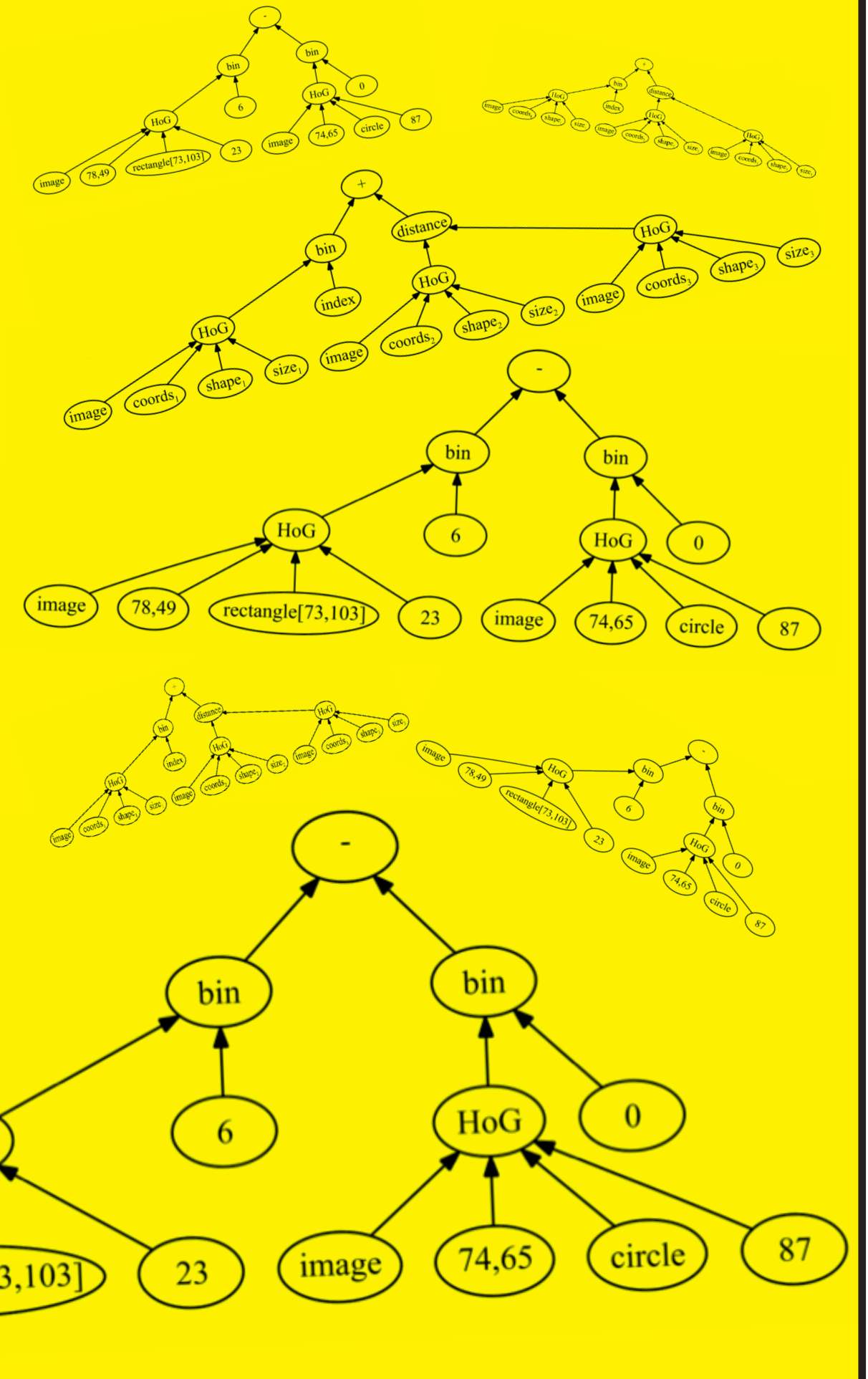
SIGEVOlution

newsletter of the ACM Special Interest Group on Genetic and Evolutionary Computation

Volume 10
Issue 1



in this issue

Evolutionary Feature Manipulation in Data Mining/Big Data

SIGEVO Studentship for ACM Turing Awards

MIT Press Blog article on new EiC of Evolutionary Computation

Editorial

As this newsletter goes to press, Spring is arriving and with it, the start of the EC conference season is about kick-off. EvoSTAR takes place in Amsterdam just after Easter from 19th-21st April, with keynotes from Kenneth De Jong (EC: Past, Present and Future) and Arthur Kordon (EC in Industry: A Realistic Overview) to look forward to. GECCO notifications have recently been released - congratulations to al those whose work has been accepted for poster or paper presentation. For anyone intending to go – whether as a presenter or an attendee – don't forget to register by the Early Deadline of 1st May.

Inside this issue, we are delighted to feature an article on Evolutionary Feature Manipulation in Data Mining contributed by Dr Bing Xue and Prof. Mengjie Zhang from the Victoria University of Wellington. The article describes a broad spectrum of issues in data-mining that can be tackled with EC techniques. If you would like the SIGEVO newsletter to feature your research in a future issue, please do get in touch! The newsletter also features a blog post from MIT Press featuring my recent appointment as EiC for Evolutionary Computation – again do get in touch if you have any questions or suggestions regarding the journal. Finally, if you are a student, don't miss the opportunity to apply for a SIGEVO scholarship to attend the ACM Turing event – full details on page 2.

Emma Hart

GECCO 2017

Early Registration Deadline: 1st May
http://gecco-2017.sigevo.org/index.html/Registration

Confirmed Keynote Speakers
Francesca Ciccarelli from King's College, London, UK
Drew Purves & Chrisantha Fernando from Google DeepMind, London, UK
Hod Lipson from Columbia University, New York, US

Call for STUDENT SCHOLARSHIP APPLICATIONS: 50 Years of the ACM Turing Award Celebration

In June this year, ACM will celebrate 50 Years of the A.M. Turing Award, which recognizes major contributions of lasting importance in computing. Through the years, it has become the most prestigious award in the field, often referred to as the "Nobel Prize of computing."

ACM will celebrate both the award and the visionaries who have received it with a conference on June 23 – 24, 2017 at the Westin St. Francis in San Francisco. ACM Turing laureates will join other award recipients and ACM experts in moderated panel discussions exploring how computing has evolved and where the field is headed. Panels include:

- Advances in Deep Neural Networks
- Moore's Law is Really Dead: What's Next?
- Quantum Computing: Far Away? Around the Corner? Or Maybe Both at the Same Time?
- Challenges in Ethics and Computing
- Augmented Reality: From Gaming to Cognitive Aids and Beyond

The program for the conference can be found at:
http://www.acm.org/awards/turing-award-50-conference

SigEVO is sponsoring four student members to attend the conference. To be eligible, applicants must be:

- a student registered at an accredited educational institution
- a student member of SigEVO as of April 14th.
  You join SigEVO at https://campus2.acm.org/public/qj/quickjoin/interim.cfm
- available to attend the event on June 23-24, 2017

Each scholarship includes:

- guaranteed conference registration (registration is free, but spaces are limited)
- 2 nights at the Westin St. Francis Hotel in San Francisco
  (Thursday, Friday June 23-24, 2017)
- up to $900 to help offset the cost of travel/subsistence.

To apply, please send:

- a short CV (up to 1 page)
- a short statement explaining why you should represent SigEVO at the event (up to ½ page)
- your ACM member number
- your previous relationship with SigEVO's GECCO and FOGA conferences, such as attendances, published papers or posters, prizes won, etc.

Send your application by email to the SigEVO secretary: juergen.branke@wbs.ac.uk.

The deadline for applications is April 14th, 2017. Winners will be selected by the SigEVO Executive Committee and notified by April 21th, 2017.

Evolutionary Feature Manipulation in Data Mining/Big Data

by Bing Xue and Mengjie Zhang

ABSTRACT

Known as the GIGO (Garbage In, Garbage Out) principle, the quality of the input data highly influences or even determines the quality of the output of any machine learning, big data and data mining algorithm. The input data which is often represented by a set of features may suffer from many issues. Feature manipulation is an effective means to improve the feature set quality, but it is a challenging task. Evolutionary computation (EC) techniques have shown advantages and achieved good performance in feature manipulation. This paper reviews recent advances on EC based feature manipulation methods in classifcation, clustering, regression, incomplete data, and image analysis, to provide the community the state-of-the-art work in the field.

INTRODUCTION

Machine learning and big data/data mining methods have shown great success in many real-world applications. Their performance highly depends on the quality of the input data, which is represented by a set of features describing different properties of a problem [52]. However, the feature set is often not of adequate quality. The common issues are high-dimensionality (i.e. ”the curse of dimensionality”) and containing redundant features, useless features or even noisy features. They often lead to poor performance, i.e. low accuracy, long processing time, over complex and incomprehensible models [24]. Feature manipulation, including feature selection and feature construction or extraction, can improve the performance [50], where feature selection is to select a subset of useful (relevant) features from originals while feature construction or extraction is to create new high-level informative features.

Evolutionary computation (EC) techniques have powerful search ability, do not make any assumptions and do not require domain knowledge, which make them promising in feature manipulation [9]. EC has been increasingly popular in feature manipulation in recent years [51]. The most widely used methods are genetic algorithms (GAs), genetic programming (GP), particle swarm optimisation (PSO), and ant colony optimisation (ACO). Most evolutionary feature manipulation work is on feature selection in classification, although GP has been used for feature construction or extraction.

There are a number of papers reviewing feature manipulation for data mining, mainly on classification and clustering [2, 24– 26, 47, 50, 51]. There exist only two surveys on evolutionary feature manipulation,[8] in 2013 and [51] in 2016, and both focus on feature selection in classification. However, evolutionary feature manipulation has been successfully applied to clustering, regression, incomplete data, and image analysis. This article aims to give an overview of evolutionary feature manipulation in the above areas.

FEATURE MANIPULATION

Feature manipulation is a challenging task, where the two main reasons are the large search space and the interactions between features. In feature selection, the size of the search space is 2^n if there are n features available in the dataset. It is even larger in feature construction or extraction since both tasks involve operators in addition to the features. There are often interactions among features, including both positive interactions and negatives interactions. Feature manipulation needs to find complementary features and utilise the positive interactions in order to improve the learning performance.

Figure 1 shows the overall system of feature manipulation. Note that for supervised learning tasks, such as classification or regression, the feature manipulation procedure should be performed on the training set only, not the whole dataset. Otherwise, there will be a feature manipulation bias issue [17].

The detailed feature manipulation procedure is shown in the orange colour box in the figure. The key steps are the feature set discovery and the feature set evaluation, where a powerful search technique and a good performance evaluation measure are needed, respectively. There are many different ways to evaluate a feature set. Depending on whether a learning algorithm is involved, feature manipulation methods can be grouped into filter, wrapper, and embedded methods [24, 25], where filters do not

select or construct features during the learning process of the algorithm. Wrappers often achieve the highest accuracy and use the longest time while filters is in the opposite side and the embedded methods are in the middle. More detailed discussions can be seen from [24, 25].

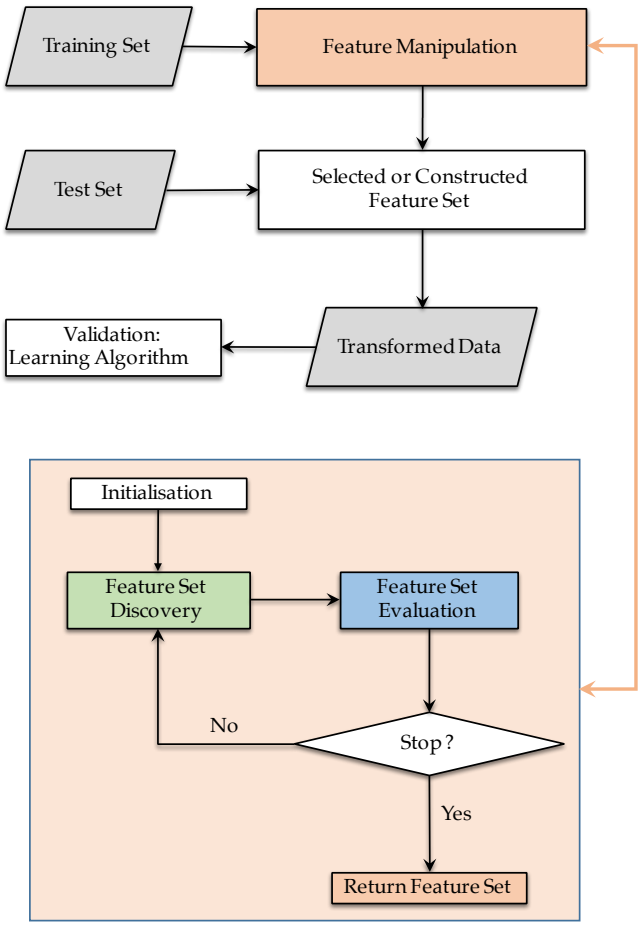


Figure 1: Overall System of Feature Manipulation

EC FOR FEATURE MANIPULATION

Classification

Classification owns the most work in EC for feature manipulation. A survey on EC for feature selection in classification was published in 2016 [51]. This section will only review typical work published in recent three years.

In EC for feature selection, PSO and GAs are the most widely used methods due to their natural representations for solving the task, but ACO and DE have become popular in recent years. In [35], a GA is used for feature selection with neural networks in credit risk assessment. The results show that the proposed method can select the most relevant features. GAs have also been used with a local search algorithm to form a memetic algorithm for feature selection in multi-label classification tasks [20] and achieve good performance.

Since the original binary PSO has some limitations, Banka and Dara [3] developed a hamming distance based binary PSO for improving feature selection performance. Nguyen et al [31] also develop a binary PSO algorithm that by introducing a sticky parameter to determine when the position should take value 1 or 0. To avoid high computational cost, a surrogate model is introduced to PSO for feature selection [33], where only a subset of instances are used in the fitness evaluation step to speed up the process without reducing the performance. In [13], PSO and GA are used with mutual information for both single-objective and multi-objective feature selection. Later, a mutual information estimation method is used in PSO for feature selection on continuous datasets [32].

Forsati et al. [10] consider the previously traversed edges in the earlier executions in ACO to adjust

the pheromone values and prevent premature convergence. This ACO algorithm successfully improves the feature selection performance. Besides the large number of features, the large number of instances is also an important issue in large-scale classification. A differential evolution (DE) based method is developed for simultaneous feature selection and instance selection, which significantly reduces the size of the dataset while maintaining the classification performance [48].

Nag and Pal [29] develop a GP based multi-objective algorithm, which can perform embedded feature selection to select relevant features and simultaneously build a classifer. Given the success of multi-objective PSO, a new method for updating and maintaining the archive in multi-objective PSO is developed in [34]. The proposed method considers the similarity between solutions in addition to the their objective values. The results show that the proposed algorithm achieved better Pareto-front solutions.

Most evolutionary feature construction methods are based on GP. Tran et al. [40] compared different GP methods for feature selection and feature construction in high-dimensional classification tasks. The results show that all the methods can reduce the number of features and improve the classification performance. The results also suggest that feature selection and construction have their own advantages and disadvantages. A good combination of them may lead to further improvement on the accuracy, speed and the interpretability of the evolved solutions.

Most GP algorithms only produce one constructed feature from the single tree representation. Tran at al. [41] develop a multi-tree based GP algorithm for feature construction, where each individual is a group of trees and each tree constructs a new feature. In this way, multiple features are constructed and the classification performance is significantly improved. Further, to reduce the redundancy among the constructed features, a feature clustering method that groups similar features to a single group is used as a preprocessing step in GP for multiple feature construction in high-dimensional data [42]

Besides the achieved success, there are still open issues in evolutionary feature manipulation in classification, such as the scalability of the algorithm, the high computational cost, and the multi-objective feature manipulation.

Clustering

EC methods have been widely used for clustering in recent years, especially swarm intelligence methods, such as PSO and artificial bee colony optimisation (ABC). However, the use of EC for feature manipulation in clustering is much less prevalent.

One of the most influential works in EC for simultaneous clustering and feature selection is the NMA_{CFS} method proposed in [38]. NMA_{CFS} uses a GA to simultaneously perform feature selection, determine the number of clusters and perform clustering. NMA_{CFS} also uses niching and local search techniques to improve the performance and consistency of the method. While the method produced good results, the datasets used in the experiments contained a small number of clusters (a maximum of 7) and a small number of features (a maximum of 30). In practice, datasets may have many more clusters and features, and how well NMA_{CFS} can scale needs further investigation. PSO has been shown to be a promising method for clustering, but there is not much work on PSO for feature selection in clustering. Lensen et al. [22] investigate the medoid and centroid representations that allow PSO to perform simultaneous clustering and feature selection. The experiments on a variety of real-world and synthetic datasets show that on several different criteria, the medoid representation can achieve superior results to the widely used centroid representation, and also out-perform other clustering methods. Later, Lensen et al. [23] further improve the method [22] by using the Silhouette Metric to estimate the number of clusters before selecting features and performing clustering. The results show that this method further improves the performance in terms of both the clustering and the number of features.

GP has also shown to be effective for performing clustering only [4, 30], but there is not much work on using GP for feature selection or construction in clustering.

Compared with classification, there is much less work on EC based feature selection or construction in clustering, since the task is much more difficult. The two important measures in clustering are hard to determine for different types of clustering methods. One is the measure used *during the clustering*

process to determine which cluster an instance belongs to, such as distance measures. The other measure is to evaluate how good the formed cluster is, such as to measure the compactness, connectedness, and separability or exclusiveness. Since there are many different types of clusters, such as hyper-spherical clusters and non-hyper-spherical clusters, there is no performance measure appropriate for evaluating all types of clusters.

Regression

Regression is another popular data mining task, which builds a model by estimating the relationships among features/variables. Symbolic regression is a special type of regression tasks that needs to build the regression model and simultaneously optimise the parameters of the model, which is the primary application of GP.

EC techniques have been used in regression tasks, e.g. GAs have been used to optimise the kernel function and parameters in support of vector regression [49], a PSO based regression approach is proposed to generate fuzzy nonlinear regression models [5], and artificial bee colony has been used for symbolic regression [15]. However, there are a limited number of papers on EC for feature manipulation in regression.

Most of the work on EC for feature manipulation in regression is GP for feature construction in symbolic regression, since GP has a built-in ability for feature construction. Back in the 1990s, the automatically defined functions introduced by Koza [18] can be seen as a feature construction method for symbolic regression. In [28], the concept of latent variable, which is a linear combination of the input variables, i.e. a constructed feature, is introduced into symbolic regression to transform the input space into a reduced-dimensionality space. Kattan et al. [16] also uses GP for feature construction, and the constructed features are used to improve the generalisation of a set of regression methods.

Recently, Chen et al. [6] develop an embedded feature construction method for improving the performance of symbolic regression. Later, Chen et al. [7] develop a permutation based feature selection method to select a subset of relevant features, which successfully improves the generalisation performance of GP for symbolic regression on a number of high-dimensional datasets. Besides GP, GAs have also been used for feature selection for partial least squares regression, and the results show that the regression performance is improved by selecting only a small number of important features [19].

High-dimensional complex (symbolic) regression tasks appear more and more in important real-world problems. It is needed and there are opportunities for developing EC based effective feature manipulation methods to improve the regression performance.

Incomplete Data

Missing values are a common issue in many real-world datasets. Many machine learning algorithms cannot be directly applied to incomplete data. Imputation based methods are one of the most powerful approaches to handling missing data, which use the complete features to build a regression model for predicting the missing values.

Since GP is good at symbolic regression, it has been used to impute missing values in incomplete data with good success, and the built-in feature selection and feature construction ability in GP also helps improve the performance [43]. Tran et al. [43] applies GP as a regression method to impute missing values, and the results show that the GP method achieves better performance than other imputation methods. In [44], GP is used for constructing multiple high-level features by introducing new interval functions to cope with incomplete data. The results show that the proposed method can substantially improve the accuracy and reduce the complexity of the classifiers.

The widely used filter feature selection measure, mutual information, is investigated in [36] to select relevant features on incomplete datasets, and show good results. A PSO based wrapper feature selection method is proposed in [45], where C4.5 is used to evaluate the classification accuracy of the selected features. The results show that by using feature selection, the classification accuracy of the incomplete data is improved. Later, Tran et al. [46] propose a PSO based wrapper feature selection method and the selected features are used with the bagging ensemble method for classification. The results show that the combination of feature selection and bagging improve the classification accuracy

and reduce the complexity of the learned classifiers.

Evolutionary feature manipulation has only been used on incomplete data in recent years. More work needs to be done, since incomplete data itself leads to poor feature set, and the increase of the dataset size makes it even worse.

Image Analysis

Feature manipulation plays a key role in image Analysis, especially feature extraction or feature construction. Among EC technique, GP is the most widely used method. Some early work can be seen from [51]. This section will introduce typical work published in recent three years.

Ryan et al. [37] propose a general GP approach to image classification, where each image is segmented to sub-images. Features are extracted from sub-images and the most promising features are used in GP for classification. A hierarchical feature construction method, which includes constructing features from both the primitive image processing filters and the evolved filters, is used to create high-level features in GP for classification [39]. In [27], GP is used to evolve motion feature descriptors on a population of primitive operators so that scale and shift invariant features can be effectively extracted. The results show that the proposed method significantly outperforms other types of features, either hand-designed or machine-learned.

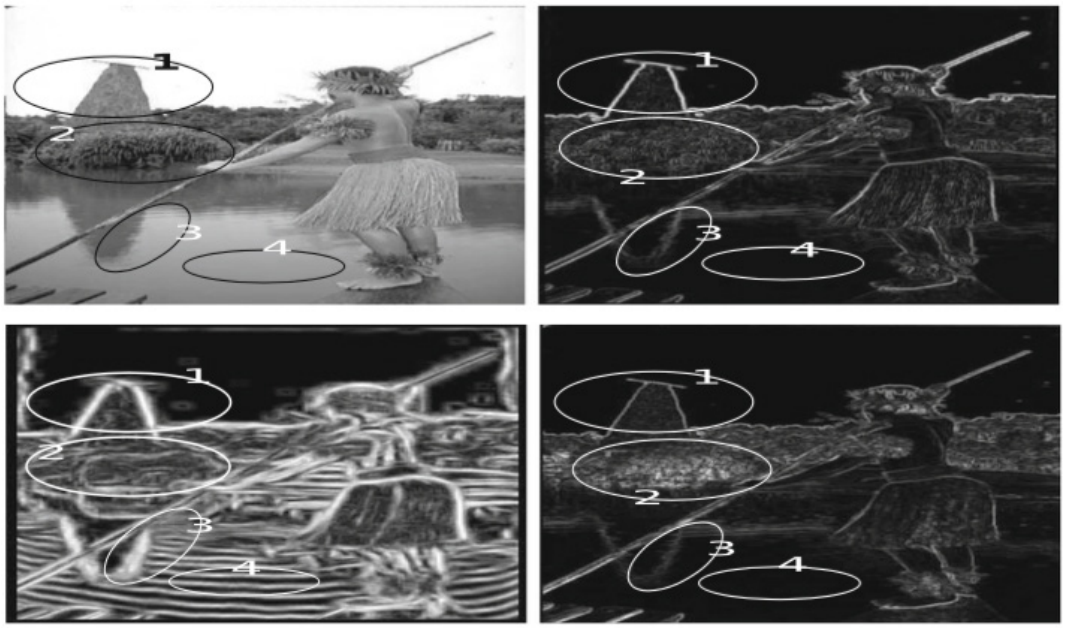


Figure 2: Image analyses from [12]

Lensen at al. [21] designed a new GP tree structure which can select important regions in the images, extract features from regions, and further construct and select the features for image classification. Experiments show that the proposed GP system achieved better classification accuracy than other methods. GP is also used to evolve an image descriptor [1], where a special function node is developed to allow the extraction of pixel values to features. The results show that the proposed method outperforms other GP and non-GP methods. Later, the structure of the algorithm [1] is further developed in transfer learning to solve difficult image classification tasks [14]. The results show that the GP method with transfer learning can solve difficult tasks that most other algorithms cannot solve. Fu et al. [11, 12] propose a GP based approach which can search the pixel space to automatically construct features for edge detection. Further, a GP based feature construction method is developed to use the estimated distribution of observations for constructing features for improving the edge detection performance [12].

Feature manipulation is a key step in almost all tasks in image analysis, but very challenging. There have been a large number of non-EC based feature manipulation methods, which can be used together with EC methods to further improve the performance.

SUMMARY

Evolutionary feature manipulation is an emerging research area with a lot of challenges and opportunities. Due to the complexity and characteristics of the data in different problems, a general method does not work well. Since new challenges and opportunities appear, new approaches are still needed to utilise the advantages of different EC methods to solve feature manipulation tasks in problems in data mining.

ACKNOWLEDGMENT

This article describes evolutionary feature manipulation undertaken at the the Evolutionary Computation Research Group (ECRG), Victoria University of Wellington, New Zealand. Led by Prof Mengjie Zhang, ECRG is an interdisciplinary research group, crossing computer science, software engineering, network engineering, electronic engineering, statistics, operations research and biology. It consists of over 10 staff members and over 20 PhD students carrying out research related to evolutionary computation and machine learning. The Group now has a number of strategic research directions: the first is evolutionary feature manipulation (selection, construction, extraction) and big dimensionality reduction in data mining (classification, clustering, regression, missing data, etc.), the second is evolutionary computer vision and image processing, and the third is evolutionary scheduling and combinatorial optimisation (including job shop/production scheduling, routing, web service composition, cloud/grid computing scheduling, etc.).

The Group has involved the organisations of major EC conferences such as GECCO, IEEE CEC, EvoStar and SEAL, and is an Associate Editor or member of the Editorial Board for major EC journals such as IEEE TEVC, ECJ, IEEE TCYB, GPEM, IEEE TETCI, and Applied Soft Computing. The Group has been chairing the IEEE CIS Evolutionary Computation Technical Committee, Emergent Technology Technical Committee, and Intelligent Systems Applications Technical Committee, and (co) chairing Task Forces on Evolutionary Feature Selection and Construction, Evolutionary Scheduling and Combinatorial Optimisation, and Evolutionary Computer Vision and Image Processing. More details about ECRG can be seen from http://ecs.victoria.ac.nz/Groups/ECRG/.

This work was supported in part by the Marsden Fund of New Zealand Government under Contracts VUW1209, VUW1509 and 16-VUW-111, and the University Research Fund of Victoria University of Wellington under contracts 209861/3580, 209862/3580 and 213150/3663, and the Huawei NZ Industry Program under grant number E2880/3663.

REFERENCES

[1] H. Al-Sahaf, A. Al-Sahaf, B. Xue, M. Johnston, and M. Zhang. 2017. Automatically Evolving Rotation-Invariant Texture Image Descriptors by Genetic Programming. IEEE Transactions on Evolutionary Computation 21, 1 (2017), 83–101.

[2] Salem Alelyani, Jiliang Tang, and Huan Liu. 2013. Feature Selection for Clustering: A Review. In Data Clustering: Algorithms and Applications. 29–60.

[3] Haider Banka and Suresh Dara. 2015. A Hamming distance based binary particle swarm optimization (HDBPSO) algorithm for high dimensional feature selection, classification and validation. Pattern Recognition Letters 52 (2015), 94–100.

[4] Neven Boric and Pablo A. Este´vez. 2007. Genetic programming-based clustering using an information theoretic fitness measure. In Proceedings of the IEEE Congress on Evolutionary Computation (CEC). 31–38.

[5] K. Y. Chan, T. S. Dillon, and C. K. Kwong. 2011. Modeling of a Liquid Epoxy Molding Process Using a Particle Swarm Optimization-Based Fuzzy Regression Approach. IEEE Transactions on Industrial Informatics 7, 1 (2011), 148–158.

[6] Qi Chen, Mengjie Zhang, and Bing Xue. 2016. Genetic Programming with Embedded Feature Construction for High-Dimensional Symbolic Regression. In the 20th Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES). Springer, 87–102.

[7] Qi Chen, Mengjie Zhang, and Bing Xue. 2017. Feature Selection to Improve Generalisation of Genetic Programming for High-Dimensional Symbolic Regression. IEEE Transactions on Evolutionary Computation 99, 1 (2017), to appear.

[8] Beatriz de la Iglesia. 2013. Evolutionary computation for feature selection in classification problems. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 3, 6 (2013), 381–407.

[9] Agoston E Eiben and Jim Smith. 2015. From evolutionary computation to the evolution of things. Nature 521, 7553 (2015), 476–482.

[10] Rana Forsati, Alireza Moayedikia, Richard Jensen, Mehrnoush Shamsfard, and Mohammad Reza Meybodi. 2014. Enriched ant colony optimization and its application in feature selection. Neurocomputing 142 (2014), 354 – 371.

[11] W. Fu, M. Johnston, and M. Zhang. 2014. Low-Level Feature Extraction for Edge Detection Using Genetic Programming. IEEE Transactions on Cybernetics 44, 8 (2014), 1459–1472.

[12] Wenlong Fu, Mark Johnston, and Mengjie Zhang. 2015. Distribution-based invariant feature construction using genetic programming for edge detection. Soft Computing 19, 8 (2015), 2371–2389.

[13] Min Han and Weijie Ren. 2015. Global mutual information-based feature selection approach using single-objective and multi-objective optimization. Neurocomputing 168 (2015), 47–54.

[14] M. Iqbal, B. Xue, H. Al-Sahaf, and M. Zhang. 2017. Cross-Domain Reuse of Extracted Knowledge in Genetic Programming for Image Classi cation. IEEE Transactions on Evolutionary Computation 99 (2017). DOI:http://dx.doi.org/10. 1109/TEVC.2017.2657556

[15] Dervis Karaboga, Celal Ozturk, Nurhan Karaboga, and Beyza Gorkemli. 2012. Artificial bee colony programming for symbolic regression. Information Sciences 209 (2012), 1 – 15.

[16] Ahmed Kattan, Michael Kampouridis, and Alexandros Agapitos. 2014. Generalisation Enhancement via Input Space Transformation: A GP Approach. Springer Berlin Heidelberg, 61–74.

[17] Ron Kohavi and George H. John. 1997. Wrappers for feature subset selection. Artificial Intelligence 97 (1997), 273–324.

[18] John R. Koza. 1992. Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge, MA, USA.

[19] Riccardo Leardi and Amparo Lupiez Gonzlezb. 1998. Genetic algorithms applied to feature selection in PLS regression: how and when to use them. Chemometrics and Intelligent Laboratory Systems 41, 2 (1998), 195 – 207.

[20] Jaesung Lee and Dae-Won Kim. 2015. Memetic feature selection algorithm for multi-label classification. Information Sciences 293 (2015), 80 – 96.

[21] Andrew Lensen, Harith Al-Sahaf, Mengjie Zhang, and Bing Xue. 2016. Genetic Programming for Region Detection, Feature Extraction, Feature Construction and Classification in Image Data. In European Conference on Genetic Programming. Vol. 9594. Springer International Publishing, 51–67.

[22] Andrew Lensen, Bing Xue, and Mengjie Zhang. 2016. Particle swarm optimisation representations for simultaneous clustering and feature selection. In IEEE Symposium Series on Computational Intelligence (SSCI). 1–8.

[23] Andrew Lensen, Bing Xue, and Mengjie Zhang. 2017. Using Particle Swarm Optimisation and the Silhouette Metric to Estimate the Number of Clusters, Select Features, and Perform Clustering. In Proceeding of the 20th European Conference on the Applications of Evolutionary Computation. Springer, to appear.

[24] Huan Liu, Hiroshi Motoda, Rudy Setiono, and Zheng Zhao. 2010. Feature Selection: An Ever Evolving Frontier in Data Mining. In Feature Selection for Data Mining (JMLR Proceedings), Vol. 10. JMLR.org, 4–13.

[25] Huan Liu and Lei Yu. 2005. Toward integrating feature selection algorithms for classification and clustering. IEEE Transactions on Knowledge and Data Engineering 17, 4 (2005), 491–502.

[26] Huan Liu and Zheng Zhao. 2009. Manipulating Data and Dimension Reduction Methods: Feature Selection. In Encyclopedia of Complexity and Systems Science. Springer, 5348–5359.

[27] L. Liu, L. Shao, X. Li, and K. Lu. 2016. Learning Spatio-Temporal Representations for Action Recognition: A Genetic Programming Approach. IEEE Transactions on Cybernetics 46, 1 (2016), 158–170.

[28] Trent McConaghy. 2010. Latent variable symbolic regression for high-dimensional inputs. Springer.

[29] K. Nag and N.R. Pal. 2016. A Multiobjective Genetic Programming-Based Ensemble for Simultaneous Feature Selection and Classi cation. IEEE Transactions on Cybernetics 46 (2016), 499–510.

[30] Enrique Naredo and Leonardo Trujillo. 2013. Searching for novel clustering programs. In Genetic and Evolutionary Computation Conference (GECCO). 1093– 1100.

[31] Bach Hoai Nguyen, Bing Xue, and Peter Andreae. 2016. A Novel Binary Particle Swarm Optimization Algorithm and Its Applications on Knapsack and Feature Selection Problems. In the 20th Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES). Springer, 319–332.

[32] Hoai Bach Nguyen, Bing Xue, and Peter Andreae. 2016. Mutual information for feature selection: estimation or counting? Evolutionary Intelligence 9, 3 (2016), 95–110. Conference on the Applications of Evolutionary Computation. Springer International Publishing, to appear.

[33] Hoai Bach Nguyen, Bing Xue, and Peter Andreae. 2017. Surrogate-model based Particle Swarm Optimisation with Local Search for Feature Selection in Classification. In Proceeding of the 21th European Conference on the Applications of Evolutionary Computation. Springer International Publishing, to appear.

[34] Hoai Bach Nguyen, Bing Xue, Ivy Liu, Peter Andreae, and Mengjie Zhang. 2016. New mechanism for archive maintenance in PSO-based multi-objective feature selection. Soft Computing (2016), 1–20.

[35] Stjepan Oreski and Goran Oreski. 2014. Genetic algorithm-based heuristic for feature selection in credit risk assessment. Expert Systems with Applications 41, 4, Part 2 (2014), 2052 – 2064.

[36] Wenbin Qian and Wenhao Shu. 2015. Mutual information criterion for feature selection from incomplete data. Neurocomputing 168 (2015), 210–220.

[37] Conor Ryan, Jeannie Fitzgerald, Krzysztof Krawiec, and David Medernach. 2015. Image Classification with Genetic Programming: Building a Stage 1 Computer Aided Detector for Breast Cancer. Springer International Publishing, 245–287.

[38] Weiguo Sheng, Xiaohui Liu, and Mike Fairhurst. 2008. A niching memetic algorithm for simultaneous clustering and feature selection. IEEE Transactions on Knowledge and Data Engineering 20, 7 (2008), 868–879.

[39] M. Suganuma, D. Tsuchiya, S. Shirakawa, and T. Nagao. 2016. Hierarchical feature construction for image classification using Genetic Programming. In IEEE International Conference on Systems, Man, and Cybernetics (SMC). 1423– 1428.

[40] Binh Tran, Bing Xue, and Mengjie Zhang. 2015. Genetic programming for feature construction and selection in classification on high-dimensional data. Memetic Computing 8, 1 (2015), 3–15.

[41] Binh Tran, Mengjie Zhang, and Bing Xue. 2016. Multiple feature construction in classification on high-dimensional data using GP. In IEEE Symposium Series on Computational Intelligence (SSCI). 1–8.

[42] Binh Ngan Tran, Bing Xue, and Mengjie Zhang. 2017. Using Feature Clustering for GP-Based Feature Construction on High-Dimensional Data. Springer International Publishing, to appear.

[43] Cao Truong Tran, Mengjie Zhang, and Peter Andreae. 2016. A Genetic Programming-Based Imputation Method for Classification with Missing Data. Springer International Publishing, 149–163.

[44] Cao Truong Tran, Mengjie Zhang, Peter Andreae, and Bing Xue. 2016. Directly Constructing Multiple Features for Classification with Missing Data using Genetic Programming with Interval Functions. In Genetic and Evolutionary Computation Conference (GECCO).

[45] Cao Truong Tran, Mengjie Zhang, Peter Andreae, and Bing Xue. 2016. Improving performance for classification with incomplete data using wrapper-based feature selection. Evolutionary Intelligence 9, 3 (2016), 81–94.

[46] Cao Truong Tran, Mengjie Zhang, Peter Andreae, and Bing Xue. 2017. Bagging and Feature Selection for Classification with Incomplete Data. In Proceeding of the 20th European Conference on the Applications of Evolutionary Computation. Springer, to appear.

[47] Jorge R. Vergara and Pablo A. Estevez. 2014. A review of feature selection methods based on mutual information. Neural Computing and Applications 24, 1 (2014), 175–186.

[48] Jiaheng Wang, Bing Xue, Xiaoying Gao, and Mengjie Zhang. 2016. A Di erential Evolution Approach to Feature Selection and Instance Selection. Springer Interna- tional Publishing, 588–602. DOI:http://dx.doi.org/10.1007/978- 3- 319- 42911- 3

[49] Chih-Hung Wu, Gwo-Hshiung Tzeng, and Rong-Ho Lin. 2009. A Novel hybrid genetic algorithm for kernel function and parameter optimization in support vector regression. Expert Systems with Applications 36, 3, Part 1 (2009), 4725 – 4735.

[50] Bing Xue and Mengjie Zhang. 2016. Evolutionary computation for feature manipulation: Key challenges and future directions. In 2016 IEEE Congress on Evolutionary Computation (CEC). 3061–3067.

[51] Bing Xue, Mengjie Zhang, Will N. Browne, and Xin Yao. 2016. A Survey on Evolutionary Computation Approaches to Feature Selection. IEEE Transactions on Evolutionary Computation 20, 4 (2016), 606–626.

[52] Yiteng Zhai, Yew-Soon Ong, and I.W. Tsang. 2014. The Emerging ”Big Dimensionality”. IEEE Computational Intelligence Magazine 9, 3 (2014), 14–26.

BIOGRAPHIES

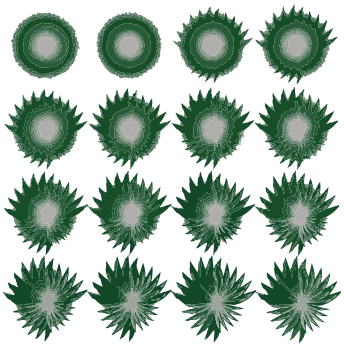
Bing Xue is currently a Senior Lecturer in School of Engineering and Computer Science at Victoria University of Wellington. Her research focuses mainly on evolutionary computation, feature selection, feature construction, multi-objective optimisation, data mining and machine learning.

Mengjie Zhang is currently Professor of Computer Science, Head of the Evolutionary Computation Research Group, and the Associate Dean (Research and Innovation) in the Faculty of Engineering at the Victoria University of Wellington, Wellington, New Zealand, where he His current research interests include evolutionary computation, particularly genetic programming, particle swarm optimization, and learning classifier systems with application areas of image analysis, multiobjective optimization, feature selection and reduction, job shop scheduling, and transfer learning.

MIT Press welcomes Emma Hart as the new EiC of Evolutionary Computation

Reproduced with kind permission from the MIT Press Blog
https://mitpress.mit.edu/blog/welcome-emma-hart

The New Year welcomes Emma Hart to the helm of Evolutionary Computation. She takes over the role of Editor-in-Chief from Hans-Georg Beyer (who had assumed the role himself in 2010). Professor Hart is the Director of the Centre for Algorithms, Visualisation and Evolving Systems at Edinburgh Napier University and her research is focused on biologically inspired computing. Professor Hart answered a few questions for us about her work with the journal and her hopes for its future.



You've published a number of articles in Evolutionary Computation (and various other journals) over the years. How did you move from contributor to editor?

I think it has helped to take as many opportunities as possible to be actively involved in the EC community—this has enabled me to get to know a lot of people across the world. I've moved gradually from chairing workshops in smaller conferences to more prominent roles such as Track Chair at GECCO (Genetic and Evolutionary Computation Conference), Technical Chair at CEC (IEEE Congress on Evolutionary Computation), and General Chair of PPSN (International Conference on Parallel Problem Solving from Nature) in 2016. I also serve on the SIGEVO (ACM Special Interest Group on Genetic and Evolutionary Computation) board and edit the SIGEVO newsletter, which has helped raise my profile. Of course, acting as an Associate Editor of Evolutionary Computation for several years has been incredibly useful in getting a better understanding of how the journal works!

What's your history with/interest in the content covered by EC?

Actually, I started my academic life as chemist. During my final year of undergraduate study, I undertook a research project that involved writing a computer program to model gas solubility in blood. This was my first real experience of using computing in science and I enjoyed it much more than mixing chemicals in labs! I went on to the University of Edinburgh to do a postgraduate course in Artificial Intelligence, and by chance, took a course in Evolutionary Computing taught by Prof. Peter Ross. The idea of combining computing with concepts from physical and biological sciences really appealed to me. A Master's dissertation led on to a PhD and then a permanent academic post, and I've never looked back! I work in a number of bio-inspired areas, particularly immune-inspired computing and evolutionary algorithms, and still love the opportunities it gives to continually learn and apply new ideas.

What are your hopes for the journal?

To make it the leading journal in the field! I hope to build on the excellent work of the previous editor Hans-Georg Beyer, which improved the journal's impact factor to its current standing of 3.600 to increase this even further. I'd like to expand the readership by raising the profile of the journal in other fields that have connections to EC (and other nature-inspired algorithms), for example, encouraging interdisciplinary articles that integrate EC with other approaches. I'd also like to encourage more articles that reflect uses of EC in real-world applications. I hope to see Evolutionary Computation pushing boundaries in reproducible and open science, encouraging publication of code and data alongside papers, which I

believe will help increase its reputation.

Any favorite articles from Evolutionary Computation?

I have to mention two articles from Stephanie Forrest's early work in Artificial Immune Systems that really inspired my own work in the field. Stephanie was one of the first pioneers in this field, and it was after reading her work that I decided to do a PhD in this area. I wouldn't be taking on the EiC role today if it hadn't been for these papers!

Using Genetic Algorithms to Explore Pattern Recognition in the Immune System

Stephanie Forrest, Brenda Javornik, Robert E. Smith, Alan S. Perelson

Evolutionary Computation 1:3 (Fall 1993)

Architecture for an Artificial Immune System

Steven A. Hofmeyr, Stephanie Forrest

Evolutionary Computation 8:4 (Winter 2000)

Swarm Robotics PhD Studentship

A funded PhD positon is available in the Service Robotics Group at the University of Luebeck, Germany in the topics of swarm intelligence, swarm robotics, swarm modeling, and evolutionary robotics.

The prospective candidate should have:

- a Master's degree in Computer Science, Electrical Engineering, or Physics,
- basic knowledge about machine learning (e.g., neural networks, evolutionary computation), or mathematical models of collective behavior
- programming experience
- English speaking and writing skills (German not required).

The research will be done within the new Service Roboticcs group of **Prof. Dr. Heiko Hamann** which is currently set up. Our group is and will be international.

The University of Luebeck is located in the very North of Germany close to Hamburg.Other attractive cities, such as Copenhagen or Berlin, are reachable within a few hours. The recreational opportunities are vast, especially because of Luebeck's location at the Baltic Sea.

The positon is fully funded; the salary is based on pay scale "E13" (roughly 44k EUR per year).

The initial contract will be limited to three years, a contract extension is optional.

Application deadline: April 10, 2017 start: May 2017 or later.

Please indicate your interest to me (hamann@iti.uni-luebeck.de), also to learn about the actual application procedure.

About this newsletter

SIGEVOlution is the newsletter of SIGEVO, the ACM Special Interest Group on Genetic and Evolutionary Computation. To join SIGEVO, please follow this link: [WWW]

Contributing to SIGEVOlution

We solicit contributions in the following categories:

Art: Are you working with Evolutionary Art? We are always looking for nice evolutionary art for the cover page of the newsletter.

Short surveys and position papers: We invite short surveys and position papers in EC and EC related areas. We are also interested in applications of EC technologies that have solved interesting and important problems.

Software: Are you are a developer of an EC software and you wish to tell us about it? Then, send us a short summary or a short tutorial of your software.

Lost Gems: Did you read an interesting EC paper that, in your opinion, did not receive enough attention or should be rediscovered? Then send us a page about it.

Dissertations: We invite short summaries, around a page, of theses in EC-related areas that have been recently discussed and are available online.

Meetings Reports: Did you participate to an interesting EC-related event? Would you be willing to tell us about it? Then, send us a short summary, around half a page, about the event.

Forthcoming Events: If you have an EC event you wish to announce, this is the place.

News and Announcements: Is there anything you wish to announce, such as an employment vacancy? This is the place.

Letters: If you want to ask or to say something to SIGEVO members, please write us a letter!

Suggestions: If you have a suggestion about how to improve the newsletter, please send us an email.

Contributions will be reviewed by members of the newsletter board.

We accept contributions in LATEX, MS Word, and plain text.

Enquiries about submissions and contributions can be emailed to editor@sigevolution.org

All the issues of SIGEVOlution are also available online at: www.sigevolution.org

Notice to Contributing Authors to SIG Newsletters

By submitting your article for distribution in the Special Interest Group publication, you hereby grant to ACM the following non-exclusive, perpetual, worldwide rights:

- to publish in print on condition of acceptance by the editor
- to digitize and post your article in the electronic version of this publication
- to include the article in the ACM Digital Library
- to allow users to copy and distribute the article for noncommercial, educational or research purposes

However, as a contributing author, you retain copyright to your article and ACM will make every effort to refer requests for commercial use directly to you.

Editor: **Emma Hart**
Associate Editors: **Darrell Whitley**, **Una-May O-Reilly**, **James McDermott**, **Gabriela Ochoa**
Design & Layout: **Callum Egan**