韓國컴퓨터情報學會 論文誌
第14卷 第7號, 2009. 7.

2009-14-7-1-3

소프트웨어 비용산정을 위한 면역 알고리즘 기반의 서포트 벡터 회귀

권 기 태*, 이 준 길**

Support Vector Regression based on Immune Algorithm for Software Cost Estimation

Kwon Ki-Tae*, Lee Joon Gil**

요 약

정보시스템에 대한 이용이 늘어남에 따라 소프트웨어 개발 요구와 개발 비용이 증가하게 되었다. 기존에는 통계적 알고리즘 기반의 회귀분석을 이용하여 소프트웨어 개발비용을 산정하였으나 오늘날은 기계학습 방법들이 많이 연구되고 있다. 본 논문에서는 기계학습 기술의 하나인 SVR를 사용하여 소프트웨어 비용을 산정하였고, 이 때 SVR에서 사용하는 파라미터들의 최적 조합을 면역계의 동작원리를 적용한 면역 알고리즘을 적용하여 최적 조합을 찾았다. 소프트웨어 비용산정을 위해 세대수, 기억세포수, 대립유전자수를 변경해 가면서 면역 알고리즘 기반의 SVR을 적용하였고, 그 실험 결과를 기존 연구된 다른 기계학습 방법과 비교 분석하였다.

Abstract

Increasing use of information system has led to larger amount of developing expenses and demands on software. Until recent days, the model using regression analysis based on statistical algorithm has been used. However, Machine learning is more investigated now. This paper estimates the software cost using SVR(Support Vector Regression), a sort of machine learning technique. Also, it finds the best set of parameters applying immune algorithm. In this paper, software cost estimation is performed by SVR based on immune algorithm while changing populations, memory cells, and number of allele. Finally, this paper analyzes and compares the result with existing other machine learning methods.

▸ Keyword : 서포트 벡터 회귀(Support Vector Regression), 소프트웨어 비용산정(Software Cost Estimation), 기계 학습(Machine Learning), 면역 알고리즘(Immune Algorithm), 매개변수(Parameters), 회귀(Regression).

• 제1저자 : 권기태
• 투고일 : 2009. 06. 23, 심사일 : 2009. 06. 25, 게재확정일 : 2009. 07. 21.
* 강릉대학교 컴퓨터공학과 교수 ** 강릉대학교 정보전산원 근무

Ⅰ. 서 론

소프트웨어 개발 초기 단계에서 소프트웨어 개발 비용을 정확하게 예측하는 것은 프로젝트의 성패를 결정짓는 중요한 요소이다. 비용 초과로 고객이 프로젝트를 취소할 수도 있고, 개발 업체는 실제 소요될 비용보다 비용을 적게 예측함으로써 이윤을 남기지 못하고 많은 시간을 소모하게 될 수도 있다. 소프트웨어 생명주기 초기에 개발 비용을 정확히 예측하면, 프로젝트 관리자들은 어떤 자원이 필요한지, 적합한 자원들을 언제 배정해야 할지를 알 수 있다.

소프트웨어 비용산정 모델에 관한 주요 연구는 1965년 169개 소프트웨어 프로젝트의 104가지의 속성에 관한 SDC의 광범위한 연구로 시작되었다. 이 모델을 기초로 1960년대 후반과 1970년대 초반 부분적으로 유용했던 일부 모델들이 유도되었다. 1970년대 후반에 SLIM, Checkpoint, PRICE-S, SEER, COCOMO 등과 같은 알고리즘 모델이 개발되었다. 이들 비용산정 모델의 개발자 대부분이 동일한 시기에 비용산정 모델을 개발하기 시작했지만, 그들은 모두 유사한 딜레마에 빠졌다. 즉, 소프트웨어의 크기가 커지고 복잡해짐에 따라, 소프트웨어 개발비용을 정확하게 예측하기가 점점 더 어렵다는 것이다. 알고리즘 모델 자체의 문제점과 더불어 매우 빠르게 변화하는 개발 환경의 영향으로 정확도가 높은 알고리즘 모델을 개발하기란 매우 어렵다[1,2].

1980년대에는 알고리즘 모델이 폭넓게 이용되었으며, 이 시기의 모델들은 다양한 규모와 환경적인 데이터 집합을 이용하여 비교하였다. 이러한 연구에서 얻은 주요한 결론은 비용 산정 모델들은 환경이 다른 경우에 측정하지 못한 인자들이 적용된다면 성능이 떨어진다는 것이다. 1990년에는 Abdel-Hamid와 Madnick 같은 연구자들은 소프트웨어 개발은 복잡한 동적인 프로세스이며 복잡함과 생산성에서 나타나는 다양성을 설명할 수 있는 변경과 관련된 관계성을 거의 알지 못함을 알게 되었다. 따라서, 1990년대에는 기계 학습 알고리즘에 기반한 비모수 모델링 기법의 소개 및 산정법등이 등장하였다[3].

기계학습은 환경과의 상호작용에 기반한 경험적인 데이터로부터 스스로 성능을 향상시키는 시스템으로 축적된 데이터에 기반하여 수행모델을 자동으로 생성시키는 기술이다[4].

기계학습에 의한 소프트웨어 비용산정 방법으로 가장 먼저 사용된 기법은 신경망에 의한 비용예측이다. 이어서 사례기반추론 방법을 도입하여 소프트웨어 개발비를 예측하였고, 또한 트리 기반 방법으로 회귀트리, 의사결정트리 등을 활용하여 소프트웨어 비용산정을 시도한 연구들도 수행되었다. 신경망, 사례기반추론 또는 회귀모형을 이용한 소프트웨어 비용산정의 정확도를 비교하면 연구자에 따라 정확도가 다소 차이가 있는 것으로 보고되고 있다[5,6,7].

본 논문에서는 통계적 기계학습 기술인 SVR을 이용하여 소프트웨어 개발비용을 산정하고, 이 때 SVR에서 사용하는 사용자 정의 파라미터들의 최적 조합을 면역 알고리즘을 이용하여 찾고자 한다. 면역 알고리즘은 인간의 행동을 결정하는 정보처리 메카니즘들 중의 하나이다. 인간의 정보처리 메카니즘은 유전계, 신경계, 면역계, 내분비계의 네 가지로 분류할 수 있는데, 이 중 유전계와 신경계는 유전자 알고리즘과 인공신경망으로 다양한 분야에서 응용되고 있으나 면역계는 이에 비하여 공학적으로 응용된 예가 많지 않다[8].

본 논문의 구성은 제1장 서론에 이어서, 제2장 연구의 배경에서는 면역 알고리즘과 SVR에 대하여 소개하고, 제3장에서는 본 논문에서 제안하는 면역 알고리즘 기반의 SVR를 이용한 소프트웨어 비용산정을 소개하고, 제4장에서는 실제의 데이터를 대상으로 실험하고 그 결과를 기존의 연구 결과와 비교 분석한다. 그리고 마지막 제5장에서는 본 논문의 결론과 추후 연구과제를 제시한다.

Ⅱ. 연구의 배경

많은 소프트웨어 비용추정 알고리즘 모델이 지난 30여년 이상 개발되어 왔지만, 알고리즘 모델 자체에 근본적인 한계를 가지고 있으므로 기계학습 알고리즘에 기반한 비모수 기법에 대한 관심이 제기되고 있다.

2.1 면역 알고리즘

면역 시스템은 생물학적 측면에서 볼 때 외부의 병원체에 대응하여 생체의 방어 및 유지를 수행하는 자율 분산 시스템으로 시스템의 요소들이 뇌의 명령을 따르는 것이 아니라 각 요소가 자율적으로 환경에 대응한다[9]. 일반적으로 면역 시스템은 외부의 항원(antigen)에 반응하는 항체(antibody)를 구성하고 그 항체들이 기억세포를 형성하고 분화한다. 면역 알고리즘은 이러한 반응체계를 공학적으로 적용하는 시스템이다.[9,10]

면역 알고리즘은 다른 비결정론적인 알고리즘과 마찬가지로 동시에 여러 개의 가능해로서 최적화를 진행해 나가며, 해의 값 자체를 그대로 사용하는 것이 아니라, 코드화된 수의 배열을 사용한다. 그리고 최적화의 목적함수를 미분값과 그 외 다른 정보를 요구하지 않고 그대로 사용한다는 장점을 갖

는다. 또한, 이러한 비결정론적 알고리즘의 특징 이 외에도 면역 알고리즘만이 갖는 가장 큰 특징은 최적해로의 수렴을 보장하기 위하여 기억세포군(memory cell)을 갖고 최적화 과정을 수행한다[11,12].

면역 알고리즘을 최적화 문제에 대응시켜 보면 면역 알고리즘의 항원은 최적화 문제의 제약조건과 목적함수로 대응되고, 항체는 최적화 문제의 해집단 후보가 된다. 그리고 면역 알고리즘의 기억세포는 최적화 문제의 해집단이 되며, 면역 시스템에서의 항원과 항체간의 친화도는 최적화 문제에서의 적합도로 계산된다[12].

2.2 서포트 벡터 머신

VM은 Vapnik에 의해 제안된 통계적 학습이론으로 두 범주를 갖는 객체들을 분류하는 방법이다[14]. SVM은 분류 오류 확률을 최소화 하는 구조적 위험 최소화(structural risk minimization) 방법에 기초하고 있다. SVM은 인공신경망과 비슷한 수준의 높은 예측력을 나타낼 뿐만 아니라 인공신경망의 한계점으로 지적되었던 과대적합, 국소최적화와 같은 한계점들을 완화하는 장점을 갖고 있다. 하지만, 한 가지 단점은 최적 커널 변환 함수와 이 함수의 인자들이 모든 데이터 집합마다 약간씩 달라 매번 이들을 찾아야 한다는 것이다.

SVM은 기본적으로 학습 데이터를 분류하고 그 분류할 때 사용하는 결정함수에 의해 발생하는 일반화 오차(generalization error)를 최소로 하는 결정함수를 찾는 것이다.

주어진 학습 데이터 $(x^1,y^1),...,(x^l,y^l)$, $x\in R, y\in\{-1,+1\}$를 분류하는 결정함수를 D라 하면 결정함수 D는 식(1)과 같다.

$$D(X) : W^T \bullet X + b = 0 \quad \cdots\cdots (1)$$

결정함수를 초평면(hyperplane)이라 하는데 SVM에서는 최종적으로 학습 데이터를 두 그룹의 데이터로 분리할 수 있는데, 이때 최적분리 경계면과 각 그룹의 가장 근접한 데이터를 SV(Support Vector)라 하며, 각 그룹의 SV간의 거리 $\frac{2}{\|w\|}$가 최대가 되는 지점에서 최적 분리 경계면이 설정된다.

그러나 일반적으로 실세계에서는 선형분리가 가능하지 않으며, 이러한 경우에 결정된 최적 분리 경계면은 높은 분류능력을 갖지 못한다. 선형분류가 어려운 경우에 SVM은 커널함수라는 매핑함수를 이용하여 고차원 특징공간을 선형분류 가능한 공간으로 매핑하여 분류능력을 일반화시킨다.

2.3 서포트 벡터 회귀(SVR)

SVM 분류를 회귀 문제에 적용하여 훈련 데이터에 의존한 SVM Regression(SVR) 예측 모델을 만들 수 있다. SVR은 분류 최적화에 대한 일반화 능력이 뛰어나다. SVR은 SVM Classification과 같은 방식으로 구하며 그 절차는 (그림 1)과 같다[14,15].

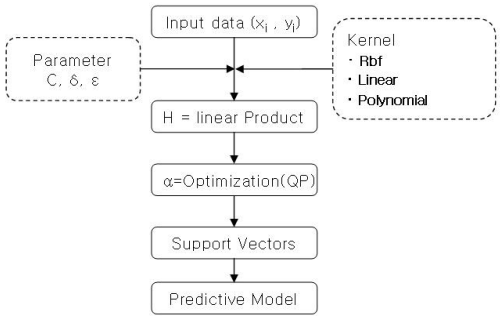

그림 1. SVR 처리 순서도
Fig. 1. Flowchart of SVR

SVR은 학습데이터 $D=\{(x_i,y_i)\in R^n\times R, i=1,2,...,l\}$ $x\in R^n$, $y\in R$가 있을때 선형회귀 초평면은 식(2)와 같다.

$$f(x,w) = W^T X + b \quad \cdots\cdots (2)$$

SVR에서는 분류의 마진 대신에 근사값 오류를 측정한다. 이 ε-insensitivity zone을 갖는 오류함수는 식(3), (그림 2)와 같다.

$$E(x,y,f) = |y-f(x,w)|_e = \begin{cases} 0 & \text{if } |y-f(x,w)| \le \epsilon \\ |y-f(x,w)|-\epsilon, & otherwise \end{cases} \quad \cdots\cdots (3)$$

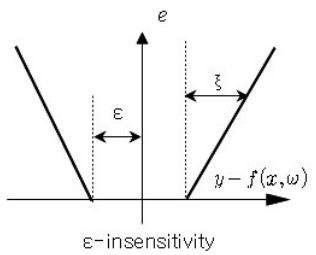

그림 2. ε-insensitivity 함수
Fig. 2. ε-insensitivity Function

(그림 3)은 SVR에서 사용되는 파라미터들, 실데이터, 예측 데이터들 사이의 관계를 보여준다. (그림 3)에서 검은 사각형은 실제값이며 Support Vector를 나타낸다.

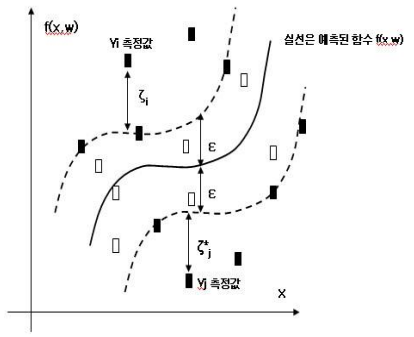

그림 3. SVR에서 파라미터 관계
Fig. 3. Parameter Relation of SVR

식(6)의 실험적 오류를 식(4)와 같이 나타내고, 식(4)와 $\| W \|^2$을 동시에 최소로 함으로써 식(2)의 초평면을 구할 수 있다. 식(4)를 최소로 하는 것은 식(5), 식(6), 식(7)을 조건으로 식(5)를 최소화하는 것과 같다.

$$R_{emp}^{e}(w,b) = \frac{1}{l}\sum_{i=1}^{l} \left| y_i - W^T x_i - b \right|_e \quad \cdots\cdots (4)$$

$$R_{w,\xi,\xi^*} = \left[\frac{1}{2} \| w \|^2 + C\left(\sum_{i=1}^{l} \xi_i + \sum_{i=1}^{l} \xi_i^* \right) \right] \quad \cdots\cdots (5)$$

$$y_i - W^T x_i - b \le \epsilon + \xi_i, \ i = 1,2,...,l \quad \cdots\cdots (6)$$

$$W^T x_i + b - y_i \le \epsilon + \xi_i^*, \ i = 1,2,...,l \quad \cdots\cdots (7)$$

$$\xi_i \ge 0, \xi_i^* \ge 0, \ i = 1,2,...,l \quad \cdots\cdots (8)$$

이후 SVM Classification과 같이 최적화 문제를 풀면 식(10), 식(11) 조건하에 회귀식(9)를 구할 수 있다.

$$f(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^{'}) K(x_i, x) + b \quad \cdots\cdots (9)$$

$$\sum_{i=1}^{n} (\alpha_i - \alpha_i^{'}) = 0 \quad \cdots\cdots (10)$$

$$0 \le \alpha_i, \alpha_i^{'} \le C, \ i = 1,...,n \quad \cdots\cdots (11)$$

III. 본 론

본 논문에서는 SVR를 사용하여 소프트웨어 개발비용을 추정하고, 이 때 SVR에서 사용되는 파라미터들의 최적 조합을 면역 알고리즘을 적용하여 찾았다. 소프트웨어 비용산정의 정확성을 비교하기 위한 척도로는 식 (12)의 MMRE(Mean Magnitude of Relative Error)와 PRED(Prediction at Level)[16]를 사용한다. PRED(v)는 예측값의 v% 범위에 포함되는 추정치의 개수의 비율을 말한다.

$$MMRE = \frac{1}{n}\sum_{i=1}^{n} \frac{\left| Y_i - \overline{Y_i} \right|}{Y_i} \quad \cdots\cdots (12)$$

본 논문에서 수행한 면역 알고리즘 기반의 SVR를 이용한 소프트웨어 비용산정은 (그림 4)와 같이 6단계의 순서에 의해 시행된다.

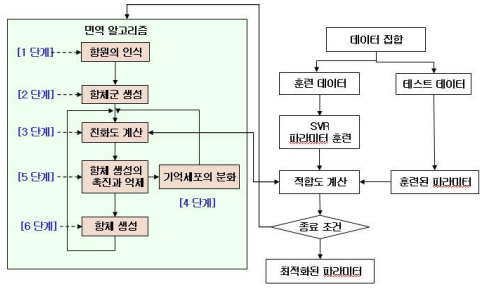

그림 4 면역 알고리즘이 적용된 SVR 흐름도
Fig. 4. SVR Flowchart based on IA

〔1단계〕 항원의 인식

항원, 목적함수, 제한 조건들이 입력되고, 최적화 문제를 정의하는 부분이다.

〔2단계〕 초기 항체군의 생성

최초의 과정은 유효한 항체를 무작위로 생성한다. 이 때 각 항체는 43개의 유전자로 이루어져 있고, 대립유전자수는 2로 하여 기억세포수 만큼의 항체를 생성한다. 각 항체는 SVR의 파라미터 조합이 되며 이 항체를 이용하여 주어진 소프트웨어 비용을 산정한다.

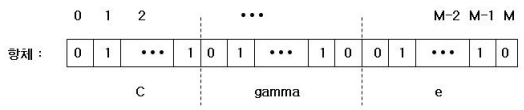

그림 5 항체구성도
Fig. 5. The structure of antibody

〔3단계〕 친화도 계산

친화도는 항체들 간 또는 항원과 항체 간의 결합력으로 알 수 있다. 본 논문에서는 정보 엔트로피 계산법을 이용하여 생산된 항체의 다양성을 측정하는 친화도를 정의한다[9,13]. (그림 6)과 같이 가정하면, 유전자좌 j의 유전자 정보 엔트로피는 식(13)으로 계산된다.

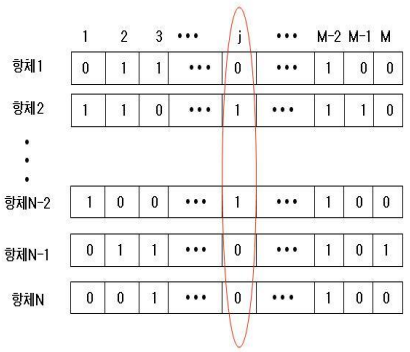

그림 6 유전자 정보 엔트로피 개념도
Fig. 6. The informative entropy of antigens

$$H_j(N) = \frac{1}{M}\sum_{i=0}^{p}(-P_{ij}*\log P_{ij}) \quad \cdots\cdots (13)$$

여기서 P_{ij}는 유전자좌j에 나타난 대립유전자의 출현 확률로 식(14)와 같이 나타낼 수 있다.

$$P_{ij} = \frac{\text{유전자좌 } j\text{에 출현한 대립유전자 } i\text{의 총수}}{N} \quad \cdots\cdots (14)$$

각 항체의 유전자 정보 엔트로피의 계산식은 식(15)이며 면역 시스템의 다양성의 평균 정보 엔트로피는 식(15)에 의해 계산할 수 있다.

$$H(N) = \frac{1}{M}\sum_{j=0}^{M} H_j(N) \quad \cdots\cdots (15)$$

친화도는 항체의 유사성의 척도이며, 항체와 항체간 그리고 항원과 항체간의 친화도가 있다. 두 항체간의 친화도 즉 항체a와 항체b간의 친화도는 식(16)으로 정의한다.

$$affinity(a,b) = \frac{1}{(1+H(2))} \quad \cdots\cdots (16)$$

$H(2)$는 항체a와 항체b만의 정보 엔트로피를 의미하며, $H(2)=0$일 때 항체a와 항체b간의 유전자가 완전히 일치하는 경우이다. $affinity(a,b)$가 1인 경우는 두 항체가 완전히 일치하는 경우이며 $affinity(a,b)$가 0에 가까울수록 기억세포 내에 유사한 항체가 없는 것을 의미한다. 또 항원과 항체v간의 친화도는 식(17)로 나타내며 항체의 평가치로 항체와 항원간의 결합강도를 나타낸다.

$$ax_v = opt_v \quad \cdots\cdots (17)$$

여기서 opt_v는 이 논문에서 목적함수의 해에 해당하며 추정값이 된다. 이 추정값과 실제값의 MMRE를 적합도로 사용하였다. 항체와 항체간의 친화도 계산은 다음 순서에서 이루어지는 기억세포로 분화하기 위해 필요한 과정이다.

〔4단계〕 기억세포로 분화

〔3단계〕에서 계산된 친화도가 높은 항체를 기억세포에 추가한다. 항체간의 친화도가 가장 높은 항체들을 소멸시키고 항원과의 친화도가 높은 상위 50%를 기억세포에 저장한다.

〔5단계〕 항체 생성의 촉진과 억제

본 논문에서는 항체의 생성과 억제를 기대치에 의하지 않고 〔3단계〕에서 계산된 친화도에 의하여 친화도가 가장 높은 1개를 기억세포에 저장한다. 그리고 항체간의 친화도를 계산하여 친화도가 높은 항체들을 상위부터 두 개 중 하나를 삭제한다. 따라서 항원과의 친화도가 높은 항체의 생산을 촉진하며 면역 시스템 전체에 항체간의 적합도가 높은 항체의 생성을 억제하여 면역 시스템에 있어서 다양성의 조절기구로 작용하게 된다.

〔6단계〕 항체의 생성

친화도가 가장 높은 항체와 기억세포의 분화로 저장된 차세대 기억세포를 구성하기 위하여 현재 기억세포에 존재하는 항체들을 추출하여 돌연변이만 일으키고, 새로운 항체를 처음 기억세포의 수와 같아질 때까지 추가한다.

IV. 실험 및 분석

면역 알고리즘 기반의 SVR을 이용한 소프트웨어 비용산정 알고리즘은 LIBSVM Version 2.86[17]과 Python 2.5를 사용하여 구현하였으며, 실험 데이터는 비용산정 연구 분야에서 가장 널리 이용되고 있는 데이터 중의 하나인 〈표 1〉의 Jean Marc Desharnais 데이터[18]를 사용하였다. 동일한 데이터를 사용한 기존 연구[19,20,21]와 비교하기 위해, 기존 연구와 동일한 방법으로 81개 프로젝트 데이터 중 63개를 훈련 데이터로, 18개를 실험 데이터로 사용하여 소프트웨어 비용을 산정하였다.

표 1. Desharnais 데이터 집합
Table 1. Features of Desharnais Data Set

Features	설명	사용여부
Team exp	팀 개발 경험기간	X
Manager Exp	프로젝트 관리자 경험기간	X
Year End	개발 기간	O
Length	규모	X
Effort (Y)	공수	O
Transactions	트랜잭션 수	O
Entities	엔티티 수	O
PointAdjust	조정 기능점수	O
Adjustment	조정 인자	X
PointNonadjust	미조정 기능점수	O

훈련 데이터로 SVR 모델을 생성 후 실험 데이터로 테스트한 결과인 세대수, 기억세포수, 대립유전자수에 따른 실험결과는 〈표 2〉와 같다. 〈표 2〉의 실험결과를 보면 MMRE는 세대수 보다는 기억세포수에 영향을 많이 받으며, PRED는 세대수, 기억세포수, 대립유전자수에 따라 근소하지만 각각이 증가함에 따라 좋은 결과를 보이고 있다.

표 2. 세대수, 기억세포수, 대립유전자수에 따른 SVR 파라미터, MMRE, PRED 결과
Table 2. Results of IA-SVR

세대수	기억세포수	대립유전자수	C	σ	ε	MMRE	PRED (25)
10	10	2	7327	0.133	0.172	0.4112	0.39
		3	18400	0.079	0.001	0.4060	0.44
		10	17684	0.084	0.010	0.4087	0.44
	20	2	13613	0.095	0.189	0.4075	0.39
		3	13391	0.097	0.069	0.4083	0.39
		10	32183	0.056	0.069	0.4078	0.50
	40	2	14813	0.090	0.319	0.4074	0.39
		3	19364	0.075	0.025	0.4068	0.44
		10	22213	0.071	0.028	0.4048	0.50
20	10	2	6589	0.132	0.147	0.4124	0.39
		3	14117	0.091	0.176	0.4086	0.39
		10	24788	0.067	0.087	0.4049	0.50
	20	2	14030	0.089	0.010	0.4094	0.39
		3	19546	0.076	0.192	0.4060	0.50
		10	27732	0.062	0.079	0.4059	0.50
	40	2	15199	0.088	0.146	0.4075	0.39
		3	19288	0.076	0.012	0.4068	0.44
		10	27722	0.063	0.095	0.4046	0.56
50	10	2	13233	0.095	0.057	0.4085	0.39
		3	18709	0.077	0.043	0.4066	0.44
		10	30289	0.061	0.095	0.4066	0.56
	20	2	14779	0.089	0.265	0.4080	0.39
		3	18511	0.079	0.098	0.4057	0.44
		10	24089	0.068	0.018	0.4054	0.50
	40	2	16081	0.086	0.181	0.4073	0.44
		3	19245	0.078	0.189	0.4055	0.50
		10	35218	0.056	0.072	0.4052	0.56
100	10	2	15814	0.087	0.296	0.4071	0.44
		3	16922	0.084	0.079	0.4066	0.44
		10	28533	0.062	0.093	0.4049	0.56
	20	2	15989	0.087	0.144	0.4077	0.44
		3	17627	0.081	0.006	0.4061	0.44
		10	29776	0.061	0.055	0.4054	0.56
	40	2	15524	0.087	0.042	0.4074	0.39
		3	19204	0.077	0.023	0.4053	0.50
		10	25989	0.065	0.030	0.4049	0.56

〈표 2〉를 보면 MMRE가 40% 정도, PRED가 39%부터 56%까지 값을 보이고 있으며, 세대수 20, 기억세포수 40, 대립유전자수 10일 때 MMRE = 0.4046, PRED = 0.56으로 가장 좋은 결과를 보이고, 이 때의 최적 파라미터 조합은 C = 27722, σ = 0.06328, ε = 0.09495이다.

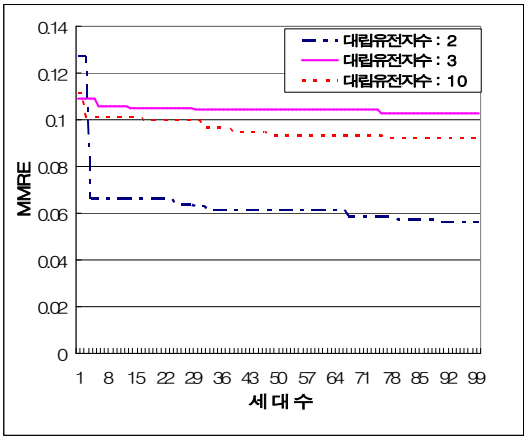

그림 7. 훈련 데이터의 세대수에 따른 MMRE 비교
Fig. 7. MMRE comparison by populations of training set

〈그림 7〉은 훈련 시 대립유전자수와 세대수 증가에 따른 MMRE의 변화를 나타낸 그래프이다. 면역 알고리즘의 성능 평가를 위하여 세대수, 기억세포수, 대립유전자수를 변경해 가면서 실험한 결과 기억세포수와 대립유전자수가 많은 것이 좋은 결과를 보였고, 세대수에 대한 변화는 초반인 10세대 이내에서 많은 변화가 있었고 이 후로는 변화가 미미해 나중에는 세대수에는 큰 영향이 없었다.

표 3. 본 연구의 결과와 기존 예측모델의 비교
Table 3. The Comparison of IA-SVR with existing predicting model

	모델	MMRE	PRED(25)
1	SVEG[20]	0.52	0.46
2	2단계 인접 이웃[19]	1.62	0.44
3	5단계 인접 이웃[19]	1.68	0.44
4	인공 신경망[19]	0.61	0.56
5	유전 프로그래밍[19]	0.45	0.23
6	퍼지 논리[21]	0.54	0.30
7	면역 알고리즘 기반의 SVR	0.40	0.56

〈표 3〉은 본 논문에서 제안한 면역 알고리즘 기반의 SVR를 가지고 계산한 MMRE와 PRED 값을 기계학습 기법을 중심으로 기존 연구된 모델들의 값과 비교한 표이다. 〈표 3〉에서 1은 GSS 방식과 그리드 방식이 결합된 SVR 모델, 2,3은 인접성 기준 분류 모델, 4는 인공신경망 모델, 5는 유전 프로그래밍, 그리고 6은 퍼지 논리를 이용한 모델이다.

위 〈표 3〉을 보면 본 연구에서 제안한 면역 알고리즘 기반의 SVR 모델이 기존의 SVR뿐만 아니라 다른 기계학습방법인 인접성 기준 분류모델, 인공 신경망, 유전 프로그래밍, 퍼지 논리보다 우수한 것을 알 수 있다.

V. 결 론

본 논문에서 소프트웨어 비용산정을 위하여 면역 알고리즘과 SVR를 결합하여 사용하였다. 본 논문에서 사용한 면역 알고리즘은 항체들 간의 친화도가 높은 것은 항체 생성을 억제하는 자기조절기능으로, 항원 간의 친화도가 높은 것은 새로운 항체를 생성함으로써 우수한 항체의 보존과 다양성을 통해 파라미터들을 최적화했다.

면역 알고리즘에 의해 찾은 최적의 파라미터 조합으로 소프트웨어 비용산정을 한 결과을 보면 기존 연구된 결과보다 우수한 결과를 보여 면역 알고리즘 기반의 SVR를 이용한 소프트웨어 비용산정이 기존의 SVR뿐만 아니라 다른 기계학습 방법인 사례기반추론, 인공 신경망이나 유전 프로그래밍, 퍼지 논리보다 우수한 것을 알 수 있었다.

본 논문에서 면역 알고리즘 기반의 SVR를 사용하여 소프트웨어 비용산정에 적용하는 것은 소프트웨어 공학 분야에 최초의 사례라는 점에서도 의의가 있다.

향후 연구과제로는 본 논문에서 보는 것 같이 면역 알고리즘에서 대립유전자의 길이가 기호화하여 표현하는 항체들에 많은 영향을 미침으로 적절한 대립유전자와 항체를 기호화하는 연구가 필요하다. 또한 면역 알고리즘 기반의 SVR를 이용한 모델을 유지보수 데이터에 적용하는 것과 소프트웨어 신뢰성과 품질관리 분야에도 적용하고, 다양한 데이터마이닝 연구[22]와 결합하는 것도 차후 과제라 할 수 있다.

참고문헌

[1] Linda M. Laird and M. Carol Brenman., *"Software Measurement and Estimation"* Wiley-Interscience, 2006.

[2] Barry W. Boehm et al., *"Software Cost Estimation with COCOMO II"* Prentice-Hall, 2000.

[3] M. A. Parthasarathy, *"Practical Software Estimation,"* Addison Wesley, 2007.

[4] 장병탁, "차세대 기계학습 기술", 정보과학회지 제25권, 제3호, 2007년 3월.

〔5〕 Mukhopadhyay et al., "Examining the Feasibility of a Case-based Reasoning Model for Software Effort Estimation," MIS Quarterly, 16, pp. 155-171, June 1992.
〔6〕 Martin Shepperd and Chris Schofield, "Estimating Software Project Effort Using Analogies," IEEE Trans. Software Eng., Vol. 23. No. 12, pp. 736-743, November 1997.
〔7〕 M. Shin and A.L. Goel, "Empirical Data Modeling in Software Engineering using Radial Basis Functions," IEEE Trans. Software Eng. Vol. 26, No. 6, pp. 567-576, June 2000.
〔8〕 Jean Mare Alliot E, Lutton M.Schoenauer, "*Artificial Evolution*", Springer, 1996.
〔9〕 Dipankar Dasgupta, "*Artificial immune systems and their applications*", Springer, 1999.
〔10〕 심귀보 외, "컴퓨터 면역시스템 개발을 위한 인공면역계의 모델링과 자기인식 알고리즘", 한국 퍼지 및 지능시스템학회 논문지, 제11권, 제10호, 910-918쪽, 2001년 10월.
〔11〕 정형환 외, "면역 알고리즘을 이용한 전력 계통 안정화 장치의 최적 파라미터 선정", 전기학회 논문지 제49A권 제9호, 433-445쪽, 2000년 9월,
〔12〕 박진현 외, "DC 모터 파라미터 변동에 대한 면역 알고리즘 제어기 설계", 한국 퍼지 및 지능시스템학회 논문지, 제12권, 제4호, 353-360쪽, 2002년 4월,
〔13〕 Pang-Ning Tan et al., "*Introduction to Data Mining*" Addison Wesley, 2006.
〔14〕 Changha Hwang., "Support Vector Median Regression," Data and Information Science, Vol. 14, No. 1, pp. 67-74, 2003.
〔15〕 Toby Segaran, "*Programming Collective Intelligence*," O'relly, 2007.
〔16〕 권기태 외, "소프트웨어 비용산정을 위한 SVM의 파라미터 선정과 응용에 관한 연구", 한국컴퓨터정보학회 논문지, 제14권, 제3호, 209-216쪽, 2009년, 3월.
〔17〕 Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, "A Practical Guide to Support Vector Classification", *http://www.csie.ntu.edu.tw/~cjlin/libsvm/*, 2008
〔18〕 J.M. Desharnais, "Analyse Statistique de la Productivities des Projects Informatique a Partie de la Technique des Point des Function", Masters Thesis, Univ. of Montreal, 1989.
〔19〕 Colin J. Burgess and Martin Lefley, "Can genetic programming improve software effort estimation? A comparative evaluation", Information and Software Technology, Vol. 43, Issue 14, pp. 863-873, December 2001
〔20〕 Hojung Lim and Amrit L. Goel, "Support Vector Machines for Data Modeling with Software Engineering Applications," in Springer Handbook of Engineering Statistics, Springer, pp. 1023-1037, 2006.
〔21〕 A. R. Gray and S. G. MacDonell, "Applications of Fuzzy Logic to Software Metric Models for Development Effort Estimation," Proceedings of NAFIPS'97, pp. 394-399, 1997.
〔22〕 이영호 외, "약물부작용감시스템에서 재현성 평가를 통한 마이닝 모델 개발", 한국컴퓨터정보학회 논문지, 제14권, 제3호, 183-192쪽, 2009년, 3월.

저 자 소 개

권 기 태

1986년 서울대학교 계산통계학과 학사
1988년 서울대학교 계산통계학과 석사
1993년 서울대학교 계산통계학과 박사
1996년 Univ. of Southern California, Post-Doc.
1990.9 ~ 현재
강릉대학교 컴퓨터공학과 교수
관심분야 : 소프트웨어 비용산정, 소프트웨어 메트릭스

이 준 길

1987년 서울시립대학교 전산통계학과 학사
2000년 강릉대학교 산업대학원 컴퓨터과학과 석사
2009년 강릉대학교 컴퓨터공학과 박사
현재 강릉대학교 정보전산원 근무
관심분야 : 소프트웨어 비용산정, 소프트웨어 메트릭스,