AI 시대의 범주형 데이터 처리: Entity Embedding과 Cat2Vec의 효과성 분석

김동현

(지도교수 : 김선용)

Handling Categorical Data in the AI Era: Analyzing the Effectiveness of Entity Embedding and Cat2Vec

Dong-Hyeon Kim

요약

본 연구는 AI 시대에서 범주형 데이터를 효과적으로 처리하는 방법에 대한 통찰을 제공하고자 한다. 특히, Entity Embedding과 Cat2Vec, 그리고 기존의 One-hot encoding 방법의 성능 차이를 중점적으로 분석한다. Rossmann Store Sales 데이터셋을 활용하여 각 방법으로 데이터를 인코딩 및 임베딩한 후, K평균 군집화 알고리즘을 통해 성능을 평가한다. 이를 통해 각 임베딩 기법이 정형 데이터의 표현과 이해에 미치는 영향을 분석하고, Tabular Learning 분야의 발전에 기여할 수 있는 기초 연구로서의 의의를 가진다. 또한, 실제 비즈니스 데이터를 활용함으로써 실무적 적용 가능성에 대한 시사점도 제공할 것으로 기대된다.

Keyword: 정형 데이터, *Tabular Learning*(정형 데이터 이해), 임베딩, 범주형 변수

I. 서 론

정형 데이터는 현대 사회의 다양한 분야에서 중요한 역할을 수행하고 있다. 특히 인공지능, 금융, 의료, 전자상거래 등 데이터의 가치가 높은 분야에서 그 중요성이 두드러진다. 최근 기계학습 분야에서는 HuggingFace, Kaggle, AIHub 등을 비롯한 여러 플랫폼에서 데이터를 공유하고 활용하는 사례가 증가하고 있으며, 이러한 현상은 데이터의 중요성을 잘 반영하는 사회적 현상이다.

정형 데이터의 수요가 증가함에 따라 인공지능이 정형 데이터를 얼마나 잘 이해하는지 평가하는 Tabular Learning 분야에 대한 관심도 높아지고 있다. 이에 따라 트리 기반, 인코더-디코더 기반, 신경망 기반, 통계 기반 등 다양한 방법론이 제안되고 있다. 하지만 정형 데이터에 혼재된 범주형 변수는 순서 관계, 대소 관계 등이 존재하지 않는 경우가 많다는 점이 Tabular Learning 문제의 난이도를 높이고 있다. 또한, 기계학습에 사용되는 현실 데이터는 불균형, 저차원, 이 또한 Tabular Learning의 난이도를 한층 더 높이는 데 기여한다.

임베딩은 데이터 속의 의미 관계를 보존하면서 차원을 축소할 수 있는 방법을 말한다. 이는 고차원 데이터를 저차원 공간으로 사영함으로써 데이터 연산량을 줄이는 데 도움을 준다. 더불어, 모델의 일반화 성능을 향상시키는 데도 기여한다.

2016년에는 정형 데이터를 신경망 모델을 사용하여 연속 공간에 임베딩하는 Entity Embedding과 Cat2Vec에 대한 연

구 논문이 발표되었다. 그러나 이들이 실제로 기계학습에 얼마나 큰 영향을 미치는지 비교하는 논문은 현재까지 존재하지 않는 상황이다.

본 연구는 각 임베딩 모델이 인공지능 모델의 정형 데이터 이해도에 어떤 영향을 주는지 파악하기 위해 원핫 인코딩, Entity Embedding, Cat2Vec 모델로 인코딩 및 임베딩된 데이터를 K평균 군집화 모델을 통해 군집화하고 군집화 성능을 평가한다. 이를 통해 각 임베딩 기법이 인공지능 모델에 미치는 영향을 분석함으로써 각 모델의 임베딩 성능에 대한 더욱 깊은 이해를 얻고자 한다.

II. 관련 연구

2-1. One-hot Encoding

One-hot encoding은 Label encoding과 함께 범주형 데이터를 처리하는 가장 기초적인 방법 중 하나이다. 이 방법은 각 범주를 고유한 이진 벡터로 표현하며, 해당 범주에 해당하는 위치만 1로, 나머지는 0으로 표시한다. 예를 들어 “사과”, “바나나”, “콜라”를 one-hot 인코딩하면 각각 (1, 0, 0), (0, 1, 0), (0, 0, 1)로 표현된다.

One-hot encoding은 구현이 간단하고 직관적이지만 범주형 변수 간의 관계를 표현하기 어렵고, 차원의 저주 문제를 야기하기 쉽다는 한계가 있다.

2-2. Entity Embedding

Entity Embedding은 범주형 변수를 저차원 연속 벡터 공간에 사영하는 기법 중 하나이다. 신경망을 사용하여 각 범주에 대한 임베딩을 학습하여, 범주의 의미를 표현하고자 시도하였다. Entity Embedding은 One-hot encoding의 한계를 극복하고, 범주형 데이터를 효율적으로 저차원 공간에 표현할 수 있도록 한다.

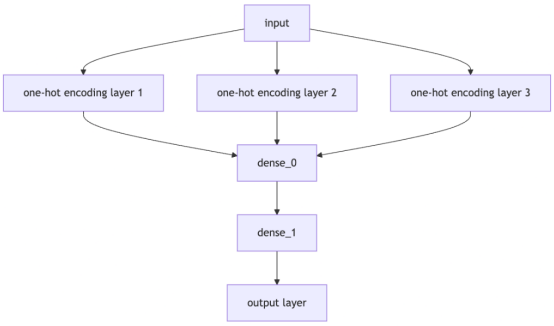

그림 1 Entity Embedding 모델 구성도

위 이미지는 Entity Embedding의 모델 구조이다. Entity Embedding은 Tabular Learning 문제를 “함수 추론”의 관점에서 접근하여 상관관계에 있는 변수들을 원-핫 벡터로 변환한 후, 신경망을 통해 학습하는 방식이다.

Rossmann Store Sales 데이터셋을 사용한 연구[2]에서는 도시 간의 위치가 독일의 실제 도시 위치와 유사하게 임베딩되는 사례가 있었다.

2-3. Cat2Vec

Cat2Vec[1]은 ICLR 2017에서 발표된 범주형 변수의 임베딩 기법 중 하나이다. 이 방법은 범주형 특징 간의 상관성을 포착하여, 이를 연속 공간에 사영함으로써 임베딩 작업을 수행하는 방식이다.

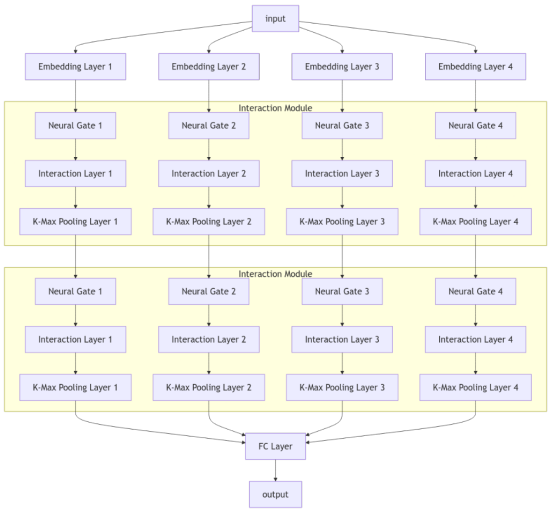

그림 2 Cat2Vec 모델 구성도

위 이미지는 Cat2Vec 모델의 구성도이다. 입력 직후에 임베딩 계층이 존재하며, 이후에는 2개의 Interaction Module을 거치며 상관관계를 학습하고, 최종적으로 완전 연결 계층을 거쳐 출력을 하게 되는 구조이다.

임베딩 계층에서는 원핫 인코딩 또는 PyTorch 등에서 제공하는 임베딩 모듈을 통해 데이터의 벡터화를 수행한다. 이후 뉴럴 게이트에서는 상관관계를 포착하기 위해 데이터를 SUM, MUL, HIGHWAY 등의 게이트에 입력하여 값을 얻어낸다. 최종적으로 이 값을 K-Max 풀링 계층에 입력하여 가장 높은 값을 골라내며 관계성이 높은 값만 걸러내고, 임베딩을 학습하는 구조이다.

Cat2Vec은 국내외의 다양한 연구[6]에서 정형 데이터의 차원 축소에 활용된 바 있다. CTGAN과 함께 사용되어 고차원 데이터에서 취약한 생성 성능을 보였던 CTGAN의 성능을 개선하는 데 효과를 보였다.

III. 실험 방법

본 연구는 임베딩 방법의 효과를 평가하기 위해 Rossmann Store Sales 데이터셋을 One-hot encoding, Entity Embedding, Cat2Vec 방법으로 인코딩 및 임베딩한다. 이후 임베딩된 데이터를 K평균 군집화 알고리즘에 입력하여 군집화 성능을 평가한다. 연구에 사용된 Rossmann Store Sales 데이터셋은 fast.ai 플랫폼에서 획득하였다.

Rossmann Store Sales 데이터셋은 train.csv, store.csv, state.csv 파일로 구성된다. 이 데이터셋을 실험에 적합하게 가공하기 위해 다음과 같은 전처리 과정을 거쳤다:

1. "Open"과 "Sales" 특성을 기준으로 운영 중이 아니거나 판매량이 없는 데이터를 제거
2. state.csv의 가게 지점 정보를 "Store" 식별자를 이용해 train.csv와 병합
3. 분석에 불필요한 열 제거
4. 계산 효율성을 위해 200,000개의 샘플만 무작위 추출

이후 각각 One-hot encoding, Entity Embedding, Cat2Vec 방법을 적용하여 임베딩을 생성하고 별도로 저장하여 분석에 사용하였다.

Entity Embedding과 Cat2Vec은 원논문 [1], [2]를 참고하여 torch 2.5.1을 통해 직접 구현하였다. 또한, KMaxPooling은 PyTorch 내에 구현된 것이 존재하지 않아 직접 구현하였다. KMeans(K 평균 군집화 모델) 구현체는 scikit-learn 1.5.2(stable) 버전의 구현체를 사용하였다.

임베딩 차원은 Guo와 Berkhahn[1]의 연구를 참고하여 다음과 같이 결정하였다.

표 1 Rossmann Store Sales의 차원 정보

특징	원래 차원	임베딩 차원
DayOfWeek	7	6
Promo	2	1
Year	3	2
Month	12	6
Day	31	10
Open	2	1
Store	1115	10
State	12	6

KMeans의 하이퍼파라미터는 n_clusters만 사용하였으며, 12로 설정하였다. 이는 임베딩 과정에서 타겟 값으로 준 데이터가 "State"이기 때문이다.

실험 결과의 신뢰성을 높이고 평가 오차를 최소화하기 위해, 각 임베딩 방법에 대해 5번의 임베딩을 수행한 후, 각 임베딩 결과를 KMeans에 입력하여 군집화를 수행하였다. 또한, "State" 열을 타겟 데이터로 사용하고 나머지를 입력으로 활용하여 학습을 진행하였다.

군집화 성능 평가에는 Adjusted Rand Index (ARI)를 활용하였다. ARI는 군집화 결과와 실제 레이블 간의 유사성을 측정하는 지표로, 우연히 일치하는 경우를 보정하여 더 정확한 평가가 가능하다. 개별 임베딩 집합을 군집화 한 후 계산한 ARI 점수의 평균을 최종 평가지표로 사용하였으며, 이는 단일 실험 결과에 의한 편향을 줄이고 더 안정적인 성능 평가를 가능하게 한다.

IV. 실험 결과

본 연구는 Rossmann Store Sales 데이터에서 랜덤하게 200,000개만큼 샘플링하여 원핫 인코딩, Entity Embedding, Cat2Vec을 통해 인코딩 및 임베딩을 수행하고, 그 다음 인코딩/임베딩 결과를 K평균 군집화 모델에 입력하여 그 성능을 Adjusted Rand Index(ARI)로 평가하였다.

ARI는 군집화 알고리즘의 성능 평가에 널리 사용되는 지표이며, -1부터 1 사이의 값을 가진다. 1에 가까울수록 완벽하게 군집을 분류해냈음을 의미한다. 0은 무작위 군집화를 수행하는 군집화기의 점수로 가정된다.

아래는 각 인코딩 및 임베딩 기법에 따른 ARI 점수이다. 원핫-인코딩은 1회, Entity Embedding과 Cat2Vec은 각각 5번회 독립적으로 임베딩을 수행하여 군집화 모델에 입력하였다.

표 2 ARI 점수 (군집화 평가 점수)

항목	ARI 점수	차원 수
onehot	0.0	1,171
EE #1-#5	1.0	12
Cat2Vec #1-#5	1.0	12

임베딩 모델을 사용한 경우는 1.0 (100% 올바르게 분류), 원핫 인코딩을 수행한 경우는 0.0 (랜덤 군집화기와 같은 수준의 분류)이라는 점수가 나왔다.

One-hot 인코딩 사례는 차원의 저주에 빠진 것처럼 보인다. 반면, Entity Embedding과 Cat2Vec 사례는 모든 경우에서 완벽한 군집화를 수행했다고 계산되었다. ARI는 타겟 데이터를 모두 가지고 평가하기에 100% 완벽하게 분류했다는 의미이기도 하다.

이는 임베딩 기법들이 단순히 차원을

축소하는 것이 아니라 데이터에 내재된 유의미한 특징을 효과적으로 포착하고 있음을 시사한다. 특히, Entity Embedding과 Cat2Vec은 범주형 변수 간의 복잡한 관계와 잠재적 패턴을 저차원 공간에 투영함으로써 군집화 알고리즘이 더 유의미한 그룹을 식별하게 한 것으로 보인다.

IV. 결 론

본 연구는 Rossmann Store Sales 데이터셋을 사용하여 One-hot encoding, Entity Embedding, Cat2Vec 방법의 효과를 비교 분석하였다. 실험 결과, Entity Embedding과 Cat2Vec이 One-hot encoding에 비해 월등히 우수한 성능을 보였으며, 두 방법 모두 완벽한 군집화 결과(ARI 점수 1.0)를 달성하였다.

이러한 결과는 Entity Embedding과 Cat2Vec이 범주형 데이터의 복잡한 관계를 효과적으로 포착하고, 저차원 공간에 의미 있게 투영할 수 있음을 보여준다. 반면, One-hot encoding은 고차원 공간에서의 희소성 문제, 차원의 저주 문제로 인해 효과적인 군집화를 수행하지 못했다.

본 연구 결과는 다양한 머신러닝 작업 및 정형 데이터를 활용하는 작업에서 Entity Embedding과 Cat2Vec이 유용함을 입증한다. 이러한 임베딩 기법을 활용하면 데이터의 차원을 효과적으로 축소하면서도 중요한 정보를 보존하여 후속 분석 작업에서의 성능을 크게 향상시킬 수 있다.

그러나 본 연구에는 몇 가지 한계점이 있다. 첫째, 단일 데이터셋과 단일 군집화 모델에 대한 평가지표이므로 다양한 도메인과 특성을 가진 데이터셋에 대한 검증이 필요하다. 둘째, 군집화 성능만을 평가 지표로 사용하기 때문에 회귀, 분류 등 다른 머신러닝 작업에서의 성능은 추가적으로 연구되어야 한다.

향후 연구에서는 이러한 한계점을 보완하고, 다양한 데이터 특성(불균형, 희소, 편향 등)에 따른 임베딩 기법의 성능 변화를 탐구할 필요가 있다. 또한, 임베딩 기법의 해석 가능성과 실제 비즈니스 문제에의 적용 가능성에 대한 연구도 진행할 수 있을 것이라 기대된다.

References:

[1] Wen, Y., Wang, J., Chen, T., & Zhang, W. (2016). Cat2Vec: Learning distributed representation of multi-field categorical data.

[2] Guo, C., & Berkhahn, F. (2016). Entity embeddings of categorical variables. arXiv Preprint arXiv:1604.06737.

[3] Zhao, Z., Kunar, A., Birke, R., & Chen, L. Y. (2021). CTAB-GAN: Effective Table Data Synthesizing (Vol. 157; V. N. Balasubramanian & I. Tsang, Eds.). Retrieved from https://proceedings.mlr.press/v157/zhao21a.html

[4] Li, S.-C., Tai, B.-C., & Huang, Y. (2019). Evaluating Variational Autoencoder as a Private Data Release Mechanism for Tabular Data. Presented at the 2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC). doi:10.1109/PRDC47002.2019.00050

[5] Kiran, A., & Kumar, S. S. (2023). A Comparative Analysis of GAN and VAE based Synthetic Data Generators for High Dimensional, Imbalanced Tabular data. Presented at the 2023 2nd International Conference for Innovation in Technology (INOCON). doi:10.1109/INOCON57975.2023.10101315

[6] 김석준. (2022). 범주형과 연속형 변수가 혼합된 불균형 데이터 분류를 위한 CAT2VEC과 Conditional Tabular GAN의 활용. Retrieved from http://www.riss.kr/link?id=T16068647

[7] Dai, B., & Wipf, D. (2019). Diagnosing and enhancing VAE models. arXiv Preprint arXiv:1903.05789.

[8] Dahouda, M. K., & Joe, I. (2021). A Deep-Learned Embedding Technique for Categorical Features Encoding. IEEE Access, 9, 114381-114391. doi:10.1109/ACCESS.2021.3104357

[9] Meulemeester, H. D., & Moor, B. D. (2020). Unsupervised Embeddings for Categorical Variables. Presented at the 2020 International Joint Conference on Neural Networks (IJCNN). doi:10.1109/IJCNN48605.2020.9207703

[10] Kotelnikov, A., Baranchuk, D., Rubachev, I., & Babenko, A. (2023). Tabddpm: Modelling tabular data with diffusion models. Presented at the International Conference on Machine Learning.