공학 석사학위 논문

CTGAN 및 TabNet 기법을 활용한 불균형 정형 데이터 이진분류 모델링 개발

Development of Imbalanced tabular data binary classification modeling using CTGAN and TabNet techniques

아 주 대 학 교 대 학 원

인 공 지 능 학 과

성 성 민

CTGAN 및 TabNet 기법을 활용한 불균형 정형 데이터 이진분류 모델링 개발

Development of Imbalanced tabular data binary classification modeling using CTGAN and TabNet techniques

지도교수 권 순 선

이 논문을 공학 석사학위 논문으로 제출함.

2022 년 2 월

아 주 대 학 교 대 학 원

인 공 지 능 학 과

성 성 민

성성민의 공학 석사학위 논문을 인준함.

심사위원장 이 슬 인

심 사 위 원 [illegible] 인

심 사 위 원 도경아 인

아 주 대 학 교 대 학 원

2022 년 01 월 07 일

논문 요약

불균형 정형 데이터(Imbalanced Tabular data)란 관계형 데이터베이스 테이블에 담을 수 있는 데이터의 클래스 간 관측치가 현저하게 차이나는 데이터를 뜻한다. 이러한 불균형 데이터는 학습 시 다수 클래스의 예측에 편향되어 예측 정확도가 낮아지는 문제가 발생한다. 또한 딥러닝을 이용한 모델의 예측률이 우수하지만 정형 데이터에서의 연구는 상대적으로 저조한 편으로 현재까지 의사결정 나무 기반 앙상블 모델을 선호하는 경우가 많다. 그러나 최근 정형 데이터에서도 기존 모델의 성능을 개선한 딥러닝 알고리즘이 증가하고 있다.

본 논문에서 불균형 정형 데이터 이진 분류 성능의 향상을 위한 딥러닝 혼합 모델을 제안한다. 클래스 분포가 균일한 상태의 데이터를 만들기 위해 생성적 적대 신경망을 기반한 CTGAN 을 활용해 소수 클래스의 데이터를 증강하고 의사결정 나무 기반의 이점을 가진 TabNet 과 결합한 분류모델과 원본 데이터에 TabNet 을 적용한 결과와 기존 데이터 샘플링 중 오버 샘플링 기법인 SMOTE(Synthetic Minority Over-sampling Technique)과 TabNet 을 접목한 분류모델과 비교 분석한다. 또한 불균형의 비율이 다른 실제 데이터셋을 적용하여 성능을 비교하여 제안 방법론의 성능이 효과적임을 증명한다.

주제어: 불균형 정형 데이터, 데이터 증강, SMOTE, CTGAN, TabNet

목차

그림 목차

표 목차

제 1 장 서론

오늘날 IT 기술의 발달로 인해 빅데이터와 기계 학습을 통한 예측모형의 정확도를 향상하는 연구가 증가하고 있다. 기본적으로 인공지능은 성능 향상을 위해 많은 양의 데이터셋을 필요로 한다. 하지만 개인 정보 보호 문제, 특정 집단에서의 자료수집 혹은 사용된 도구 등에 따라 데이터가 편향될 수 있다. 또한 데이터를 확보해도 실제 연구목적으로 쓰이는 데이터는 중복된 값, 누락된 값, 불균형한 레이블 분포 등 일관성이 없는 경우가 대다수이다. 이러한 문제를 해결하기 위해 기계 학습 방법의 하나인 분류 분석이 자주 적용된다. 선행연구로 KNN[1], AdaBoost[2] 등이 있다. 분류 분석의 주요 목적은 정확도가 높은 모델을 찾는 것이기 때문에 선행논문의 알고리즘은 데이터의 레이블 분포가 어느 정도 균형을 이루고 있다는 가정을 기반으로 한다. 불균형 데이터 문제는 한 레이블이 다른 레이블의 샘플 수보다 훨씬 크거나 작을 때 발생한다[3]. 일반적으로 샘플 수에 따라 다수 클래스(Majority Class)와 소수 클래스(Minority Class)로 나누어진다. 두 클래스의 비율은 1:10, 1:1000, 1:10000 혹은 그 이상이 될 수 있다[4]. 데이터가 불균형한 상태에서 예측모형을 만들 때 과적합 하는 문제가 발생할 수 있다. 모델은 데이터 분포도가 높은 다수 클래스에 가중치를 크게 두기 때문에 다수 클래스의 정확도는 높아질 수 있지만, 분포도가 낮은 클래스의 정확도는 낮아지므로 소수 클래스의 예측률이 낮아지는 문제가 발생하게 된다. 또한 학습(Train) 단계에서 높은 예측률을 보이다가 테스트(Test) 세트 혹은 새로운 데이터에 적용했을 때 예측 성능 더 낮아지는 현상도 발생할 수 있다. 따라서 데이터 전처리하지 않고 선행연구의 표준 알고리즘을 사용한다면 제대로 된 결과값을 얻기 어렵다. 불균형 데이터를 다룬 선행연구로 신용카드 사기 감지[5], 고객 이탈 예측[6], 웹 스팸 탐지[7]와 의료 의사 결정[8] 등이 있다.

불균형 데이터[9, 10]의 분포를 균일하게 해결하기 위해 주로 사용되는 방법은 데이터 샘플링(Data Sampling) 이다. 전체 데이터 중에서 분석에 필요한 데이터만 뽑아내는 과정을 샘플링이라고 하며 샘플링 된 데이터가 원래의 데이터의 특징을 유지되도록 하는 것이 중요하다. 불균형 데이터는 분류된 데이터 분포가 균일하지 않은 클래스들이고 일반적으로 다수 클래스(Majority Class)와 소수 클래스(Minority Class) 두 가지가 존재하며 소수 클래스를 예측하는 모델을 만드는 경우가 많다.

데이터가 불균형한 상태에서 예측모형을 만들 때 과적합 하는 문제가 발생할 수 있다. 모델은 데이터 분포도가 높은 다수 클래스에 가중치를 크게 두어 다수 클래스의 정확도는 높아질 수 있지만 분포도가 낮은 클래스의 정확도는 낮아지므로 소수 클래스의 예측률이 낮아지는 문제가 발생하게 된다. 데이터 샘플링은 두 클래스 중 어느 쪽의 클래스의 샘플 수를 조절하는지에 따라 언더 샘플링(Under Sampling)과 오버 샘플링(Over Sampling)으로 분류된다. 언더 샘플링은 정보 손실을 가져올 수 있으므로 대부분은 언더 샘플링 보다 오버 샘플링 기법을 선호하며 효율적인 방법을 제안하는 연구들이 존재한다[11,12]. 이 연구에서 사용된 오버 샘플링 기법은 SMOTE(Synthetic Minority Over-sampling Technique)[13], Borderline-SMOTE[14]와 ADASYN(Adaptive Synthetic Sampling)[15] 등이 있다. 비슷한 기법으로 두 인공신경망인 생성기와 판별기가 상호 경쟁하며 훈련하는 데이터 증강 기법(Data Augmentation)인 GAN(Generative Adversarial Networks)[16]이 있다. GAN 은 생성자와 판별자가 서로 경쟁하며 학습하는 기업으로 생성자는 실제 데이터와 유사한 데이터를 생성하고, 판별자는 실제 데이터와 생성자가 생성한 데이터를 구분한다. 데이터의 불균형 문제를 해결하기 위해 소수 클래스의 데이터를 생성자를 통해 생성하는 연구에서 SMOTE 보다 더 우수한 성능을 보였다[17]. 하지만 기존의 GAN 은 생성되는 데이터의 제어가 불가능했기에 이를 보완시킨 기법인 조건부 확률적 생성 모델 CGAN(Conditional GAN)[18]이 제안되었다. 그 밖에 WGAN(Wasserstein GAN)[19], WGAN-GP(Wasserstein GAN with gradient penalty)[20], CTGAN[21] 등 효과적인 성능을 보이는 연구가 있다. 그 중 CTGAN 은 이미지, 음성 혹은 영상에서 주로 사용하는 증강기법인 GAN 을 불균형 정형데이터에 효과적으로 적용한 방법을 제안한다. 따라서 본 논문은 소수 클래스에 CTGAN 을 적용한 데이터와 기존의 리샘플링 기법인 SMOTE 로 오버 샘플링한 데이터를 각각 제안한 모델에 접목시켜 비교한다.

정형 데이터(Tabular data)는 데이터베이스의 정해진 형식과 구조에 따라 수치만으로 표현된 테이블 형태의 데이터를 뜻한다. 가장 많이 사용되는 정형 데이터 모델링 방법으로 LightGBM[22]과 XGBoost[23] 같은 Gradient Boosting 모델이 있다. 반대 개념인 비정형 데이터(Unstructured data)는 정해진 규칙이 없는 데이터로 이미지, 영상과 자연어 처리 등이 이에 속한다. 인공신경망의 발전으로 비정형 데이터를 사용한 연구에서 우수한 성능을 보이고 있지만 그에 비해 정형 데이터에서의

연구가 상대적으로 저조하며 현재까지도 의사결정나무 기반의 앙상블 모델을 선호하는 경우가 많다. 그러나 최근 정형 데이터에서도 기존 모델의 성능을 개선한 딥러닝 알고리즘이 증가하고 있다. 그 중 캐글(Kaggle)의 대회에서 뛰어난 성능을 보인 TabNet[24]가 있다. TabNet 은 정형 데이터에서 활용하는 의사결정나무 기반 모델의 장점을 가진 딥러닝 모델을 제안한다. 일반적인 정형 데이터와 달리 어떠한 전처리도 필요 없으며 여러 변수 중 가장 효율이 좋은 변수를 선택하여 딥러닝에 반영시키므로 해석에 용이하다.

본 연구에서 제안하는 이진 분류 예측 모델링에는 불균형 정도에 따른 여러 데이터 분류 예측을 위해 캐글(Kaggle)에서 주로 사용되며 선행연구로 언급했던 신용카드 사기 감지[5]에 사용한 데이터셋, 스페인 글로벌 은행인 산탄데르 은행 고객 데이터셋[35], 금융거래 사기 데이터셋[36], 미국 국립 당뇨병 연구소의 당뇨병 데이터셋[37]을 활용하였다. 데이터의 불균형을 해결하기 위해 소수 클래스의 데이터를 증강하는 GAN 과 TabNet 의 혼합 모델을 구조화한 후 SMOTE 로 오버 샘플링한 데이터를 TabNet 에 접목시킨 혼합 모델과 분류성능을 비교한다.

본 논문은 구성은 다음과 같다. 서론에 이어 제 2 장 선행 연구는 불균형한 데이터를 다룰 수 있는 오버 샘플링 기법, 데이터를 인공적으로 만들어 내는 기법인 생성적 적대 신경망에 관한 관련 연구와 딥러닝을 활용한 정형 데이터 네트워크에 대해 기술되어 있다. 제 3 장 제안 방법론에는 CTGAN 을 기반한 TabNet 혼합 모형 구조와 분류 성능을 평가 지표를 설명하였다. 제 4 장 데이터 분석에서 데이터셋과 구현 과정을 서술하고 제안하는 모델과 기존 모델의 분류 성능을 비교 및 분석하였다. 마지막 제 5 장 결론에서 한계점과 향후 연구에 관하여 기술하였다.

제 2 장 연구 방법론

제 1 절 불균형 데이터

불균형 데이터(Imbalanced data)는 분류된 데이터 분포가 균일하지 않은 데이터이며 다수 클래스(Majority Class)와 소수 클래스(Minority Class) 두 가지가 존재한다. 소수 클래스를 예측하는 모델을 만드는 경우가 많으므로 이를 양의 클래스(Positive Class), 다수 클래스를 음의 클래스(Negative Class)라 부르기도 한다. 클래스 불균형 문제를 발견할 수 있는 가장 직관적인 방법은 클래스 불균형 비율(Imbalance ratio, IR)를 계산하는 것이다. IR 은 다음과 같이 정의된다.

$$IR = \frac{I_{maj}}{I_{min}} \tag{2.1}$$

I_{min}은 소수 클래스의 표본 크기이고, I_{maj}는 다수 클래스의 표본 크기를 나타낸다. 일반적으로 $IR < 1$이면 클래스 불균형이 없는 상태이며 $IR \geq 1$ 일 때 숫자가 커질수록 불균형의 정도가 심해짐을 의미한다. 본 논문에서 표본 재추출 방법을 구현한 파이썬 패키지 imblearn 을 사용한다[25].

제 2 절 오버 샘플링

일반화가 낮은 모델이 생성되지 않도록 불균형 데이터의 분포를 균일하게 해결하려는 방법으로 표본 재추출 방법이 있다. 크게 두 가지 방법으로 다수 데이터셋을 소수 데이터셋 수준으로 감소시키는 언더 샘플링 기법과 소수 데이터 셋을 증식하여 충분한 데이터를 확보하는 오버 샘플링 기법이 있다. 언더 샘플링은 무작위로 샘플링하는 랜덤 언더 샘플링(Random Under Sampling)과 최근 접하는 데이터를 삭제하면서 샘플링하는 CNN(Condensed Nearest Neighbor)[26] 등이 존재한다. 의미 있는 데이터만 사용하며 시간을 단축할 수 있지만 정보의 손실이 발생할 수 있다. 반대의 개념인 오버 샘플링 기법으로 랜덤 오버 샘플링(Random Over Sampling, ROS),

SMOTE[13]과 적응 샘플링(adaptive sampling) 기법인 Borderline-SMOTE[14]와 ADASYN[15] 등이 존재한다.

1. SMOTE (Synthetic Minority Over-sampling Technique)

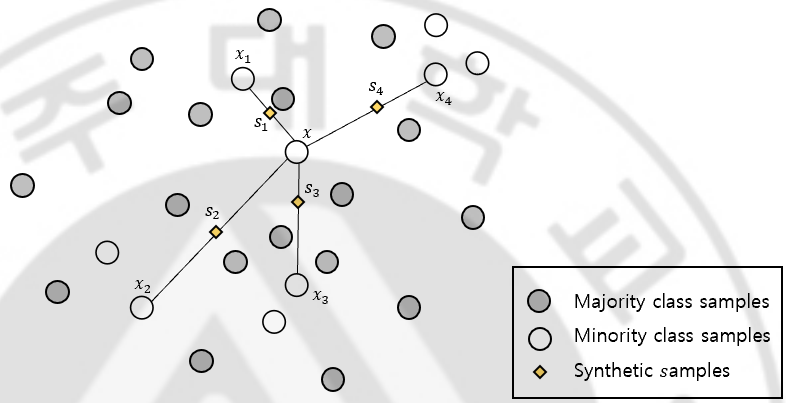

그림 1 SMOTE 기법 예시

기존에 존재하는 데이터를 단순 복제하는 것과 달리 SMOTE(Synthetic Minority Over-sampling Technique)[13]는 가장 대표적인 오버 샘플링 기법으로 소수 클래스의 데이터들을 서로 보강하여 새로운 인공 데이터를 합성한다. 이 기법은 KNN(K-Nearest Neighbors)[1] 알고리즘을 활용하여 소수 클래스 내 개별 데이터들이 k 개 이웃 간의 차이를 일정 값으로 만들고 기존 데이터와 주변 이웃을 고려하여 새로운 합성 샘플을 생성하는 방식이다. 합성 샘플의 수식과 SMOTE 알고리즘은 다음과 같다.

$$X_{syn} = X_i + (X_k - X_i) \times \delta, \delta \in [0,1] \quad (2.2)$$

X_i은 소수 클래스의 데이터 중 특정 샘플이고 X_k은 K-최근접 이웃 (K-Nearest Neighbor, KNN)의 샘플이며 δ은 0 과 1 사이의 난수이다. 기본적인 작동 원리로 소수

클래스 표본인 I_{min} 에 속하는 데이터 중 하나의 샘플 X_i에 대해 K-최근접 이웃을 수행한다. K-최근접 이웃 중 임의의 샘플 X_k을 랜덤하게 선택하고, N개의 합성 샘플들을 샘플 X_k와 X_i 사이를 잇는 지점에 랜덤하게 생성한다. 위 과정을 I_{min}과 I_{maj}이 균형을 이룰 때까지 반복한다.

오버 샘플링은 데이터 정보 손실이 없고 언더 샘플링에 비해 분류 정확도가 높지만, 계산 시간이 증가하며 이상치에 민감하게 반응할 수 있다. 이분류 불균형 데이터에서 최상의 성능을 얻는 방법으로 가장 대표적인 방법인 SMOTE 기법을 선택하였다.

제 3 절 생성적 적대 신경망

1. GAN (Generative Adversarial Networks)

생성적 적대 신경망이라고도 불리는 GAN(Generative Adversarial Networks)[16]은 두 신경망 모델인 생성자(Generator, G)와 판별자(Discriminator, D)가 경쟁적으로 대립하면서 학습한다. 생성자(G)의 목표는 실제 데이터를 미리 학습하고 실제와 가까운 가상 데이터를 생성하는 것이다. 판별자(D)는 생성자가 생성한 가상 데이터와 원본 데이터를 판별하도록 학습한다. 생성자는 판별자가 자신이 만든 가상 데이터가 원본 데이터와 유사하게 만들고 판별자는 정확하게 두 데이터를 판별하는 방향으로 학습을 진행한다. 이와 같은 학습 과정을 반복하면 원본 데이터와 완벽하게 유사한 가상의 데이터가 생성된다.

$$min_G max_D V(D,G) = E_{x\sim P_{data}(x)}[logD(x)] + E_{z\sim P_z(z)}[log(1-D(G(z)))] \quad (2.3)$$

(2.3)은 GAN의 손실함수 수식이다. V는 가치함수, E는 기댓값, D와 G는 각각 판별자와 생성자를 의미한다. x는 실제 데이터의 샘플이고 P_{data}는 실제 데이터의 분포이다. z는 노이즈 데이터의 샘플이고 P_z는 노이즈 데이터의 분포이다. $D(x)$는 실제 데이터로 판단할 확률, $G(z)$는 생성자가 만든 데이터이고 $D(G(z))$는 생성자가 만든 데이터를 판별하여 실제 데이터가 들어왔을 때 1, 생성자가 만든 가상 데이터가 들어왔을 때 0 을 출력한다. 우선 판별자는 V(D,G)를 최대화하기 위해 두 항의

$logD(x)$와 $log\,(1-D(G(z)))$가 모두 최대가 되도록 학습한다. 반대로 생성자는 첫 번째 항에 G가 포함되어 있지 않아 생략할 수 있고 두 번째 항의 $log\,(1-D(G(z)))$가 최소가 되도록 학습한다. 따라서 판별자는 V(D,G)가 최대화되도록 학습하고, 생성자는 V(D,G)를 최소화가 되도록 훈련시켜 최소한의 손실함수 값을 가지게 한다. GAN 은 데이터 분포 간의 거리를 구할 때 KLD(Kullback-Leibler divergence)나 JSD(Jensen-Shannon divergence)를 사용하며 학습데이터의 수가 적어도 생성자를 활용하여 성능을 높이는 방향으로 학습할 수 있지만, 생성자가 다양한 데이터를 만들지 못하고 비슷한 데이터만 생성하는 경우의 편향적인 학습을 하는 모드 붕괴 현상(Mode Collapse)이 나타날 수 있다. 또 다른 이유로 이미지와 영상 같은 비정형 데이터 분야에서 높은 성능을 보여주는 연구 결과는 많지만 정형 데이터 혹은 자연어 같은 텍스트 생성에서 한계점이 있다. 따라서 위 문제점을 해결하기 위한 새로운 기술인 CGAN(Conditional GAN)[18], WGAN(Wasserstein GAN)[19]과 WGAN-GP(Wasserstein GAN with gradient penalty)[20]가 제안되었으며 CGAN와 WGAN-GP의 손실함수를 활용한 CTGAN(Conditional Tabular GAN)[21]이 제안되었다.

2. CGAN (Conditional GAN)

기존의 GAN 은 생성되는 데이터의 제어가 불가능했다면 그를 보완하기 위해 CGAN(Conditional GAN)[18]은 조건부 확률적 생성 모델을 사용한다. GAN 과 학습 방식은 유사하며 손실함수의 생성자와 판별자에 조건 y가 추가되었다는 차이가 있다. CGAN 손실함수는 다음과 같다.

$$min_G max_D V(D,G) = E_{x\sim P_{data}(x)}[logD(x|y)] + E_{z\sim P_z(z)}[log\,(1-D(G(z|y)))] \quad (2.4)$$

y는 보조적인 정보로 다양한 형태를 가질 수 있고 생성자와 판별자의 입력 레이어에 구성되어 있다. 생성자에서 노이즈 $P_z(z)$와 y를 동시에 입력하고 판별자에는 원 핫 벡터(One-hot vector)로 이루어진 y와 데이터 x를 입력한다. 사용자가 원하는 조건을 설정할 수 있지만 GAN에 비해 학습 결과의 질이 떨어질 수 있다.

3. WGAN (Wasserstein GAN)

WGAN(Wasserstein GAN)[19]은 GAN 에서 발생하는 모드 붕괴 현상(Mode Collapse) 문제를 해결하기 위해 제안되었다. WGAN 은 판별자(D) 대신 비평자(Critic, C)를 설정하고 최적화를 위해 생성적 적대 신경망의 비용함수에 WL(Wasserstein loss)를 사용한다. WL 은 실제 데이터의 확률 분포 간 거리를 측정하는 척도로 EMD(Earth Mover Distance)를 사용했으며 출력값을 0 과 1 로 한정 짓지 않는다[27]. 하지만 너무 큰 출력을 피하고자 비평자가 립셔츠 제약조건(1-Lipschitz constraint function)을 걸어 기울기의 절대값을 최대 1 로 제한한다[28]. 이 조건을 위해 가중치 클리핑(Weight clipping)을 이용해 함수 가중치의 최댓값을 제한한다. WGAN 의 손실함수는 다음과 같다.

$$W(P_r, P_g) = \inf_{\gamma \in \Pi(P_r, P_g)} E_{(x,y)\sim\gamma}[\,\|x-y\|\,] \\ = \sup_{\|f\|_L \le 1} E_{x\sim P_r}[f(x)] - E_{x\sim P_\theta}[f(x)] \tag{2.5}$$

Inf(Infimum)은 최대하계, Sup(Supremum)은 최소상계를 의미한다. P_r은 실제 데이터 분포, P_g 는 가상 데이터 분포, $\Pi(P_r, P_g)$ 는 주변 분포(Marginal Probability Distribution)가 각각 P_r, P_g인 모든 결합 분포 $\gamma(x,y)$의 집합이다. 하지만 (2.5) 첫 번째 수식의 Inf 는 계산할 수 없으므로 변형시킨 두 번째 수식을 이용한다. $f(x)$은 립셔츠 함수이고 $\|f\|_L \le 1$은 랜덤한 두 점 x와 y 사이의 평균변화율이 1 을 넘지 않음을 뜻한다. 따라서 WGAN 은 생성자와 판별자의 균형 문제를 해결하고 더 부드러운 기울기를 가질 수 있지만 기울기 소실(Gradient Vanishing)과 잘못된 샘플 생성에 주의해야 한다.

4. WGAN-GP (Wasserstein GAN with Gradient Penalty)

WGAN 의 문제점을 고려하여 개선된 방법으로 WGAN-GP(Wasserstein GAN with Gradient Penalty)[20]을 제안한다. 립셔츠 제약을 실행할 수 있는 대안으로 가중치 클리핑 대신 비평자의 손실함수에 경사 패널티(Gradient Penalty)를 적용한다. 미분이 가능한

함수는 모든 곳에서 노름(Norm)이 1 이어야만 하고 벗어나면 모델에 패널티를 부과한다. 손실함수는 다음과 같다.

$$L = E_{\tilde{x}\sim P_g}[D(\tilde{x})] - E_{x\sim P_r}[D(x)] + \lambda E_{\hat{x}\sim P_{\hat{x}}}[(\|\nabla_{\hat{x}} D(\hat{x})\| - 1)^2] \quad (2.6)$$

첫 항과 두 번째 항은 WGAN 의 손실함수와 동일하며 마지막 항은 경사 패널티를 의미한다. 실제 데이터 분포인 P_r에서 샘플링 된 샘플의 쌍과 가상 데이터 분포 P_g 간의 사이 직선을 따라$P_{\hat{x}}$를 정의하고 랜덤 샘플 $\hat{x}\sim P_{\hat{x}}$에 패널티 λ를 부여한다.

5. CTGAN (Conditional Tabular GAN)

일반적으로 테이블 형식의 데이터는 연속형 변수와 범주형 변수로 이루어져 있는데 GAN 의 경우 연속형 변수와 범주형 변수를 동시에 생성하는 것과 카테고리 변수에서 심한 불균형을 띄고 있다는 문제점이 발생한다. CTGAN 는 CGAN 를 사용하여 위의 문제점을 개선했다. CTGAN(Conditional Tabular GAN)[21]은 표 형식 데이터를 합성하도록 설계된 GAN 기반 아키텍처이다. 다음은 CTGAN 의 구조이다.

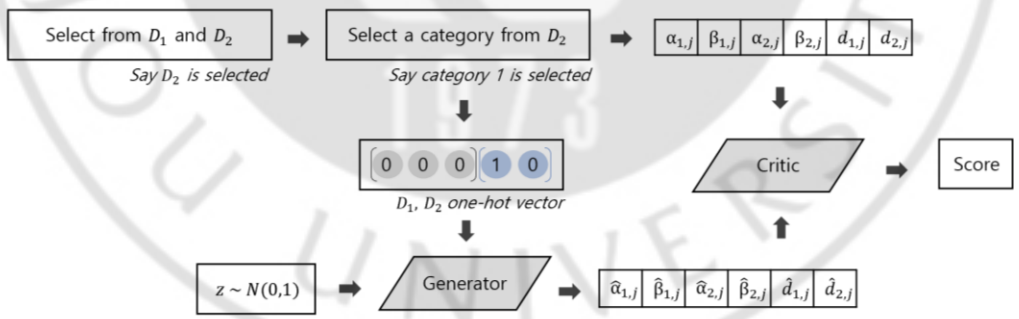

그림 2 CTGAN 의 구조

CTGAN 은 CGAN 의 조건부 방식과 WGAN-GP 의 손실함수를 사용했으며 두 가지 새로운 기법을 제안한다. 첫 번째 기술은 임의분포의 연속형 변수를 모드 별 N(0,1)

정규화하여 비가우시안(Non-Gaussian) 및 다중 모드 분포(Multimodal distribution)를 다루는 것이다. 이 전에 제안된 GAN 모델은 학습 시 가우시안 분포(Gaussian distribution)[29]을 따르므로 최소-최대 정규화 기법을 사용하여 연속된 값을 [-1,1]로 정규화 한다. 하지만 CTGAN 은 변동 가우스 혼합모델(Variational Gaussian mixture Model, VGM)을 단위 분포 모델로 사용하여 각 변수의 분포 수를 추정하고 추정된 분포에 따라 변수값을 정규화한다. 이 과정에서 인코딩된 값을 훈련 중에 원래 데이터 대신 사용한다. 훈련 후 가상 데이터를 생성할 때 생성된 데이터를 원래 규모로 변환시킨다. 두 번째 기술은 범주형 변수의 불균형 정도를 처리하기 위한 조건부 벡터(Conditional Vector)를 이용한 조건부 생성자이다. 기존의 GAN 의 생성자는 빈도 수가 큰 변수를 높은 확률로 생성한다. 따라서 다양한 데이터를 생성하기 위해 테이블 데이터의 각 열과 범주형 변수 D_1와 D_2를 [0,0,0], [1,0] 같은 조건부 벡터로 원 핫 인코딩한다. 조건부 벡터는 범주의 로그 빈도에 따라 표본 추출되어 희소(Sparse) 범주형 변수 데이터가 균등하게 추출되도록 하여 생성자의 입력 단계에 사용한다. 또한 조건부 판별자는 원하는 조건을 모방하도록 학습되는 동안 조건부 벡터 뿐만 아니라 임의의 노이즈도 입력으로 받는다. CTGAN 의 손실함수는 WGAN 과 경사 패널티를 결합한 손실함수인 WGAN-GP 손실함수[20]와 동일하다. 조건부 생성자 출력은 학습된 조건부 분포와 실제 조건부 분포 사이의 거리를 계산하면서 비평자에 의해 평가된다.

제 4 절 딥러닝 정형 데이터 네트워크

1. TabNet (Attentive Interpretable Tabular Learning)

TabNet(Attentive Interpretable Tabular Learning)[24] 알고리즘은 정형 데이터에서 예측 성능을 개선한 딥러닝 모델이다. 의사결정나무 기반의 앙상블 모델들이 우수한 성능을 보였기 때문에 정형 데이터의 딥러닝 연구가 저조했었다. 트리 기반의 모델들은 학습이 빠르고 쉽게 모델링 할 수 있고 트리 기반 모델의 특성인 변수 중요도를 구할 수 있으므로 해석에 용이하다. 반면에 딥러닝 모델은 과적합 된다는 단점으로 인해 정형 데이터에 부합하지 않았다.

TabNet 은 의사결정나무 기반의 모델과 딥러닝의 이점을 활용한 DNN 아키텍처이다. 모델은 순차적인 어텐션(Sequential Attention)을 사용하여 각 의사결정에서 추론할 변수(Feature)를 선택한다. 데이터 전처리가 불필요하며 경사 하강법(Gradient descent) 기반 최적화를 사용하여 훈련된다. 그림 3 은 TabNet 알고리즘의 전체적인 구조이다.

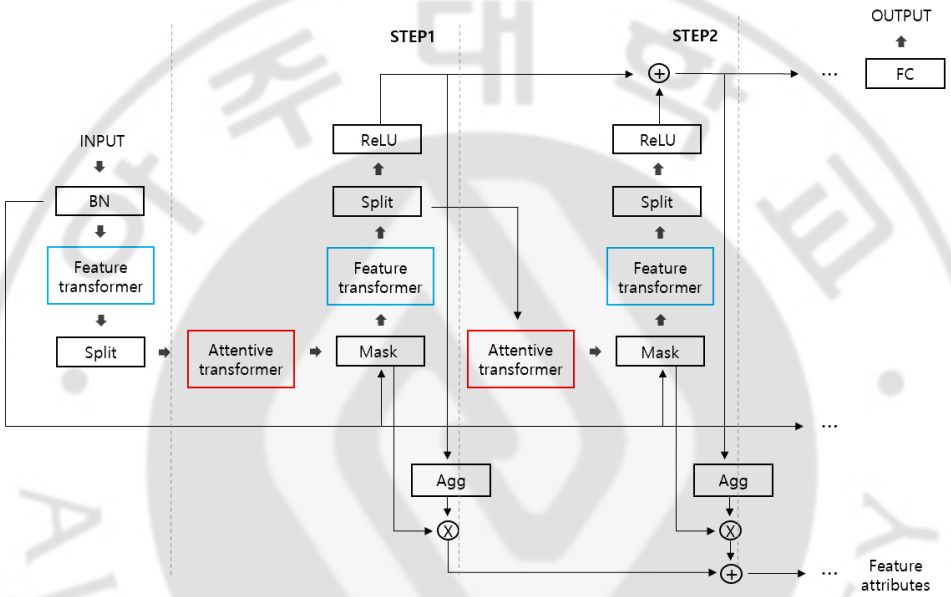

그림 3 TabNet 인코더 구조

TabNet 의 인코더는 단계가 나누어져 있고 각 단계에 대해서 크게 3 가지 변수변환기(Feature transformer), 어텐티브 변환기(Attentive transformer), 변수 마스킹(Feature masking) 블록으로 구성되어 있다. 스플릿(spilt)은 변수변환기에서 도출된 값을 2 개로 나누어 1 개는 ReLU 활성화 함수로, 나머지는 어텐티브 변환기로 전달된다. 그 이후 마스크(Mask)는 각 단계에서 작용하는 변수의 설명을 제공하고 Agg(Aggregate)를 통해 최종적으로 중요한 변수를 가린다. 이처럼 전반적인 인코더 구조는 이전 단계의 학습 결과가 다음 단계의 마스크 학습에 영향을 주는 연결형 구조이다. 따라서 TabNet 의 네트워크는 변수변환기와 어텐티브 변환기 블록을

통과하며 최적의 마스크를 학습함을 간략하게 알 수 있다. 그림 4과 그림 5는 각각 블록의 구체적인 레이어 구조를 보여준다.

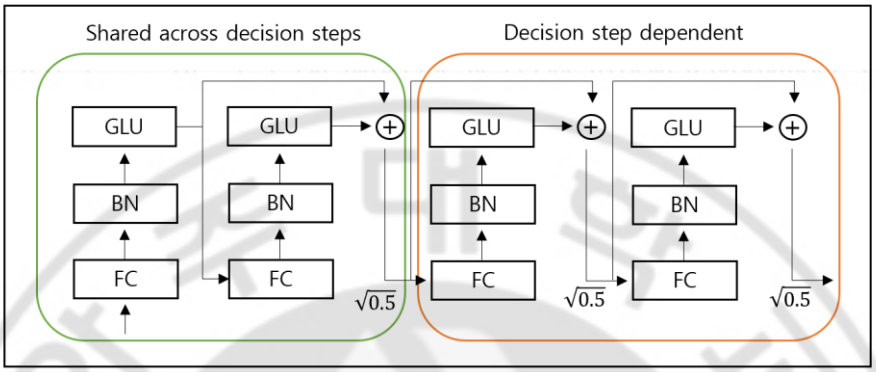

그림 4 변수변환기(Feature transformer) 레이어 구조

정형 데이터의 수치형 변수는 관련 없으나 범주형 변수는 원 핫 인코딩 등을 실행해야 한다. 위 모델은 임베딩 레이어를 구성하여 범주형 변수를 임베딩하고 임베딩 레이어 또한 학습 레이어로 구성한다. 같은 구성이 4번 반복되는 4개의 레이어 네트워크 구조로 이루어져 있는 변수변환기는 2개의 레이어는 모든 결정 단계(Decision step)에서 공유하고 나머지 2개의 레이어는 해당 결정 단계에서만 사용한다. 각 레이어는 3가지 FC(Fully-Connected Layer), BN(Batch Normalization), GLU(Gated Linear Unit)[30]로 이루어져 있다. BN은 배치를 분할한 나노 배치를 사용하여 잡음을 추가하여 지역 최적화를 예방하면 사이즈가 큰 배치로 학습 속도를 향상시키고, GLU는 이전 레이저에서 전달되는 데이터의 크기를 제어하는 임무를 수행한다. $\sqrt{0.5}$로 정규화하면 전반적으로 분산이 극적으로 변하지 않도록 학습되며 출력은 ReLU 활성화 함수를 사용한다. 변수가 변환되면 변수 선택(Feature selection)을 위해 어텐티브 변환기와 마스크로 전달된다.

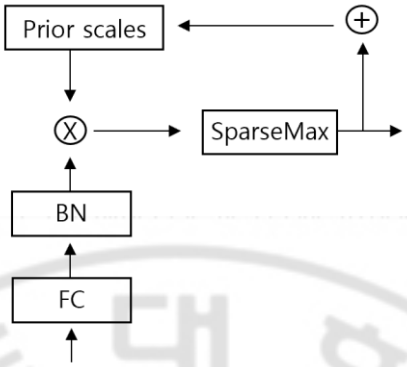

그림 5 어텐티브 변환기(Attentive transformer) 레이어 구조

그림 5는 어텐티브 변환기 레이어 구조를 보여준다. 이전 척도(prior scale)는 이전 단계에서 각 변수가 어느 정도 사용했는지 집계한 척도이고 단일 레이어에 매핑하여 사용한다. 계수의 정규화는 각 결정 단계에서 영향이 크게 미치지 않는 변수를 줄이고 가장 두드러진 특징을 희소(Sparse)하게 선택하기 위해 스펄스 맥스(Sparsemax)를 사용하여 학습한다. 스펄스 맥스는 소프트 맥스(Softmax)의 좁은 버전으로 희소한 데이터셋에 적용했을 때 우수한 성능을 보인 정규화 기법이다[31]. 마지막으로 마스크(Mask)는 가장 중요한 변수만 주목하며 설명 가능성을 도출할 때 사용된다. 이는 어텐티브 변환기에 의해 중요하다고 간주한 기능만 사용하도록 한다. 마스크를 구할 때 사용되는 수식은 다음과 같다.

$$P[i] = \prod_{j=1}^{i}(\gamma - M[j]) \tag{2.7}$$

$$M[i] = Sparsemax(P[i-1] \times h_i(a[i-1])) \tag{2.8}$$

P[i]는 γ에서 이전 마스크를 뺀 값들의 곱으로 표현되며 이전 결정 단계에서 처리된 변수와 마스크를 고려하여 새로운 마스크를 만든다. γ는 완화 매개변수(Relaxation parameter)로 $\gamma = 1$일 때 변수가 하나의 결정 단계에서만 사용되도록 만들고 γ의 값이

커질수록 여러 결정 단계에서 사용된다. 마스크(M)는 스펄스 맥스을 통해 정규화를 수행하여 결정단계에서 가장 영향력이 큰 변수를 선택하도록 한다.

TabNet 은 분류[32] 및 회기 문제[33]에서 의사결정 나무모델 대비 우세한 성능을 보였고 용량이 큰 [32]의 Higgs Boson 데이터셋에서 선행 학습의 성능도 추가로 입증했다. 본 연구는 정형 데이터셋에서의 성능 향상을 위해 CTGAN 으로 조건부 데이터셋을 만들고 TabNet 의 장점에 초점을 둔 혼합 모델로 응용한다.

제 3 장 제안 방법론

제 1 절 CTGAN 및 TabNet 혼합 모델

본 논문에서는 불균형 정형 데이터 분류의 정확도를 효과적으로 향상시키기 위해 테이블 데이터 합성에 특화된 CTGAN 과 의사결정 나무기반의 변수 선택을 특징으로 하는 딥러닝 모델 TabNet 을 혼합한 모델을 제안한다. 데이터를 증강하는 방법으로 전통적이며 성능이 좋은 오버 샘플링 기법이 있지만, 딥러닝의 이점을 활용하여 GAN 기반 모델로 데이터를 증강한다. 따라서 불균형 정형 데이터의 소수 클래스를 증강하는 방법으로 CTGAN 을 이용한다. 연속형 변수와 범주형 변수로 이루어져 있는 정형 데이터의 데이터 생성 시 발생하는 문제점을 개선한 CTGAN 은 표 형식 데이터를 합성하도록 설계된 GAN 기반 아키텍처이다. 또한 딥러닝에서 정형 데이터는 과적합 된다는 단점과 전체적인 성능이 저조하다는 이유로 딥러닝을 활용한 비정형 데이터 분석과 상대적으로 연구 성과가 적다. TabNet 은 딥러닝의 이점과 앙상블 모델의 이점을 결합한 DNN 아키텍처이다. 따라서 CTGAN 으로 생성한 데이터를 TabNet 에 적용하여 향상된 성능을 입증한다. 그림 6 은 제안 방법론의 전체 구조이다.

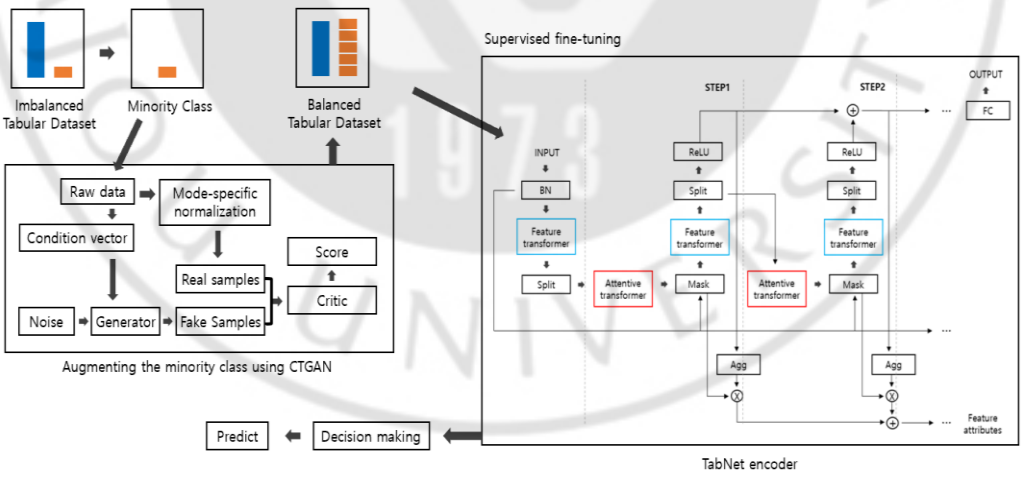

그림 6 제안 방법론의 전체 구조

제안하는 모델의 성능을 증명하기 위해 두 가지 비교모델과 함께 실험한다. 첫 번째로 아무런 전처리를 하지 않은 원본 데이터에 TabNet 모델을 사용한다. 두 번째는 원본 데이터를 SMOTE 로 오버 샘플링 한 후 TabNet 모델을 적용한다. 세 가지의 서로 다른 모델을 비교하여 CTGAN 과 TabNet 을 결합한 모델 성능의 우수함을 증명하는 것을 목표로 한다. 모델에 사용되는 모든 데이터는 Train : Test : Validation 의 비율은 80:10:10 으로 설정한다. 그림 7 는 기존 방법론과 제안방법론 실험과정의 도식화이다.

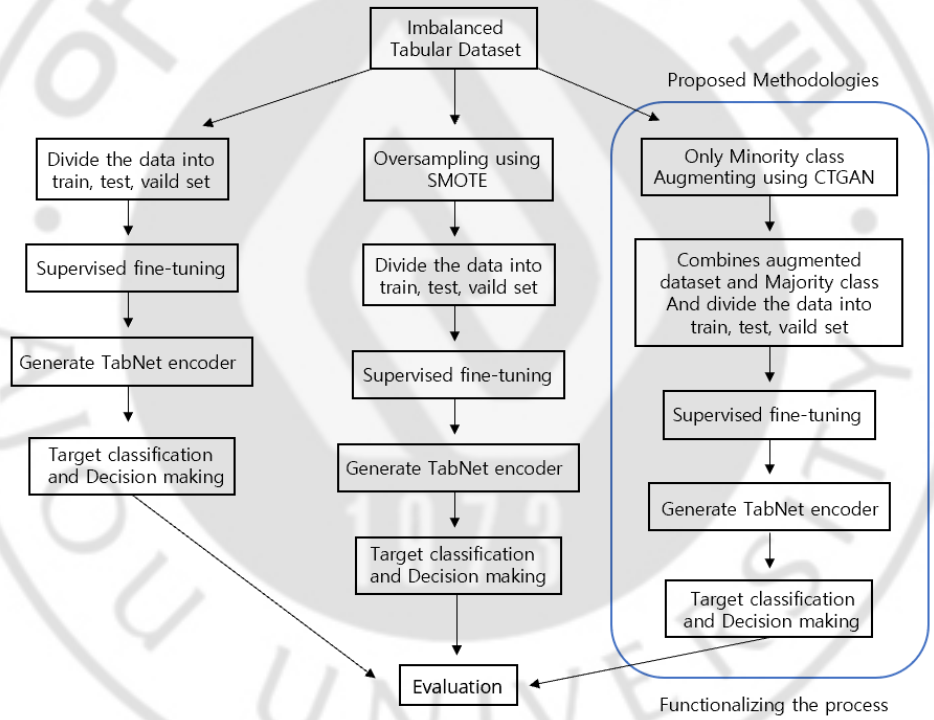

그림 7 실험과정 도식화

제 2 절 구현 세부사항

표 1 사전 정의된 매개변수

Network	Hyperparameter		Value
CTGAN	epochs		15
TabNet	n_d		24
	n_a		24
	n_steps		3
	gamma		1.3
	optimizer		Adam
	learning_rate		0.02
	scheduler_params	gamma	0.9
		step_size	50
	mask_type		Sparsemax
	max_epochs		15
	batch_size		1024
	virtual_batch_size		128

표 1은 본 연구에서 사전 정의한 매개변수의 테이블이다. CTGAN의 파라미터인 에폭(epochs=15)는 반복학습 횟수이다. TabNet의 파라미터인 n_d는 결정 예측 레이어의 너비로 값이 클수록 과적합 위험이 있는 모델에 더 큰 용량을 제공한다. n_a는 각 마스크에 대한 어텐션 임베딩의 너비이다. [24]에 따라 n_d와 n_a의 값이 같을 때 더 좋은 선택이라 설명하므로 n_d와 n_a 모두 24로 설정했다. n_steps=3는 아키텍처의 단계 수, gamma=1.3는 마스크의 기능 재사용에 대한 계수이며 1에 가까울수록 마스크 선택이 레이어 간의 상관관계가 낮아진다. Optimizer은 Adam, Learning rate는 0.02으로 최적화하였고 mask_type은 스펄스 맥스(Sparsemax)을 사용했다. scheduler_parms는 모델 학습 중 학습률을 변경하는 Pytorch 스케줄러의 매개변수를 설정하는 변수로 gamma는 0.9, step_size는 50으로 설정하였다. 학습의 최대 epoch인 max_epochs는 15, 배치당 예제 수인 batch_size는 1024, 미니 매치 크기인 virtual_batch_size는 128로 설정하였다.

SMOTE와 CTGAN을 활용하여 오버 샘플링시 소수 클래스와 다수 클래스 데이터 수의 비율(ratio)은 0.4, 0,7, 1이 되도록 설정했다. 기존 방법론인 SMOTE의 데이터 샘플링을 위해 매개변수 sampling_strategy를 사용한다[25]. 이 매개변수는 소수 클래스를 다수 클래스의 일정 비율만큼 리샘플링한다. 비율은 $\frac{I_{rmin}}{I_{maj}}$로 I_{rmin}는 리샘플링 이후의 소수 클래스 표본 크기이며 I_{maj}는 다수 클래스의 표본 크기이다. 본 논문에서 사용할 변수 ratio는 위 비율로 정의한다.

개별환경은 주피터 노트북(Jupyter Notebook)에서 이루어졌으며 Python 3.8.3과 파이썬 패키지 numpy 1.18.5, pandas 1.1.4, imblearn[25] 0.7.0와 ctgan[21], sklearn 0.24.2, torch 1.9.0+cpu, pytorch-tabnet [34]을 사용한다.

제 3 절 분류 성능 평가 지표

표 2는 이진 분류 오차 행렬(Binary classification Confusion matrix)로 모델의 예측값이 실제 관측값을 얼마나 정확하게 예측했는지 보여주기 위한 행렬이다. 혼동 행렬은 양성을 양성으로 정확하게 예측하는 TP(True Positive), 양성을 양성이 아니라고 잘못 예측하는 FP(False Positive), 음성을 음성이라고 정확하게 예측하는 TN(True Negative)와 음성을 양성으로 잘못 예측하는 FN(False Negative)이 있으며 총 4가지로 구성되어 있다.

표 2 이진 분류 오차 행렬

	Positive	Negative
Positive	True Positives (TP)	False Negatives (FN)
Negative	False Positives (FP)	True Negatives (TN)

이진 분류 혼동 행렬로부터 평가점수를 계산하여 그 점수를 최종적으로 모델을 평가하는 기준으로 자주 사용된다. 불균형 데이터에서 많이 쓰이는 연구 결과의 평가지표로 정확도(Accuracy), 정밀도(Precision), TPR(True Positive Rate),

FPR(False Positive Rate) 그리고 F1 점수(F1 Score) 등이 있다. 본 논문은 데이터셋의 불균형 정도가 크기 때문에 분류 성능 평가 지표로 정확도보다 ROC-AUC Score 로 평가하는 것이 적합하다고 판단하였다.

$$TPR = \frac{TP}{TP + FN}, \qquad FPR = \frac{FP}{FP + TN} \tag{3.1}$$

ROC 곡선(Receiver Operating Characteristic Curve)과 이에 기반한 AUC(Area Under Curve)는 이진 분류 모델의 성능 평가로 중요하게 사용된다. ROC curve의 x축은 FPR과 TPR이며 AUC는 아래의 면적(Area Under the ROC curve)을 이용하여 측정한다. 지표에 사용되는 변수 중 FPR은 실제 음성을 잘못 예측한 비율이고 TPR은 재현율과 같은 뜻으로 실제 양성인 것 중에 예측과 실제 값이 양성으로 일치하는 데이터의 비율을 의미한다. 면적은 0에서 1사이 값으로 1에 가까울수록 예측률이 높은 모델이다.

제 4 장 제안한 방법론 적용

제 1 절 데이터셋

본 논문에서 제안하는 CTGAN과 TabNet 기법을 활용한 이진 분류 모델을 검증하기 위해 불균형 정도가 각기 다른 불균형 정형 데이터셋을 사용하여 연구를 진행하였다. 실험에 사용된 데이터셋은 네 종류로 IR 크기 순서대로 당뇨병 데이터[37], 금융거래 사기 데이터[36], 은행 고객 데이터[35], 신용카드 사기 감지[5]이다. 표 3은 실험에 사용된 네 데이터셋의 세부 사항이다.

표 3 데이터셋 세부사항

Dataset	Diabetes	Fraud	Santander	Credit
Features	9	113	371	31
Total data	768	20,468	76,020	284,807
Majority Class	500	15,030	73,012	284,315
Minority Class	268	5,438	3,008	492
IR	1.87	2.76	24.27	577.88

당뇨병 데이터셋(Diabetes)[37]은 미국 국립 당뇨병 연구소(National Institute of Diabetes and Digestive and Kidney Diseases)에서 수집되었고 진단 측정을 기반으로 환자의 당뇨병 유무를 예측한다. 총 데이터는 768개, 변수 9개이고 타겟 레이블이 0이면 정상, 1이면 당뇨병으로 500명은 정상, 268명은 당뇨병으로 예측한다. 네 개의 데이터 셋 중 불균형 정도가 가장 낮다.

금융거래 탐지 데이터셋(Fraud)[36]은 실제 은행 데이터에 PySpark를 사용하여 만든 가상 데이터셋으로 총 20,468개의 데이터와 113개의 변수로 구성되어 있다. 타겟 레이블이 0이면 정상, 1이면 비정상 거래를 의미하고 각각 데이터 수는 15,438와 5,439로 불균형임을 알 수 있다.

은행 고객 데이터셋(Santander)[35]은 스페인 글로벌 은행인 산탄데르(Santander) 은행 고객 데이터로 총 76,020개의 데이터와 371개의 변수로 구성되어 있고, 주어진 데이터셋을 기반으로 고객의 만족 여부를 예측한다. 변수는 모두 익명 처리되어 어떤 속성

인지 추정할 수 없으며 타겟 레이블이 0이면 만족, 1이면 불만족한 고객이다. 클래스 분포는 만족 73,012건, 불만족 3,008건으로 불균형 정도가 크다.

2013년 9월 유럽의 신용카드 사용자들의 실제 거래기록인 신용카드 사기 감지 데이터셋(Credit)[5]은 총 284,807건의 거래 내역이다. 이 데이터셋은 머신 러닝 그룹(Machine Learning Group)과 Worldline의 빅데이터 마이닝 및 사기 탐지에 관한 연구 협력 중 수집되었고 데이터는 보안상의 문제로 시간과 양을 제외한 변수는 PCA(Principal Component Analysis)로 변환된 결과인 수치 입력 변수만 사용되었다. 변수는 31개이고 타겟 레이블은 0이면 정상, 1이면 사기를 의미하며 총 정상은 284,315건, 사기는 492건으로 매우 심한 불균형을 보인다.

제 2 절 데이터 분석 결과

본 절은 TabNet, SMOTE 과 TabNet, 제안하는 모델인 CTGAN 과 TabNet 의 결합 모델의 분류 성능 결과를 비교하고 분석한다. 표 4 는 분류 결과의 AUC 점수를 나타낸다. 표의 비율(Ratio)는 오버 샘플링 이후의 소수 클래스 표본 크기 분의 다수 클래스의 표본 크기이다. 그림 8 은 AUC 점수를 시각화한 결과이다.

다만, 당뇨병 원본 데이터의 다수 클래스와 소수 클래스의 비율이 0.536 으로 비율을 0.4 로 오버 샘플링 하는 것은 불가능하므로 당뇨병 데이터는 비율 0.4 를 제외하고 나머지 데이터셋에 대해서 리샘플링 비율 0.4, 0.7, 1 로 설정하였다. 표 4 와 그림 8 은 각 데이터셋의 원본 데이터에 TabNet 을 적용한 결과와 소수 클래스를 증강한 후 SMOTE 와 TabNet 을 결합한 모델과 CTGAN 과 TabNet 을 결합한 모델을 적용한 결과를 보여준다.

표 4 TabNet, SMOTE + TabNet 및 제안하는 모델의 분류 결과 AUC 점수

Dataset	Model	AUC Score		
	Ratio	0.4	0.7	1
Diabetes	TabNet	-	0.645	0.403
	SMOTE + TabNet	-	**0.656**	0.550
	CTGAN + TabNet	-	0.609	**0.732**
Fraud	TabNet	0.947	0.954	0.955
	SMOTE + TabNet	0.962	0.952	0.950
	CTGAN + TabNet	**0.999**	**0.999**	**0.998**
Santander	TabNet	0.789	0.810	0.812
	SMOTE + TabNet	0.796	0.814	0.795
	CTGAN + TabNet	**0.994**	**1.000**	**0.999**
Credit	TabNet	0.939	0.966	0.987
	SMOTE + TabNet	0.961	0.962	0.967
	CTGAN + TabNet	**1.000**	**0.999**	**0.999**

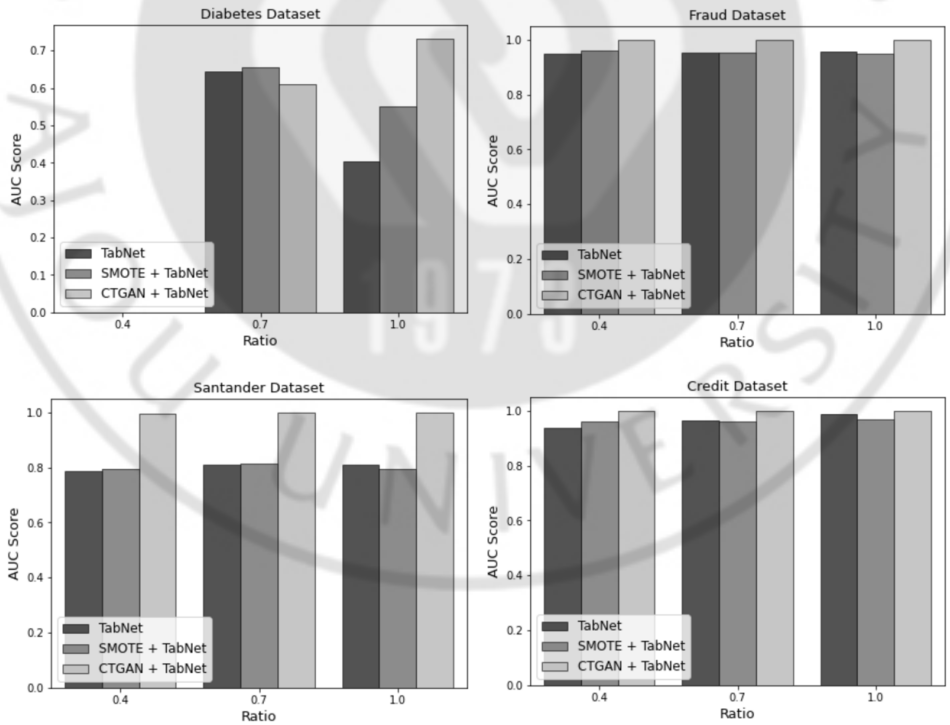

그림 8 데이터별 기존방법론과 제안방법론의 AUC Score 를 통한 성능비교

표 4의 굵은 숫자는 데이터별 비율별 가장 높은 점수를 의미한다. 분류 성능 분석 결과 당뇨병을 0.7 비율로 오버 샘플링한 결과를 제외하고 모든 데이터셋에서 제안하는 모델이 우수한 성능을 보인다. 당뇨병을 0.7의 비율로 리샘플링시 결과가 가장 좋지 않은 반면 1의 비율로 데이터를 증강시켰을 때 비교 방법론보다 결과가 확연히 높은 성능을 보이며, 산탄데르 데이터셋의 결과에서도 제안 방법론이 큰 차이로 높은 성능을 보였다. 이는 제안방법론이 데이터셋이 적을 때 다수클래스와 비슷한 비율로 소수 클래스를 증강하면 예측률이 크게 향상됨을 나타낸다. 또한 가장 데이터수가 적었던 당뇨병 데이터셋을 제외했을 때 모든 경우의 AUC 점수가 1에 가깝게 평가되었다. 이는 데이터 개수가 많을수록 모델의 분류성능이 향상됨을 알 수 있다.

표 5 TabNet, SMOTE + TabNet 및 제안하는 모델의 분류 결과 도출 시간(시:분:초)

Dataset	Model	Time		
	Ratio	0.4	0.7	1
Diabetes	TabNet	-	**0:00:02**	0:00:03
	SMOTE + TabNet	-	**0:00:02**	**0:00:02**
	CTGAN + TabNet	-	0:00:03	0:00:03
Fraud	TabNet	0:08:22	0:08:26	0:08:32
	SMOTE + TabNet	0:08:12	**0:08:16**	**0:08:19**
	CTGAN + TabNet	**0:07:19**	0:09:45	0:11:03
Santander	TabNet	1:39:44	1:45:39	1:44:25
	SMOTE + TabNet	1:39:34	1:43:34	**1:42:24**
	CTGAN + TabNet	**1:09:31**	**1:39:49**	2:39:18
Credit	TabNet	0:18:21	0:17:27	0:17:08
	SMOTE + TabNet	**0:15:48**	**0:16:17**	**0:16:17**
	CTGAN + TabNet	0:27:02	0:31:32	0:29:50

표 5는 TabNet, SMOTE 및 TabNet 결합모델과 제안하는 모델인 CTGAN과 TabNet 결합모델을 시,분,초 단위로 전체 학습의 도출 시간을 나타낸다. 굵은 숫자는 데이터별 비율별 가장 소요 시간이 적은 것을 의미한다. 표 4에서 제안모델은 AUC 점수를 비교하

였을 때 뛰어난 성능을 보였지만 대부분의 경우 소요되는 학습 시간이 길다는 것을 확인할 수 있다. SMOTE 와 TabNet 을 결합한 모델이 소요되는 시간이 가장 적다. 이는 두 개 이상의 딥러닝 모델을 결합했을 때 수많은 파라미터를 고려해야 함으로 시간적 한계점을 확인할 수 있다.

표 6 TabNet, SMOTE + TabNet 및 제안하는 모델의 분류 결과 최적의 에폭

Dataset	Model	Best epoch (Max epoch = 15)		
	Ratio	0.4	0.7	1
Diabetes	TabNet	-	**12**	1
	SMOTE + TabNet	-	2	4
	CTGAN + TabNet	-	**14**	**14**
Fraud	TabNet	**14**	**14**	**14**
	SMOTE + TabNet	**14**	**14**	**13**
	CTGAN + TabNet	**14**	**13**	**14**
Santander	TabNet	5	6	6
	SMOTE + TabNet	9	5	8
	CTGAN + TabNet	**12**	7	3
Credit	TabNet	**12**	1	8
	SMOTE + TabNet	1	**13**	**12**
	CTGAN + TabNet	2	1	**12**

표 6 은 최대 에폭을 15 으로 설정했을 때 최적의 에폭 수를 나타낸다. 굵은 숫자는 데이터별 비율별 가장 높은 에폭 수를 표시하였으며, 데이터별 최적의 에폭 수가 12 이상인 것은 당뇨병 데이터 3 개, 금융거래 사기 데이터 9 개 산탄데르 은행 데이터 1 개, 신용카드 사기 데이터에서 4 개인 것을 확인할 수 있다. 이는 분명한 패턴은 없지만, 데이터 수가 상당하거나 변수의 개수가 많을수록 최적 에폭 수가 커진다는 것을 추측할 수 있다.

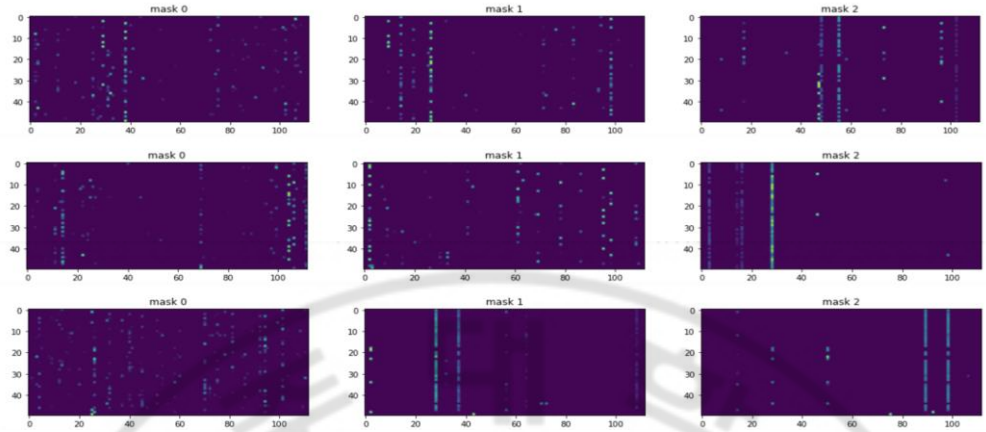

그림 9 Fraud 데이터셋 비율별 변수 중요도 마스크 시각화

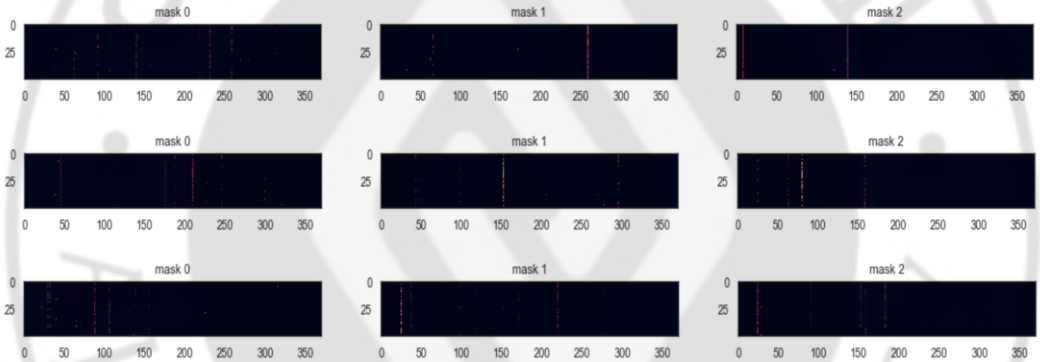

그림 10 Santander 데이터셋 비율별 변수 중요도 마스크 시각화

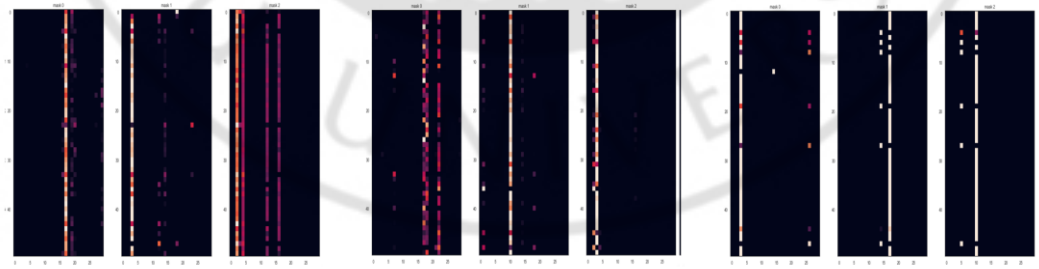

그림 11 Credit 데이터셋 비율별 변수 중요도 마스크 시각화

그림 9, 그림 10, 그림 11 각각 금융거래 사기, 산탄데르 은행, 신용카드 사기 데이터셋에서 비율별 마스크 값을 활용한 변수 중요도 시각화 자료이다. 표 1 의 매개변수 중 n_steps 를 3 으로 지정했으므로 각 데이터의 시각화는 3 개씩 생성되었고, 50 으로 설정한 step_size 는 테스트 샘플 수이다. 따라서 시각화 그래프의 x 축은 데이터셋의 변수 개수이고 y 축은 테스트 샘플 수임을 알 수 있으며 선이 밝을수록 변수의 영향력이 큰 것을 의미한다. 마스크는 모든 검증 데이터에 대해 각 어텐티브 변환기 단계에서 마스크를 적용한 후 활성화 비율로 표현하며 지역적인 특징을 확인할 수 있다. 이는 마스킹을 이용해 예측단계에 사용된 변수의 해석 가능성을 의미한다.

위 그림들을 살펴보았을 때 비율별 세 가지 샘플마다 변수의 영향력을 확인할 수 있다. 위 3 가지 변수 중요도 시각화 중 가장 변수의 개수가 적은 그림 11 의 신용카드 사기 데이터셋을 보면 리샘플링 비율이 1 일 때 0 번 샘플은 3, 1 번 샘플은 17, 2 번 샘플은 10 번 변수, 리샘플링 비율이 0.7 일 때 0 번 샘플은 22, 1 번 샘플은 10, 2 번 샘플은 3 번 변수, 리샘플링 비율이 0.4 일 때 0 번 샘플은 17, 1 번 샘플은 3, 2 번 샘플은 2 번째 변수가 영향력이 높은 것을 확인 할 수 있다. 이 중 반복적으로 영향력이 높은 변수는 3 번째 변수이며 17 번과 10 번째 변수도 중복되어 나왔으므로 중요하다고 평가할 수 있다.

제 5 장 결론

본 논문에서는 불균형한 정형 데이터의 분류성능을 높이기 위해 두 가지 딥러닝 모델인 조건부 방식을 활용한 GAN 기반 모델과 심층 데이터 분석 아키텍처를 결합한 방법을 제안하였다. 이를 통해 제안하는 모델이 불균형 데이터의 문제를 해결함과 동시에 기존의 분류성능보다 향상된 결과를 보였다. 성능 평가를 위해 불균형 정도가 다른 네 개의 데이터셋인 당뇨병 데이터셋, 신용카드 사기 데이터셋, 산탄데르 은행 데이터셋, 금융거래 사기 데이터셋을 활용하였다. 모델 학습의 결과, 본 논문에서 제안하는 모델의 분류성능이 나머지 두 비교 모델보다 뛰어난 성능을 보였다. 특히 데이터 샘플수가 적을 때 큰 편차로 높은 성능을 보였다. 또한 데이터 수가 적었던 당뇨병 데이터셋의 결과를 제외하고 다른 데이터셋에서 제안된 모델의 AUC 점수가 1 에 가깝게 나온 것을 미루어 보아 데이터의 양이 많을수록 우수한 성능을 보임을 알 수 있다. 하지만 기존의 모델과 다르게 변수 선택과 모델 학습 단계가 나누어지지 않고 한 번에 가능하다는 장점이 있지만 그만큼 조절해야 하는 파라미터의 수가 많아 소요되는 시간이 늘어난다는 단점을 보였다. 변수 중요도 마스크 시각화는 어떠한 변수에 높은 우선순위를 주었는지 확인할 수 있지만 이러한 결과가 실제 변수값과 관련이 있는지 불분명하다. 또한 TabNet 은 비지도학습에서도 상당한 성능 향상을 보이지만 본 논문에서는 지도학습만 이루어졌다는 한계점도 있다. 향후 연구는 비지도학습을 포함한 데이터셋을 입력 데이터로 사용하고 정형 데이터에서 딥러닝을 활용한 관련 연구를 지속하여 모델의 학습 소요시간을 단축할 수 있는 방안을 고려한다. 마지막으로 제안하는 모델은 아직도 특정 데이터셋에서만 우수한 성능을 보이므로 광범위한 데이터셋에서 성능을 높일 수 있는 방안을 제안하고자 한다.

참고문헌

[1] N. S. Altman (1992) An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression, The American Statistician, 46:3, 175-185, DOI: 10.1080/00031305.1992.10475879

[2] Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 55(1), 119-139.

[3] Luque, A., Carrasco, A., Martín, A., & de las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognition*, *91*, 216-231.

[4] Chawla, N. V., Japkowicz, N., & Kotcz, A. (2004). Special issue on learning from imbalanced data sets. *ACM SIGKDD explorations newsletter*, *6*(1), 1-6.

[5] Panigrahi, S., Kundu, A., Sural, S., & Majumdar, A. K. (2009). Credit card fraud detection: A fusion approach using Dempster-Shafer theory and Bayesian learning. Information Fusion, 10(4), 354-363.

[6] Xie, Y., Li, X., Ngai, E. W. T., & Ying, W. (2009). Customer churn prediction using improved balanced random forests. Expert Systems with Applications, 36(3), 5445-5449.

[7] Lu, X. Y., Chen, M. S., Wu, J. L., Chang, P. C., & Chen, M. H. (2018). A novel ensemble decision tree based on under-sampling and clonal selection for web spam detection. Pattern Analysis and Applications, 21(3), 741-754.

[8] Mazurowski, M. A., Habas, P. A., Zurada, J. M., Lo, J. Y., Baker, J. A., & Tourassi, G. D. (2008). Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. Neural networks, 21(2-3), 427-436.

[9] Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent data analysis*, *6*(5), 429-449.

[10] He, H., & Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on knowledge and data engineering, 21(9), 1263-1284.

[11] Cho, H. Y., & Kim, Y. H. (2020, July). A genetic algorithm to optimize SMOTE and GAN ratios in class imbalanced datasets. In Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion (pp. 33-34).

[12] H. Y. Cho. (2020). Optimization of Data Oversampling Ratio Using a Genetic Algorithm. Master's thesis. Kwangwoon University, Seoul.

[13] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357.

[14] Han, H., Wang, W. Y., & Mao, B. H. (2005, August). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In International conference on intelligent computing (pp. 878-887). Springer, Berlin, Heidelberg.

[15] He, H., Bai, Y., Garcia, E. A., & Li, S. (2008, June). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence) (pp. 1322-1328). IEEE.

[16] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.

[17] Douzas, G., & Bacao, F. (2018). Effective data generation for imbalanced learning using conditional generative adversarial networks. Expert Systems with applications, 91, 464-471.

[18] Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784.

[19] Arjovsky, M., Chintala, S., & Bottou, L. (2017, July). Wasserstein generative adversarial networks. In International conference on machine learning (pp. 214-223). PMLR.

[20] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. (2017). Improved training of wasserstein gans. arXiv preprint arXiv:1704.00028.

[21] Xu, L., Skoularidou, M., Cuesta-Infante, A., & Veeramachaneni, K. (2019). Modeling tabular data using conditional gan. arXiv preprint arXiv:1907.00503.

[22] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 30, 3146-3154.

[23] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).

[24] Arık, S. O., & Pfister, T. (2020). Tabnet: Attentive interpretable tabular learning. arXiv.

[25] Lemaître, G., Nogueira, F., & Aridas, C. K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. The Journal of Machine Learning Research, 18(1), 559-563.

[26] Hart, P. (1968). The condensed nearest neighbor rule (corresp.). IEEE transactions on information theory, 14(3), 515-516.

[27] Rubner, Y., Tomasi, C., & Guibas, L. J. (2000). The earth mover's distance as a metric for image retrieval. International journal of computer vision, 40(2), 99-121.

[28] Weaver, N. (2018). Lipschitz algebras. World Scientific.

[29] Chhikara, R. (1988). *The Inverse Gaussian Distribution: Theory: Methodology, and Applications* (Vol. 95). CRC Press.

[30] Dauphin, Y. N., Fan, A., Auli, M., & Grangier, D. (2017, July). Language modeling with gated convolutional networks. In International conference on machine learning (pp. 933-941). PMLR.

[31] Tran, K. (2020). From english to foreign languages: Transferring pre-trained language models. arXiv preprint arXiv:2002.07306.

[32] Dua, D., & Graff, C. (2019). UCI Machine Learning Repository, University of California, School of Information and Computer Science, Irvine, CA, 2019.

[33] Vijayakumar, S., & Schaal, S. (2000, June). Locally weighted projection regression: An o (n) algorithm for incremental real time learning in high dimensional space. In Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000) (Vol. 1, pp. 288-293). Morgan Kaufmann.

[34] PyTorch implementation of TabNet, https://github.com/dreamquark-ai/tabnet

[35] https://github.com/Roweida-Mohammed/Code For Santander Customer Transaction Prediction

[36] https://www.kaggle.com/volodymyrgavrysh/fraud-detection-bank-dataset-20k-records-binary For Fraud detection bank Transaction Prediction

[37] Bennett, PMH., T.A. Burch, and M. Miller. (1971). Diabetes mellitus in American (Pima) Indians. Lancet 2:125-128.