Summary of Existing uses of GP and Databases

2.1 Data Mining

In the GP literature there is much reference made to "symbolic regression". This means inducing a program which describes some data. This description may be approximate. The program is a symbolic description of the data. There are many papers on symbolic regression, the following seem more directly concerned with databases as their source of data than most.

"Data mining" is more-or-less the same as symbolic regression but the emphasis is not on complete description of the data but on extracting salient (and ideally small) nuggets of information from potentially large data sources (e.g. databases) [Tac95] [HD95], [TV95].

Symbolic regression can also be viewed as GP generating a model of some process. For example successful models of journeys made within a city where constructed by a GP and validated against a database of historical data [OT94]. Other modeling work is described in [Bab95] and [MWB95].

A further use of symbolic regression is in the abstraction of formulae or models of protein shapes from large online protein databases. These have considerable commercial potential in the drug industry as a proteins shape can be used to predict is interaction with other biomolecules and so potential medicinal use.

2.2 GP and Query Optimization

Genetic programming has been used experimentally to optimize database queries both in terms of obtaining the required data more efficiently [HL94] and to decide which data to present to the user (so avoiding overloading them) [KPBS94]. Finally some work has been done on using GP to find records by example (of positive cases and possibly also negative cases) [Tel95].

2.3 Structured Data Types

My experiments have shown that GP can evolve simple data structures [Lan96b] and gain benefits by using them [Lan96a] (but these are a long way from databases).

3 Possible Future uses of GP and Databases

Database where not much discussed at the AAAI fall 1995 GP symposium [SK95] but the possible combination of GP and knowledge bases was `http://www.cs.columbia.edu/~evs/gpsym95/memory.html`. Perhaps a suitable starting point would be to see if GP could evolve programs to make sense of existing knowledge bases, i.e. treating them as read-only. Perhaps modification of knowledge bases could be investigated once read-only access had been shown to be effective.

There has been some work on using GP with Prolog.

References

[Bab95] Vladan Babovic. Genetic model induction based on experimental data. In *Proceedings of the XXVIth Congress of International Association for Hydraulics Research*, 1995.

[HD95] Les M. Howard and Donna J. D'Angelo. The GA–P: A genetic algorithm and genetic programming hybrid. *IEEE Expert*, 10(3):11–15, June 1995.

[HL94] Alex Ho and George Lumpkin. The genetic query optimizer. In John R. Koza, editor, *Genetic Algorithms at Stanford 1994*, pages 67–76. Stanford Bookstore, Stanford, California, 94305-3079 USA, December 1994.

[KPBS94] D. H. Kraft, F. E. Petry, W. P. Buckles, and T. Sadasivan. The use of genetic programming to build queries for information retrieval. In *Proceedings of the 1994 IEEE World Congress on Computational Intelligence*, pages 468–473, Orlando, Florida, USA, 27-29 June 1994. IEEE Press.

[Lan96a] W. B. Langdon. Using data structures within genetic programming. Research Note RN/96/1, UCL, Gower Street, London, WC1E 6BT, UK, January 1996.

[Lan96b] William B. Langdon. Data structures and genetic programming. In Peter J. Angeline and K. E. Kinnear, Jr., editors, *Advances in Genetic Programming 2*, chapter 20. MIT Press, Cambridge, MA, USA, 1996.

[MWB95] Ben McKay, Mark J. Willis, and Geoffrey W. Barton. Using a tree structured genetic algorithm to perform symbolic regression. In A. M. S. Zalzala, editor, *First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, GALESIA*, volume 414, pages 487–492, Sheffield, UK, September 1995. IEE.

[OT94] S. Openshaw and I. Turton. Building new spatial interaction models using genetic programming. In T. C. Fogarty, editor, *Evolutionary Computing*, Lecture Notes in Computer Science, Leeds, UK, 11-13 April 1994. Springer-Verlag.

[SK95] Eric V. Siegel and John R. Koza. Working notes AAAI-95 fall symposium series genetic programming. Technical note, The American Association for Artificial Intelligence, 10–12 November 1995. Held at MIT, Cambridge, MA, USA, 10–12 November 1995.

[Tac95] Walter Alden Tackett. Mining the genetic program. *IEEE Expert*, 10(3):28–38, June 1995.

[Tel95] Astro Teller. The discovery of algorithms for automatic database retrieval. In Justinian P. Rosca, editor, *Proceedings of the Workshop on Genetic Programming: From Theory to Real-World Applications*, pages 76–88, Tahoe City, California, USA, 9 July 1995.

[TV95] Astro Teller and Manuela Veloso. Program evolution for data mining. *The International Journal of Expert Systems*, 8(3), 1995.