

WEB 2.0 ENVIRONMENTAL SCANNING AND ADAPTIVE DECISION SUPPORT FOR BUSINESS MERGERS AND ACQUISITIONS

Raymond Y. K. Lau, and Stephen S. Y. Liao

Department of Information Systems, City University of Hong Kong, Tat Chee Avenue, Kowloon,
Hong Kong SAR, CHINA {raylau@cityu.edu.hk} {issliao@cityu.edu.hk}

K. F. Wong

Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong,
Shatin, N.T., Hong Kong SAR, CHINA {kfwong@se.cuhk.edu.hk}

Dickson K. W. Chiu

Department of Information Systems, City University of Hong Kong, Tat Chee Avenue, Kowloon,
Hong Kong SAR, CHINA {kwchiu@cityu.edu.hk}

Appendix A

A Hierarchical Coevolution Genetic Algorithm for Adaptive M&A Decision Support

Figure A1 depicts an overview of the coevolution process controlled by the proposed HCGA algorithm. The Level II (high-level) population consists of individuals representing the feasible values of the application parameters. The Level I (low-level) populations represent two sets of system parameters (i.e., two species). One of the low-level populations represents the system parameters that drive the sentiment analysis process, whereas the second low-level population represents the system parameters that control the business relation mining process. The HCGA algorithm controls the evolutionary processes among all of the populations. At the end of the coevolution process, a set of near-optimal application parameter values and low-level NLP features (e.g., the use of specific sentiment lexicons) with respect to a particular M&A situation are obtained to refine the M&A target scoring function (i.e., the decision support mechanism).

The fitness of an individual from each population is assessed in terms of a high-level measure, that is, the system's performance on M&A target recommendations. In particular, the precision regarding making top 10 recommendations (i.e., P@10) is assessed to determine an individual's fitness. The rationale for using P@10 (Ounis et al. 2008) as the fitness function instead of other quantitative measures such as ROI is that it may take years for an M&A deal to generate the anticipated ROI, and thus this kind of information may not be available to assess the system's performance. The P@10 score is computed with respect to a set of training M&A cases retrieved from the real-world or recommended by M&A experts. The fitness of an individual (chromosome) c is defined by the following:

$$fitness(c) = \frac{TruePositive_{top-10}}{FalsePositive_{top-10} + TruePositive_{top-10}}$$

where $TruePositive_{top-10}$ and $FalsePositive_{top-10}$ represent the true positive and false positive recommendations at the top 10 positions, respectively. In other words, the system's M&A scoring module (i.e., the decision support model) is invoked to generate the top 10 recommendations whenever the fitness of a chromosome needs to be evaluated.

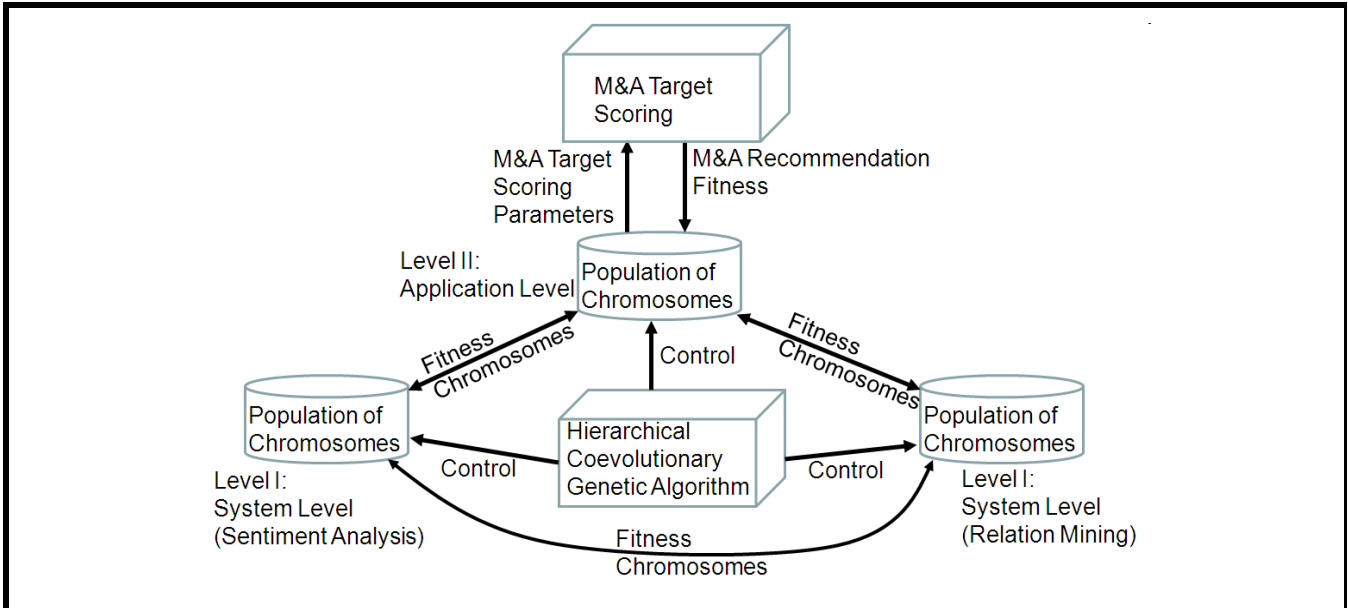


Figure A1. Overview of the Hierarchical Coevolutionary Process

When an HCGA-coordinated evolutionary process takes place, it always begins with the high-level population. To assess the fitness of each individual in the Level II population, the fittest individual from each Level I population is passed to the Level II population. In other words, the sets of system parameters and the set of application parameters are combined to drive the eventual M&A target scoring process. The P@10 score of the resulting M&A recommendations is used to assess the fitness of each individual. Similarly, the fittest individual of the Level II population is passed to a Level I population when the fitness of an individual in a Level I population is assessed. In addition, the fittest individual is exchanged between the two Level I populations. These interactions among the species (populations) drive the coevolution process. The advantage of the coevolutionary approach is that a large solution space can be divided into subspaces for a parallel and diversified search, which improves both the efficiency and the effectiveness of the heuristic search process (Delgado et al. 2004; Olsson 2001).

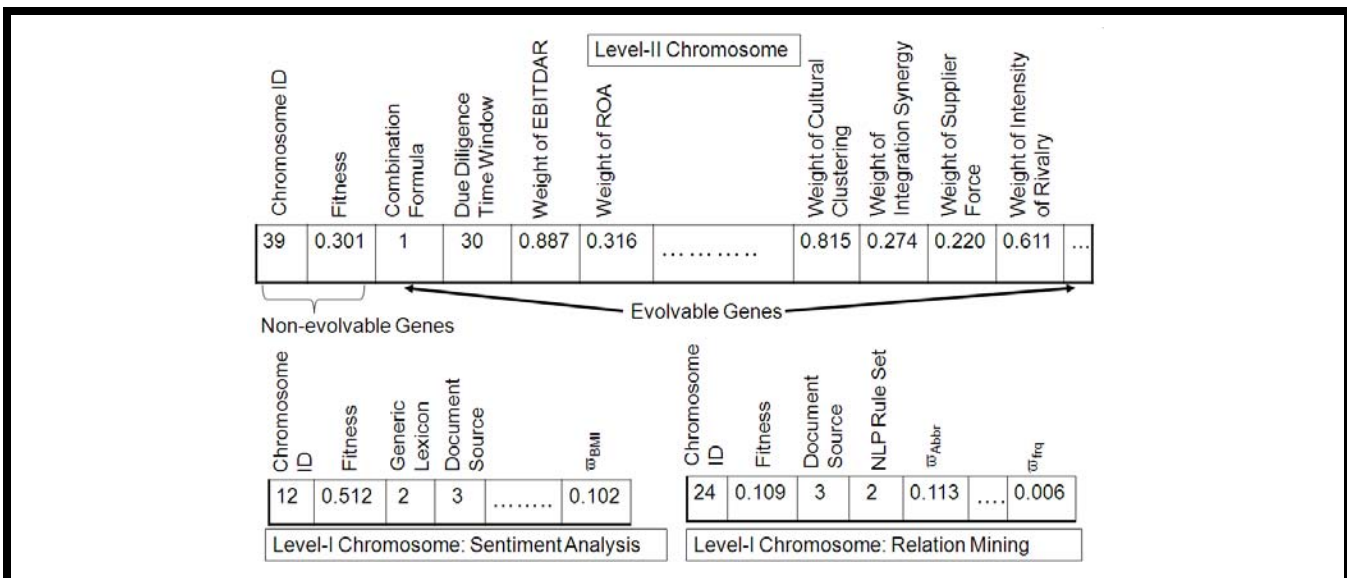


Figure A2. Gene Encoding of the HCGA Algorithm

```

Algorithm HCGA(Avgfit, Maxgen, NP, Psize[x], pm[x], pc[x], we[x])
Inputs:      Avgfit      /* the average fitness threshold to terminate the algorithm
                Maxgen     /* the maximum no. of generations re-produced
                NP        /* no. of populations
                Psize[x]   /* the population size of a specific population x
                pm[x]     /* the mutation probability of a specific population x
                pc[x]     /* the crossover probability of a specific population x
                we[x]     /* the elitism rate of a specific population x
Output:     cout[x]   /* the fittest chromosomes returned
Main Procedure:
1. gen := 1; /* initialize the generation index
2. for x = 1 to NP
3. randomly assign feasible value to each gene of each chromosome according to population
   size Psize[x];
4. end
5. exchange chromosomes among high-level and low-level populations;
6. compute the fitness of each chromosome in each population P[x]gen for generation gen
   according to the fitness function;
7. while gen <= Maxgen and Fitness(P[x]gen) of each population < Avgfit
8. for x = 1 to NP
9. select we[x] percentage of fittest chromosomes of a population P[x]gen to build
   P[x]gen+1 starting from the high-level population;
10. while size(P[x]gen+1) <= Psize[x]
11. apply roulette-wheel selection to select two chromosomes to produce P[x]gen+1;
12. generate a random number r for each pair of selected chromosome;
13. if r < pc[x], apply two point crossover to the selected chromosomes;
14. generate a random number r for each evolvable gene of each chromosome;
15. if r < pm[x], apply mutation to the evolvable gene of each selected chromosome;
16. end
17. end
18. exchange the fittest chromosomes among high-level and low-level populations;
19. compute the fitness of each chromosome in each population P[x]gen+1 for generation gen+1
   according to the fitness function;
20. gen := gen + 1;
21. end
22. for x = 1 to NP
23. return the fittest chromosome cout[x] from population x;
24. end
25. exit

```

Figure A3. The HCGA Algorithm

Figure A2 shows the gene encoding of individuals in both Level I and Level II populations. Basically, a Level II chromosome carries the genes representing various application parameters, such as the scoring formula for each due diligence dimension, the time window of due diligence, the weights of various financial metrics, the weights of various sociocultural factors, the weights of various factors related to business fitness, and the weights of various factors related to competitive advantage of an M&A target. All together, there are 36 evolvable genes of each Level II chromosome. A Level I chromosome carries the genes related to various system parameters and low-level NLP features (e.g., a specific generic sentiment lexicon to be used) controlling the sentiment analysis process or the business relation mining process. There are 9 evolvable genes of the Level I chromosome representing the system parameters controlling sentiment analysis, and 12 that are encoding the parameters for business relation mining. Decimal gene encoding is used for both Level I and Level II chromosomes in our system (Goldberg 1989; Lau et al. 2006).

Figure A3 shows the computational details of the HCGA algorithm. At the beginning of a coevolutionary process (i.e., the first generation), the chromosomes at each level are initialized by randomly assigning feasible values to each evolvable gene. Then, the fitness of each individual (chromosome) in a population is evaluated starting from level II. During fitness evaluation, the fittest individuals among the populations are exchanged according to the interaction pattern depicted in Figure A1. For the first generation, a randomly chosen individual from each Level I population is passed to the Level II population for fitness computation because the fitness of each individual of a Level I population has not yet been determined. For the subsequent generations, only the fittest individuals are exchanged among the populations. For each population, standard genetic operators such as selection, crossover, and mutation are applied to reproduce individuals of the next generation (Goldberg 1989; Huang et al. 2009; Lau et al. 2006). Moreover, the elitism rate w_e (i.e., the elitism factor) is applied to directly transfer a certain percentage of the fittest chromosomes from the current generation to the next generation P^{t+1} in order to retain the fittest chromosomes that represent good solutions for a problem domain (Goldberg 1989; Lau et al. 2006).

Roulette wheel selection (Goldberg 1989; Huang et al. 2009; Maiti and Maiti 2008) is applied to choose relatively fitter chromosomes from the current generation to produce individuals of the next generation. This type of selection is analogous to a roulette wheel, where each individual occupies an area on the wheel. The larger area the individual occupies, the more likely the ball will land there (i.e., the individual will be chosen). To tie the fitness of a chromosome to its probability of being chosen for reproduction of the next generation, a probabilistic selection function is defined according to the following:

$$\Pr(c) = \frac{fitness(c)}{\sum_{i=1}^{P_{size}} fitness(c_i)}$$

where c is a chromosome under consideration and $fitness$ is the fitness function defined earlier. P_{size} is the predefined size of a population P . To implement roulette wheel selection, a random number r in the unit interval is generated for each chromosome under consideration. If the selection probability $\Pr(c)$ of a chromosome c is greater than the random number (i.e., $\Pr(c) > r$), the corresponding chromosome is selected for reproduction. In other words, a fitter chromosome has a better chance to be selected for reproduction. However, chromosomes with low fitness values may also have a chance to be selected to maintain a good balance between exploitation- and exploration-oriented search.

After two chromosomes are selected for re-production, the genetic operation of two-point crossover (Goldberg 1989; Lau et al. 2006; Ruiz-Torribiano 2010) is applied according to a predefined crossover probability p_c . Specifically, a random number r in the unit interval is generated for the pair of chromosomes under consideration. If $r < p_c$ is true, a two-point crossover is applied to the selected pair of chromosomes. Basically, two points along the evolvable genes of the pair are randomly selected. Then, the genes within the two-point boundary are exchanged between the two parent chromosomes to produce two child chromosomes. If $r < p_c$ is not established, the crossover operation will not be applied to the pair.

Each chromosome of the selected pair is then considered for the mutation operation after the crossover operation. First, a random number r in the unit interval is generated for each evolvable gene of each chromosome of the selected pair. If $r < p_m$ is true, where p_m is a predefined mutation rate, a mutation operation will be applied to the particular gene. With decimal gene encoding, the current value of the selected gene is replaced by another feasible gene value randomly. The evolutionary process (i.e., selection, crossover, and mutation) is repeated until the number of individuals of the next generation reaches the predefined number P_{size} . The aforementioned evolutionary process is applied to each population from high-level to low-level. If the average fitness of each population reaches a predefined threshold AVG_{fit} , or the number of generations reproduced exceeds the maximum number of generations MAX_{gen} , the HCGA algorithm will be terminated. At this stage, the fittest chromosome from each population is chosen to drive the operation of the M&A target scoring module of the ABIMA system. Table A1 lists the genetic parameters of the HCGA algorithm; these parameters were applied to the experiments reported in this paper.

HCGA Parameters	Level II Population (Application Parameter)	Level I Population (Sentiment Analysis Parameter)	Level I Population (Relationship Mining Parameter)
Size of population	$P_{size}[1] = 90$	$P_{size}[2] = 40$	$P_{size}[3] = 40$
Gene encoding	decimal	decimal	decimal
Elitism rate	$w_e[1] = 10\%$	$w_e[2] = 10\%$	$w_e[3] = 10\%$
Crossover probability	$P_c[1] = 0.83$	$P_c[2] = 0.83$	$P_c[3] = 0.83$
Mutation probability	$P_m[1] = 0.05$	$P_m[2] = 0.05$	$P_m[3] = 0.05$
Type of crossover	two-point	two-point	two-point
Type of mutation	uniform	uniform	uniform
Max number of generations	$Max_{gen} = 500$		
Max average fitness	$AVG_{fit} = 0.95$		

References

- Delgado, M., Zuben, F., and Gomide, F. 2004. "Coevolutionary Genetic Fuzzy Systems: A Hierarchical Collaborative Approach," *Fuzzy Sets and Systems* (141:1), pp. 89-106.
- Goldberg D. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading, MA: Addison-Wesley.
- Huang, H., Pasquier, M., and Quek, C. 2009. "Financial Market Trading System with a Hierarchical Coevolutionary Fuzzy Predictive Model," *IEEE Transactions on Evolutionary Computation* (13:1), pp. 56-70.
- Lau, R. Y. K., Tang, M., Wong, O., Milliner, S., and Chen, Y. 2006. "An Evolutionary Learning Approach for Adaptive Negotiation Agents," *International Journal of Intelligent Systems* (21:1), pp. 41-72.
- Maiti, A. K., and Maiti, M. 2008. "Discounted Multi-Item Inventory Model Via Genetic Algorithm with Roulette Wheel Selection, Arithmetic Crossover and Uniform mutation in Constraints Bounded Domains," *International Journal of Computer Mathematics* (85:9), pp. 1341-1353.
- Olsson, B. 2001. "Co-evolutionary Search in Asymmetric Spaces," *Information Sciences* (133:3-4), pp. 103-125.
- Ounis, I., Macdonald, C., and Soboroff, I. 2008. "Overview of the TREC-2008 Blog Track," in *Proceedings of the 17th Text Retrieval Conference*, Gaithersburg, MD (<http://trec.nist.gov/pubs/trec17/>).
- Ruiz-Torrubiano, R., and Suarez, A. 2010. "Hybrid Approaches and Dimensionality Reduction for Portfolio Selection with Cardinality Constraints," *IEEE Computational Intelligence Magazine* (5:2), pp. 92-107.

Appendix B

Affect Analysis for Sociocultural Fitness Assessment of M&A Targets

Affect analysis is very useful for modeling financial phenomena from both theoretical and pragmatic perspectives (Bollen et al. 2011; Deresky 2011). Recently, affect analysis has been successfully applied to predict the movement of the Dow Jones Industrial Average (Bollen et al. 2011). The main function of our affect analysis module is to estimate the sociocultural fitness of the targeted M&A companies, the targeted industrial sectors, or the entire targeted nation. For instance, affect analysis can be applied to assess the public's feelings (e.g., happiness or fear) about a potential M&A deal after it is announced. Specifically, the WordNet-affect lexicon (Valitutti et al. 2004) is applied to build our affect analysis module. From among the big six classes of affect (i.e., anger, fear, happiness, sadness, surprise, and neutral) that are often applied to affect analysis (Calix et al. 2010), four of them—anger, fear, happiness, and sadness—are used to estimate the emotion score of a potential M&A deal. The affect classes of surprise and neutral are not used by our affect analysis module because our preliminary experiments show that these classes cannot improve the performance of M&A affect analysis.

Each token of a financial document (e.g., a financial news article or an investor comment about a potential M&A deal) is matched against the WordNet-affect lexicon to identify its emotion class. Then, the emotion score of the document is computed according to the following:

$$emotion(d) = \frac{|happy| - (|anger| + |fear| + |sad|)}{|happy| + |anger| + |fear| + |sad|}$$

where *happy*, *anger*, *fear*, and *sad* represent the sets of emotional indicators extracted from a financial document *d*, which covers the potential M&A deal. With respect to the predefined time window of due diligence (i.e., an input parameter of an M&A query), each emotion score is then weighted using an exponential decay function (Barari and Mitra 2008; Jo et al. 1997). In particular, we apply the following exponential decay function to weight the affect scores:

$$emotion(d, t) = emotion(d) \times e^{-\left(\frac{t_{current} - t}{\tau/2}\right)}$$

where $emotion(d)$ the term is the original affect score of a document *d* (i.e., without weighting), and $emotion(d, t)$ is the weighted emotion score at time point *t*. The term τ is the due diligence time window specified in months, and the term $(t_{current} - t)$ is the elapsed time (in months) between the time *t* when a financial document containing affects is posted and the time $t_{current}$ when M&A target scoring is conducted. For an emotion score derived from a document posted in the same month when M&A target scoring is conducted, the elapsed time $(t_{current} - t)$ is zero.

The weighted emotion score of a potential M&A deal $emotion(deal)$ is the mean of the weighted emotion scores of the set of relevant financial documents D containing affects about a deal over each time point of the due diligence window, and it is defined by

$$emotion(deal) = \frac{\sum_{i=0}^{\tau-1} \sum_{j=1}^{|D|} emotion(d_j, t_i)}{\sum_{i=0}^{\tau-1} |D_{t_i}|}$$

where $emotion(d_j, t_i)$ represents the emotion score of a document d_j at time point t_i . The term D_i refers to the set of relevant documents at each time point t_i . Finally, the sociocultural fitness of an M&A target is estimated by taking into account the weighted emotion score of the potential M&A deal and other sociocultural factors.

References

- Barari, M., and Mitra, S. 2008. "Power Law Versus Exponential Law in Characterizing Stock Market Returns," *Atlantic Economic Journal* (36:3), pp. 377-379.
- Bollen, J., Maoa, H., and Zengb, X. 2011. "Twitter Mood Predicts the Stock Market," *Journal of Computational Science* (2:1), pp. 1-8.
- Calix, R. A., Mallepudi, S. A., Chen, B., and Knapp, G. M. 2010. "Emotion Recognition in Text for 3-D Facial Expression Rendering," *IEEE Transactions on Multimedia* (12:6), pp. 544-551.
- Deresky, H. 2011. *International Management: Managing Across Borders and Cultures, Text and Cases* (7th ed.), New York: Prentice Hall.
- Jo, H., Han, I, and Lee, H. 1997. "Bankruptcy Prediction Using Case-Based Reasoning, Neural Networks, and Discriminant Analysis," *Expert Systems with Applications* (13:2), pp. 97-108.
- Valitutti, A., Strapparava, C., and Stock, O. 2004. "Developing Affective Lexical Resources," *Psychology* (2:1), pp. 61-83.