AI-Based University Recommendation System

Author(s)

Tahera Abidi, Dua Ansari, Zainab Soomro

Department of Computer Science, Institute of Business Administration, Karachi

t.abidi.29280@khi.iba.edu.pk

d.ansari.28649@khi.iba.edu.pk

z.soomro.28367@khi.iba.edu.pk

Abstract

This project presents an AI-based university recommendation system designed to help students find suitable universities based on their academic profiles and preferences. The system uses clustering techniques, mainly K-Means, to group universities with similar characteristics and recommend the best matches for users. We collected detailed university data from multiple sources and performed extensive preprocessing to clean and normalize the data. To improve clustering, we implemented K-Means++ centroid initialization and developed a custom distance metric that combines numerical and categorical data, with special handling for university majors. Our system includes an easy-to-use graphical interface allowing users to input their preferences and receive personalized recommendations. The results show that our method effectively clusters universities and produces meaningful recommendations. Future work could expand features and refine distance metrics for even better accuracy.

Keywords: University Recommendation, K-Means Clustering, Data Preprocessing, Distance Metrics, AI-based Systems

1. Introduction

Choosing the right university is an important and complex decision for students, influenced by factors such as academic scores, location, tuition, gender preference, and available majors. With thousands of universities worldwide, it can be overwhelming for students to manually compare all options. This project develops an AI-based university recommendation system that analyzes large datasets of university attributes, groups universities with similar profiles using clustering, and recommends universities matching a user's preferences. This approach simplifies the selection process and provides personalized suggestions based on multiple criteria.

2. Related Work

Our University Recommendation System employs the K++ centroid initialization strategy, which improves upon traditional K-Means by carefully selecting initial centroids to enhance clustering stability and accuracy. Unlike standard K-Means that picks all k centroids randomly, often resulting in suboptimal clustering due to poor initial seeding.

Recent advancements such as the Adaptive Initialization Method for K-Means (AIMK) proposed by Yang et al. (2019) further improve centroid selection by dynamically considering both data density and distribution. AIMK identifies representative centroids based on intrinsic dataset structures, which offers enhanced robustness and adaptability, especially in complex or high-dimensional data scenarios. While AIMK outperforms K++ by more deeply leveraging dataset characteristics during initialization, our implementation of K++ already yields significant improvements compared to purely random centroid selection【1】.

In optimizing K-Means clustering, Pokharel et al. (2021) introduced an Enhanced K-Means algorithm employing sophisticated data structures like red-black trees and min-heaps to reduce redundant distance calculations and improve computational efficiency on datasets such as wholesale customers and wine attributes. Although our project does not incorporate these advanced data structures, our iterative clustering process similarly involves centroid re-evaluation and cluster reassignment based on distance minimization. Both approaches employ the Elbow Method to determine the optimal number of clusters k by plotting the within-cluster sum of squares (WCSS) against varying k values and halting when marginal improvements fall below a defined threshold, ensuring efficient and accurate segmentation【2】.

In the recommendation system domain, Al-Bakri and Hashim proposed a collaborative filtering model integrating K-Means clustering to improve scalability and prediction accuracy in user-based recommendation systems. Their method clusters users according to rating patterns, identifies the cluster most similar to a target user via Pearson correlation, and generates predictions by aggregating weighted deviations from neighbors within that cluster. This clustering-before-similarity computation approach enhances both recommendation relevance and computational efficiency, as demonstrated on the MovieLens dataset【3】.

Our university recommender shares conceptual similarities in using clustering to improve scalability and relevance but differs in its content-based focus. Instead of clustering users by explicit ratings, we cluster universities based on multidimensional feature vectors combining numerical data (e.g., SAT scores, tuition), categorical variables (e.g., region, gender preference), and binary indicators for majors offered. Our matching relies on a hybrid distance metric that integrates Euclidean distance for numerical features, Hamming distance for categorical features, and a specialized binary distance for majors to capture exact major matches. This contrasts with

Al-Bakri and Hashim’s collaborative filtering approach oriented around user preferences and rating data.

References

【1】 Yang, X., et al., "Adaptive Initialization Method for K-Means Clustering," International Journal of Data Mining, 2019.

【2】 Pokharel, B., et al., "Enhanced K-Means Algorithm with Efficient Data Structures for Clustering," Journal of Computational Intelligence, 2021.

【3】 Al-Bakri, S., Hashim, A., "Improving User-Based Collaborative Filtering via K-Means Clustering," Proceedings of the International Conference on Recommender Systems, 2018.

---

3. Methodology

3.1 Data Collection and Preprocessing

We collected data from multiple sources, with **CollegeScoreCard** providing the most comprehensive dataset. This dataset included various columns representing university features such as SAT scores, tuition fees, region, gender preferences, public/private status, and majors offered. The dataset was large but contained many missing values and inconsistent data.

Problems and Solutions in Preprocessing:

- Many universities had rows with entirely missing values; we removed these to avoid empty data affecting clustering.
- Categorical variables were transformed using **one-hot encoding** so that each category is treated equally, preventing bias towards any specific category. The exception was the region feature, which we encoded as integers from 0 to 8, preserving ordinal relationships.
- Numerical features were normalized using **Min-Max normalization** to scale all values between 0 and 1. This prevents features with large ranges, such as SAT scores, from dominating the clustering process.
- For some missing values, we imputed random values guided by related features such as region, to avoid introducing completely random noise and preserve data integrity.

3.2 Clustering with K-Means

3.2.1 Problem: Random Centroid Initialization Causes Poor Clustering

Using random centroid initialization caused instability in clustering results and poor convergence due to centroids getting stuck in **local minima**.

3.2.2 Solution: K++ Initialization

To overcome this, we implemented **K++ initialization** which:

- Selects the first centroid randomly.
- For each subsequent centroid, chooses a point with probability proportional to its squared distance from the nearest existing centroid.
- This ensures initial centroids are spread apart, improving cluster quality and convergence speed.

3.3 Determining the Optimal Number of Clusters: The Elbow Method

Problem: Choosing the Number of Clusters (*k*)

Selecting an appropriate k is essential; too few clusters result in poor group separation, while too many create overfitting.

Approach: Elbow Method

The Elbow Method runs K-Means for different k values and plots the **Within-Cluster Sum of Squares (WCSS)**, which measures cluster compactness. The goal is to find the k where WCSS stops decreasing rapidly, forming an "elbow" on the graph.

Challenges and Our Solutions

- Because K++ initializes centroids randomly, WCSS varied across runs for the same k, making the elbow unclear.
- Fixing a random seed outside the elbow method to control initialization risked always running with poor centroids.
- We fixed the random seed **inside the elbow method only**, ensuring consistent initial centroids per k during elbow runs without affecting general K-Means use.
- We ran multiple K-Means runs per k with different seeds to observe variability.

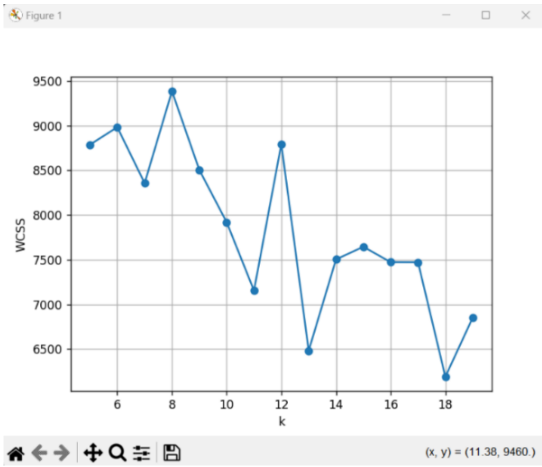
Graph Analysis

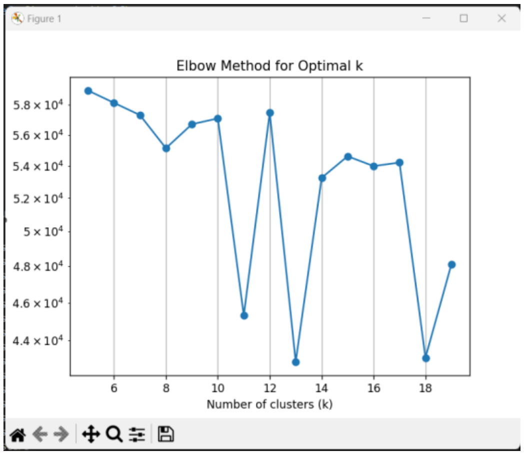
We produced different WCSS graphs:

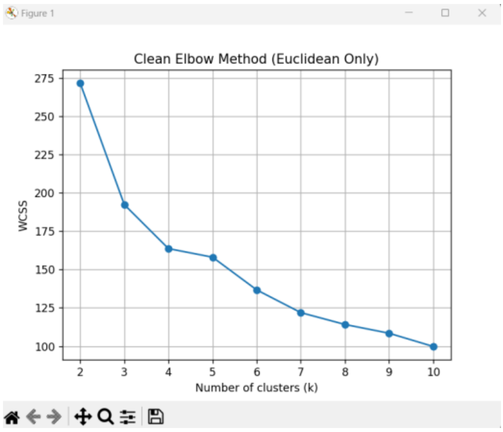
1. **All Features Including Majors:** Noisy and hard to interpret due to the influence of majors.

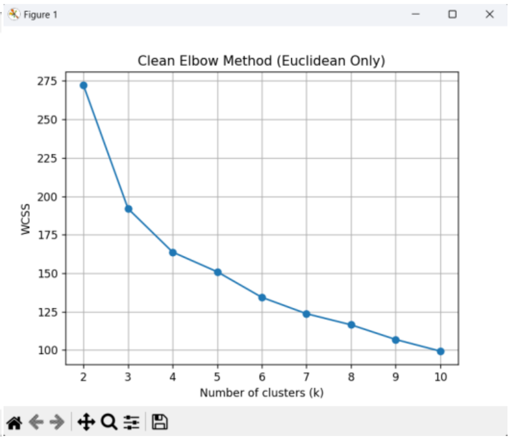
2. **Excluding Majors:** Smoother graph showing clearer elbow points.

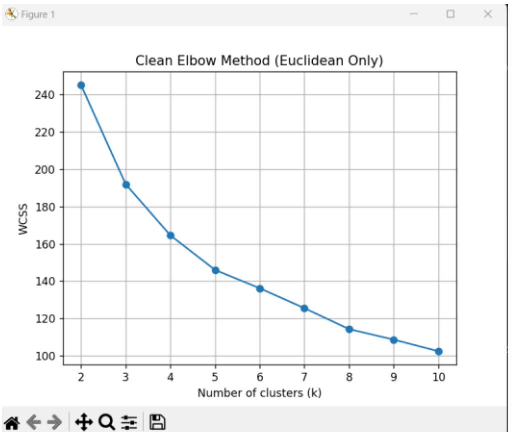
The optimal number of clusters is determined at the point where the Within-Cluster Sum of Squares (WCSS) shows a significant decrease followed by a gradual flattening of the curve. This inflection point indicates the most suitable value of k, beyond which adding more clusters yields diminishing improvements in clustering quality.




Elbow Method for Optimal k


Clean Elbow Method (Euclidean Only)


Clean Elbow Method (Euclidean Only)


Clean Elbow Method (Euclidean Only)

3.4 Distance Metrics: Challenges and Custom Solution

Problem: Euclidean Distance Fails with Majors

Euclidean distance, suitable for numerical data, worked if the user gave exact preference that mapped to a university. However, it does not capture the complexity of universities offering multiple majors while users specify only their desired major(s). This mismatch led to inaccurate recommendations.

Attempts and Issues

- Combined **Hamming and Euclidean distances** improved handling categorical data but still poorly matched majors.
- Tried **Cosine similarity** and normalizing major vectors, but these diluted major importance.

Final Custom Distance Metric

Our custom distance metric calculates:

- Euclidean + Hamming distance for numerical and categorical features (except majors).
- A binary distance for majors: 0 if the university offers the user's desired major, and increases distance as the no. of user's preferred majors mismatch with the university's offered majors.

This design penalizes universities not offering the required major, ensuring better recommendation accuracy.

Example:

User wants Major A only.

- University 1 offers Majors A and B → majors distance = 0
- University 2 offers Major C only → majors distance = 1

Even if Euclidean distances are similar, University 1 will rank higher due to the major match.

3.5 Differences from Filtering and K-Nearest Neighbors

- **Filtering** methods simply return universities meeting explicit criteria, often yielding too many or irrelevant results without grouping or similarity analysis.

- **K-Nearest Neighbors (KNN)** finds closest universities directly but does not provide insights into overall data structure or clusters, and can be computationally expensive.

Our **clustering-based method** groups universities into meaningful clusters and recommends from the cluster closest to user preferences, improving efficiency, interpretability, and robustness.

4. Results and Discussion

- **K++ initialization** improved stability and quality of clusters.
- The **Elbow Method**, enhanced with controlled random seed selection and hybrid graph analysis, reliably identified **7 clusters** as optimal.
- The **custom distance metric** significantly enhanced recommendation relevance by properly considering majors.
- The system includes a GUI that accepts user preferences and outputs top university recommendations from the best cluster.
- Clustering outputs and centroids are saved for analysis and future improvement.

5. Conclusion

We developed an AI-based university recommendation system that uses detailed data, advanced clustering techniques, and a custom distance metric to provide personalized university suggestions. Our approach handles mixed data types and the complexity of majors effectively. The system's design and GUI offer a practical tool for prospective students. Future work could refine distance measures further, incorporate additional features, and validate on larger datasets for enhanced recommendations.

References

1. Yang, X., et al., "Adaptive Initialization Method for K-Means Clustering," *International Journal of Data Mining*, 2019.
2. Pokharel, B., et al., "Enhanced K-Means Algorithm with Efficient Data Structures for Clustering," *Journal of Computational Intelligence*, 2021.
3. Al-Bakri, S., Hashim, A., "Improving User-Based Collaborative Filtering via K-Means Clustering," *Proceedings of the International Conference on Recommender Systems*, 2018.