Distance-Weighted k-Means Clustering for Class-Imbalance Problem

Hyesoo Shin and Ki Yong Lee

October 23, 2023

클래스 불균형 문제에 대한 거리 기반 가중치 *k-* 평균 클러스터링 기법*

신혜수○, 이기용
숙명여자대학교 컴퓨터과학과
{seawater, kiyonglee}@sookmyung.ac.kr

Distance- weighted k- means clustering for class- imbalance problem

Hyesoo Shin○, Ki Yong Lee
Dept. of Computer Science, Sookmyung Women's University

요 약

본 논문에서는 클래스(class) 불균형 문제를 해결하기 위해 거리 기반 가중치를 활용한 k-평균 클러스터링(distance- weighted k- means clustering) 기법을 제안한다. k-평균 클러스터링은 데이터 포인트를 클러스터(cluster)로 그룹화하는 대중적인 기술 중 하나로, 동일한 클러스터에 속하는 모든 데이터 포인트들의 평균(mean)을 통해 각 클러스터의 중심점(centroid)을 업데이트하는 특징이 있어 클래스 간의 데이터 불균형이 있는 경우 성능 저하 문제가 발생할 수 있다. 이러한 문제에 대응하여 제안하는 모델은 클러스터의 중심점을 업데이트할 때마다 모든 데이터 포인트와 클러스터 중심점과의 거리를 계산하고, 이를 기반으로 가중평균으로 계산하여 새로운 중심점을 얻는다. 과정의 반복을 통해 불균형한 클래스 간의 클러스터링 결과를 개선하는 것을 목표로 하였으며, 실데이터를 사용한 실험 결과에서 제안 모델은 평균 또는 중앙값을 사용하여 클러스터 중심점을 계산하는 기존 연구들보다 클러스터링 품질이 우수함을 확인하였다. 특히 클러스터의 내부 간 응집도와 클러스터 간 분리도를 측정하는 실루엣 계수(silhouette coefficient) 지표에서 평균 또는 중앙값을 이용한 비교 모델들이 음의 값으로 측정된 반면에 제안 모델은 0.1919로 측정되었음을 확인하였다.

1. 서 론

클러스터링(clustering)은 비슷한 데이터 포인터들을 하나의 클러스터(cluster)로 분류하는 데이터 마이닝 기술로서, k-평균(k-means) 클러스터링[1] 기법은 클러스터링 분야에서 대표적인 클러스터링 기술로 활용되고 있다. k-평균 클러스터링은 각 클러스터에 속한 데이터 포인터들의 평균(mean)값 계산을 통해 중심점(centroid)을 업데이트하는 과정을 반복한다. 하지만 실제 데이터셋에서는 클래스 간의 불균형이 발생하는 경우가 매우 흔하며, 특정 클래스는 다른 클래스보다 데이터 포인트 수가 훨씬 많을 수 있다. 이러한 클래스 불균형 문제는 k-평균 클러스터링 알고리즘의 성능을 저하시키는 주요 요소 중 하나로 알려져 있다.

클래스 불균형 문제를 해결하기 위해 다양한 선행 연구[2-4]가 이미 진행된 바 있다. 하지만 이들 연구는 대부분 가중치를 반영할 수 있는 근접 이웃 점 개수를 모델 학습 전에 지정해야 하거나, 지정하더라도 모든 클러스터에 대해 동일한 범위만을 반영하기 때문에 클래스 불균형이 있는 실제 데이터셋에서는 고질적인 문제점을 크게 극복하지 못했으며, 이를 해결하기 위해 복잡한 연산 과정이 추가되어 계산 비용이 크게 증가하게 되는 경우가 많았다.

따라서 본 논문에서는 거리 기반 가중치를 활용한 k-평균 클러스터링 기법을 제안하고자 한다. 제안하는

* 이 성과는 2021년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. NRF- 2021R1A2C1012543).

모델은 클러스터링의 중심점을 업데이트할 때 평균값을 계산하는 것이 아닌 클러스터에 속한 모든 데이터 포인트들과 중심점 간의 거리를 계산하여 가까운 데이터 포인트는 더 큰 가중치를, 먼 데이터 포인트는 작은 가중치를 배정받도록 하여 가중평균으로 새 중심점을 얻도록 한다. 그림 1(a)을 통해 보면 알 수 있듯이 평균값으로 중심점을 지정할 경우에는 클러스터 내에 소수의 데이터 포인트가 중심점을 크게 치우치는 영향력을 미칠 수 있으나 그림 1(b)에서처럼 가중 평균을 사용하면 기존의 중심점에 가까운 클러스터들을 통해 비교적 안정적인 클러스터 형태를 보이는 것을 확인할 수 있다. 이를 통해 중심점을 업데이트하면 데이터 포인터들을 재할당하였을 때 거리가 먼 데이터 포인트들은 다른 클래스로 재배치될 확률이 높아지기 때문에 모든 데이터 포인트의 가중치를 클러스터 중심 업데이트에 고려함으로써 불균형한 클래스 간의 클러스터링 결과를 향상시키도록 한다.

실제 데이터셋을 사용한 실험 결과에서 기존의 진행된 연구들과 평균값, 중앙값으로 계산한 k-평균 클러스터링 기법보다 우수한 클러스터링 퀄리티를 보임을 성능 평가를 통해 확인할 수 있었다.

본 논문의 구성은 다음과 같다. 2장에서는 기존에 진행된 연구를 살펴본다. 3장에서는 제안하는 기법을 자세히 설명한다. 4장에서는 실험을 통해 제안 방법의 성능 평가 결과를 제시하며, 5장에서는 결론을 맺는다.

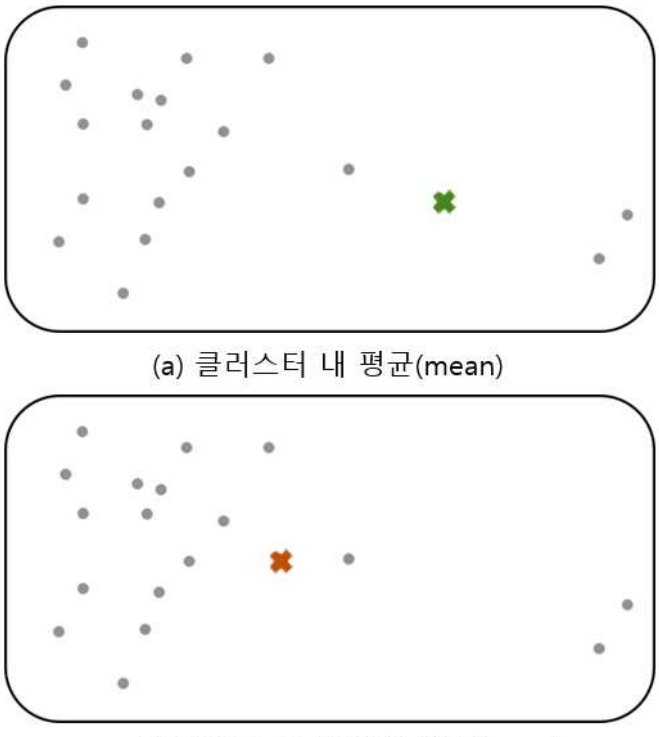

그림 1 (a)클러스터 내부의 중심점을 평균으로 계산하였을 때와 (b)가중 평균으로 계산하였을 때의 시각화

2. 관련 연구

1장에서 언급한 것처럼 데이터 포인트에 가중치를 부여하는 연구들이 일부 진행되고 있다. 본 장에서는 이러한 관련 연구를 살펴본다.

[2]은 *k*-NN(*k*-nearest neighbor) 알고리즘[5] 기반의 예측 모델인 W-K-NN(weighted k-nearest neighbor)을 제안하였다. W-K-NN은 모델이 고려할 근접 이웃점 개수 k를 미리 정의한 후, 데이터 포인트 간의 거리를 계산한 후 거리를 순서로 정렬한다. 그런 다음 현재 데이터 포인트로부터 거리가 가장 짧은 k개의 점을 선택하여 유클리드 거리의 역수를 가중치로 사용하여 예측을 수행하여 k개의 데이터 포인트가 위치한 범주의 발생 빈도를 결정한다.

[3]는 [2]과 유사하게 k-NN 알고리즘 기반의 모델인 FWKNN (feature weighted knearest neighbor)을 사용한다. 여기에 특성에 대한 가중 서포트 벡터 머신인 FWSVM (feature weighted SVM)를 결합하여 예측하는 프레임워크를 제안하였다. 하지만 [2]과 [3]은 모델을 학습하기 이전에 미리 근접 이웃점 k의 값을 미리 정의해야 하며, 모든 클러스터에 동일한 k 값을 적용하기 때문에 데이터 포인트 간의 불균형을 고려하지 못하는 한계가 있다.

[4]는 불균형이 있는 데이터 포인트를 고려하기 위해서 k-NN과 유사한 접근 방식을 사용하지만, 여기에 정보 이득과 대칭 불확실성과 같은 통계적 기법을 추가적으로 사용하여 특성을 선택하고 최적의 특성 하위 집합을 찾는 방법으로 AWKNN(adaptive weighted k-nearest neighbors) 모델을 제안하였다. 하지만 통계적 기법은 모든 가능한 특성 조합을 평가하기 위해 계산 비용이 많이 들 수 있으며, 주어진 특성의 개별적인 영향을 고려하므로 특성 간의 복잡한 상호작용이 중요한 경우 이를 고려하지 못하는 한계가 있다.

이와 같이 관련 연구의 대부분은 k개의 가중치가 적용된 가장 가까운 이웃을 계산하여 값을 예측하는 방식을 사용하고 있다. 그러나 본 연구에서는 클러스터의 크기에 관계없이 모든 데이터 포인트의 가중치가 클러스터 중심 업데이트에 반영되도록 하며, 별도의 특성 평가 과정 없이 거리 기반의 가중치 클러스터링 알고리즘을 제안하고자 한다.

3. 제안 방법

본 장에서는 기법을 제안한다. 제안 방법은 크게 (1) 최초의 초기 중심점 설정과 할당 단계, (2) 클러스터 내부의 가중치 계산 단계, (3) 새로운 중심점 설정과 재할당 단계, (4) 반복 단계로 구성된다. 아래에서는 각 단계를 상세히 설명한다.

3.1 최초의 초기 중심점 설정과 할당

i개의 데이터 포인트 집합 X가 주어진다고 가정한다. k-평균 클러스터링의 학습을 위해서는 모델의 클러스터 개수 K를 정의하고, 데이터 포인트 X에 대한 k-평균 클러스터링의 초기 중심점 m_k를 얻는다.

일반적으로 첫 번째 중심점 m_1을 무작위로 선택하고, 나머지 중심점 $m_2, m_3, \dots, m_k$ 들은 이전 중심점과 멀리 떨어진 데이터 포인트를 확률적으로 선택하여 설정한다.

각 데이터 포인트 $x_i \in X$는 가장 가까운 중심점 m_c에 할당되며, 이를 수식으로 나타내면 다음과 같다.

$$c_i = argmin(\| x_i - m_c \|^2) \quad (1)$$

여기서 c_i는 데이터 포인트 x_i의 클러스터 할당을 나타내며, $argmin$은 가장 작은 값을 가진 클러스터를 선택하는 연산을 의미한다.

3.2 클러스터 내부의 가중치 계산

3.1절에서 초기 중심점 설정과 할당을 거친 결과, 데이터 포인터 x_i들이 초기 클러스터 c_i를 얻게 되었다. 이제 제안 방법은 각 데이터 포인터 x_i가 현재 클러스터 중심점 c_i로부터 부여받는 가중치 점수를 얻기 위하여 각 데이터 포인터 x_i에 대해 다음과 같이 계산한다.

$$d_i = \sqrt{(x_i - c_i)^2} \quad (2)$$

$$w_i = \frac{1}{1+d_i^2} \quad (3)$$

여기서 $x_i \in c_i$ 이며, d_i는 i번째 데이터 포인트와 데이터 포인트가 속한 클러스터 중심 간의 유클리디안(euclidean) 거리다. w_i은 각 데이터 포인트가 가지는 가중치 점수로, 거리가 가까울수록 더 높은 가중치를 부여하도록 역수를 취하고, 분모가 0이 되는 것을 방지하기 위해 1을 더한다. 즉, 거리가 작은 데이터 포인트에 높은 가중치를 부여하고 거리가 멀어질수록 가중치가 낮아진다.

3.3 새로운 중심점 설정과 재할당

3.2절을 통해 계산된 모든 데이터 포인트에 대한 가중치 w_i를 활용하여 가중평균 계산을 통해 중심점을 계산한다. 식은 다음과 같다.

$$m_c^{t+1} = \sum_{x_i \in c_i} \frac{w_i \times x_i}{w_i} \quad (4)$$

여기서 $x_i \in c_i$를 의미하며, 각 클러스터에 속한 모든 데이터 포인트들의 가중치가 새로운 중심점을 얻을 때 사용되도록 한다. 새로운 중심점을 기준으로 식 (1)을 통해 가장 가까운 중심점에 다시 데이터 포인트가 할당되도록 한다.

3.4 반복

3.2 절과 3.3절을 반복하면서 반복에 대한 임계값 ϵ 에 대해 $\sum_{c \in k} ||m_c^{t+1} - m_c^t|| < \epsilon$ 으로 수렴할 때까지 중심점 m_c^{t+1}을 재설정하고, 클러스터링을 반복한다.

4. 성능 평가

본 장에서는 제안 방법이 기존의 방법과 비교하였을 때 클러스터링 결과를 얼마나 향상시키는지 평가한 결과를 제시한다.

4.1 데이터셋 및 실험 방법

본 실험에서는 3개의 실제 데이터셋을 사용하였다.

kdd-cup-1999[6] 데이터는 4,898,431개의 군사 네트워크 환경에서 다양한 종류의 침입이 포함된 데이터로, 데이터셋의 침입 종류는 normal, satan, portsweep, Neptune, ftp_write 등 11개의 클래스가 주어져 있다. 다수가 속한 정상 클래스는 37,873개의 데이터 포인트들이 포함된 반면에 일부 비정상 클래스는 각각 50개 미만의 데이터 포인트를 가진다.

NSL-KDD[7] 데이터는 네트워크 보안에서 사용되는 데이터로 125,973개의 데이터가 포함되며 상태에 대한 23개의 클래스를 가진다. Normal 클래스는 67,343개를 포함하고 있지만 perl, spy와 같은 비정상적인 접근에 대한 클래스는 각각 10개 미만의 데이터 포인트를 가진다.

Cora[8] 데이터는 그래프 연구에서 사용되는 논문 인용 데이터로 1,433개의 데이터 포인트를 가지며 한 논문이 다른 논문을 인용하는 경우 간선이 형성된다. 논문 분류에 대한 클래스로 Neural Networks, Genetic Algorithms 등 7개의 클래스가 존재하며 818개가 존재하는 클래스가 있는 반면에 120개가 존재하는 클래스가 있어 클래스 간에 불균형이 존재한다.

실험을 위해 일부 속성(feature)들은 MinMaxScaler를 통한 Normalization 과정을 거쳤으며, 데이터셋 각각에 대해 클러스터 개수 k는 11, 23, 7로 설정하였고, 반복에 대한 임계값 ϵ은 모두 0.4로 설정하였다.

실험에서 비교한 모델은 일반적인 k-means[1] 기법과 중앙값을 사용하는 k-median[9], 앞서 관련 연구에서 언급한 바 있던 W-K-NN[2] 모델과 비교하였다.

4.2 성능 평가 결과

클러스터링 결과를 평가하는 지표로는 조정 랜드 지수(adjusted rand index, ARI), 실루엣 계수(silhouette coefficient, SC), 정규화된 상호 정보량(normalized mutual information, NMI)를 사용하였다. ARI는 클러스터링 결과와 실제 클래스 레이블 간의 일치 정도를 측정하며, SC는 클러스터링의 응집도와 분리도를 활용해 클러스터링의 성능을 측정한다. 마지막으로 NMI는 클러스터링 결과가 실제 데이터 구조를 얼마나 잘 반영했는지 측정하는 지표다.

세 가지 평가 기준을 두고 실험에서는 다음 방법들의 성능을 제안 방법과 비교하였다. 이러한 지표를

사용하여 클러스터링 결과의 품질을 측정하고 비교하는 데 있어서 필요한 정답 레이블은 데이터셋에서 제공된 사전 정보를 활용하였다.

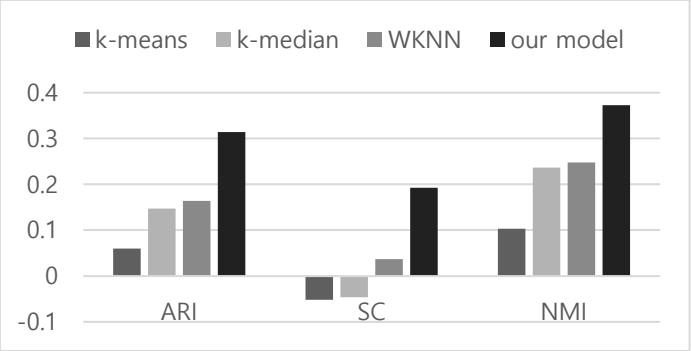

그림 2 Cora 데이터셋에 대한 성능평가 결과

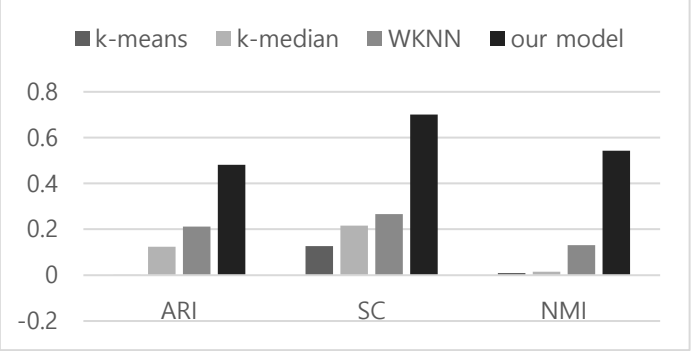

그림 3 ***kdd-cup-1999*** 데이터셋에 대한 성능평가 결과

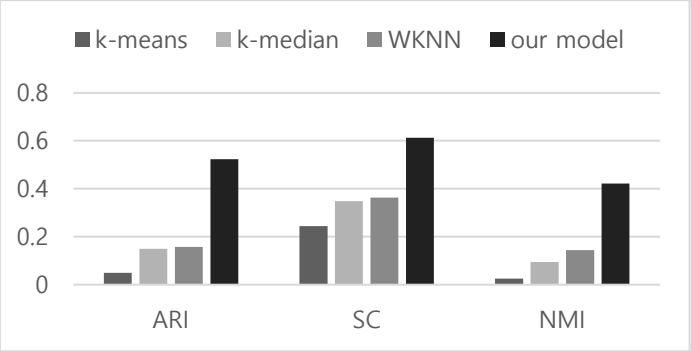

그림 4 ***NSL-KDD*** 데이터셋에 대한 성능평가 결과

그림 [2]에서 볼 수 있듯이 제안하는 논문의 방법은 모든 데이터셋에서 세 가지 성능 지표에서 다른 제안 방법보다 본 논문의 제안 방법이 가장 우수한 클러스터링 결과를 얻을 수 있음을 관찰할 수 있다. 특히 Cora 데이터셋에서 k-means와 k-meadian은 오히려 SC가 음수의 값으로 측정되며, WKNN은 0.0367로 클러스터링 품질이 매우 좋지 않다. 반면에 제안 방법은 0.1919의 값으로 보다 우수한 품질을 보임을 확인하였다. 이를 통해 각 클러스터 내에 속한 비정상적인 값들이 중심점에 덜 영향을 미치게 함으로써 중심점을 재배치할 때 클러스터의 형태가 더욱 더 안정화될 수 있도록 개선되었음을 확인할 수 있다.

5. 결론

본 논문에서는 기존의 k-평균 클러스터링 기법이 해결하지 못하는 클래스 불균형 문제를 해결하기 위해 거리 기반 가중치를 활용한 k-평균 클러스터링 기법을 제안하였다. 제안하는 모델은 클러스터의 중심을 업데이트할 때마다 해당 클러스터의 모든 데이터 포인트 거리를 계산하고, 거리 기반 가중치를 고려하여 중심점을 업데이트 하기 때문에 이를 통해 불균형한 클래스 간의 클러스터링 결과를 개선하는데 큰 영향을 미칠 수 있다. 실제 실험 결과를 통해 제안된 모델이 기존 연구보다 우수한 클러스터링 품질을 보여주며, 성능이 향상하는 것을 확인할 수 있었다.

참 고 문 헌

[1] MacQueen, J. B. Some Methods for classification and Analysis of Multivariate Observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1, 281-297, 1967.

[2] Fan, G.-F., Guo, Y.-H., Zheng, J.-M., & Hong, W.-C. Application of the Weighted K-Nearest Neighbor Algorithm for Short-Term Load Forecasting. Energies, 12(5), 916, 2019. https://doi.org/10.3390/en12050916

[3] Chen, Y., & Hao, Y. A. feature weighted support vector machine and K-nearest neighbor algorithm for stock market indices prediction. Expert Systems with Applications, 80, 340-355, 2017. https://doi.org/10.1016/j.eswa.2017.02.044

[4] Sun, L., Zhang, J., Ding, W., & Xu, J. Feature reduction for imbalanced data classification using similarity-based feature clustering with adaptive weighted K-nearest neighbors. Information Sciences, 593, 591-613, 2022. https://doi.org/10.1016/j.ins.2022.02.004

[5] Cover, T., & Hart, P. Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13(1), 21-27, 1967.

[6] kddcup99, http://kdd.ics.uci.edu/databases/kddcup99

[7] NSL-KDD, https://www.unb.ca/cic/datasets/nsl.html

[8] Cora, https://relational.fit.cvut.cz/dataset/CORA

[9] Har-Peled, S., & Mazumdar, S. On coresets for k-means and k-median clustering. In Proceedings of the thirty-sixth annual ACM symposium on Theory of computing, pp. 291-300, 2004.