CLUSTERING DI CARTE DI MAGIC

Bortolussi Lorenzo
SM3201370

STRUTTURA DEL LAVORO

Costruzione del dataset

Database di Scryfall

Download tramite API

Database locale

Preprocessing dati

Diversi dataset locali

Testo descrittivo delle carte

Embedding delle carte

Elaborazione dei cluster

Embedding delle carte

Riduzione della dimensione

Rappresentaizone in bassa dimensione

Clustering

Classi predette

Metriche di validazione e visualizzazioni 2D

Risultati

I dati in questione: carte di Magic

Le carte di MTG sono un dato **semi-strutturato**, con tipi di carta che vengono aggiunti ogni anno.

Ognuno dei **cinque colori** dovrebbe avere un suo **stile di gioco**, ma si possono trovare carte con effetti simili se non uguali sparse tra colori diversi.

Tramite Scryfall è possibile accedere ad una lista di tutte le carte mai stampate con associate numerose informazioni.

Preprocessing del testo

Il dataset originale, contenente tutte le carte mai stampate in tutte le ristampe, in lingua inglese. I **filtri** applicati a questo dataset includono:

- rimozione di campi non utili all'analisi (prezzo di mercato, link all'immagine...)
- rimozione di carte che non soddisfano certi criteri (carte solo digitali, carte senza testo oracle...)

Da questo dataset "pulito" viene poi estratto l'insieme di carte che ha un **unico colore** (o identità di colore): questo è il punto di partenza della fase di embedding.

❗ La stessa carta può comparire più volte, verrà considerata una volta per ogni anno di ristampa

Fase di embedding

Tra tutti i campi forniti, ho scelto di rappresentare le carte concatenando:

- **Type line**: utile a livello di gioco
- **Oracle text**: campo importante con cui la carta interagisce nel gioco
- **Keywords**: per ridondanza, importanti a livello di gioco

Questo testo é poi stato passato ad un sentence transformer (scaricato da hugging face) che ha computato gli embedding delle carte in vettori di dimensione 768.

Fase di clustering: Riduzione della dimensionalità

La fase di clustering è caratterizzata dalla seguente pipeline:

- Riduzione di dimensionalità tramite UMAP, riducendo i dati intorno a R^5
- Applicazione di un metodo di clustering sui dati in bassa dimensione
- Applicazione di misure di validazione per i label trovati (comparati con i colori delle carte)
- Riduzione del dataset originale in R^2 per poter plottare i risultati

Prima di applicare la riduzione della dimensionalità, é possibile scegliere se **normalizzare** i vettori di embedding delle carte per dare più peso alla direzione dei vettori.

Algoritmi utilizzati

Clustering:

HDBSCAN

Utilizzato perché non é limitato a cluster convessi, ma non è possibile imporre il numero di cluster a priori e la performance dipende molto dal parametro min_cluster_size.

Spectral clustering

Trova cluster non convessi e permette di specificare il numero di cluster, in questo caso noto. Non contempla punti di noise ma in questo caso non è un problema.

Riduzione della dimensionalità:

UMAP

Noto algoritmo per ridurre la dimensione dei dati, veloce a computare e capace di preservare la struttutra locale e globale dei dati stessi tramite il parametro n_neighbors.

Metriche di validazione utilizzate

- **Adjusted Rand Index**

  Misura le coppie di valori correttamente assegnate allo stesso cluster o a cluster diversi.

- **Normalized Mutual Information**

  Misura la dipendenza tra assegnazione al cluster predetto e a quello reale.

- **Homogeneity**

  Verifica se ogni singolo cluster contiene esclusivamente membri di una sola classe reale.

- **Completeness**

  Verifica se ogni punto che appartiene ad un cluster reale viene assegnato allo stesso cluster.

- **V-Measure**

  E’ la media armonica di Homogeneity e Completeness.

- **Contingency table**

  Permette di vedere esatamente dove l’algoritmo di clustering sta commettendo degli errori

Risultati

I risultati migliori non variano sensibilmente in meglio variando i parametri degli algoritmi utilizzati, parametri che sono stati riportati di seguito.
Il clustering è stato eseguito su carte stampate in un singolo anno alla volta. Come ordine di grandezza si contano intorno alle duemila carte per anno, quantità che cresce leggermente assieme al tempo.

UMAP

n_neighbors = 12
n_components = 5
metric = ‘cosine’

HDBSCAN

min_cluster_size = 40
min_samples = 10

UMAP

n_neighbors = 80
n_components = 5
metric = ‘cosine’

Spectral clustering

n_clusters = 5
affinity = 'rbf’

Risultati: HDBSCAN

Con normalizzazione:

Number of clusters found (excluding noise): 14
Fraction of noise points: 355 / 1606
Adjusted Rand Index (ARI): 0.0519
Normalized Mutual Information (NMI): 0.1175
Homogeneity: 0.1515
Completeness: 0.0959
V-Measure: 0.1175

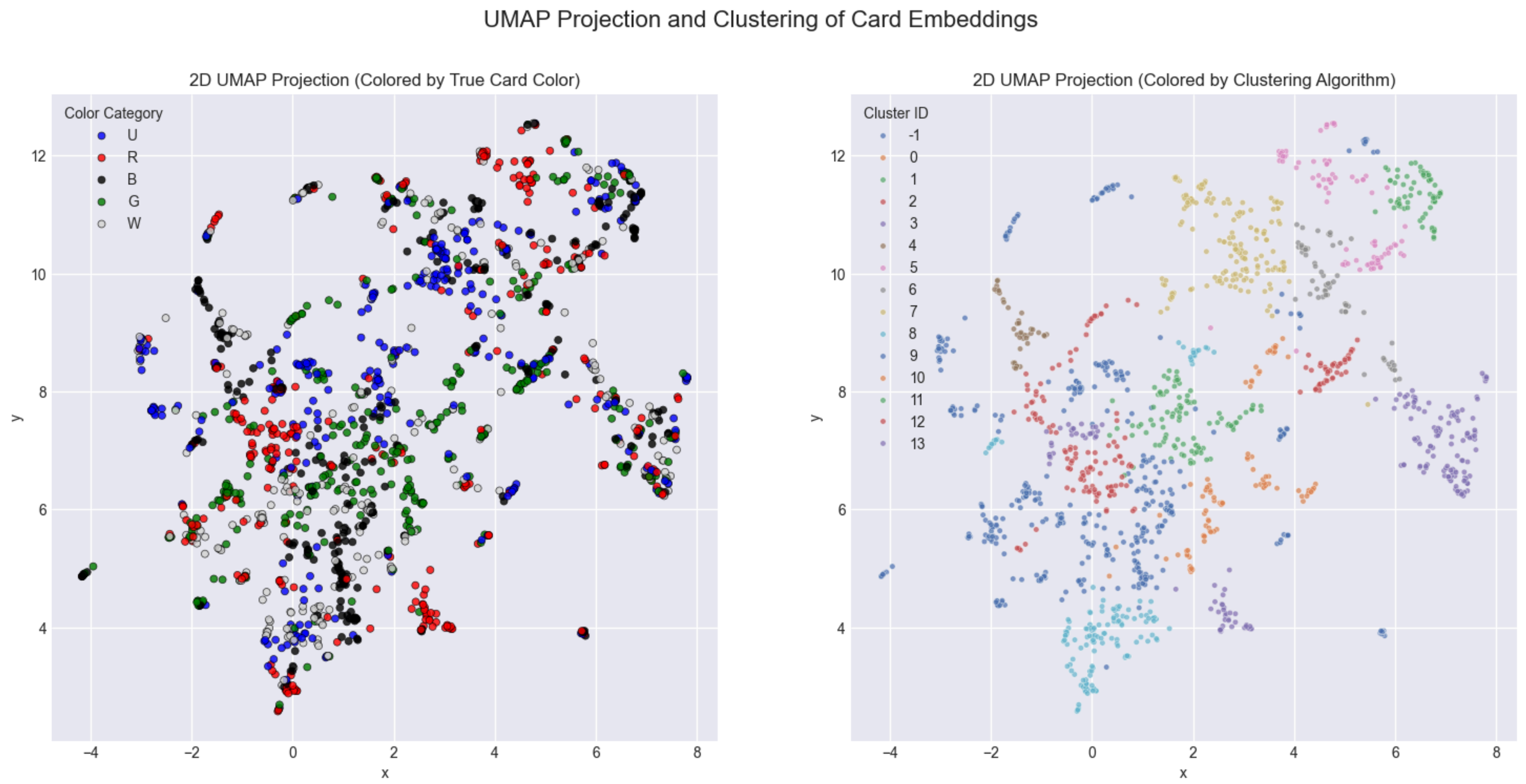
Contingency Matrix (rows=true, cols=predicted):
[[6 30 10 19 28 15 11 32 32 50 3 11 19 4]
[14 13 24 27 0 12 11 24 7 12 27 41 34 2]
[9 3 2 24 4 53 11 20 25 2 2 7 29 56]
[12 23 9 29 3 2 11 68 39 1 4 24 15 2]
[8 5 1 31 9 19 18 24 51 29 9 16 24 0]]

Senza normalizzazione:

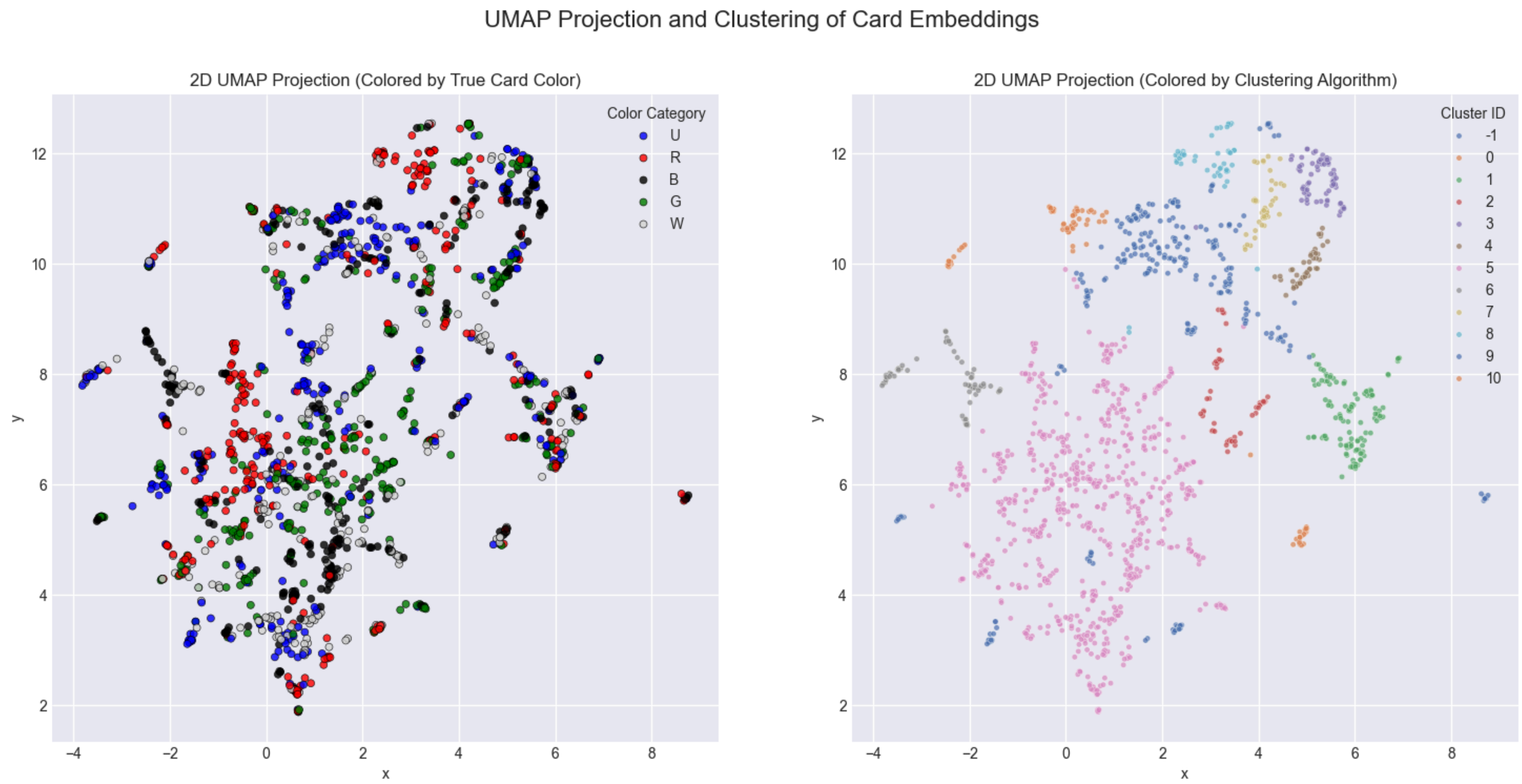
Number of clusters found (excluding noise): 11
Fraction of noise points: 142 / 1606
Adjusted Rand Index (ARI): 0.0149
Normalized Mutual Information (NMI): 0.0612
Homogeneity: 0.0638
Completeness: 0.0587
V-Measure: 0.0612

Contingency Matrix (rows=true, cols=predicted):
[[8 19 6 30 10 136 29 13 2 33 6]
[6 27 15 13 24 190 2 10 2 15 15]
[7 24 9 3 2 159 6 14 39 11 13]
[4 29 13 23 9 120 14 1 1 61 4]
[15 31 11 5 1 155 19 11 8 23 8]]

Con normalizzaizone



Senza normalizzaizone



Risultati: Spectral Clustering

Con normalizzazione:

Adjusted Rand Index (ARI): 0.0193
Normalized Mutual Information (NMI): 0.0253
Homogeneity: 0.0240
Completeness: 0.0268
V-Measure: 0.0253

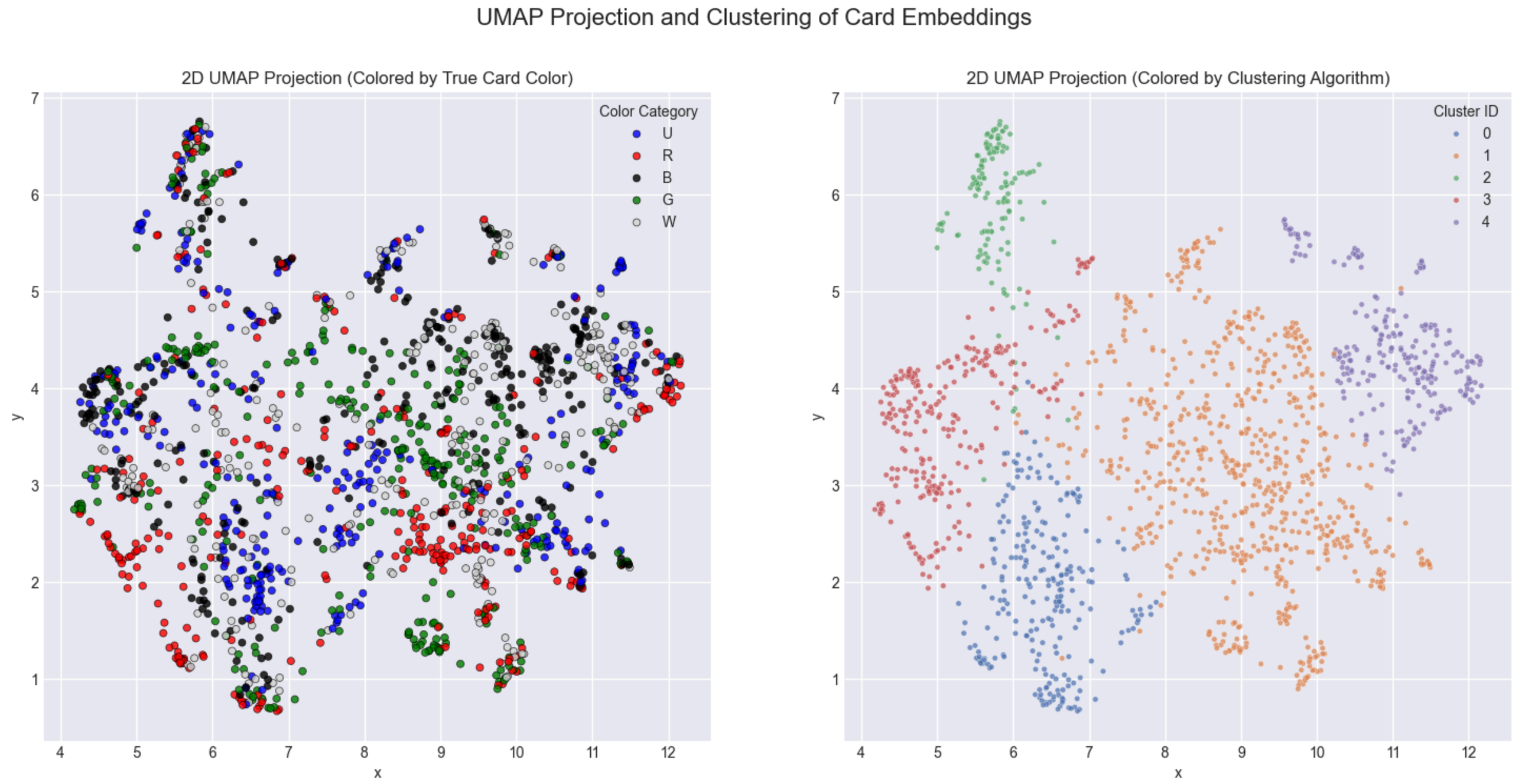
Contingency Matrix (rows=true, cols=predicted):
[[38 125 21 71 61]
[42 208 26 56 19]
[51 149 26 54 33]
[73 105 30 56 53]
[44 125 34 28 78]]

Senza normalizzazione:

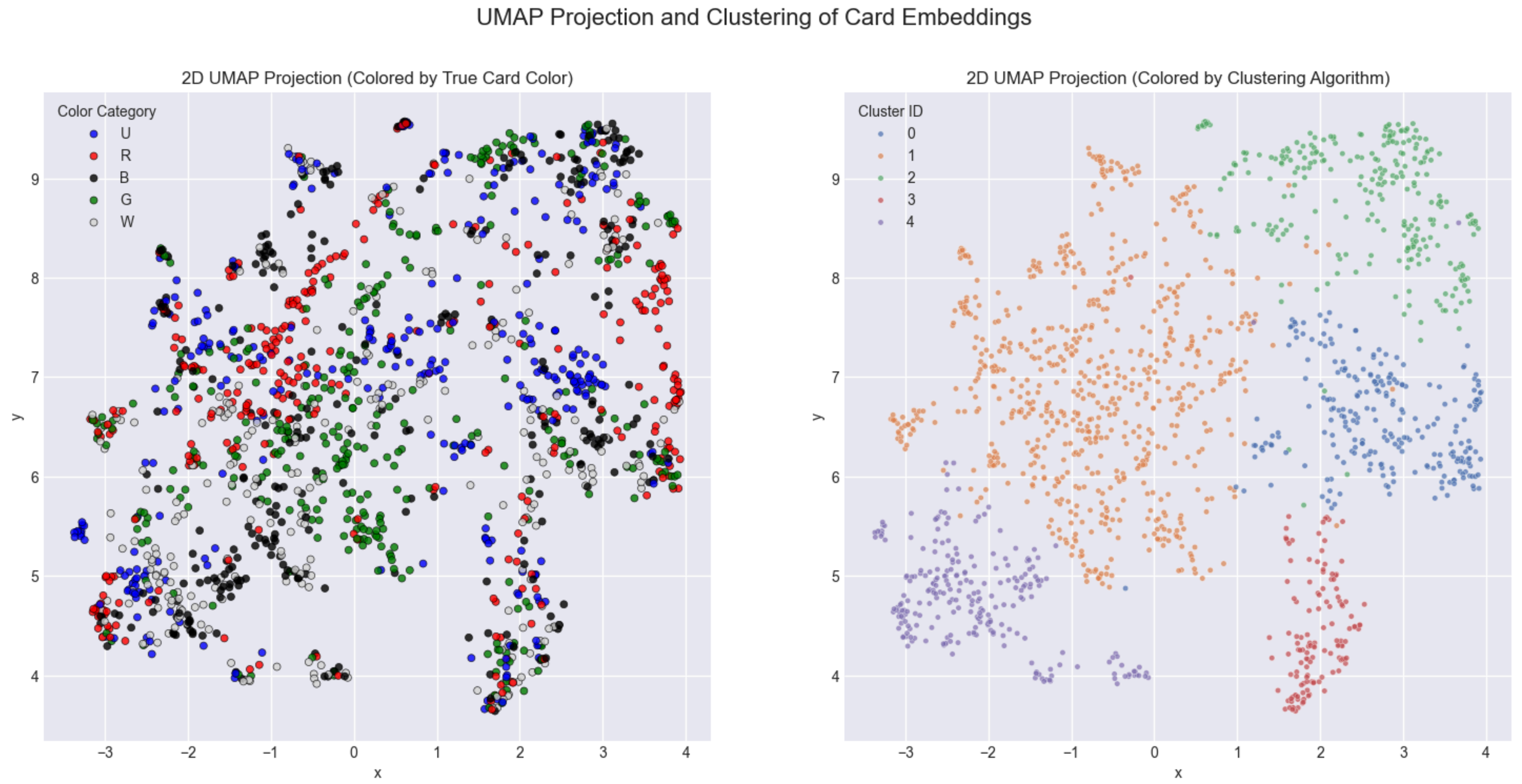
Adjusted Rand Index (ARI): 0.0177
Normalized Mutual Information (NMI): 0.0242
Homogeneity: 0.0230
Completeness: 0.0256
V-Measure: 0.0242

Contingency Matrix (rows=true, cols=predicted):
[[38 124 71 21 62]
[42 201 62 26 20]
[50 151 54 25 33]
[73 105 56 30 53]
[46 124 28 33 78]]

Con normalizzaizone



Senza normalizzaizone



Limitazioni principali

Natura complessa dei dati

Le carte non sono facilmente rappresentabili tramite testo, e sono presenti diverse contraddizioni della ‘color pie’ (e.g. creature senza effeti con le stesse statistiche in colori diversi)

Natura semplificata degli embedding

Utilizzare i campi scelti all’inizio è un buon punto di partenza, ma non è sufficiente passarli attraverso S-BERT per ottenere degli embedding significativi