Journal of the Korea Academia-Industrial cooperation Society
Vol. 21, No. 8 pp. 521-529, 2020

https://doi.org/10.5762/KAIS.2020.21.8.521
ISSN 1975-4701 / eISSN 2288-4688

기상 데이터를 이용한 데이터 마이닝 기반의 산불 예측 모델

김삼근*, 안재근
한경대학교 컴퓨터응용수학부(컴퓨터시스템 연구소)

Data Mining based Forest Fires Prediction Models using Meteorological Data

Sam-Keun Kim*, Jae-Geun Ahn
School of Computer Engineering & Applied Mathematics, Hankyong National University

요 약 산불은 경제, 자연환경, 건강과 같은 삶의 여러 측면에서 몇 가지 악영향을 주는 가장 핵심적인 환경위험 중의 하나이다. 산불의 조기발견, 빠른 예측, 신속한 대응은 산불 위험으로부터 재산과 생명을 구하는데 본질적인 역할을 할 수 있다. 산불의 빠른 발견을 위해 기상청에서 각 지역에 설치한 로컬 센서를 통해 획득한 기상 데이터를 이용하는 방법이 있다. 기상 조건(예: 온도, 바람)은 산불 발생에 영향을 미친다고 알려져 있다. 본 논문에서는 산불의 피해 면적을 예측하기 위해 데이터 마이닝(DM) 기법을 적용한다. 다섯 종류의 DM 모델, 예를 들어 Stochastic Gradient Descent(SGD), Support Vector Machines(SVM), Decision Tree(DT), Random Forests(RF), Deep Neural Network(DNN)과 네 가지 입력 특성 그룹(공간, 시간, 기상 데이터 이용)을 최근 5년간의 경기도 지역에서 수집한 실제 산불 발생 데이터에 적용하였다. 실험결과는 기상 데이터만을 이용한 DNN 모델이 가장 우수한 성능을 보였다. 제안한 모델은 빈도수가 높은 작은 규모의 산불 예측에 더 효과적이었다. 제안한 예측 모델을 통해 도출된 이러한 지식은 소방 자원 관리를 개선하는데 특히 유용하다.

Abstract Forest fires are one of the most important environmental risks that have adverse effects on many aspects of life, such as the economy, environment, and health. The early detection, quick prediction, and rapid response of forest fires can play an essential role in saving property and life from forest fire risks. For the rapid discovery of forest fires, there is a method using meteorological data obtained from local sensors installed in each area by the Meteorological Agency. Meteorological conditions (e.g., temperature, wind) influence forest fires. This study evaluated a Data Mining (DM) approach to predict the burned area of forest fires. Five DM models, e.g., Stochastic Gradient Descent (SGD), Support Vector Machines (SVM), Decision Tree (DT), Random Forests (RF), and Deep Neural Network (DNN), and four feature selection setups (using spatial, temporal, and weather attributes), were tested on recent real-world data collected from Gyeonggi-do area over the last five years. As a result of the experiment, a DNN model using only meteorological data showed the best performance. The proposed model was more effective in predicting the burned area of small forest fires, which are more frequent. This knowledge derived from the proposed prediction model is particularly useful for improving firefighting resource management.

Keywords : Data Mining, Deep Neural Network Model, Support Vector Machine, Meteorological Data, Prediction

*Corresponding Author : Sam-Keun Kim(Hankyong National Univ.)
email: skim@hknu.ac.kr
Received July 21, 2020 Revised July 29, 2020
Accepted August 7, 2020 Published August 31, 2020

1. 서론

산불은 경제, 자연환경, 건강과 같은 삶의 여러 측면에서 몇 가지 악영향을 주는 가장 핵심적인 환경위험요소 중의 하나이다[1]. 우리나라는 최근 10년 평균('10~'19년) 440건의 산불이 발생하여 857ha의 산림이 소실되었으며, 최근에는 기후변화 등의 원인으로 전 세계적으로 초대형 산불이 자주 발생하여 산불 예방과 관리가 국제적 이슈로 부각되고 있다[2]. 그러나 조기발견, 빠른 예측, 신속한 대응은 산불 위험으로부터 재산과 생명을 구하는데 본질적인 역할을 할 수 있다.

산불 발생의 빠른 탐지는 성공적인 소방 활동을 위한 핵심 요소이다. 전통적인 인간 감시는 비용이 많이 들고 주관적인 요소에 의해 영향을 받기 때문에 자동 솔루션을 개발하는 것이 강조되어 왔다. 자동 솔루션을 세 가지 부류로 그룹 지을 수 있다[3]: 인공위성 기반, 적외선/연기 스캐너, 로컬 센서(기상데이터 측정용). 인공위성 데이터는 지역화 하는데 지연이 발생하며 해상도가 떨어진다. 스캐너는 장비 구입 및 유지관리에 고비용이 든다. 기상조건(예: 온도, 바람세기 등)은 화재 발생에 영향을 준다고 알려져 있으며[4], 자동기상관측 장비가 거의 대부분 지역에 설치되어 있으므로 저비용으로 실시간 기상 데이터를 획득할 수 있다.

기상데이터를 이용한 DM(Data Mining) 기반의 산불 예측 모델을 구축하기 위한 많은 연구들이 있었다. Paulo Cortez[5]는 포르투갈 북동지역의 산불 피해지역을 예측하기 위해 해당 지역의 공간데이터, 시간데이터, 그리고 기상데이터를 이용하여 다양한 DM 기법들을 적용하였고, 그 중에서도 기상데이터(예: 온도, 바람 등)에만 적용한 SVM(Support Vector Machine) 기법이 가장 우수함을 보여주었다. [6]은 온도, 습도, 바람세기 등의 기상 데이터를 이용하여 MLP(Multilayer Perceptron), RBFN(Radial Basis Function Network), SVM 등에 적용하였다. [7]은 기상 데이터를 이용하여 인간의 개입 없이 자동화 및 지능적인 방식으로 산불을 예측하는 스마트 센서 노드 아키텍처의 일부분으로 통합된 DT(Decision Tree) 기반의 산불 예측 시스템을 제안하였다. 이 연구에서 예측 성능은 82.92%의 결과를 보여주었다. Yudong Li 등[8]은 중국 광시 자치구 2010-2018 인공위성 데이터, 기상데이터, 식물 종류 등의 데이터에 대해 SVM과 BPNN(Back-Propagation Neural Networks)을 적용하여 주요 산불 원인을 결정하는 모델을 제안하였다. BPNN과 SVM의 예측 결과는 각각 92.16%, 89.89%의 정확도를 얻었다.

한편, 산불 탐지 모델을 구축하기 위해서도 다양한 DM 기법들이 연구되었다. [3]은 적외선 스캐너와 ANN(Artificial Neural Networks)을 결합하여 산불 오경보(false alarms)를 90%이상 줄일 수 있었다. [9]는 산불 위험을 모델링하고 산불 발생 고위험 지역을 인지하기 위해 로지스틱 회귀 및 ANN 기법을 적용하였다. 적용결과 로지스틱 회귀 및 ANN의 정확도는 각각 65.76%, 93.49%를 얻었다. Seong-Wook Park[10]은 DNN(Deep Neural Network) 분류기를 활용하여 산불 피해지 영상으로부터 사람의 주관이 개입되지 않고 신속히 산불 피해지를 탐지할 수 있는 모델을 제안하였다.

본 논문에서는 최근 5년간의 경기도 지역 산불 데이터와 기상 데이터를 이용하여 4개의 기계학습 알고리즘과 DNN의 DL(Deep Learning)을 적용한 산불 예측 모델을 제안한다. 적용한 기계학습 알고리즘으로는 SGD(Stochastic Gradient Descent), SVM, DT, RF(Random Forests)이고, DNN의 최적화 알고리즘으로는 SGD[11], Adagrad[12], RMSprop[13], Adam[14]을 적용한다. 본 논문에서 제안하는 모델은 단순히 5개의 날씨 변수(즉, 평균 온도, 최소 온도, 최대 온도, 최대 바람세기, 평균 바람세기)만을 이용한 DNN 모델이다. 제안 모델은 적용한 데이터 셋이 대다수가 작은 규모의 산불로 구성되어 있어서 빈도수가 높은 작은 규모의 산불 예측에 더 효과적이었다. 제안한 예측 모델을 통해 도출된 지식은 소방 자원 관리를 개선하는데 특히 유용하다.

본 논문의 구성은 다음과 같다: 2장에서는 산불 데이터에 대해 분석하고, 3장에서는 데이터 마이닝 모델을 제시한다. 4장에서는 실험 결과를 제시하고 분석한다. 마지막 5장에서는 결론을 기술한다.

2. 산불 데이터

본 논문에서는 경기도 지역의 산불 데이터에 대해 실험한다. 실험에 사용된 데이터는 2014년 1월 1일부터 2018년 12월 31일까지의 기간에 발생한 산불 발생 현황을 수집하였고, 3개의 소스를 이용하여 구축하였다. 첫 번째 데이터베이스는 산림청 연도별 산불발생 현황[2]으로부터 수집한 것이다. 이 데이터에는 산불 발생일시, 진화 종료시간, 최초 발생지 공간 위치(7×7 그리드로 표현)(Fig. 1), 발생원인, 피해면적 등이 포함되어 있다. 두 번째 데이터베이스는 산불 발생 지역에 설치된 자동기상

관측장비(AWS, Automatic Weather System)로부터 산불 발생 시점의 기상 데이터(예: 온도, 바람세기 등)[15]를 수집한 것이다. 세 번째 데이터는 Geocoder-Xr 주소변환 툴[16]을 이용하여 산불 발생지 주소를 경위도 좌표값으로 변환하여 획득한 것이다. 세 개의 데이터 소스로부터 획득한 데이터를 456개의 엔트리를 갖는 1개의 데이터 셋으로 통합하였다.

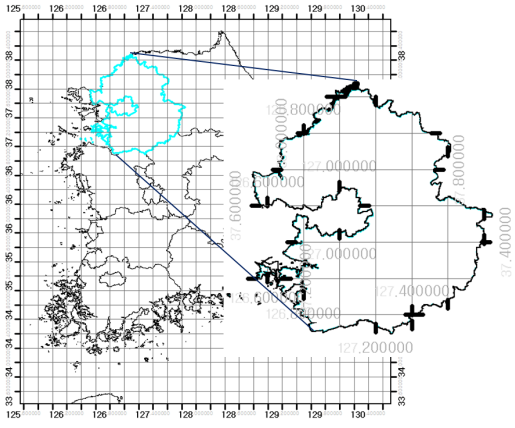
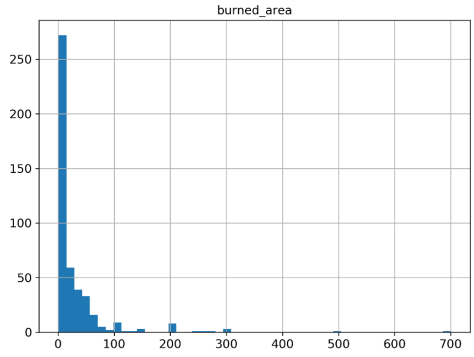


Fig. 1. The map of the Gyeonggi-do

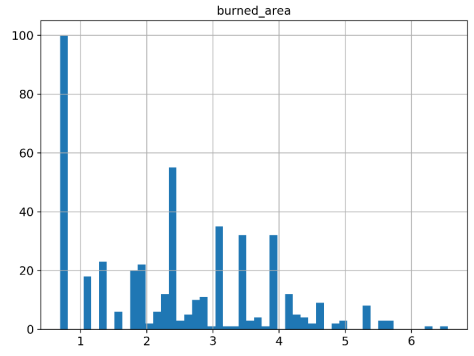
Table 1은 선택된 데이터 특성들을 보여준다. 처음 4개의 행은 공간과 시간 특성을 나타낸다. 산불이 발생한 위치를 나타내는 경도(longitude)와 위도(latitude)는 데이터 적용지역을 7×7 구역(Fig. 1)으로 분할한 셀 인덱스를 의미한다. 또한 매월 기상 조건은 확연히 다르고 요일마다 산불 발생 위험이 다르기 때문에(예: 평일 대 주말) 시간 정보를 나타내는 month와 day를 포함시켰다. 5~9행까지는 기상 데이터(평균 온도, 최저 온도, 최고 온도, 최대 바람 속도, 평균 바람 속도)를 나타낸다. 산불 발생 당시의 강수량은 96%가 0값을 가져서 강수량 속성은 제외시켰다. 마지막 행(burned_area)은 출력 클래스 속성을 의미한다.

Table 1. The preprocessed dataset attributes

Attribute	Description
longitude	x-axis coordinate (from 1 to 7)
latitude	y-axis coordinate (from 1 to 7)
month	Month of the year
day	Day of the week
avg_temp	Average temperature (℃)
min_temp	Minimum temperature (℃)
max_temp	Maximum temperature (℃)
max_wind_speed	Maximum wind speed(km/h)
avg_wind	Average wind speed(km/h)
burned_area	Total burned area (in hectares)



(a) Burned area (in hectares)



(b) Ln(area+1)

Fig. 2. The histogram for the burned area (a) and respective logarithm transform (b)

3. 데이터 마이닝 모델

Fig. 2(a)의 산불 피해지역(burned area)은 왼쪽으로 왜곡되어 있다. 이는 대다수의 산불 피해지역의 규모가 작음을 의미한다. 현재의 데이터 셋의 경우 산불 피해 규모가 1ha/100 = 100m^{2}보다 낮게 소실된 건수가 100개이다. 이러한 왜곡 현상을 줄이고 균형을 개선시키기 위해 산불 피해지역(burned area)을 $y = \ln(x+1)$단위로 변환하였다(Fig. 2 (b)). 이렇게 변환된 변수가 최종 출력 타깃이 된다.

데이터 셋 D는 $k \in \{1,...,N\}$ 인스턴스로 구성된다. 각 인스턴스의 입력 벡터 $(x_1^k,...,x_A^k)$는 타깃 벡터 y_k와 매핑된다. A는 입력 특성의 개수를 의미한다. 에러는 $e_k = y_k - \hat{y_k}$로 주어진다. 여기서 $\hat{y_k}$는 k번째 입력 패턴에 대해 예측한 값을 의미한다. 성능 측정 도구로는 MAE(Mean Absolute Error)와 RMSE(Root Mean

Square Error)를 사용한다. 이들은 모델에 의해 생성된 예측 데이터와 실제 데이터 사이의 차이를 보여주는 좋은 방법이다. MAE와 RMSE는 아래 식 (1)처럼 계산한다:

$$MAE = \sum_{i=1}^{N} \left| y_i - \hat{y_i} \right| / N$$
$$RMSE = \sqrt{\sum_{i=1}^{N} \left(y_i - \hat{y_i} \right)^2 / N} \quad (1)$$

여기서 N은 인스턴스 개수, y_i는 i번째 인스턴스의 전체 특징 벡터(feature vector)이다. 두 측정 도구 모두 값이 0에 가까우면 더 좋은 성능을 의미한다. 그러나 RMSE는 에러 값이 큰 경우에 대해 더 높은 가중치 값을 갖는다. 따라서 아웃라이어가 많은 경우에는 MAE가 더 선호된다.

회귀분석에 적절한 다양한 DM 알고리즘들이 제안되었다. 본 논문에서는 5가지 DM 알고리즘을 적용한다: SGD, DT, RF, SVM, DNN. SGD는 해석하기가 단순하고 회귀분석에 널리 사용되어 왔지만 선형 매핑만 학습할 수 있다. 이 결점을 해결하기 위한 대안으로 DT와 RF 같은 트리 구조에 기반을 둔 방법, 또는 SVM과 DNN 같은 비선형 매핑이 가능한 방법들을 적용한다.

DT는 특정 기준에 따라 계층적인 형태로 값들을 구분해주는 규칙들의 집합을 표현하는 모델이다. 이런 표현은 IF-THEN 규칙들로 변환될 수 있으며 인간이 이해하기 쉽다. RF는 DT의 앙상블이다. RF 예측자는 T개 트리의 출력 값들의 평균값으로 구축된다. 일반적으로 RF는 1개의 DT보다 개선된 성능을 보여준다.

SVM 회귀 모델[17]은 일반적인 ANN보다 이론적으로 이점을 갖는다. 즉, SVR(Support Vector Regression)은 ANN의 문제점 중의 하나인 국부적 지역해(local minima) 문제가 없다. SVR에서 입력 $x \in R^A$가 비선형 매핑 함수를 사용하여 m차원 특징 공간으로 변환된다. SVR은 변환된 특징 공간에서 최상의 선형 분리 하이퍼평면(hyperplane)을 탐색한다:

$$\hat{y} = w_0 + \sum_{i=1}^{m} w_i \psi_i(x) \quad (2)$$

여기서 $\psi_i(x)$는 $K(x,x') = \sum_{i=1}^{m} \psi_i(x)\psi_i(x')$에 따른 비선형 변환을 의미한다. SVR에 의해 생성된 모델은 모델을 구축하기 위한 비용 함수가 모델 예측 값과 가까운 데이터를 무시하기 때문에 훈련 데이터셋의 서브셋에만 의존한다. SVR 모델 학습은 아래 식 (3)의 비용 함수를 최소화시키는 것이다:

$$minimize \frac{1}{2} \| w \|^2$$
$$subject\ to\ \left| y_i - \langle w, x_i \rangle - b \right| \le \epsilon \quad (3)$$

여기서 x_i는 타깃 값 y_i인 훈련 인스턴스이다. $\langle w, x_i \rangle + b$는 해당 인스턴스에 대한 예측 값이고, ϵ는 임계값으로 동작하는 파라미터이다: 모든 예측은 실제 예측의 ϵ 범위 안에 있어야 한다.

SVR의 성능은 주로 2가지 파라미터에 의해 영향 받는다: C - 정규화(regularization) 정도; ϵ - 도로($\epsilon - insensitive$ 구역)의 폭. SVR은 가능한 한 많은 인스턴스를 도로상에 적합시키려고 하면서 마진 위반(도로 밖의 인스턴스)을 제한시킨다. 도로의 폭은 하이퍼파라미터 ϵ에 의해 제어된다. SVR에서 비선형 회귀 태스크를 수행하려면 커널 SVM 모델을 이용하면 된다. 정규화는 C 값에 의해 제어된다. C 값이 크면 정규화가 적게 적용된 것이고, C 값이 작으면 정규화가 훨씬 더 많이 적용된 것이다.

DNN은 입력층과 출력층 사이에 여러 개의 은닉층을 포함하는 ANN 구조를 가지며, DL에 의해 학습된다[18]. Fig. 3의 DNN 구조는 노드 사이의 연결 방식, 적용되는 층 개수, 사용된 활성함수의 타입, 여러 종류의 하이퍼파라미터 등에 따라 다르다. 한 층에 있는 노드들은 완전 연결(fully-connected)되어 있고, 이 연결 구조와 활성함수에 대한 유일한 요구조건은 미분 가능해야 한다는 것이다.

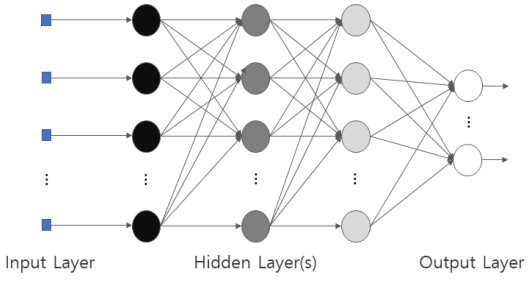


Fig. 3. Deep Neural Network structure

DL의 목적은 예측된 출력 값과 실제 출력 값 사이의 차이(비용함수)를 줄이는 것이다(Fig. 4). 따라서 네트워크의 최적화된 가중치를 찾음으로써 비용함수를 최소화하고 일반화 성능을 보장할 필요가 있다. 일반화 성능이

좋으면 더 좋은 예측을 수행한다. DNN의 구조와 무관하게 공통적인 DL 학습 과정은 입력 데이터를 네트워크 층과 활성함수에 제공하고 출력을 생성하는 것이다. 네트워크의 출력 값은 실제 값과 비교되어 에러 측정값(비용함수)을 반환하게 되고, 이 에러는 모델 성능을 평가하기 위한 도구로 사용된다. DNN 회귀 문제의 경우 에러는 흔히 예측된 출력 값과 실제 타깃 레이블 사이의 MSE(Mean Square Error)로 계산된다. 네트워크의 모든 파트가 미분 가능하기 때문에 전체 네트워크에 대해 기울기를 계산할 수 있다. 이 기울기는 네트워크의 성능을 개선시키는 방향으로 비례적으로 가중치를 증가시킨다. 결국 BP(backpropagation) 알고리즘[19]을 이용하여 모든 가중치가 순차적으로 갱신된다.

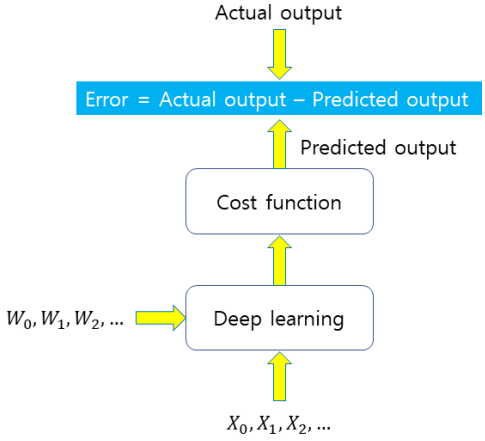


Fig. 4. Gradient based learning

대규모 DNN의 학습은 매우 느려질 수 있다. DNN의 학습을 개선시키기 위해 연결 가중치의 좋은 초기 값 설정 전략, 좋은 활성함수 사용, BN(Batch Normalization) 방법을 사용할 수 있다. 또한 일반적인 GD(Gradient Descent) 최적화 알고리즘 대신 더 빠른 알고리즘을 사용하여 속도와 성능을 크게 개선시킬 수 있다. GD는 비용함수를 최소화하는 최적화된 가중치를 찾기 위해 다양한 가중치로 수많은 반복을 수행한다. GD는 증가하는 방향을 의미한다. 그러나 여기서의 목적은 최소점을 탐색하는 것이므로 기울기의 반대방향으로 따라갈 필요가 있다. 즉, 비용함수를 최소화하기 위해 음의 기울기 방향으로 가중치를 갱신한다:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \tag{4}$$

여기서 $\boldsymbol{\theta}$는 가중치, η는 학습률, $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$는 가중치 $\boldsymbol{\theta}$의 기울기이다. SGD는 한 번에 하나씩 인스턴스를 취급하는 GD의 일종이다.

모멘텀 최적화 알고리즘[12]은 이전 스텝의 기울기뿐만 아니라 현재 스텝의 기울기를 고려하여 학습을 가속화시키는 역할을 한다: 모멘텀 벡터 $\boldsymbol{m}$으로부터 로컬 기울기만큼을 감소시킨다:

$$\begin{aligned} \boldsymbol{m} &\leftarrow \beta \boldsymbol{m} - \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \\ \boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} + \boldsymbol{m} \end{aligned} \tag{5}$$

여기서 β는 0과 1 사이의 값을 갖는 모멘텀이다.

AdaGrad[13] 알고리즘은 모멘텀에서 학습률을 적응적으로 조정하는 방법이다:

$$\begin{aligned} \boldsymbol{s} &\leftarrow \boldsymbol{s} + \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \otimes \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \\ \boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \oslash \sqrt{\boldsymbol{s} + \varepsilon} \end{aligned} \tag{6}$$

여기서 첫 번째 단계는 기울기 제곱을 벡터 $\boldsymbol{s}$에 누적시킨다. $\otimes$는 요소별 곱셈을 의미한다. 두 번째 스텝은 기울기 벡터를 $\sqrt{\boldsymbol{s} + \varepsilon}$만큼 크기를 줄인다. $\oslash$는 요소별 나눗셈을 의미한다. 결론적으로 빈도수가 낮은 가중치에 대해서는 큰 학습률을 적용하고, 빈도수가 높은 가중치에 대해서는 더 작은 값의 학습률로 학습시키게 된다. AdaGrad는 학습률을 수작업으로 조정해야 하는 필요성을 제거해준다. 단점으로는 학습률이 급격하게 사라지는 경우가 발생할 수 있다는 것이다. 이를 해결하기 위해 RMSProp[14]과 Adam[15] 알고리즘이 제안되었다.

RMSProp은 기울기 제곱의 이동평균을 이용하여 AdaGrad의 학습률이 급격하게 사라지는 문제를 해결한다.

$$\begin{aligned} \boldsymbol{s} &\leftarrow \beta \boldsymbol{s} + (1-\beta) \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \otimes \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \\ \boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \oslash \sqrt{\boldsymbol{s} + \varepsilon} \end{aligned} \tag{7}$$

Adam 알고리즘은 기울기의 1차 모멘텀과 2차 모멘텀 추정치로부터 각 가중치에 대한 적응적인 학습률을 계산함으로써 AdaGrad의 문제점을 완화시켜 준다. AdaGrad의 모멘텀들에 대한 단순 이동평균 대신에 지수 이동평균을 이용한다.

$$
\begin{aligned}
\boldsymbol{m} &\leftarrow \beta_1 \boldsymbol{m} - (1-\beta_1)\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \\
\boldsymbol{s} &\leftarrow \beta_2 \boldsymbol{s} + (1-\beta_2)\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \otimes \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \\
\hat{\boldsymbol{m}} &\leftarrow \frac{\boldsymbol{m}}{1-\beta_1^t} \\
\hat{\boldsymbol{s}} &\leftarrow \frac{\boldsymbol{s}}{1-\beta_2^t} \\
\boldsymbol{\theta} &\leftarrow \boldsymbol{\theta} + \eta \hat{\boldsymbol{m}} \oslash \sqrt{\hat{\boldsymbol{s}} + \varepsilon}
\end{aligned} \tag{8}
$$

여기서 하이퍼파라미터 $\beta_1, \beta_2 \in [0,1]$는 이동평균들의 지수 decay 비율을 제어한다.

본 논문에서는 DNN의 최적화 알고리즘으로 SGD 뿐만 아니라 속도와 성능을 크게 개선시킨 Adagrad, RMSProp, Adam을 적용한다.

4. 실험 및 고찰

기계학습 알고리즘(SGD, SVM, DT, RF) 실험은 Scikit-Learn 라이브러리[19]를 이용하였다. DNN에 관한 실험은 TensorFlow[20]와 Keras 라이브러리[21]를 이용하였다.

모델을 학습시키기 전에 카테고리형 입력 특성들을 전처리하였다. 4개의 카테고리형 특성들(latitude, longitude, month, day)을 원핫 인코딩(one-hot encoding) 기법을 사용하여 인코딩하였다. 또한 모든 수치 특성들은 평균 0(zero)과 표준편차 1인 분포로 표준화 하였다. 이렇게 변환된 특성들을 적용하여 모든 모델을 학습시켰다. 출력은 DM 모델을 적합시킨 후에 다시 로그 함수의 역(inverse)으로 변환하였다.

입력 특성들의 효과를 추정하기 위해 4개의 입력 특성 그룹으로 나누어 각 DM 알고리즘을 테스트하였다: STM, SM, TM, M (S - 공간 데이터(latitude, longitude); T - 시간 데이터(month, day); M - 5개의 기상 데이터 (avg_temp, min_temp, max_temp, max_wind_speed, avg_wind). 각 모델의 최상의 일반화 성능을 얻기 위해 하이퍼파라미터 탐색에 10-fold 그리드 탐색 기법을 적용하였다. SVM의 경우 하이퍼파라미터 탐색을 위해 다음처럼 설정하였다: $kernel \in \{linear, poly, rbf\}$, $C \in \{0.1, 1, 10, 100\}$, $degree \in \{2,3,4\}$, $\epsilon \in \{0.1, 1.0, 1.5\}$. 최상의 kernel, C, degree, ϵ 값을 선택한 후 모든 훈련 데이터에 대해 재학습시켰다. Table 2는 SVM에 대해 선택된 최상의 하이퍼파라미터 값들을 보여준다.

Table 2. The best hyperparameters for SVM

SVM Model hyperparameters	Feature Selection Setup			
	STM	SM	TM	M
kernel	rbf	rbf	rbf	rbf
C	1	1	0.1	10
degree	2	2	2	2
epsilon	1.5	1.5	0.1	0.1

DNN의 경우 하이퍼파라미터 탐색을 위해 환경변수들의 범위를 Table 3처럼 설정하였다. 예측 성능을 확인하기 위해 각 입력 특성 셋업과 DNN optimizer 설정에 대해 10-fold 교차검증으로 30개의 런(run)을 적용하였다(300개의 시뮬레이션).

Table 3. Proposed DNN architecture

Hyperparameter	Value
optimizers	['sgd', 'adagrad', 'rmsprop', 'adam']
# input neurons	One per input feature
# hidden layers	[2 - 5]
# neurons per hidden layer	[32 - 64]
# output neurons	1 per prediction dimension
learning_rate	[1e-4, 1e-3, 1e-2]
drop_rate	[0.1, 0.2, 0.3, 0.4, 0.5]
Hidden activation	ReLU
Output activation	None

DNN 모델의 경우 일반화 성능에 비해 훈련 데이터에 대한 성능이 뛰어나면 과적합(overfitting)이 발생했다고 볼 수 있다. 본 논문에서는 과적합을 피하기 위해 DNN

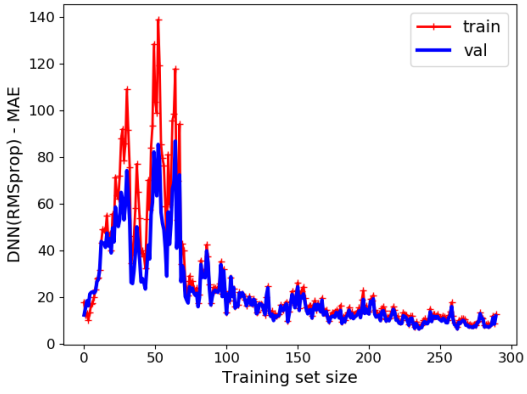


Fig. 5. Learning curves on the training set and validation set as a function of the training set size; DNN (optimizer=RMSprop) and M setup.

Table 4. The predictive results in terms of the MAE errors (RMSE values in parentheses; underline- best model; bold - best within the feature selection)

DM Model		Feature Selection Setup							
		STM		SM		TM		M	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
SGD		1.44 (1.97±0.03)	**1.46** (2.23±0.05)	1.48 (2.52±0.11)	**1.49** (2.23±0.04)	1.53 (1.96±0.03)	1.49 (2.19±0.04)	1.49 (1.98±0.03)	1.51 (2.22±0.04)
SVM		1.49 (1.95±0.03)	1.51 (2.16±0.04)	1.52 (1.98±0.03)	1.54 (2.15±0.04)	1.43 (1.94±0.03)	**1.47** (2.16±0.05)	1.38 (1.88±0.03)	**1.50** (2.08±0.04)
DT		1.72 (2.12±0.03)	1.76 (2.16±0.04)	1.58 (2.07±0.03)	1.73 (2.29±0.05)	1.45 (2.00±0.03)	1.53 (2.26±0.06)	1.69 (2.07±0.03)	1.72 (2.32±0.05)
RF		1.54 (1.91±0.02)	1.61 (2.12±0.04)	1.50 (1.92±0.02)	1.60 (2.13±0.04)	1.47 (1.89±0.02)	1.53 (2.13±0.04)	1.53 (1.92±0.02)	1.60 (2.16±0.04)
DNN	SGD	1.17 (4.60±0.27)	1.88 (3.00±0.09)	**1.17** (4.49±0.27)	1.84 (2.96±0.10)	**1.15** (4.44±0.27)	1.85 (2.91±0.09)	1.14 (4.20±0.24)	1.82 (2.75±0.07)
	Adagrad	**1.14** (7.06±0.41)	1.86 (23.76±2.4)	**1.17** (7.09±0.42)	1.90 (24.28±2.5)	**1.15** (6.88±0.40)	1.86 (25.04±2.6)	1.14 (6.68±0.4)	1.77 (21.79±2.2)
	RMSProp	1.17 (2.63±0.04)	1.87 (3.04±0.09)	**1.17** (2.51±0.03)	1.88 (2.78±0.06)	**1.15** (2.49±0.04)	1.77 (2.73±0.06)	**1.10** (2.35±0.04)	1.79 (2.46±0.05)
	Adam	1.17 (2.69±0.04)	1.87 (3.06±0.10)	**1.17** (2.57±0.04)	1.91 (2.88±0.08)	1.18 (2.54±0.04)	1.87 (2.86±0.09)	1.11 (2.39±0.04)	1.77 (2.55±0.07)

네트워크 구조에 BatchNormalization 층[22]과 Dropout 층[23]을 네트워크 구조에 추가하였다. Fig 5는 과적합 발생 여부를 확인하기 위한 방법으로 전체 데이터 셋에서 나누어진 훈련 데이터 셋(80%)을 다시 훈련 데이터 셋(80%)과 검증 데이터 셋(20%)으로 나눈 후 훈련 데이터 셋 크기를 늘려가면서 모델의 성능을 그래프로 표현한 것이다. 훈련 데이터 셋 학습 곡선과 검증 데이터 셋 학습 곡선이 낮은 에러로 근접해 있으면 과적합이 발생하지 않은 것으로 간주할 수 있다.

Table 4는 DM 모델들에 대한 실험 결과를 최상의 결과 값과, 평균과 t-student 95% 신뢰구간($\bar{x} \pm 1.96\widehat{\sigma_{\bar{x}}}$) 방식으로 보여준다[24]. 벤치마킹의 목적을 위해 기계학습 SGD 알고리즘의 예측결과(첫 번째 줄)를 추가하였다. MAE 기준에서는 모든 DNN 모델이 SGD 벤치마크 모델보다 성능이 우수했다. RMSE 기준에서는 SVM 모델이 우수했다. 또 다른 흥미로운 결과는 데이터 셋에서 공간/시간 데이터를 제외했을 경우 오히려 성능이 더 개선된다는 점이다. M 셋업과 DNN(RMSprop) 모델의 경우가 가장 좋은 성능을 보였다.

5. 결론

산불은 인간 생명을 위협하는 심각한 환경 재해를 야기한다. 과거 20여 년간 Fire Management System(FMS)을 지원해 주는 산불 자동 탐지 툴을 구축하기 위한 많은 연구들이 진행되어 왔다. 본 논문에서는 기상청에서 각 지역에 설치한 기상 센서들로부터 획득 가능한 기상 데이터를 이용하는 DM 기반의 산불 예측 모델을 제안한다. 날씨는 산불 발생에 영향을 미친다고 알려져 있다. 기상 데이터를 이용하는데 있어서 장점은 실시간/저비용으로 데이터를 획득할 수 있다는 점이다. 본 논문의 실험에서는 경기도 지역의 최근 5년간의 실제 데이터를 적용하였다. 적용된 데이터베이스는 공간, 시간, 기상 데이터를 결합하여 구축하였고 이를 회귀문제로 모델링하였다. 목적은 산불 피해면적의 예측이었다. 전체 5가지 DM 모델(DNN 모델의 4가지 종류의 최적화 알고리즘 포함)과 4가지 입력 특성 그룹에 대해 적용해 보았다. 제안한 최적의 산불 예측 모델은 기상 데이터 M 셋업과 DNN(RMSprop)에 기반을 둔 예측 모델이 가장 우수한 성능을 보였다. 본 논문은 오프라인 학습에 기반을 두었다. 즉, 데이터가 수집된 후에 DM 기법을 적용하였다.

향후 연구과제는 전국 규모의 데이터 셋으로 확장하는 것과 저비용의 자동기상관측장비를 활용하여 실시간으로 획득한 기상 데이터에 기반을 둔 실시간 산불 예측 서비스를 구축하는 것이다.

References

[1] Samaher Al_Janabi, Ibrahim Al_Shourbaji, and Mahdi A. Salmana, "Rating and Mapping Fire Hazard in the Hardwood Hyrcanian Forests using GIS AND Expert Choice Software", Applied Computing and Informatics, Vol 14, Issue 2, pages 214-224, 2018.
DOI: https://doi.org/10.1016/j.aci.2017.09.006

[2] Meteorological Agency, Forest fire statistics by year, http://www.forest.go.kr/

[3] B. Arrue, A. Ollero, and J. Matinez de Dios, "An Intelligent System for False Alarm Reduction in Infrared Forest-Fire Detection", IEEE Intelligent Systems, 15(3):64-73, 2000.

[4] J. Terradas J. Pinol and F. Lloret, "Climate warming, wildfire hazard, and wildfire occurrence in coastal eastern Spain", Climatic Change, 38:345-357, 1998.

[5] Paulo Cortez, A.D.J.R. Morais, "A data mining approach to predict forest fires using meteorological data", Proceedings of the 13th Portuguese Conference on Artificial Intelligence (EPIA) Guimarães Portugal, pp. 512-523, 2007.
https://repositorium.sdum.uminho.pt/bitstream/1822/8039/1/fires.pdf

[6] A.M. Özbayoğlu, R. Bozer, "Estimation of the burned area in forest fires using computational intelligence techniques", Procedia Computer Science, 12, pages 282-287, 2012.
DOI: https://doi.org/10.1016/j.procs.2012.09.070

[7] Faroudja Abid and Nouma Izeboudjen, "Predicting Forest Fire in Algeria Using Data Mining Techniques: Case Study of the Decision Tree Algorithm", Advanced Intelligent Systems for Sustainable Development, pp. 363-370, 2020.
DOI: https://doi.org/10.1007/978-3-030-36674-2_37.

[8] Yudong Li, Zhongke Feng, Shilin Chen, Ziyu Zhao, and Fengge Wang, "Application of the Artificial Neural Network and Support Vector Machines in Forest Fire Prediction in the Guangxi Autonomous Region, China", Discrete Dynamics in Nature and Society Volume 2020, Article ID 5612650, 2020.
DOI: https://doi.org/10.1155/2020/5612650

[9] Y.J. Goldarag, A. Mohammadzadeh, A.S. Ardakani, "Fire risk assessment using neural network and logistic regression", Journal of the Indian Society of Remote Sensing volume 44, pages 885-894, 2016.
DOI: https://doi.org/10.1007/s12524-016-0557-6

[10] Seong-Wook Park, "Detection of forest fire burned area using Landsat satellite images and Deep learning", Pukyong National University, Department of Spatial Information Engineering, Thesis, 2020.
http://pknu.dcollection.net/common/orgView/200000294613.

[11] Boris T. Polyak, "Some methods of speeding up the convergence of iteration methods", USSR Computational Mathematics and Mathematical Physics 4(5):1-17, 1964.
DOI: https://doi.org/10.1016/0041-5553(64)90137-5.

[12] John Duchi, Elad Hazan, and Yoram Singer, "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization", Journal of Machine Learning Research, 12(61):2121-2159, 2011.
http://jmlr.org/papers/volume12/duchi11a/duchi11a.pdf.

[13] Geoffrey Hinton and Tijmen Tieleman, Slide 29 in lecture 6, 2012.
https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf.

[14] Diederik P. Kingma and Jimmy Ba, "Adam: A Method for Stochastic Optimization", arXiv:1412.6980, 2019.
https://arxiv.org/pdf/1412.6980.pdf.

[15] Korea Meteorological Administration Portal, https://data.kma.go.kr/data/grnd/selectAwsRltmList.do?pgmNo=56.

[16] Geocoder-Xr, https://gisdeveloper.co.kr/?p=4784.

[17] Harris Drucker, Chris J.C. Burges, Linda Kaufman, Alex Smola and Vladimir Vapoik, "Support Vector Regression Machines", Advances in Neural Information Processing Systems 9 (NIPS 1996), pp. 155-161, 1996.
https://papers.nips.cc/paper/1238-support-vector-regression-machines.pdf

[18] Yann LeCun, Yoshua Bengio & Geoffrey Hinton, "Deep Learning", Nature, Vol. 521, 2015.
DOI: https://doi.org/10.1038/nature14539.

[19] Scikit-Learn, https://scikit-learn.org/stable/

[20] TensofFlow, https://www.tensorflow.org/

[21] Keras, https://keras.io/.

[22] Sergey Ioffe and Christian Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift", 2015.
Available From: https://arxiv.org/abs/1502.03167.

[23] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever and Ruslan Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", Journal of Machine Learning Research, 15(56):1929-1958, 2014.
http://jmlr.org/papers/v15/srivastava14a.html.

[24] Arthur Flexer, "Statistical evaluation of neural networks experiments: Minimum requirements and current practice", In Proceedings of the 13th European Meeting on Cybernetics and Systems Research, volume 2, pp.1005-1008, Vienna, Austria, 1996.
https://researchgate.net/publication/2627930_Statistical_Evaluation_of_Neural_Network_Experiments_Minimum_Requirements_and_Current_Practice.

김 삼 근(Sam-Keun Kim) [종신회원]

- 1988년 2월 : 숭실대학교 대학원 전자계산학과 (공학석사)
- 1998년 2월 : 숭실대학교 대학원 전자계산학과 (공학박사)
- 1992년 3월 ~ 현재 : 한경대학교 컴퓨터응용수학부 교수

〈관심분야〉
인공신경망, 데이터마이닝, 기계학습, IoT

안 재 근(Jae-Geun Ahn) [종신회원]

- 1994년 2월 : 서울대학교 대학원 산업공학과 (공학석사)
- 1997년 8월 : 서울대학교 대학원 산업공학과 (공학박사)
- 1997년 9월 ~ 현재 : 한경대학교 컴퓨터응용수학부 교수

〈관심분야〉
경영정보시스템, 최적화, 데이터베이스, 데이터마이닝