Improving Product Categorization from Label Clustering

Alexander Friedman, Alexandra Porter, Alexander Rickman
CS229: Machine Learning Class Project
{ajfriedman,amporter,arickman}@stanford.edu

Motivation

In a massive online store, an intractably large set of keywords to describe books can easily be acquired by either seller input or automatic searching of the text. Our goal is to organize a massive set of labels applied to a set of books to use for categorization. We implement an algorithmic and application based project to analyze data from Amazon web-crawl data of books and their categorizations. We embed labels into a feature space, and apply clustering approaches to find interesting features such as redundancies, hierarchies, and anomalies.

Methods

- **Node2vec** The node2vec algorithm [1] samples a set of random walks and then performs stochastic gradient descent on the feature representation of the vertices. The loss function is the similarity of the pairs of representations, given that the vertices appear together.
  - $|V| \times d$ parameter matrix
  - For $u \in V$, $N_S(u) \subset V$ is neighborhood with sampling strategy S
  - Maximize objective function:

$$\max_{f} \sum_{u \in V} \left[-\log Z_u + \sum_{n_i \in N_S(u)} f(n_i) \cdot f(u) \right]$$

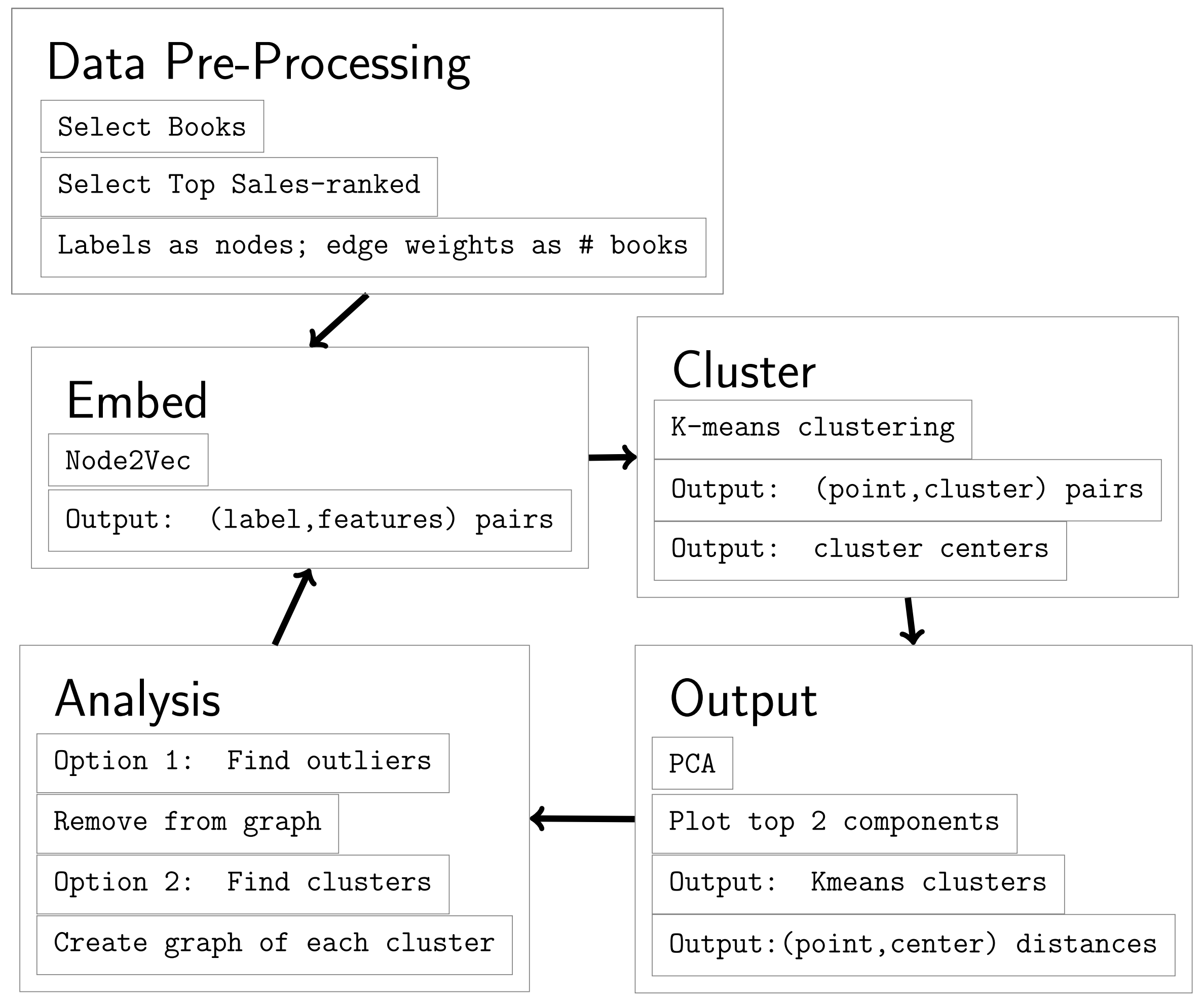
- **Clustering** Once we have node2vec representations of the network, we cluster with K-means [3]. Based on subjective observation and testing on the data set, we specified the number of clusters as 6:

```
procedure K-MEANS($k$, pointset)
    while centers change do
        clustercenters = k random points
        for $p \in$ pointset do
            center[p] = $\operatorname{argmin}_{c \in \text{clustercenters}}$ Dist(p,c)
        for $c \in$clustercenters do
            c =mean({p: center[p] = c})
```

Dataset

The Amazon dataset contains metadata on 350,000 books, including the categories ("labels") to which each book belongs. The graph dataset which we input into Node2Vec was created by using labels as nodes and generating edges between nodes whenever a book belonged to multiple labels. Labels in the original Amazon dataset can be described as a forest. These labels can often be redundant, so our model aims to detect these redundancies so they can be replaced with a cleaner labeling scheme.

Framework and Implementation



Data Pre-Processing
Select Books
Select Top Sales-ranked
Labels as nodes; edge weights as # books

Embed
Node2Vec
Output: (label,features) pairs

Cluster
K-means clustering
Output: (point,cluster) pairs
Output: cluster centers

Output
PCA
Plot top 2 components
Output: Kmeans clusters
Output:(point,center) distances

Analysis
Option 1: Find outliers
Remove from graph
Option 2: Find clusters
Create graph of each cluster

Figure 1: Framework

As seen above, we iterate through a workflow of embed, cluster, plot, analyze, and repeat. In this process we adjust parameters of both the node2vec and clustering models We can use this system to detect/remove outliers before optionally re-embedding. We can also select a cluster from the initial run, then re-embed and re-cluster that cluster, repeating numerous times in order to collect redundant categories and analyze label hierarchies. After analysis, we select an induced subgraph of the original graph to re-embed and continue the cyclic process. We use the scikit-learn package to cluster and plot [2].

Results

- **Anomaly Detection and Removal**

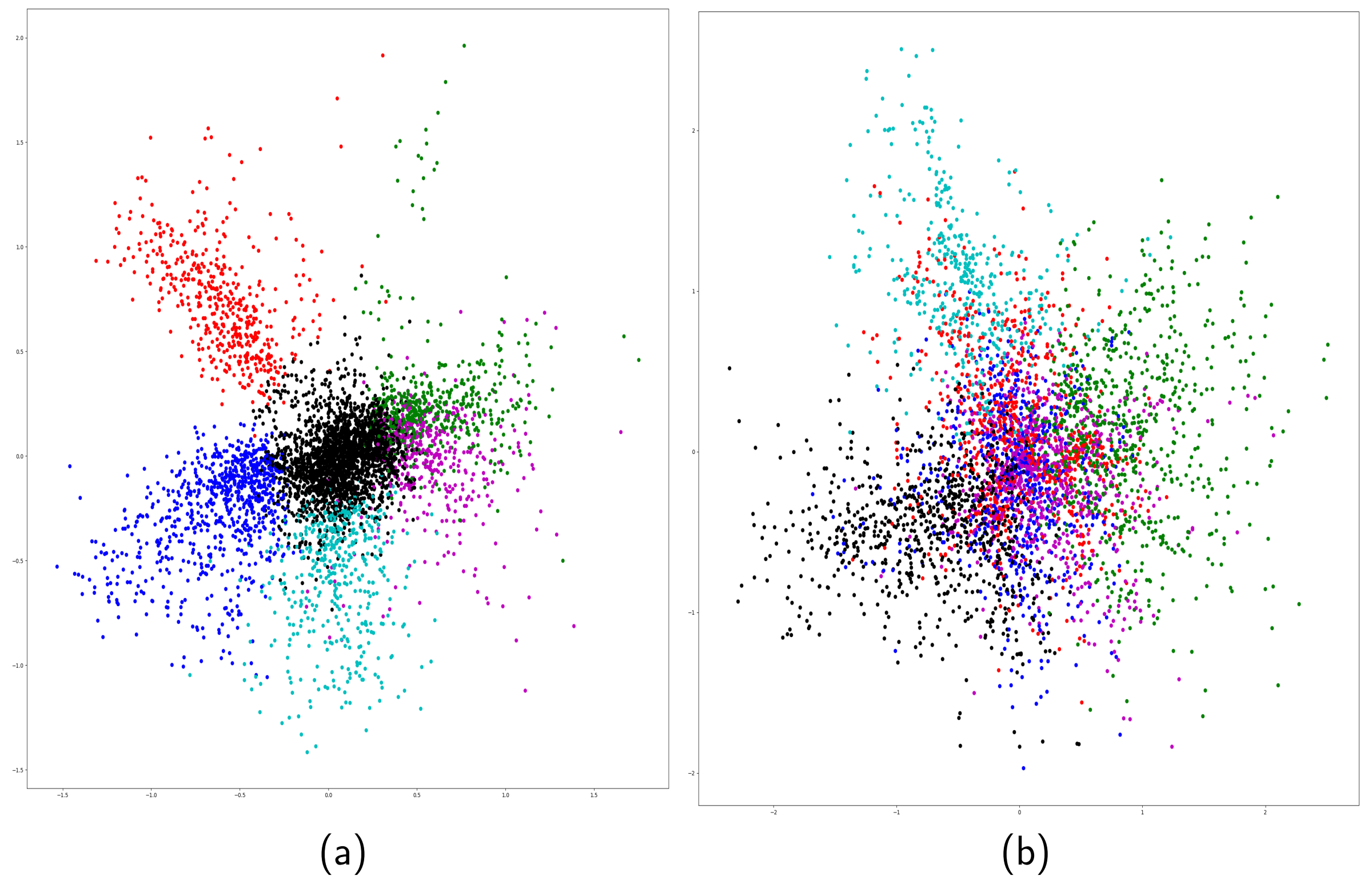
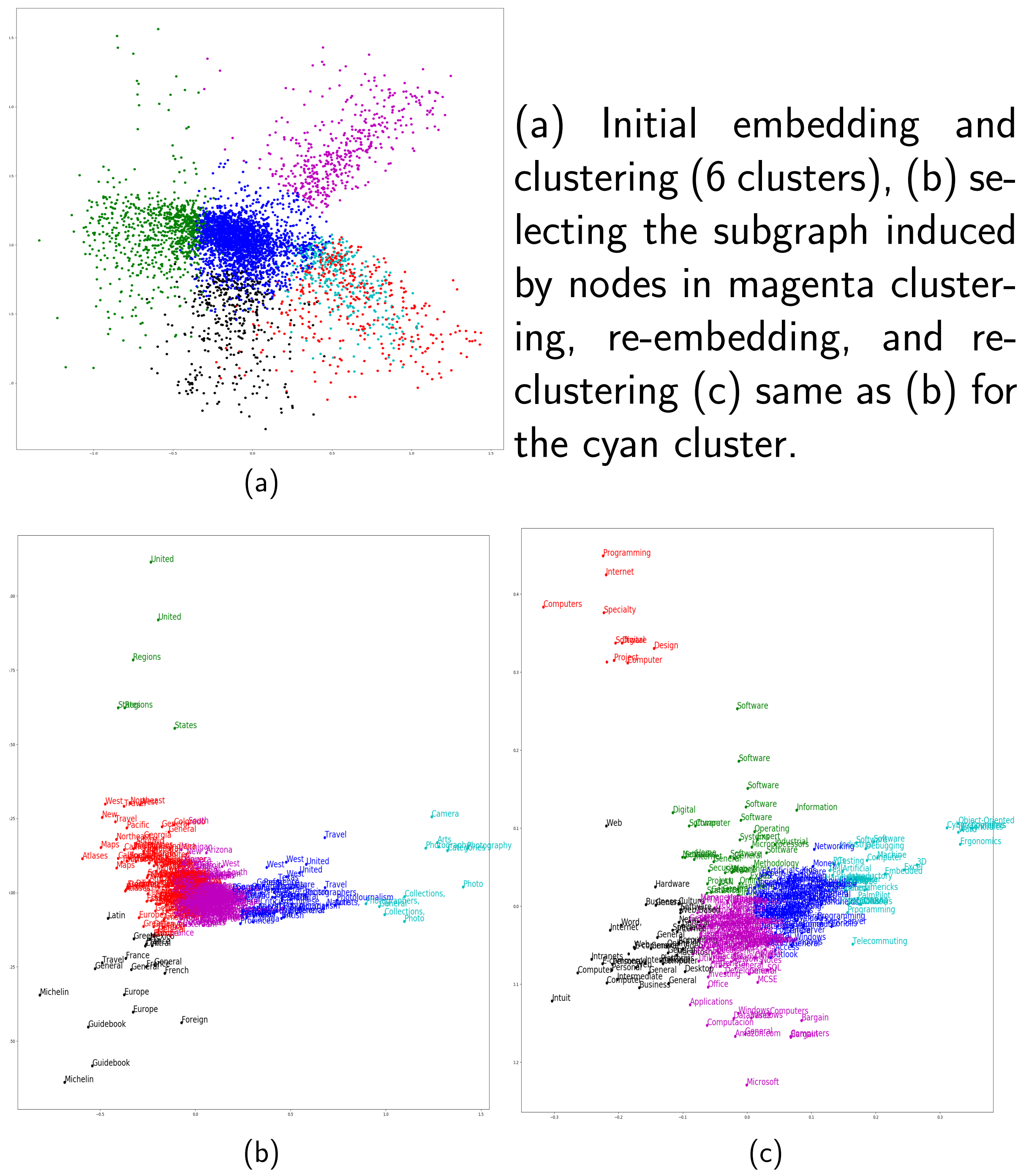


Figure 2: (a) Original clustering (6 clusters), (b) Anomalies removed from graph and re-embedded before another clustering.

- **Nested Label Associations**



(a) Initial embedding and clustering (6 clusters), (b) selecting the subgraph induced by nodes in magenta clustering, re-embedding, and re-clustering (c) same as (b) for the cyan cluster.

Figure 3: Creating nested clusters.

Analysis

- **Anomalies** We use Euclidean distances of points from K-means centroids to detect outliers (as seen in the table below). We can directly remove these outliers from the plots, but we hypothesized that removing outliers from the graph and re-embedding before re-plotting would produce more cohesive clusters. As seen in Figure 2(b), removing anomalies results in less clearly defined clusters, likely due to the cluster structure being primarily defined by the anomalies. We hypothesize that the graph induced by non-anomalous nodes is relatively uniform and thus lacks structure for our method to identify.

Distance from Center	Label
1.9203707947953235	Subjects[1000]
1.9626659852879147	Instruction[11811]
2.0156069220765436	Books[283155]
2.0276880269223216	Poetry[9966]
2.1297687095091673	Foreign Languages[11773]
2.2177811099675195	Dictionaries & Thesauruses[11475]
2.4396388017039103	General[725800]

- **Label Organization** After two iterations of embedding and clustering, we see that groups are mostly made up of labels which are redundant or closely related. Below are 3 examples of label sets (strings as they appear in the data) found in a cluster (shown in Figure 3(b),(c))

Cluster 1	Cluster 2	Cluster 3
Regions[17228]	Computer	Arts
Regions[640504]	Computers	Camera
States[17263]	Design	Categories[493964]
States[640538]	Digital	Collections,
United	Internet[768564]	Collections,
United	Programming[3839]	General[2050]
	Project	Photo
	Software	Photo
	Specialty	Photographers,
	[229534]	Photography
		Photography[2020]
		[172282]

Future Work

- Additional parameter optimization: node2vec search strategies (depth vs. breadth), kmeans clusters, outlier threshold.
- Determine necessary number of nested label clustering steps to find all redundancy.
- Additional applications: other product categorizations, financial transaction networks, telecommunications networks, pharmaceutical co-prescription data.

References

[1] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864. ACM, 2016.

[2] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python . *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[3] Saedsayad.com. K-means clustering, 2018.