https://doi.org/10.7840/kics.2024.49.9.1264

GAN을 이용한 주식 시장 데이터 시뮬레이션 및 머신러닝 기반 트레이딩 시스템 개발

유 성 주•, 장 주 현*, 김 재 윤°

Development of a Stock Market Data Simulation and a Machine Learning Trading System Using GANs

Sungju Yoo•, Juhyeon Jang*, Jaeyun Kim°

요 약

머신러닝 트레이딩 시스템을 설계할 때 필수적인 과정 중 하나는 과거 주가 데이터를 사용하여 학습 모형을 구축하는 것이다. 하지만 금융시장이라는 환경의 특성상 대량의 주가 데이터를 얻는 것은 시간과 비용이 많이 요구되는 작업이다. 모델 구축을 위한 데이터가 부족할 경우 낮은 일반화 능력, 예측 능력 저하 등 다양한 문제점을 야기할 수 있다. 따라서 본 논문에서는 적대적 생성 신경망 기반의 가상 데이터 생성 시뮬레이션을 활용한 새로운 금융 시장 분석방법론을 제안한다. 현대 금융이론의 기하 브라운 운동 모형의 제한점 중 하나인 heavy-tail 현상을 극복하기 위해 GAN으로부터 추출된 난수를 입력하여 실제 주식 수익률 분포를 근사하였고 이를 통해 실제 시장의 움직임이 반영된 미래 주가 변동을 시뮬레이션 하였다. 이후, 생성된 데이터와 실제 데이터, 두 데이터 셋을 머신러닝 모델에 훈련하고 트레이딩 전략을 수립하여 거래 성과를 비교하였다. 실험 결과 제안된 방법론을 바탕으로 생성된 가상의 주가 데이터를 활용할 경우 전반적으로 거래 평가지표에서 성능이 향상되었고 이러한 결과는 금융 데이터의 제한을 극복하는 새로운 솔루션을 제시한다. 결과적으로 GAN을 활용한 데이터 시뮬레이션과 머신러닝 기반의 트레이딩 시스템은 거래 전략과 위험 관리를 향상시킬 수 있다는 가능성을 확인하였다.

키워드: 적대적 생성 신경망, 기하브라운 운동, 머신러닝, 트레이딩 시스템
Key Words : GAN, Geometric Brownian Motion, Machine Leaning, Trading System

ABSTRACT

One of the essential steps in designing a trading system is to build a training model using historical stock data. However, due to the nature of the financial markets, obtaining large amounts of stock price data is a time-consuming and expensive endeavor. The lack of data for model building can lead to various problems such as low generalization ability and poor prediction ability. Therefore, this paper proposes a new methodology for analyzing financial markets using adversarial generative neural network-based virtual data generation simulation. In order to overcome the heavy-tail phenomenon, which is one of the limitations of the geometric Brownian motion model in modern financial theory, we approximate the actual stock return distribution

※ 이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2022R1A2C1092808). 또한 순천향대학교 학술연구비 지원으로 수행하였음.
• First Author : Soonchunhyang University, Department of Future Convergence Technology, sjyoo@sch.ac.kr, 학생회원
* Corresponding Author : Soonchunhyang University, Department of AI and Big Data, kimym38@sch.ac.kr, 정회원
° Soonchunhyang University, Department of Future Convergence Technology, kwack0202@sch.ac.kr, 학생회원
논문번호 : 202405-087-E-RN, Received May 2, 2024; Revised June 6, 2024; Accepted June 9, 2024

by inputting random numbers extracted from the GAN to simulate future stock price fluctuations that reflect the actual market behavior. Afterward, we trained the machine learning model on the two datasets, the generated data and the real data, and developed a trading strategy to compare the trading performance. The experimental results showed that utilizing the fictitious stock price data generated by the proposed methodology improved overall performance on trading metrics, and these results suggest a new solution to overcome the limitations of financial data. The results confirm the potential of data simulation and machine learning-based trading systems utilizing GANs to improve trading strategies and risk management.

Ⅰ. 서 론

최근 정보통신기술(Information and Communication Technology)의 급속한 발전에 힘입어, 머신러닝(Machine Learning)과 딥러닝(Deep Learning) 기술의 진보는 사회, 경제 및 산업 분야 등 다양한 공간에서 적극적으로 활용되고 있다. 해당 기술에 대한 관심이 높아지는 가운데, 금융AI 산업에서도 미래 주가 예측, 알고리즘 기반 트레이딩, 이상 거래 탐지 등을 포함한 다양한 컴퓨터 기술의 응용을 통해서 투자자들의 원활한 매매 환경을 지원하기위한 연구가 활발히 진행되고 있다. 이러한 기술적 발전은 대량의 금융 데이터를 분석하고 복잡한 시장 동향을 이해하여 더 정확하고 효율적인 투자 결정을 수행할 수 있도록 도와주고 있다.

머신러닝을 활용한 미래 주가 예측 연구에서는 모델을 구축하는 과정에서 과거 주가 시계열 데이터를 활용한 모델 설계 과정이 필수적이다. 이는, 주식 시장에서 양질의 데이터를 활용하는 것이 곧 투자 경쟁력으로 이어질 수 있음을 의미한다. 그러나, 주식 시장의 복잡성과 데이터 제공 플랫폼의 제한으로 인해 대량의 주가 데이터를 단기간에 수집하는 것은 매우 난이도가 높은 작업일 수 있다. 이로 인해 모델 학습을 위한 데이터를 충분히 채우지 못하게 되어 예측 모델의 성능 저하로 직접적으로 이어질 수 있다. 따라서, 이와 같은 문제를 해결하기 위해 다양한 방법들이 제안되고 있다.

첫 번째 방법은 주가와 관련된 외부 데이터를 추가적으로 수집하는 것이다. 주식 시장에서의 동적인 가격 변동은 다양한 외부 요인에 의해 영향을 받기 때문에 신문, 뉴스, 소셜 미디어 등의 다양한 플랫폼에서 수집한 정보를 연결하여 데이터 부족의 문제를 보완할 수 있다.[1] 대표적인 예로, 수집된 텍스트 데이터에 감성 분석을 수행하여 도출한 감성 score를 데이터로 추가하여 분석하거나[2], 수집한 뉴스 데이터를 기반으로 토픽 지수를 개발하여 주가 변동성을 예측하는 회귀 모델을 구축하는 방법이 있다[3]. 이러한 연구들은 기존 주가 데이터에 텍스트 데이터를 결합함으로써 주가 예측의 정확도를 향상시키는 사례로 볼 수 있다.

데이터 부족 문제에 대응하는 또 다른 접근법은 기술적 분석을 통한 파생 데이터를 생성하는 것이다. 기술적 분석은 주식 시장에서 과거의 가격 움직임을 분석하여 미래 가격 변동을 예측하는 방법이며, 부족한 데이터를 보완하는 지표로 활용될 수 있다. 예를 들어, 특정 기술적 지표들을 선택하여 모델을 구축하고 코스피 주가지수의 미래 방향성을 예측[4], 새로운 기술적 지표를 생성[5], 머신러닝 모델[6]이나 딥러닝 모델[7]과 결합하는 등 다양한 연구 방법이 시도되었다. 이러한 접근 방법들은 기존 데이터의 한계를 극복하고 주가 예측의 정확도를 향상시키기 위한 다각적인 노력의 일환이 되었다.

앞선, 대다수의 연구는 주가 시계열 데이터의 각 데이터 포인트마다 추가적인 데이터 (즉, 외부 데이터와 파생 데이터)를 통합함으로써 주가의 방향성 예측 성과를 향상시키려는 공통된 목표를 지니고 있다. 그러나 이러한 연구 방법론에는 몇 가지 문제점이 존재한다.

첫째, 외부 데이터의 통합 과정에서 직면하는 주요 문제는 데이터의 품질과 신뢰성을 보장하기 어렵다는 점이다. 외부 데이터의 수집 경로와 방법은 매우 다양하며, 데이터를 수집하는 기관의 신뢰도에 따라 데이터의 질과 정확성이 크게 달라질 수 있다. 나아가, 뉴스와 소셜 미디어 네트워크와 같은 외부 데이터 소스는 불확실한 정보를 포함할 가능성이 있으며, 이러한 불확실성은 모델 구축에 부정적인 영향을 미칠 수 있다. 둘째, 파생 데이터의 한계는 결국 추가적인 정보가 원본 데이터의 보완적인 역할에 불과하다는 점에 있다. 머신러닝 기반 모델 개발 시, 기술적 지표와 같은 파생 데이터는 원본 데이터로부터 도출된다. 이 때, 원본 데이터의 양과 품질이 부족할 경우, 외부 데이터의 통합이 모델의 성능 향상에 있어 한계를 보이며, 이는 모델의 전반적인 성능 저하로 이어질 수 있다.

본 연구는 앞선 한계점들을 보완하기 위해 기하 브라운 운동(Geometric Brownian Motion, GBM)과 생성적 적대 신경망(Generative Adversarial Networks, GAN)을 활용하여 주식 시장의 복잡한 움직임을 모델

링하는 새로운 접근법을 제안한다. 주식 가격의 움직임은 불확실성과 변동성을 내포하고 있으며, 이를 효과적으로 표현하기 위해 GBM을 기반으로 한 통계적 속성과 GAN을 통한 데이터 생성 능력을 결합한다. 이러한 결합을 통해, 실제 시장 데이터에서 관찰될 수 있는 통계적 패턴을 반영하는 고품질의 가상 주가 데이터를 생성할 수 있다. 생성된 가상의 주가 데이터는 기존의 GBM 모형과 과거 데이터가 가지고 있는 한계를 극복하고, 더욱 풍부한 시나리오를 학습 데이터에 포함시킬 수 있다. 이는, 머신러닝과 금융공학 분야에서 새로운 시각을 제공하고, 주식 시장의 분석 및 트레이딩 시스템 개발에 있어 중요한 기초 자료로 활용될 수 있을 것이다.

제안된 방법을 사용해 생성된 데이터로부터 다양한 기술적 지표를 추출하고, 이를 머신러닝 모델에 학습시켜 미래 주가의 움직임을 예측하는 트레이딩 시스템을 개발한다. 평가 과정에서는 트레이딩 평가 지표를 사용한 백테스팅을 통해 모델의 수익성과 리스크 관리 능력을 종합적으로 분석한다. 제안된 접근법이 실제 시장 조건 아래에서 얼마나 효과적으로 작동하는지 실증적으로 검증하며, 전통적인 주가 예측 모델과의 비교 과정을 통해 그 우수성을 입증한다.

본 논문은 다음과 같이 구성된다. 2절에서는 본 연구와 관련된 연구에 대해 설명한다. 3절에서는 본 연구에서 제안한 프레임워크와 접근 방법을 소개하며 4절에서는 실험 방법 및 결과를 다룬다. 마지막으로 5절에서 연구 결과를 요약하고 추후 연구에 대해 논의한다.

Ⅱ. 관련 연구

2.1 기하 브라운 운동

1973년 피셔 블랙(Fischer Black)과 마이런 숄즈(Myron Scholes)에 의해 소개된 블랙-숄즈 모형은 금융 시장의 기본적 작동방식을 바탕으로 옵션 가격을 결정하는 이론적 기반을 제공한다. 해당 모형에선 파생상품의 위험분석과 가격결정을 위해 기하 브라운 운동(GBM)을 사용하였다. GBM은 주식 및 자산 가격 변동의 모델링 과정에서 사용되는 중요한 확률적 추론 과정으로, 현대 금융이론에서 핵심적인 역할을 차지한다. 이는 주가의 움직임을 랜덤워크(Random walk)의 개념과 같이 임의의 방향으로 향하는 연속적이고 무작위적인 과정으로 모델링하되, 실제 주가 움직임에서 포착할 수 있는 추세(trend)의 개념을 함께 고려한다. 또한, GBM은 주가를 로그 수익률로 변환할 때 정규분포의 성질을 보유하여 미래의 주가 변동이 정규분포의 특성 아래에서 결정된다고 가정한다. 따라서 GBM은 주가 변동의 두 가지 주요 측면인 기대수익률(또는 추세)과 변동성을 종합적으로 고려하여 주가의 미래 가치에 대해 연속적인 시간 내에서 정규분포를 따르는 확률적 과정으로 표현해 보다 현실적으로 주가 변동을 설명하게 된다.

실제 금융 시장에서 GBM은 수학적 정교함 및 주가 시계열 데이터와의 높은 적합성을 보유하기 때문에 다양한 금융 상품의 가격 결정, 위험 관리, 포트폴리오 최적화 등에 널리 활용되며, 연구자들은 이 모형의 기본 가정을 개선하고, 적용 범위를 확장하기 위한 노력을 지속하고 있다.[8,9] 미래 주가 예측을 위한 GBM의 주가 모형은 식 (1)과 같이 수학적으로 표현될 수 있다. 이는 변동성이 일정한 경우에서 GBM의 일반항을 나타내며 S_t는 특정 t시점의 주가, S_0은 초기 주가를 의미하며, e는 자연상수, μ는 로그 수익률의 기대값(추세), σ는 로그 수익률의 표준편차(변동성), W_t는 기하브라운 모형의 위너 프로세스를 의미한다. 이 때, 위너 프로세스라 함은 자산 가격의 랜덤한 변동성을 도입하는 역할을 통해 시간에 따라 불규칙하게 변동되는 주가를 모델링하게 된다. 따라서, σW_t는 주가의 변동성을 나타내는 확산항으로 GBM의 변동성을 조정한다.

$$S_t = S_0 \times e^{(\mu - \frac{\sigma^2}{2})t + \sigma W_t} \tag{1}$$

GBM은 주식의 로그 수익률이 정규분포를 따른다고 가정하지만, 실제 주식 시장에서의 로그 수익률 분포는 이론적 모델과 다소 차이를 보인다. 실제 관측된 로그 수익률 분포는 뾰족한 정점과 무거운 꼬리(heavy-tail) 현상을 나타내며, 이는 금융 시장에서 극단적인 사건의 발생 가능성을 내포한다.[10] 이러한 분포의 특성은 금융 시장의 불안정한 변동성을 반영하며, 예측하기 어려운 큰 손실의 가능성이 존재한다. 이로 인해 리스크 관리와 금융 모델링 분야에서 금융 시장의 현실적인 특성을 반영하려는 노력이 증가하고 있다.[11,12]

2.2 머신러닝을 활용한 트레이딩 시스템

머신러닝 기술의 발전은 금융 분야에서 트레이딩 전략의 혁신과 자동화된 거래 시스템의 발전을 주도하고 있다. 특히, 머신러닝 알고리즘을 기반으로 한 트레이딩 시스템은 고빈도 거래와 투자 전략 최적화에 중점을 두어 높은 수익성을 달성하고 운영 효율성을 향상시키도록 목표하여 연구가 활발히 진행되고 있다. 최근 연구들은 트레이딩 시스템 구축에 있어서 데이터 라벨링의 중요성을 강조하며, 전통적인 주가 상승 및 하락의 이진

라벨링 방식 대신 N-기간 변동성을 활용한 라벨링 기법을 적용해 모델의 예측 성과를 개선하는 방안을 제시하고 있다.[13] 또한, 그래디언트 부스팅과 같은 고급 머신러닝 기법과 유전 알고리즘을 결합하여 변수 선택과 예측 모델의 정확도를 더욱 향상시키는 연구가 진행되었다.[14] 이러한 연구는 트레이딩 전략의 시뮬레이션 과정에서 실증적으로 검증되었으며, 특히 당일 종가에 매수하여 다음 날 시초가에 매도하는 전략을 적용했을 때, 뛰어난 거래 성과를 보여주었다.

이와 같은 연구 결과들은 머신러닝이 주식 가격 예측과 트레이딩 시스템 구축에 있어 효과적으로 작동될 수 있으며, 금융 시장에서 중대한 잠재력이 내포된 것을 알 수 있다. 이와 같이 주가 시계열 데이터를 활용한 알고리즘 트레이딩은 금융 시장 분석과 투자 전략 개발에 있어 머신러닝의 응용 범위를 확장하고, 기존 모델에 비해 더 정밀하고 다양한 분석이 가능함을 의미한다.

Ⅲ. 연구 방법

본 논문의 주요 프레임워크는 그림 1과 같으며 3단계로 구성되어 있다. 첫 번째 단계에서는 머신러닝 트레이딩에 필요한 학습 데이터를 구축한다. GAN과 GBM을 활용한 가상의 주가 데이터 및 기술적 지표를 생성하며, 학습 데이터 구축을 위한 데이터 라벨링으로 구성된다. 두 번째 단계에서는 구축된 학습 데이터를 바탕으로 머신러닝 알고리즘을 적용하고 이를 기반으로 트레이딩 신호를 생성한다. 마지막 단계에서는 트레이딩 신호를 바탕으로 거래 전략을 설계하고, 해당 전략의 거래 성과를 평가한다.

3.1 데이터 생성

3.1.1 GAN을 활용한 GBM

GBM은 금융 시장에서 주가의 시간에 따른 변화를 모델링하는 데 널리 사용되는 방법이다. 이 모형은 주가

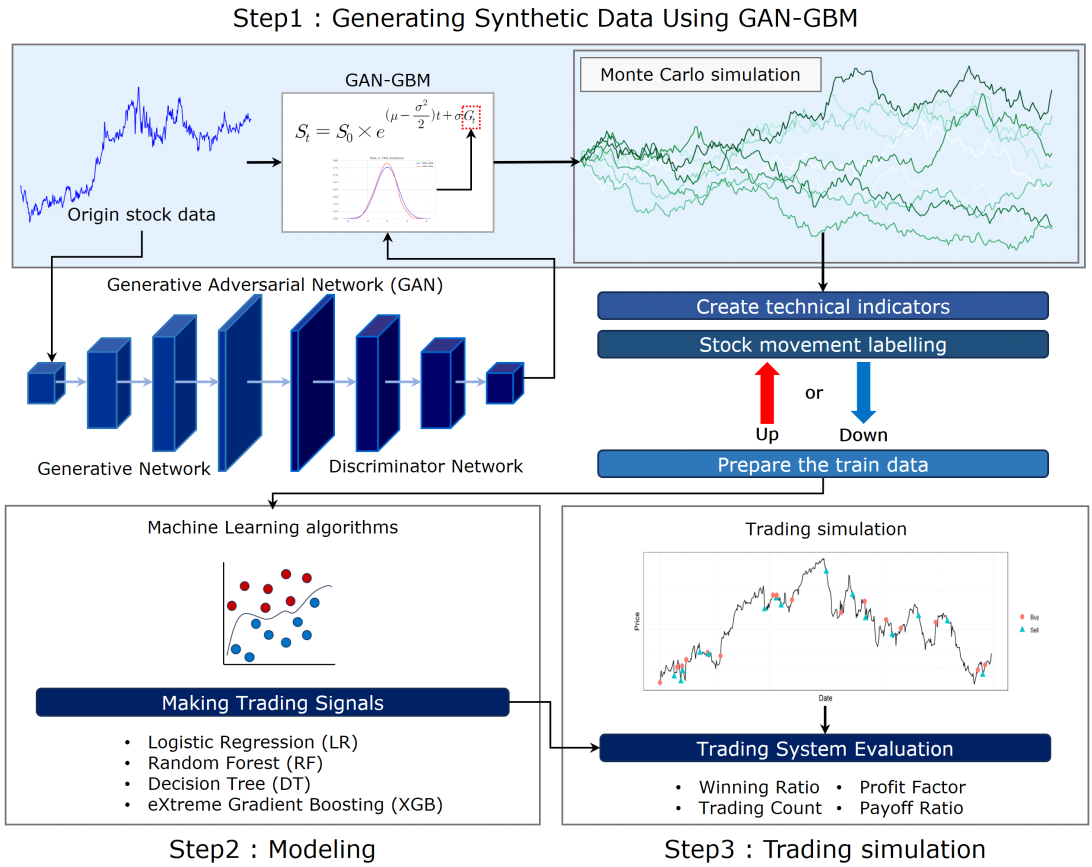

그림 1. 제안 모델의 프레임워크
Fig. 1. Proposed model Framework

가 로그 정규분포를 따른다고 가정하고 미래의 주가에 대해 정해진 정규분포 아래에서 예측하게 된다. 하지만 실제 금융 시장의 급격한 주가 변동, 극단적 사건 등으로 인하여 실제 주가 데이터는 비정규성을 보이는 경우가 많기 때문에 GBM 모형만으로 금융 시장의 변동성을 충분히 반영하기엔 효과적이지 못할 수 있다.

따라서, 이와 같은 제한점을 보완하기 위해 본 논문에서는 생성적 적대 신경망(GAN)을 활용하였다. GAN은 두 개의 신경망, 즉 생성자(Generator)와 판별자(Discriminator)를 경쟁시키며 학습하는 구조로, 주가 시계열 데이터와 같이 복잡한 데이터의 분포를 효과적으로 모델링할 수 있다. 본 논문은 GAN을 사용하여 실제 주가 데이터의 분포와 유사한 가상의 주가 데이터를 생성하게 된다. GBM 모형의 전통적인 난수 생성 방식 대신에 GAN으로부터 학습된 주가 데이터의 분포를 GBM 모형에 적용하여 실제 주가의 로그 수익률 분포를 효과적으로 표현하는 난수를 생성하게 된다. GAN으로부터 생성된 난수는 시장의 상황을 효과적으로 반영하여 GBM에 입력되기 때문에 보다 현실적으로 시뮬레이션이 진행될 수 있다. 식(2)는 제안된 GAN을 기반으로 GBM모형으로부터 가상의 주가 데이터를 생성하는 수식을 표현한다.

$$S_t = S_0 \times e^{(\mu - \frac{\sigma^2}{2})t + \sigma G_t} \tag{2}$$

본 논문에서 GAN을 활용해 난수를 생성하는 방법은 다음과 같다. 우선 실제 주가 데이터의 수익률 분포에서 μ(평균) 과 σ(표준편차)를 추정한다. 이후에 GAN 모델에 입력하여 현재 주식 시장에서 실제로 관찰되는 주가 수익률 분포를 모방하는 난수 그룹 G_t를 저장한다. 마지막으로, 저장된 그룹으로부터 무작위로 난수를 추출해 주어진 식 (2)에 적용하여 다음 주가의 움직임을 시뮬레이션한다. 해당 접근법을 통해 생성된 가상의 주가 데이터는 금융 시장의 복잡성과 비정규성을 포함한 실제 주가 변동의 다양한 특성을 더 정확하게 반영 할 수 있다.

3.1.2 몬테카를로 시뮬레이션(Monte carlo simulation)

실제 주식 시장의 수익률 분포에 기반한 GAN을 바탕으로 설계된 GBM은 미래의 주가 움직임에 대해서 시뮬레이션 과정이 가능하다. 본 논문에서 적용된 시뮬레이션 방법은 몬테카를로 방식으로, 이는 가상 데이터 생성을 위해 난수를 무작위로 추출하는 과정을 반복해 함수의 값을 수치적으로 추정하는 확률적 기법이다. 본 논문은 미래 주가 데이터 생성을 위해 시뮬레이션 횟수당 250일치의 종가데이터가 생성되도록 설정하여 총 500회의 시뮬레이션 과정을 진행하였다. 설정된 값과 같이 시뮬레이션 횟수가 500회인 경우 종목 당 250일 * 500회에 맞춰 총 125,000개의 학습 데이터가 생성되며 이는 125,000일치에 해당하는 종가 데이터를 의미한다.

3.1.3 Up/Down 라벨링

주가 데이터를 활용한 머신러닝 트레이딩 시스템 개발에서는 미래의 주가 흐름 예측을 위한 데이터 라벨링 과정이 필요하다. 이를 위한 일반적인 방법 중 하나는 Up/Down 라벨링이며 식(3)과 같이 계산된다. 이는 현재 시점을 기준으로 미래 시점의 주가의 변동을 활용해 상승 또는 하락으로 구분하는 기법이며 현재 시점을 T라고 표현할 때 T+1 시점과 비교하여, 주가가 오르면 Up(상승), 내리면 Down(하락)으로 라벨링한다.

$$\begin{cases} 1(=up) & \text{if} \quad close_{t_{+1}} > close_t \\ 0(=down) & \text{if} \quad close_{t_{+1}} < close_t \end{cases} \tag{3}$$

3.1.4 기술적 지표(Technical indicator)

기술적 지표는 과거 가격 변동과 거래량 같은 정보를 바탕으로 현재 주가에 대해 분석하는 도구이다. 기술적 지표는 실제 시장의 많은 투자자에 의해 활용되며, 특정 주식의 상대적인 평가 정도 혹은 추세 전환 여부등을 파악하는 데 유용하게 사용된다. 본 연구에서는 파이썬의 Ta-Lib를 활용하여 총 12가지의 기술적 지표를 생성하였으며, 표 1은 선정된 기술적 지표를 요약한다.

표 1. 기술적 지표 목록
Table 1. List of technical indicators

Indicator	Parameters
Absolute Price Oscillator (APO)	fastperiod=12, slowperiod=26, matype=0
Change Momentum Oscillator (CMO)	timeperiod=14
Moving Average Convergence Divergence (MACD)	fastperiod=12, slowperiod=26, signalperiod=9
MACD Signal	fastperiod=12, slowperiod=26, signalperiod=9
MACD Histogram	fastperiod=12, slowperiod=26, signalperiod=9

Indicator	Parameters
Momentum Oscillator (MOM)	timeperiod=10
Percentage Price Oscillator (PPO)	fastperiod=12, slowperiod=26, matype=0
Price Rate of Change (ROC)	timeperiod=10
Relative Strength Index (RSI)	timeperiod=14
Stochastic Fast (STOCHF)	fastk_period=5, fastd_period=3, fastd_matype=0
Stochastic RSI (STOCHRSI)	timeperiod=14, fastk_period=5, fastd_period=3, fastd_matype=0
Tripple Exponential Moving Average (TRIX)	timeperiod=30

3.2 모델링 및 매매 신호 생성

가상의 주가 시계열 데이터로부터 생성된 기술적 지표와 Up/Down 라벨링이 적용된 학습데이터를 사용해 머시러닝 기반의 트레이딩 시스템을 구축한다. 본 연구에서는 매매 신호 생성을 위해 Logistic regression(LR), Decision tree(DT), Random Forest(RF), Extreme Gradient Boosting(XGBoost) 총 4개의 알고리즘을 선정해 비교분석했다.

3.3 트레이딩 시뮬레이션

3.3.1 트레이딩 전략 설계

트레이딩 시스템을 바탕으로 생성된 예측 결과를 정제하여 매수(Buy), 매도(Sell), 홀딩(Holding), 거래 안함(No action)을 포함하는 네 가지 트레이딩 신호를 생성한다. 본 논문에서는 매수 이후 매도 신호의 발생까지 포지션을 유지하고, 매도 이후 매수 신호의 발생까지는 거래를 수행하지 않는 전략을 취한다. 표 2는 해당 전략의 이해를 돕기위한 매매 신호 생성의 예시이다. 만약 T 시점에 포지션이 없는 상태에서 T+1 시점의 예측값이 상승(Up, 1)이면 매수를 수행하고, 이후 하락이 예측되는 시점까지는 포지션을 유지하게 된다. 반대로 포지션이 없는 상태에서 T+1 시점의 예측값이 하락이라면, 이후의 시점에서 상승이 예측되기 전까지는 매매를 수행하지 않는 No action 포지션을 지속하게 된다. 이와 같은 거래 전략은 제안된 시스템을 바탕으로 생성된 매매 신호의 성과에 대해 직관적으로 평가할 수 있으며,

표 2. 거래 신호 생성 예시
Table 2. Example of trading signals

Time	Label	Signal
T	-	-
T+1	1 (Up)	Buy
T+2	1 (Up)	Holding
T+3	0 (Down)	Sell
T+4	0 (Down)	No action
T+5	0 (Down)	No action
T+6	1 (Up)	Buy
T+7	1 (Up)	Holding
T+8	0 (Down)	Sell
T+9	1 (Up)	Buy

전략을 구조화하여 객관적인 매매 의사결정을 지원할 수 있다.

3.3.2 트레이딩 성과

본 논문에서는 트레이딩 시스템의 거래 전략 성과를 평가하기 위해서 승률(Winning ratio, WR), 수익 평균, 손실 평균, payoff ratio(PR), profit factor(PF)와 같은 핵심 평가지표를 선정하였다. 승률은 설정된 기간 내에서 진행된 총 거래 중 수익이 발생된 거래의 비율을 의미한다. 식 (4)는 제안된 트레이딩 시스템을 바탕으로 진행된 거래에서 발생한 평균 수익($\overline{w}$)과 평균 손실($\overline{L}$)이다. N_{win}, N_{lose}는 전체 거래 중 수익이 발생한 거래와 손실이 발생한 거래의 수를 의미하며, $\sum w$와 $\sum L$은 수익 거래의 총 수익과 손실 거래의 총 손실을 의미한다.

식(5)은 식(4)로부터 계산된 평균 수익을 평균 손실로 나눈 Payoff ratio(P_r)와 총 수익금액을 총 손실금액으로 나눈 Profit factor(P_f)이다. P_r은 거래에서 발생된 수익과 손실 사이의 상대적 크기를 평균적으로 비교해 거래당 수익을 평가하는데 사용되며, P_f는 거래 전략의 전체적인 성과를 직관적으로 표현해 거래 전략의 수익성을 평가하는 데 사용된다. 두 지표는 제안된 거래 전략의 성과를 평가하는 중요한 척도이다. 일반적으로 이 두 가지 지표의 값은 1이상일 경우 수익성이 존재하고, 높을수록 수익성이 더욱 우수하다고 평가할 수 있다.

$$\overline{w}=\frac{\sum w}{N_{win}}, \overline{L}=\frac{\sum L}{N_{lose}} \tag{4}$$

$$P_r = \frac{\overline{w}}{\overline{L}}, \quad P_f = \frac{\sum w}{\sum L} \tag{5}$$

Ⅳ. 실 험

4.1 데이터 수집

본 논문의 제안된 시스템을 구현하기 위해, 한국종합주가지수(KOSPI)에 상장된 기업 중 시가총액 기준 상위 30개 기업을 선정하였다. 이들 각 기업의 주가 시계열 데이터를 수집하기 위해 파이썬의 금융 데이터 수집용 라이브러리인 FinanceDataReader 패키지를 활용해 데이터를 수집하였다. 표 3에선 실험을 위해 선정된 30개 기업의 목록을 요약하였다.

표 3. 주식 종목 리스트
Table 3. Stock list

Stock	Ticker	Stock	Ticker
Samsung	005930	Korea Electric Power	015760
Hyundai Motor	005380	HMM	011200
POSCO Holdings	005490	KT&G	033780
SK	034730	Samsung SDI	006400
POSCO Future M	003670	Celltrion	068270
LG H&H	051900	SK Innovation	096770
SK Hynix	000660	Doosan Enerbility	034020
Kakao	035720	Hana Financial	086790
KB Financial Group	105560	S-Oil Corp	010950
LG Electronics	066570	Naver	035420
Samsung Life Insurance	032830	Samsung C&T	028260
Korea Zinc	010130	Shinhan Financial Group	055550
LG Chemicals	051910	LG	003550
KIA	000270	SK Telecom	017670
Hyundai Mobis	012330	Korean Air Lines	003490

4.2 실험 방법

4.2.1 GAN 조기 종료

GAN 모델의 핵심적인 요소 중 하나는 조기 종료(early stopping)이다. GAN은 학습 과정에서 생성자와 판별자의 경쟁 구조가 지속되기 때문에 특정 시점 이후로 학습이 불안정해지거나 과적합(over-fitting)이 발생할 위험이 있다. 본 논문에서는 이를 위해 FID(Frechet Inception Distance)를 활용해 GAN의 적절한 조기 종료 시점을 결정하였다. FID 점수는 두 데이터 분포 간의 프레쉐(Frechet) 거리를 기반으로 산출되며, 이는 실제 데이터 분포와 생성된 데이터 분포 사이의 유사성을 측정하는 데 사용된다. 본 연구에 사용된 1D-GAN 모델에서는 실제 데이터와 생성된 데이터의 평균과 공분산을 계산하며, 이를 통해 두 분포 간의 유사성을 평가하는데 활용될 수 있다. 식 (6)은 1D-GAN에 대한 FID 점수를 계산하는 방법을 설명하며, 실제 데이터와 가상 데이터 두 가지 데이터의 분포에 대해 평균과 공분산 차이를 활용하여 유사성을 판단할 수 있다.

$$d(X,Y) = (\mu x - \mu y)^2 + (\sigma x - \sigma y)^2 \tag{6}$$

또한 본 논문에서는 GAN의 학습이 이뤄지는 epoch를 기준으로 학습종료 규칙 3가지를 생성하였다.

Ⅰ) 최소한의 학습 보장을 위해 epoch는 200회 이상
Ⅱ) 5번의 epoch 값 FID의 평균값이 0.015 보다 낮아지는 시점이 오면 학습을 강제 종료
Ⅲ) 앞선 두 가지 조건이 발생되지 않았다면 과적합 방지를 위해서 epoch 1500회에서 강제 종료

4.2.2 학습 및 평가 기간

본 논문에서 제안된 모델의 일반화 능력을 평가하기 위해 설정된 기간 내에서 학습 데이터와 테스트 데이터를 표 4와 같이 분리하여 비교분석을 수행하였다.

표 4. Train, Test data의 설정 기간
Table 4. Specification period for train and test data

Num	Train	Test
1	2017.01.01.~ 2019.12.31. (3Y)	2020.01.01.~ 2020.12.31. (1Y)
2	2018.01.01.~ 2020.12.31. (3Y)	2021.01.01.~ 2021.12.31. (1Y)
3	2019.01.01.~ 2021.12.31. (3Y)	2022.01.01.~ 2022.12.31. (1Y)

4.2.3 데이터 구축 방식 별 트레이딩 성과 비교

본 논문에서는 세 가지의 서로 다른 학습데이터 구축 방법(GAN-GBM, GBM, Historical data)을 통해 머신러닝 트레이딩 전략의 성과를 비교 분석하였다. 본 논문에서 선정한 4가지 머신러닝 알고리즘의 트레이딩 성과를 평균내어 계산했으며, 표 5는 사전에 설정한 Test기간에 해당되는 연도별 성과를 요약한 것이다.

2020년과 2021년의 실험 결과에서 본 논문의 제안 방법론인 GAN-GBM은 평균 26.207회와 26.466회의 거래를 수행했으며 각각 54.1%, 48.5%의 승률을 기록했다. 나아가, payoff ratio는 1.385, 1.243으로 나타났으며 profit factor는 1.763와 1.221로 나타났다. GBM 방법은 평균 30.722회, 30.761회의 거래가 실행됐으며, 승률은 50.5%, 43.7%로 나타났고, payoff ratio은 1.381, 1.384로 profit factor는 1.58과 1.171로 나타났다. 원본의 주식 데이터만을 사용한 경우 평균 32.899회와 34.515회의 거래 횟수가 기록됐으며, 승률은 54.6%와 48.6%, payoff ratio은 1.099와 0.983, profit factor는 1.563와 1.086이었다.

2022년 실험에서는 GAN-GBM을 활용한 전략의 거래 횟수가 평균 25.941회로 소폭 감소했으며, 승률은 48.5%로 유지하였다. 하지만, payoff ratio은 0.97, profit factor는 0.995로 감소하여 거의 손익분기점에 도달했다. GBM 모델의 거래 횟수는 평균 28.86회로 감소하였고, 승률은 51.1%로 증가했으나, payoff ratio은 0.782, profit factor는 0.91로 감소했다. 과거 주식 데이터을 사용한 전략은 거래 횟수가 평균 35.683회로 증가했으나, 승률은 47.6%로 감소했고, payoff ratio과 profit factor는 각각 0.941과 0.984로 나타나, 전반적인 성능이 떨어진 것을 확인하였다.

표 6은 본 논문에서 제안한 GAN-GBM에 대한 2020년부터 2022년까지 머신러닝 알고리즘마다 구축된 트레이딩 시스템의 비교분석 결과를 요약하며 그림 2와 3은 Test기간에서 연도별 각 머신러닝 알고리즘의 payoff ratio와 profit factor를 시각화한 것이다. 2020년 테스트 데이터 구간에서, 로지스틱 회귀 모델이 상대적으로 낮은 트레이딩 횟수에도 불구하고 높은 승률(54.2%)과 payoff ratio(2.497)로 두드러진 성과를 보였다. 이는 2.763이라는 가장 높은 profit factor로 이어졌으며, 가장 우수한 트레이딩 전략을 가진 것으로 평가할 수 있다. 그러나, 그림 2~3에서 볼 수 있듯이, 로지스틱 회귀 모델의 payoff ratio과 profit factor 분포는 비교 대상 모델들에 비해 넓은 범위에 걸쳐 있어, 더 큰 변동성을 가지고 있음을 확인할 수 있다. XGBoost와 의사결정 트리, 랜덤 포레스트 모델은 더 많은 트레이딩 기회를 탐색했지만, 로지스틱 회귀에 비해 낮은 payoff ratio과 profit factor을 기록했다.

2021년 테스트 구간에서는 XGBoost 모델의 트레이딩 평균 횟수가 29.4회로 증가하였다. 이 모델은 승률 54%와 payoff ratio은 1.204로, 1.576이라는 profit factor를 달성하였다. 의사결정 트리와 랜덤 포레스트 모델 역시 활발한 트레이딩 활동을 보였으나, payoff ratio과 profit factor에서는 XGBoost에 비해 낮은 결과를 보였다.

2022년 경우에는 로지스틱 회귀 모델의 승률이 크게 감소하여 44.4%로 떨어졌지만, 2.002의 payoff ratio을

표 5. 데이터 구축 방식 별 트레이딩 성과 비교
Table 5. Comparative trading performance by data generation methods

Year	Method	Trading count	WR	PR	PF
2020	GAN-GBM	26.207	0.541	1.385	1.763
	GBM	30.722	0.505	1.381	1.580
	Historical Data	32.899	0.546	1.099	1.563
2021	GAN-GBM	26.466	0.485	1.243	1.221
	GBM	30.761	0.437	1.384	1.171
	Historical Data	34.515	0.486	0.983	1.086
2022	GAN-GBM	25.941	0.485	0.970	0.995
	GBM	28.860	0.511	0.782	0.910
	Historical Data	35.683	0.476	0.941	0.984

표 6. 모델별 트레이딩 성과 비교
Table 6. Comparative trading performance by model

Year	Model	Trading count	WR	PR	PF
2020	LR	13.896	0.542	2.497	2.763
	XGB	24.467	0.533	0.924	1.341
	DT	30.867	0.541	1.163	1.646
	RF	35.667	0.549	0.957	1.303
2021	LR	17.923	0.456	2.269	2.382
	XGB	29.4	0.540	1.204	1.576
	DT	38.6	0.515	1.126	1.305
	RF	36.967	0.510	0.924	1.058
2022	LR	12.867	0.444	2.002	1.528
	XGB	28.033	0.486	0.931	0.976
	DT	27.833	0.509	1.053	1.290
	RF	37.133	0.502	0.985	1.090

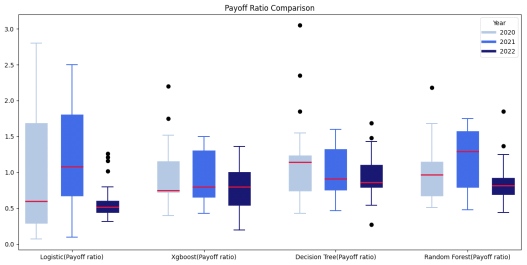

그림 2. 머신러닝 알고리즘 별 payoff ratio
Fig. 2. Payoff ratio by machine learning algorithm

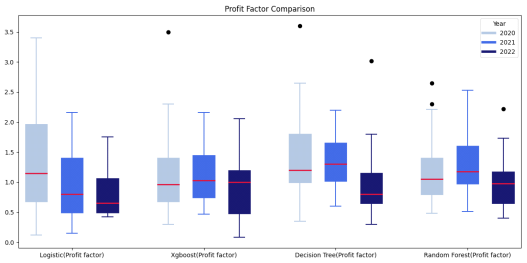

그림 3. 머신러닝 알고리즘 별 profit factor
Fig. 3. Profit factor by machine learning algorithm

유지하며 1.528의 profit factor를 기록하였다. XGBoost 모델은 이전 해에 비해 수익률이 감소하여 0.976을 기록했으며, 이는 낮은 payoff ratio(0.931)과 승률(48.6%)에 기인한다. 의사 결정 트리와 랜덤 포레스트 모델은 비슷한 트레이딩 횟수와 승률을 유지하며 안정적인 수익을 도출하였다.

V. 결론 및 향후 연구

본 논문은 머신러닝을 활용한 트레이딩 시스템 개발 시 쉽게 겪을 수 있는 학습 자료 부족의 문제를 극복하기 위해 가상의 주가 시계열 데이터를 생성하는 새로운 방법론을 제시하였다. 기존의 현대 금융시장에서 사용되던 GBM 모델의 heavy-tail 현상을 극복하기 위해, GAN으로부터 난수를 생성하는 기법을 적용하여 실제 주식 가격 변동률 분포를 효과적으로 모방해 가상의 주가 데이터를 생성한다. 생성된 가상 데이터와 실제 데이터 사이의 세밀한 비교 과정을 바탕으로 생성된 고품질의 가상 데이터를 머신러닝 모델의 입력으로 사용하여 미래 주가의 상승 및 하락(매수 및 매도 신호)을 예측하고, 이를 기반으로 거래 전략을 설계하여 트레이딩 성과를 측정하였다. 국내 주식 시장의 시총 상위 30개 종목을 대상으로 제안 시스템을 실험한 결과, 원본 데이터만을 사용할 경우에 비해 주요 거래 성과 지표에서 전반적으로 거래 성과가 향상됨을 확인하였다. 이는 제안된 방법을 바탕으로 생성된 가상의 데이터가 트레이딩 시스템 구축을 위한 데이터가 부족한 문제를 효과적으로 해결할 수 있는 가능성으로 해석하였다.

본 논문의 한계점은 다음과 같다. 첫 번째는 제안된 방법론을 바탕으로 설계된 트레이딩 시스템의 성과를 확인하기 위해 국내 시장의 시총 상위 30개 종목만을 대상으로 연구를 수행한 것이다. 이는 대형주 위주로 적용하여 결과를 확인한 것이기 때문에 국내 시장 전체를 대표하기 어려우며 상대적으로 낮은 시가총액의 종목으로 대상을 확장한다면 초과 수익 혹은 더욱 안정적인 리스크 관리 기회가 존재할 수 있다. 또한, 제안 논문의 주요한 부분에 해당되는 GAN의 학습 과정에서 조기 종료를 위한 조건 수행을 30개 종목에 일괄적으로 적용한 것이다. 만약 개별 종목의 고유한 FID값을 활용하여 최적의 값을 찾는다면 더욱 향상된 모델 성능이 나타날 수 있다. 따라서, 추후 연구에선 해외 시장, 채권, 금, 원자재 등 대상 상품을 다양하게 적용하고 개별 상품에 최적화된 FID 적용을 통해 제안된 방법의 잠재된 유용성을 살펴볼 필요성이 있다.

References

[1] P.-Y. Hao, C.-F. Kung, C.-Y. Chang, and J-B. Ou, "Predicting stock price trends based on financial news articles and using a novel twin support vector machine with fuzzy hyperplane," *Applied Soft Computing*, vol. 98, 2021. (https://doi.org/10.1016/j.asoc.2020.106806)

[2] E. Jang, H. Choi, and H. Lee, "Stock prediction using combination of BERT sentiment Analysis and Macro economy index," *J. Korea Soc. Comput. Inf.*, vol. 25, no. 5, pp. 47-56, 2020. (https://doi.org/10.9708/jksci.2020.25.05.047)

[3] K. Ko, S. Oh, and J. Baek, "Development of economic fluctuation topic indices and topic indices regression model for KOSPI200 index," *J. Korean Data and Inf. Sci. Soc.*, vol. 31, no. 4, pp. 579-594, 2020. (https://doi.org/10.7465/jkdi.2020.31.4.579)

[4] W. Lee, "A deep learning analysis of the KOSPI's directions," *J. Korean Data and Inf. Sci. Soc.*, vol. 28, no. 2, pp. 287-295, 2017.

(https://doi.org/10.7465/jkdi.2017.28.2.287)

[5] G. Ji, J. Yu, K. Hu, J. Xie, and X. Ji, "An adaptive feature selection schema using improved technical indicators for predicting stock price movements," *Expert Syst. with Appl.*, vol. 200, 2022. (https://doi.org/10.1016/j.eswa.2022.116941)

[6] J. Y. Park, R. Jaepil, and S. H. Joon, "Predicting KOSPI stock index using machine learning algorithms with technical indicators," *J. Inf. Technol. and Architecture*, vol. 13, no. 2, pp. 331-340, 2016. (UCI:G704-SER000010357.2016.13.2.013)

[7] T. Han, "Stock price prediction using LSTM: Focusing on the combination of technical indicators, macroeconomic indicators, and market sentiment," *The Soc. Convergence Knowledge Trans.*, vol. 9, no. 4, pp. 189-198, 2021. (https://doi.org/10.22716/sckt.2021.9.4.0)

[8] E. T. Mensah, A. Boateng, N. K. Frempong, and D. Maposa, "Simulating stock prices using geometric brownian motion model under normal and convoluted distributional assumptions," *Scientific African*, vol. 19, 2023. (https://doi.org/10.1016/j.sciaf.2023.e01556)

[9] C. N. Angstmann, B. I. Henry, and A. V. McGann, "Time-fractional geometric brownian motion from continuous time random walks," *Physica A: Statistical Mechanics and its Appl.*, vol. 526, 2019. (https://doi.org/10.1016/j.physa.2019.04.238)

[10] J. Beirlant, Y. Goegebeur, J. Segers, and J. L. Teugels, *Statistics of extremes: Theory and applications*, John Wiley & Sons, 2006. (https://doi.org/10.1002/0470012382)

[11] B. Kelly and H. Jiang, "Tail risk and asset prices," *The Rev. Financial Stud.*, vol. 27, Issue 10, pp. 2841-2871, Oct. 2014. (https://doi.org/10.1093/rfs/hhu039)

[12] J. Nicolau, P. M. M. Rodrigues, and M. Z. Stoykov, "Tail index estimation in the presence of covariates: Stock returns' tail risk dynamics," *J. Econometrics*, vol. 235, Issue 2, pp. 2266-2284, 2023. (https://doi.org/10.1016/j.jeconom.2023.04.002)

[13] Y. Han and J. Kim, "Developing a XGBoost trading system based on N-period volatility labeling in the stock market," *J. Korean Data And Inf. Sci. Soc.*, vol. 32, no. 5, pp. 1049-1070, 2021. (https://doi.org/10.7465/jkdi.2021.32.5.1049)

[14] P.-S. Jang, "Performance analysis of trading strategy using gradient boosting machine learning and genetic algorithm," *J. Korea Soc. of Comput. and Inf.*, vol. 27, no. 11, pp. 147-155, 2022. (https://doi.org/10.9708/jksci.2022.27.11.147)

유 성 주 (Sungju Yoo)

2022년 8월 : 순천향대학교 빅데이터공학과 학사
2024년 2월 : 순천향대학교 미래융합기술학과 석사
<관심분야> 빅데이터, 인공지능, 생성형 AI, 강화학습

장 주 현 (Juhyeon Jang)

2024년 2월 : 순천향대학교 AI·빅데이터학과 학사
2024년 3월~현재 : 순천향대학교 미래융합기술학과 석사
<관심분야> 인공지능, 머신러닝, 컴퓨터 비전

김 재 윤 (Jaeyun Kim)

2009년 8월 : 국민대학교 비즈니스IT 학과 학사
2015년 8월 : 연세대학교 정보산업공학과 박사
2018년 3월~현재 : 순천향대학교 AI·빅데이터학과 부교수
<관심분야> 금융빅데이터분석, 머신러닝, 메타휴리스틱
[ORCID:0000-0001-7855-8969]