Online Social Community City Classification

Jiarui Wang[1][0009−0007−2528−7473], George Barnett[1][0000−0002−7511−1886], Norman Matloff[1][0000−0001−9179−6785], and S. Felix Wu[2][0000−0001−6033−5353]

[1] University of California, Davis, Davis CA 95616, USA
{jrwwang,gabarnett,nsmatloff}@ucdavis.edu
[2] National Cheng-Kung University, Tainan City, Taiwan
sfelixwu@gs.ncku.edu.tw

Abstract. The public page is a popular online social community platform. These pages form a network by liking each other. Location classification of public pages has been studied at the country and state levels. In this paper, we explored the task of public page classification by cities within California. We introduced a virtual geographic structure for city clusters resembling counties in California. We developed a clustering algorithm that leverages the confusion matrix from flat city classification to construct the virtual geographic city structure. Then, adopting a two-stage hierarchical classification strategy—first classifying pages by city cluster and then within clusters by city—we enhanced the accuracy from 0.6928 of flat city classification to 0.8014.

Keywords: Online social networks · Online community classification · Location.

1 Introduction

In the digital realm of online spaces, people's behaviors remain closely linked to location. Individuals tend to show greater interest in local news, are more likely to connect with nearby friends, and have preferences for local dining options, among other location-centric activities. Location information plays a crucial role in both economic activities and public services, including targeted news dissemination, product and service recommendations, and emergency event notifications.

Since the inception of online social network platforms, automatically identifying users' geographic locations has gained popularity. A substantial body of research has explored various methods for geolocating users. Some studies predict location based on content analysis, including words in posts, comments, and tweets. Other research examines user networks, such as friendships and following relationships, to predict locations based on the tendency of users to interact with geographically close individuals.

The geolocation of online social communities, such as public pages and Reddit, which serve as digital town halls for information dissemination and user discussion, has not been extensively explored. The task of predicting the geolocation of geographically unlabeled public pages has been approached with varying levels of granularity. [8] Hong introduced the Majority Voting method to categorize public pages by country. In our previous study[20], we furthered this research by utilizing the GraphSAINT model in conjunction with neighborhood state distribution (NSD) feature vectors. This approach facilitated the more challenging task of classifying pages into specific sublocations, such as States within the U.S.

Classifying public pages by cities presents a greater challenge because a city is a much smaller and more fragmented area than a country or state. For example, our dataset includes 630 California cities, which complicates classification.

In this paper, we introduced a virtual geographic structure of cities, which are city clusters resembling counties, to enhance classification performance. This virtual geographic structure of cities is not represented in the data explicitly. The composition of cities in each cluster results from our clustering algorithm, which is based on the confusion matrix of the flat city classification. Based on the results of the clustering, we implemented a two-stage hierarchical classification method that classifies pages by city clusters first and then by cities within each cluster. These innovations have significantly improved our city classification performance.

2 Data Description

2.1 Data Acquisition

In this paper, we utilize the same dataset of public pages as presented in [20]. The metadata for each page includes its ID, name, description, category, country, city, the other pages it likes, and the number of fans, among other details. Importantly, these datasets do not contain any private user information. To collect the data, we employed snowball sampling[9], initiating the crawl from several popular public pages and progressively moving to the pages they like. This process naturally constructs a directed graph of public pages.

2.2 Data Cleaning

We differentiated between deterministic and non-deterministic pages. Deterministic pages correspond to cities with unique names across the United States, while non-deterministic pages are linked to cities whose names are shared by cities in different states. For the purposes of city classification, we focus exclusively on deterministic pages.

Public pages from California, having the highest number of deterministic pages, serve as our dataset for city classification experiments. The California page graph comprises 324,887 pages and 2,378,881 edges, encompassing 58 counties and 630 cities.

However, we encountered cities listed as ground truth that were absent from our city database, including some rural and small community areas within larger cities or counties. To retain as much data as possible, we manually relabeled these pages to their nearest recognized city. This process involved identifying these cities individually, relocating urban communities to their larger parent cities, and rural communities to the nearest cities in our database. As a result, we modified the city labels for 26,371 pages.

3 Classification Baseline

3.1 GraphSAINT Model

The GraphSAINT model addresses the challenge of processing large graphs by reducing them into smaller, sampled subgraphs through random walk sampling[23, 26]. This method significantly decreases memory usage compared to the GCN strategy, which involves loading the entire graph into GPU memory. By transforming the original graph into manageable subgraphs, GraphSAINT enhances the model's ability to process large datasets, thereby improving training efficiency and speed. Consequently, we employ two-layer GraphSAINT models for all experiments in this chapter, utilizing the PyTorch Geometric (PyG) framework, a specialized tool for Graph Neural Networks[7].

3.2 City Neighborhood Distribution Vector

To effectively classify pages by city, we require not only the GraphSAINT model to understand the page graph's topology but also node features that offer additional information to enhance performance.

Building on the findings from [20], we have demonstrated that neighborhood state distribution serves as an effective node feature for page state classification. Extending this approach to city classification, we introduce the city neighborhood distribution vector ($City - ND$) as a novel node feature.

The $City - ND$ vectors represent the ratio of a page's neighbors from each city to its total number of neighbors. Since every page in the California page graph is connected, these vectors are guaranteed to be non-zero, offering a reliable feature for machine learning-based classification. To ensure a thorough understanding of a page's local network, we calculate $City - ND$ for both one-hop and two-hop distances, considering neighbors connected through inward, outward, and undirected edges. This comprehensive, multi-faceted strategy enriches the representation of page associations, thereby enhancing the accuracy of city classification. The formulation of the $City - ND$ vector is as follows:

$$\begin{aligned} City - ND(Page) = [[\\ & City - IND_1(Page, City_i), City - IND_2(Page, City_i), \\ & City - OND_1(Page, City_i), City - OND_2(Page, City_i), \\ & City - UND_1(Page, City_i), City - UND_2(Page, City_i) \\ & \qquad] : i \in 1, ..., N_{number\ of\ cities}] \end{aligned} \tag{1}$$

where:

- $City{-}IND_k(Page, City_i)$ denotes the inward city neighborhood distribution for $City_i$, calculated within a k-hop distance from the $Page$.
- $City - OND_k(Page, City_i)$ denotes the outward city neighborhood distribution for $City_i$, calculated within a k-hop distance from the $Page$.
- $City - UND_k(Page, City_i)$ denotes the undirected city neighborhood distribution for $City_i$, calculated within a k-hop distance from the $Page$.

Furthermore, each element of the $City{-}ND$ for a page, whether $City{-}IND$, $City - OND$, or $City - UND$, is defined as the ratio of neighbors from city i within a j-hop distance, normalized by the total number of neighbors across all cities within the same hop distance:

$$City - XND_j(Page, City_i) = \frac{XNeighbor_{ij}}{\sum_{i=1}^{N_{number\ of\ cities}} XNeighbor_{ij}},$$
$$i \in \{1, ..., N_{number\ of\ cities}\}, j \in \{1, 2\}, X \in \{I, O, U\} \quad (2)$$

Where:

- i denotes the ith city.
- j denotes the one-hop or two-hop distance.
- X denotes edge directions, inward I, outward O, or undirected U.
- $XNeighbor_{ij}$: the total number of neighbors from City i within j hop distance for inward I, outward O, or undirected U edge direction.

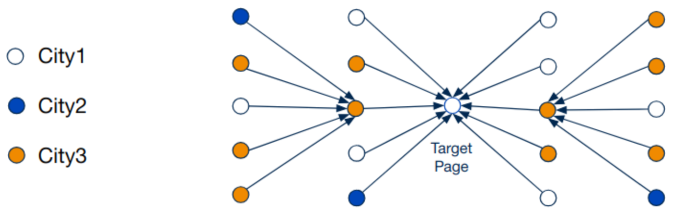

Fig. 1: Example of a two-hop inward neighborhood for a target page within a page graph covering three cities

For example, Figure 1 illustrates the two-hop inward neighborhood of a target page within a page graph that includes three cities. For the target page, the one-hop inward city neighborhood distribution, $City - IND_1(Target)$, is [0.5, 0.1, 0.4], and the two-hop inward city neighborhood distribution, $City - IND_2(Target)$, is [0.35, 0.5, 0.15]. Given the dataset encompasses 630 cities, the $City - ND$ vector features 3780 dimensions for each page. This characteristic is leveraged for our baseline experiment.

Using the GraphSAINT model and the city neighborhood distribution vectors as node features, we achieved a page city classification accuracy of 0.6928, as shown in Table 1. This accuracy is lower than the page state classification accuracy of 0.8752, reported in [20] and Table 1.

Table 1: Baseline accuracy for Pages in California Page Graph

Baseline	Overall Accuracy
Page City Classification	0.6928
Page County Classification	0.8869
Page State Classification	0.8752

3.3 County Neighborhood Distribution Vector

Page city serves as the ground truth data. We can derive the page county from the page city. Given that the city classification accuracy in Table 1 falls short of the state classification accuracy, we also undertake county classification as an additional reference point. This effort aims to explore avenues for enhancing the performance of page city classification.

We introduce the county neighborhood distribution ($County - ND$) vectors as node features for the classification of page counties. The definition is as follows:

$$\begin{aligned}
&County - ND(Page) = [[\\
&\quad County - IND_1(Page, County_i), County - IND_2(Page, County_i),\\
&\quad County - OND_1(Page, County_i), County - OND_2(Page, County_i),\\
&\quad County - UND_1(Page, County_i), County - UND_2(Page, County_i)\\
&\qquad] : i \in 1, ..., N_{number\ of\ counties}]
\end{aligned} \tag{3}$$

where:

- $County - IND_k(Page, County_i)$ denotes the inward county neighborhood distribution for $County_i$, calculated within a k-hop distance from the $Page$.
- $County - OND_k(Page, County_i)$ denotes the outward county neighborhood distribution for $County_i$, calculated within a k-hop distance from the $Page$.
- $County - UND_k(Page, County_i)$ denotes the undirected county neighborhood distribution for $County_i$, calculated within a k-hop distance from the $Page$.

In the California page graph, page cities are associated with 58 distinct counties, resulting in 364 dimensions in the county neighborhood distribution ($County - ND$) vectors. Employing the GraphSAINT model with county neighborhood distribution vectors as node features, we achieved a county classification accuracy of 0.8869, detailed in Table 1. This accuracy surpasses the city classification accuracy of 0.6928 by a large margin. This outcome suggests potential avenues for enhancing the performance of city classification.

4 City Classification Feature Engineering

4.1 Integrate Predicted County Information

To improve the page city classification performance, our initial strategy focused on analyzing and adjusting the expressiveness of features. Specifically, the $City-ND$ vectors comprise six subvectors: $City-IND_1$, $City-OND_1$, $City-UND_1$, $City-IND_2$, $City-OND_2$, and $City-UND_2$. Each subvector is a 630-dimensional vector, corresponding to the 630 cities. For a target page associated with city A, if city A's distribution in any of these subvectors is the highest and unequivocal, then there is a strong likelihood that the page will be correctly classified to city A. For instance, as shown in Table 2, within the subvector $City-IND_1$, the highest city distribution that matches the city label accounts for 70.42% of the pages. Among these pages, 73.99% have the matching city label uniquely, without ties to other city labels.

Table 2: Percentage of highest distribution match and no tie for city label

Subvector	Highest Distribution Match City Label %	no Tie %
$City-IND_1$	70.42	73.99
$City-OND_1$	82.83	38.09
$City-UND_1$	68.85	84.99
$City-IND_2$	63.65	82.51
$City-OND_2$	78.34	38.69
$City-UND_2$	57.97	98.40

Given the superior performance of page county classification over city classification in Table 1, we explored the integration of county information into the $City-ND$ vectors to potentially enhance accuracy. Initially, we predicted each page's county label through county classification, then amplified the city distribution values within the $City-ND$ vectors for cities corresponding to the predicted county. This approach aimed to highlight the cities within the predicted counties. However, this modification did not significantly improve accuracy, suggesting that amplifying the city distribution for all cities in the predicted county may not be the correct way to integrate the county information. We need to explore other options.

5 City Classification within Counties

5.1 Derived County Classification

We compare the baseline city classification and baseline county classification by analyzing the county classification accuracy derived from the baseline city classification results, as discussed in Section 3.2. Consider a scenario where $City-A$ and $City-C$ belong to $County-A$, and $City-B$ belongs to $County-B$; these represent ground truth one-to-one relationships. If $Page-A$, with a ground truth label of $City-A$ and consequently belonging to $County-A$, is correctly classified as $City-A$, it implies that $Page-A$ is also correctly classified as belonging to $County-A$. Conversely, if $Page-A$ is misclassified as $City-C$, it is incorrectly classified at the city level but correctly at the county level ($County-A$). However, if $Page-A$ is misclassified as $City-B$, it indicates an incorrect classification at both the city and county levels, as it would be incorrectly assigned to $County-B$. This derivation approach allows for an assessment of how the baseline city classification performs at the county level.

The derived county classification accuracy stands at 0.8425, lower than the baseline county classification accuracy of 0.8869 but significantly surpassing the baseline city classification accuracy of 0.6928. This observation suggests the potential benefit of first classifying pages by county using a county classifier, which demonstrates superior performance at the county level, followed by classifying pages into specific cities within those counties. This method necessitates a two-step approach: initially classifying pages by county, then further classifying pages into cities within those counties. This process requires two distinct types of classifiers: a county classifier and fifty-eight city classifiers, one for each county.

5.2 County Classifier

The county classifier employs a two-layer GraphSAINT model to predict the classification of pages across 58 counties within the California page graph dataset. Utilizing county neighborhood distribution ($County-ND$) vectors, as introduced in Section 3.3, as node feature inputs, and the California (C.A.) Page graph as the graph input, this classifier achieves a high prediction performance, with an accuracy of 0.8869

5.3 City Classifier For Each County

All city classifiers for each county utilize a two-layer GraphSAINT model, along with city neighborhood distribution ($City-ND$) vectors, as introduced in Section 3.2, as node features to predict page classifications within each county. The key differences include the graph input, which is the specific county page graph for each classifier, and the ($City-ND$) vectors for each page, calculated based on the cities within the respective county. The total number of cities within each county ranges from 1 to 60, leading to significantly fewer dimensions in the ($City-ND$) vectors compared to the baseline ($City-ND$) vectors for 630 cities. However, this approach requires the training and inference process to be executed fifty-eight times.

5.4 City Classification Accuracy

After training two kinds of classifiers—a county classifier and city classifiers for each county—we perform inference for all California pages in two steps:

1. Classify all California pages into different counties using the county classifier based on the California page graph and $County - ND$ node features. We disregard the pages misclassified at the county level, retaining only those correctly classified for the subsequent step. However, these misclassified pages at the county level are still included as errors for overall accuracy calculation.
2. For each county, we take the pages correctly classified to the respective county from step one and classify these pages into different cities within the county using the city classifier specific to that county. This classification is based on the page graph and $City - ND$ node features specific to the respective county. We record the pages correctly classified at the city level in this step to calculate the overall city classification accuracy later. This step is repeated for every county.

Table 3: Accuracy for City Classification within Counties

Algorithms	City Accuracy	County Accuracy	Total Pages
City Classification within Counties	0.7494	0.8869	324887
Baseline City Classification	0.6928	0.8425	324887
Improvement	0.0566	0.0444	324887

The overall accuracy of city classification within all counties is 0.7494, significantly higher than the baseline city classification accuracy of 0.6928, as shown in Table 3. The improvement in city level accuracy is greater than the improvement in county level accuracy between these two methods, as illustrated in Table 3. This indicates that the higher county level accuracy achieved by the county classifier in step one not only improves performance at the county level but also enhances city level classification performance within each county.

5.5 Hierarchical Classification

Page city classification within counties essentially adopts a hierarchical classification approach, commonly employed in real-world classification problems [18][24][17][22]. By contrast, the baseline city classification represents a flat classification model. Here, page cities serve as the target classes, while page counties act as meta-classes for these cities, forming a natural taxonomy based on ground truth. This method, which involves classifying pages into a meta-class followed by classification within that meta-class, exemplifies hierarchical classification. Such an approach benefits from model specialization in multi-stage classification[17], where training distinct models for data subsets or specific information leads to improved performance compared to a singular, flat classifier that struggles to encompass all information effectively. As evidenced in Table 3, ensembled specialized models demonstrate superior performance at every class level compared to a single model.

6 City Classification within Clusters

6.1 Building Hierarchical Structure

In hierarchical classification, two primary types of meta-classes are identified: the first type comprises pre-existing taxonomies related to the target classes, such as counties for cities in the context of city classification within counties. The second type involves meta-classes that are newly created based on the similarity among target classes. A common methodology entails initially conducting a flat classification of the target classes, followed by the generation of a confusion matrix for this classification. The confusion matrix serves to reveal class similarities, which are then used to construct meta-classes. Subsequently, based on the hierarchical structure of these meta-classes, hierarchical classifiers are developed to execute the hierarchical classification process.

The critical step in forming meta-classes involves determining the optimal number of these groups by analyzing the classification confusion matrix. Attempting to explore all possible clustering configurations is computationally equivalent to identifying all possible partitions of n samples. The complexity of this task is quantified by Bell numbers, which increase rapidly in a manner known as combinatorial explosion. For example, the number of possible partitions for just 10 items is 115,975. Considering the challenge involves 630 city classes, the task of assessing all potential clustering options for these classes is practically unfeasible due to the exponential growth in the number of possible partitions.

6.2 Affinity Clustering

A common strategy for constructing meta-classes involves adopting a systematic approach that leverages clustering algorithms. These algorithms cluster classes based on their similarity, as indicated within the confusion matrix. This method is documented in various studies, including those focused on the use of confusion matrices for hierarchical classification construction and others that explore semantic and probability-based approaches to understanding class similarities[3][19][27][6]. In this research, given the unknown optimal number of meta-classes, we reference the affinity clustering algorithm, which organizes classes into a suitable number of meta-classes based on their similarity distances.

Affinity clustering is particularly beneficial in scenarios where the number of clusters is not predetermined. By adjusting its configuration, we managed to group the city classes into two distinct sets of city clusters: one comprising 3 city clusters and another encompassing 75 city clusters. Subsequent hierarchical classifications were conducted based on these two varying hierarchical structures to facilitate a comparative analysis. Each hierarchical classification setup requires a dedicated city cluster classifier along with multiple city classifiers. Table 4 illustrates how the adoption of 75 city-cluster and 3 city-cluster hierarchical structures enhances the accuracy of city classification.

Table 4: Accuracy for Hierarchical Classification

Hierarchical Structure	City Accuracy	Cluster Accuracy
Baseline City Classification (1 Cluster)	0.6928	1
75 City Clusters (Affinity Clustering)	0.7512	0.8573
3 City Clusters (Affinity Clustering)	0.7744	0.9375
17 City Clusters (Our Clustering Method)	0.8014	0.9778

6.3 Our Clustering Method

Intuition of Confusion Matrix The confusion matrix of the flat page city classification reveals the extent to which pages from each city are incorrectly labeled as belonging to other cities. A high number of misclassified pages between city A and city B suggests that the flat city classifier struggles to distinguish pages between these two cities, indicating a certain level of similarity or closeness between them in the context of 630 cities. This challenge arises because the flat city classifier must differentiate among pages from all 630 cities, making it difficult to capture the subtle distinctions between any two specific cities, such as city A and city B. For instance, as depicted in Figure 2, if cities A and B have a high misclassification rate, and cities C, D, and E also share high misclassification rates among themselves, it implies that cities A and B are close to each other, and cities C, D, and E form another close group. We could interpret these findings as indicating two clusters, with the flat classifier being more capable of distinguishing between these two clusters, since cities A and B have lower misclassification rates with cities C, D, and E.

City Clustering Based on Misclassification Rates Based on an intuitive analysis of the confusion matrix, we propose an algorithm for clustering cities according to their misclassification rates for the training data. For each city, we sort the misclassification rates from its respective row in the confusion matrix (normalized by row) in descending order. By connecting a city to its neighbor with the highest misclassification rate via an edge, we cluster these two cities

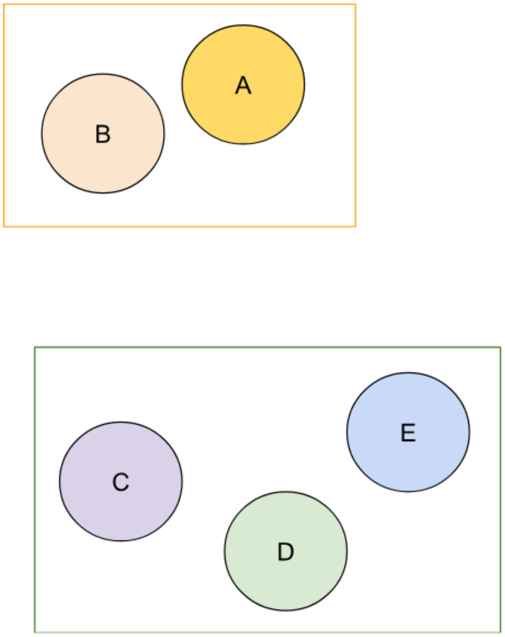

Fig. 2: City Cluster Example

Algorithm 1 City Clustering Based on Misclassification Rates

```
Require: $desc\_ordered\_rates$, $max\_edges$, $thresholds$
Ensure: Edge list $E$ with tuples (city, neighbor, rate)
$E \leftarrow []$                                                ▷ Initialize edge list
for $city \leftarrow 0$ to $number\_of\_cities - 1$ do
    $edges\_added \leftarrow 0$
    for $j \leftarrow 0$ to $number\_of\_cities - 1$ do
        $(rate, neighbor) \leftarrow desc\_ordered\_rates[city][j]$
        if $neighbor \neq city$ and $thresholds[edges\_added] \leq rate$ then
            $E$.append($(city, neighbor, rate)$)
            $edges\_added \leftarrow edges\_added + 1$
        end if
        if $edges\_added = max\_edges$ then
            break
        end if
    end for
end for
```

together in the city graph. Upon adding edges for all cities, the resulting city clusters are identified as disconnected components within the graph.

The configuration of clusters can be adjusted by two parameters: the number of edges to add based on the highest misclassification rates for each city, and the misclassification rate threshold for including an edge. Increasing the number of edges enhances the connectivity of the city graph and decreases the number of clusters. Conversely, raising the threshold for edge inclusion filters out connections with lower misclassification rates, leading to reduced connectivity and an increased number of city clusters. The detailed methodology is outlined in Algorithm 1.

By setting the number of edges to 1 and the threshold rate to 0, we exclusively link each city to its most frequently misclassified neighboring city, resulting in the formation of 17 city clusters. Subsequent hierarchical classification leverages this cluster configuration.

If we set the threshold list to [0.05] for only one edge, it leads to 55 city clusters. By introducing a second edge with a threshold of 0.5, making the threshold list [0.0, 0.5], it consolidates the cities into 11 clusters. This demonstrates the tunability of our algorithm.

6.4 Cluster Neighborhood Distribution Vector

Based on the city cluster configurations, we calculate node features for cluster classification, introducing the cluster neighborhood distribution ($Cluster - ND$) vector as the cluster-level node feature. The definition is as follows:

$$
\begin{aligned}
&Cluster - ND(Page) = [[\\
&\quad Cluster - IND_1(Page, Cluster_i), Cluster - IND_2(Page, Cluster_i),\\
&\quad Cluster - OND_1(Page, Cluster_i), Cluster - OND_2(Page, Cluster_i),\\
&\quad Cluster - UND_1(Page, Cluster_i), Cluster - UND_2(Page, Cluster_i)\\
&\qquad\qquad\qquad] : i \in 1, ..., N_{number\ of\ clusters}]
\end{aligned}
\tag{4}
$$

where:

- $Cluster - IND_k(Page, Cluster_i)$ denotes the inward cluster neighborhood distribution for $Cluster_i$, calculated within a k-hop distance from the $Page$.
- $Cluster - OND_k(Page, Cluster_i)$ denotes the outward cluster neighborhood distribution for $Cluster_i$, calculated within a k-hop distance from the $Page$.
- $Cluster - UND_k(Page, Cluster_i)$ denotes the undirected cluster neighborhood distribution for $Cluster_i$, calculated within a k-hop distance from the $Page$.

In the California page graph, with pages associated with 17 city clusters, the $Cluster - ND$ vectors result in 102 dimensions. Using the GraphSAINT model with $Cluster - ND$ vectors as node features, we achieved a cluster classification accuracy of 0.9778, as detailed in Table 4.

6.5 City Classifier within Each Cluster

City classifiers for each cluster utilize a two-layer GraphSAINT model, along with city neighborhood distribution ($City - ND$) vectors, as introduced in Section 3.2, as node features to predict page classifications within each county. The key differences include the graph input, which is the page graph for each cluster, and the ($City - ND$) vectors for each page, calculated based on the cities within the respective cluster.

6.6 City Classification Accuracy

After training two kinds of classifiers—a cluster classifier and city classifiers for each cluster—we perform inference for all California pages in two steps:

1. Classify all California pages into different clusters using the cluster classifier based on the California page graph and $Cluster - ND$ node features. We disregard the pages misclassified at the cluster level, retaining only those correctly classified for the subsequent step. However, these misclassified pages at the cluster level are still included as errors for overall accuracy calculation.
2. For each cluster, we take the pages correctly classified to the respective cluster from step one and classify these pages into different cities within the cluster using the city classifier specific to that cluster. This classification is based on the page graph and $City - ND$ node features specific to the respective cluster. We record the pages correctly classified at the city level in this step to calculate the overall city classification accuracy later. This step is repeated for every cluster.

The overall accuracy of city classification within all clusters reaches 0.8014, which is significantly higher than the baseline city classification accuracy of 0.6928 (with no hierarchy, implying a single city cluster) and the accuracies achieved using other hierarchical structures, as depicted in Table 4. The accuracy for cluster classification stands at 0.9778. This performance underscores the effectiveness of our clustering algorithm in grouping similar cities within our dataset. Such grouping facilitates the task of the cluster classifier in differentiating pages across clusters, thereby contributing to the improvement in overall city classification accuracy.

This method classifies pages into a meta-class followed by classification within each meta-class. This approach benefits from model specialization in multi-stage classification [17], where training distinct models for data subsets or specific information leads to improved performance compared to a singular, flat classifier that struggles to encompass all information effectively.

7 Related Work

7.1 User Location Prediction

Twitter user location prediction has been extensively studied, with research efforts focusing on both user home location prediction[4, 5, 14] and tweet location prediction [25]. Our interest primarily lies in user home location prediction, which aligns more closely with our objectives. There are two predominant approaches to predicting user home location. The first relies on content analysis, identifying local vernacular or place-specific words, such as "howdy" and "Phillies," which are frequently used in certain regions[4]. The second approach analyzes user networks, focusing on friendships, interactions, or other relational ties to infer location[5].

7.2 Online Community Location Studies

Online communities have been primarily studied in the context of user engagement[11, 1] and information consumption[13]. Facebook, known for its emphasis on location, prioritizes local recommendations and advertising[21]. Several studies have explored the geographical aspects of Facebook communities. For instance, [10] analyzed the location data of businesses' pages to provide geolocation recommendations for new businesses. Another study[16] found that pages belonging to news providers tend to interact more with other pages within the same geographical confines, such as continents and countries. The study [15] detects the geolocation of Twitter user communities by extracting and summarizing users' location data within each community.

7.3 Hierachical Classification

Hierarchical classification is widely utilized in various real-world classification challenges, as documented in the literature [18]. This method is particularly beneficial in domains where classes or categories inherently form hierarchical structures, including bioinformatics, text mining [2, 12], among others [24]. Typically, hierarchical classification involves the initial classification of meta-classes, followed by a more detailed classification within each meta-class. This approach leverages the advantage of model specialization in a multi-stage classification process [17], where employing distinct models for different data subsets or specific types of information can lead to superior performance compared to using a single, flat classifier that may not capture all nuances effectively.

8 Conclusion

In this paper, we looked into the task of public page classification by cities within California. We introduced city, county, and cluster neighborhood distribution vectors as distinctive features for page classifications. With an initial city classification accuracy of 0.6928, the complexity of distinguishing among 630 cities presents a significant challenge. We introduced a virtual geographic city structure resembling counties to improve the classification performance. We developed a clustering algorithm that leverages the confusion matrix from the flat city classification to construct a virtual geographic city structure. Based on this virtual structure, we implemented a two-stage hierarchical classification method, first classifying pages by virtual city clusters and then within clusters by city. This implementation of a cluster-city hierarchical classification achieves a notable improvement in city classification accuracy to 0.8014.

References

1. Kholoud Khalil Aldous, Jisun An, and Bernard J. Jansen. View, like, comment, post: Analyzing user engagement by topic at 4 levels across 5 social media platforms for 53 news organizations. *Proceedings of the International AAAI Conference on Web and Social Media*, 13(01):47–57, Jul. 2019.
2. Lijuan Cai and Thomas Hofmann. Hierarchical document categorization with support vector machines. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, CIKM '04, page 78–87, New York, NY, USA, 2004. Association for Computing Machinery.
3. Paulo Cavalin and Luiz Oliveira. Confusion matrix-based building of hierarchical classification. In Ruben Vera-Rodriguez, Julian Fierrez, and Aythami Morales, editors, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 271–278, Cham, 2019. Springer International Publishing.
4. Zhiyuan Cheng, James Caverlee, and Kyumin Lee. You are where you tweet: a content-based approach to geo-locating twitter users. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, page 759–768, New York, NY, USA, 2010. Association for Computing Machinery.
5. Clodoveu A. Davis Jr., Gisele L. Pappa, Diogo Rennó Rocha de Oliveira, and Filipe de L. Arcanjo. Inferring the location of twitter messages based on user relationships. *Transactions in GIS*, 15(6):735–751, 2011.
6. Xinyang Deng, Qi Liu, Yong Deng, and Sankaran Mahadevan. An improved method to construct basic probability assignment based on the confusion matrix for classification problem. *Information Sciences*, 340-341:250–261, 2016.
7. Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric, 2019.
8. Yunfeng Hong, Yu-Cheng Lin, Chun-Ming Lai, S. Felix Wu, and George A. Barnett. Profiling facebook public page graph. In *2018 International Conference on Computing, Networking and Communications (ICNC)*, pages 161–165, 2018.
9. Sang Hoon Lee, Pan-Jun Kim, and Hawoong Jeong. Statistical properties of sampled networks. *Phys. Rev. E*, 73:016102, Jan 2006.
10. Jovian Lin, Richard Oentaryo, Ee-Peng Lim, Casey Vu, Adrian Vu, and Agus Kwee. Where is the goldmine? finding promising business locations through facebook data analytics. In *Proceedings of the 27th ACM Conference on Hypertext and Social Media*, HT '16, page 93–102, New York, NY, USA, 2016. Association for Computing Machinery.
11. Edward Newell, David Jurgens, Haji Saleem, Hardik Vala, Jad Sassine, Caitrin Armstrong, and Derek Ruths. User migration in online social networks: A case study on reddit during a period of community unrest. *Proceedings of the International AAAI Conference on Web and Social Media*, 10(1):279–288, Aug. 2021.
12. Hao Peng, Jianxin Li, Yu He, Yaopeng Liu, Mengjiao Bao, Lihong Wang, Yangqiu Song, and Qiang Yang. Large-scale hierarchical text classification with recursively regularized deep graph-cnn. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 1063–1072, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.

13. Shahnoor Rahman. Tourism destination marketing using facebook as a promotional tool. *IOSR Journal of Humanities and Social Science*, 22:87–90, 02 2017.
14. Stephen Roller, Michael Speriosu, Sarat Rallapalli, Benjamin Wing, and Jason Baldridge. Supervised text-based geolocation using language models on an adaptive grid. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, page 1500–1510, USA, 2012. Association for Computational Linguistics.
15. Jeanette Ruiz, Jade D Featherstone, and George A Barnett. Identifying vaccine hesitant communities on twitter and their geolocations: a network approach. *Proceedings of Hawaii International Conferences on System Science (HICSS-54)*, 2021.
16. Ana Lucía Schmidt, Fabiana Zollo, Michela Del Vicario, Alessandro Bessi, Antonio Scala, Guido Caldarelli, H. Eugene Stanley, and Walter Quattrociocchi. Anatomy of news consumption on facebook. *Proceedings of the National Academy of Sciences*, 114(12):3035–3039, 2017.
17. T.E. Senator. Multi-stage classification. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, pages 8 pp.–, 2005.
18. Carlos N Silla and Alex A Freitas. A survey of hierarchical classification across different application domains. *Data mining and knowledge discovery*, 22:31–72, 2011.
19. Andrey Temko and Climent Nadeu. Svm-based-clustering-schemes. *Pattern Recognition*, 39(4):682–694, 2006. Graph-based Representations.
20. Jiarui Wang, Xiaoyun Wang, Chun-Ming Lai, and S. Felix Wu. Online social community sub-location classification. In *Proceedings of the 2023 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '23, page 276–280, New York, NY, USA, 2024. Association for Computing Machinery.
21. Rowan Wilken. Places nearby: Facebook as a location-based social media platform. *New Media & Society*, 16:1087–1103, 10 2014.
22. Yunbo Xiong. Building text hierarchical structure by using confusion matrix. In *2012 5th International Conference on BioMedical Engineering and Informatics*, pages 1250–1254, 2012.
23. Mingyu Yan, Zhaodong Chen, Lei Deng, Xiaochun Ye, Zhimin Zhang, Dongrui Fan, and Yuan Xie. Characterizing and understanding gcns on gpu. *IEEE Computer Architecture Letters*, 19(1):22–25, 2020.
24. Zhicheng Yan, Hao Zhang, Robinson Piramuthu, Vignesh Jagadeesh, Dennis DeCoste, Wei Di, and Yizhou Yu. Hd-cnn: hierarchical deep convolutional neural networks for large scale visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 2740–2748, 2015.
25. Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. Who, where, when and what: discover spatio-temporal topics for twitter users. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, page 605–613, New York, NY, USA, 2013. Association for Computing Machinery.
26. Hanqing Zeng, Hongkuan Zhou, Ajitesh Srivastava, Rajgopal Kannan, and Viktor Prasanna. Graphsaint: Graph sampling based inductive learning method, 2019.
27. Damien E. Zomahoun. A semantic collaborative clustering approach based on confusion matrix. In *2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 688–692, 2019.