Adaptive Label Propagation with Entropy-Guided Weighting for Location-Aware Graph Clustering

Doyeol Oh (20211187)
UNIST
South Korea
ohdoyoel@unist.ac.kr

Abstract

Most graph clustering methods in social networks rely solely on structural information, overlooking contextual features such as user locations. In this paper, we propose a learning-free, location-aware graph clustering algorithm that enhances label propagation through entropy-guided adaptive similarity fusion. Specifically, we combine structural and geometric similarity, where the latter is derived from each user's most recent check-in location. By computing the entropy of neighboring label distributions, the algorithm adaptively balances structural and spatial similarity, emphasizing spatial proximity when structural signals are ambiguous.

We evaluate our method on a real-world dataset (Brightkite) from a location-based social networking service. Compared to classical label propagation and the Louvain method, our approach achieves comparable performance in terms of modularity and conductance, while providing more geographically coherent clustering results. Although quantitative improvements in silhouette score are modest, visual analysis reveals that our method captures spatially localized communities more effectively than baseline methods.

Our approach is fully unsupervised, computationally lightweight, and free from any learning or training phase. It offers a scalable alternative to feature-aware models and provides a practical foundation for future extensions that incorporate temporal dynamics.

Full code is available at:

https://github.com/ohdoyoel/unist_cse304_term_project.

1 INTRODUCTION

Graph clustering reveals hidden communities in social networks. Classical methods like Label Propagation (LP) and Louvain rely purely on structural links, making them vulnerable to sparsity and noise. GNN-based approaches such as GCNs support feature-aware clustering but require training and rich node attributes, limiting their use in lightweight, unsupervised settings.

In location-based social networks, user behavior is also shaped by spatial check-ins, yet existing methods often overlook this or rely on complex, trajectory-based models.

We propose a learning-free clustering algorithm that augments LP with entropy-guided similarity fusion, blending structural and spatial signals based on uncertainty. When structural cues are weak, the method leans on recent location data; when strong, it prioritizes topology.

Tested on the Brightkite dataset, our method produces spatially coherent clusters and achieves competitive modularity and conductance, offering a scalable and interpretable alternative to training-heavy models.

Contributions:

- A novel entropy-guided weighting mechanism for combining structure and location in label propagation.
- A simple yet effective use of recent check-ins as spatial features without trajectory modeling.
- Quantitative and qualitative validation on large-scale real-world data.
- A scalable, training-free alternative to GNN-based clustering.

2 RELATED WORK

2.1 Structure-Based Clustering

Label Propagation (LP)[3] and Louvain[1] are classical clustering algorithms that rely solely on graph topology. LP assigns labels via majority vote among neighbors, offering speed but struggling in sparse or noisy graphs. Louvain optimizes modularity through hierarchical merging but likewise ignores node features.

2.2 Feature-Aware and GNN-Based Clustering

Graph Neural Networks (GNNs)[4], such as GCNs and GraphSAGE, enable clustering using both structure and features via learnable aggregation. While effective, they require training, hyperparameter tuning, and often GPU support. In our context—where Brightkite lacks rich node features or ground-truth labels—GNNs are impractical.

2.3 Location-Aware Clustering

Some studies leverage spatial information via location embeddings or trajectory-based models, often in supervised settings. While these capture fine-grained spatial patterns, they are computationally intensive. In contrast, our method uses only the most recent check-in, offering spatial awareness without modeling temporal dynamics or requiring training.

2.4 Adaptive Weighting

Inspired by multi-modal fusion[2], we adopt entropy-guided weighting to combine structure and location in a fully unsupervised setting. Unlike attention-based methods that learn weights, our approach computes fixed weights from neighborhood label entropy—requiring no labels or training.

2.5 Summary

To our knowledge, this is the first method to combine recent location data with label propagation through entropy-guided similarity fusion, offering a lightweight and training-free alternative to feature- or trajectory-based approaches.

3 PROBLEM STATEMENT

3.1 Graph and Objective

We define a social network as an undirected graph $G = (V, E)$, where each node $v_i \in V$ has a recent location l_i based on the latest check-in. The objective is to cluster nodes into $C = \{C_1, \ldots, C_k\}$ such that both structural proximity and spatial locality are preserved.

3.2 Hybrid Similarity Function

To address the limitations of structure-only propagation, we define a hybrid similarity:

$$\mathrm{sim}_{ij} = \alpha_{ij} \cdot \mathrm{sim}_{\mathrm{structure}}(i, j) + (1 - \alpha_{ij}) \cdot \mathrm{sim}_{\mathrm{geometry}}(i, j) \quad (1)$$

where $\alpha_{ij} = \frac{(\alpha_i + \alpha_j)}{2}$ is the adaptive weight. Structural similarity uses Jaccard similarity between neighbors; geometric similarity uses cosine similarity of location embeddings.

3.3 Entropy-Guided Weighting

The weight $\alpha_i \in [0, 1]$ reflects the confidence in structural coherence around node i, computed from label entropy over its neighbors:

$$H_i = -\sum_{l \in \mathcal{L}} p_i(l) \log p_i(l), \quad \alpha_i = 1 - \frac{H_i}{\log |\mathcal{L}|}$$

We normalize entropy by fixing $|\mathcal{L}|$ to the initial number of labels ($|V|$), to ensure a stable entropy range across iterations, independent of the shrinking label set. Lower entropy indicates clearer structure and results in higher α_i, increasing reliance on topology. Conversely, high entropy shifts focus to spatial similarity.

Thus, our method adaptively combines structural and spatial similarity during label propagation, guided by local uncertainty.

4 ALGORITHM

We now describe our proposed algorithm, which extends classical Label Propagation (LP) using hybrid similarity and entropy-guided weighting.

4.1 Classical Label Propagation

LP is a fast, unsupervised algorithm where each node iteratively adopts the most frequent label among its neighbors. While efficient, it is limited by its reliance on pure structure, making it less robust in sparse or noisy graphs.

4.2 Adaptive Label Propagation with Hybrid Similarity

We enhance LP by incorporating a similarity-weighted voting scheme. Each node updates its label based on a hybrid similarity score sim_{ij} (Equation (1)) between itself and its neighbors:

$$L_i^{(t+1)} = \arg\max_{l \in \mathcal{L}} \sum_{j \in N(i)} \mathrm{sim}_{ij} \cdot \mathbb{I}[L_j^{(t)} = l]$$

Here, sim_{ij} combines structural and spatial similarity, with adaptive weights α_{ij} reflecting local structural confidence (see Section 3.2). Nodes with high entropy rely more on spatial signals, while stable nodes favor structural propagation.

4.3 Algorithm Steps

(1) **Initialize**: Assign each node a unique label. Extract location embeddings.
(2) **Precompute Similarities**:
  - Compute structural (Jaccard) and geometric (cosine) similarities for all edges.

(3) **Iterative Propagation**:
  - For each node, compute label distribution and entropy.
  - Compute $\alpha_i = 1 - \frac{H_i}{\log |\mathcal{L}|}$, and $\alpha_{ij} = \frac{(\alpha_i + \alpha_j)}{2}$.
  - Calculate sim_{ij} using the hybrid formula, Equation (1).
  - Update each node's label using weighted majority vote.

(4) **Converge**: Repeat until label assignments stabilize.

Complexity. Each iteration requires $O(n \cdot d)$ computations for similarity updates and label propagation, where n is the number of nodes and d the average degree. Total complexity is $O(n \cdot d \cdot T)$.

4.4 Pseudocode

Algorithm 1 Entropy-Guided Adaptive Label Propagation (E-ALP)

Input: Graph $G = (V, E)$, locations $\{l_i\}$
Output: Labels $\{L_i\}$
for all $i \in V$ **do**
  $L_i \leftarrow i$
end for
for all $(i, j) \in E$ **do**
  Compute $\mathrm{sim}_{\mathrm{str}}(i, j)$, $\mathrm{sim}_{\mathrm{geo}}(i, j)$
end for
while labels not converged **do**
  for all $i \in V$ **do**
    Compute H_i from neighbor labels
    $\alpha_i \leftarrow 1 - \frac{H_i}{\log |\mathcal{L}|}$
  end for
  for all $(i, j) \in E$ **do**
    $\alpha_{ij} \leftarrow \frac{\alpha_i + \alpha_j}{2}$
    $\mathrm{sim}_{ij} \leftarrow \alpha_{ij} \cdot \mathrm{sim}_{\mathrm{str}} + (1 - \alpha_{ij}) \cdot \mathrm{sim}_{\mathrm{geo}}$
  end for
  for all $i \in V$ **do**
    $L_i \leftarrow \arg\max_l \sum_{j \in N(i)} \mathrm{sim}_{ij} \cdot \mathbb{I}[L_j = l]$
  end for
end while

5 EXPERIMENTS

5.1 Datasets

We use the **Brightkite** dataset from the SNAP repository, a location-based social network with user check-ins and friendship links.

The graph contains 58,228 nodes and 214,078 undirected edges, formed by keeping only mutual friendships. From 4.5M check-ins (April 2008–October 2010), we extract each user's most recent location to serve as a spatial feature, avoiding the need for full trajectory modeling.

5.2 Baselines and Evaluation Metrics

We compare our method with two baselines:

- **Label Propagation**: Unsupervised propagation based only on structural adjacency.
- **Louvain**: A modularity-optimizing hierarchical community detection algorithm.

To evaluate clustering performance, we use:

- **Modularity**: Quantifies the strength of community structure by comparing the density of intra-cluster edges to a null model.
- **Conductance**: Measures the ratio of external connections to the total degree of a cluster, indicating boundary sharpness.
- **Silhouette Score**: Captures the compactness and separation of clusters in geometric or embedded space.
- **Number of Labels**: Represents the total number of distinct clusters formed, implying over- or under-segmentation.

5.3 Experiment Result and Analysis

5.3.1 Comparison of Clustering Metrics.

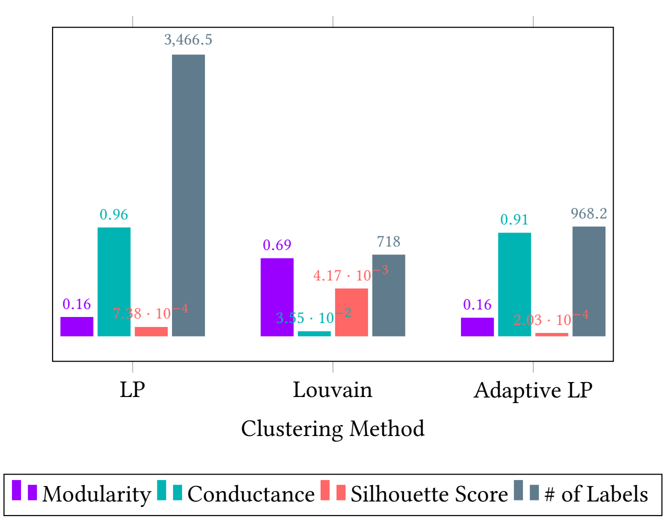


Figure 1: Averaged Clustering Metrics over 10 runs

Figure 1 summarizes 10-run averages for LP, Louvain, and ALP. Louvain yields the highest **modularity** (0.686), indicating dense internal structure, but has the lowest **conductance** (0.035), suggesting isolated clusters.

LP and ALP show higher conductance (0.960, 0.913), reflecting more connected communities. **Silhouette scores** are highest for Louvain, while ALP's lower value reflects its structural-spatial trade-off rather than geometric compactness.

In terms of **cluster count**, LP over-segments (3466.5), Louvain under-segments (718), and ALP finds balance (968.2). This shows ALP's ability to form coherent, moderately granular clusters by adaptively combining structure and spatial cues.

5.3.2 Runtime and Memory Efficiency.

Table 1 reports the average runtime and memory usage over 10 runs. All methods were executed on the same CPU environment.

Method	Runtime (s)	Memory (MB)
Label Propagation	306.0	29.3
Louvain	184.6	95.1
Adaptive Label Propagation	553.5	29.4

Table 1: Average Runtime and Memory usage over 10 runs

ALP incurs higher runtime due to dynamic recomputation of hybrid similarities and entropy-based weights at each iteration. In contrast, LP and Louvain use static or modularity-based updates.

However, ALP remains memory-efficient (29.4MB), comparable to LP and far lighter than Louvain, which maintains hierarchical structures. This trade-off makes ALP suitable for scenarios requiring low memory and moderate runtime tolerance.

5.3.3 Evolution of Label Count and Average α over Iterations.

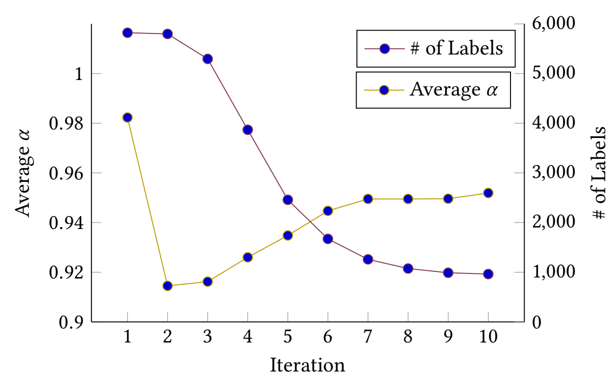


Figure 2: Evolution of Number of Labels and Average α

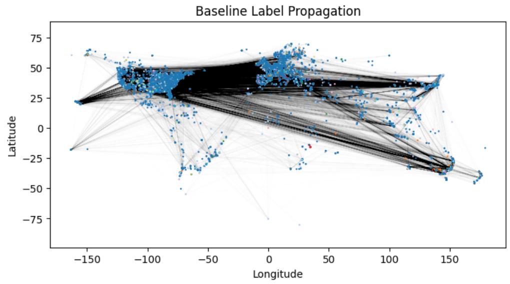
Figure 2 shows how the number of clusters and average α evolve over iterations. Initially, each node has a unique label (5,822 total), but rapid convergence reduces this to around 963 by iteration 10.

The average α starts high, drops in early iterations (0.91), and then gradually increases. This indicates that the model initially relies more on spatial cues to resolve structural uncertainty, then shifts back to structure as clusters stabilize. This dynamic reflects the effectiveness of our entropy-guided weighting in balancing spatial and topological signals.

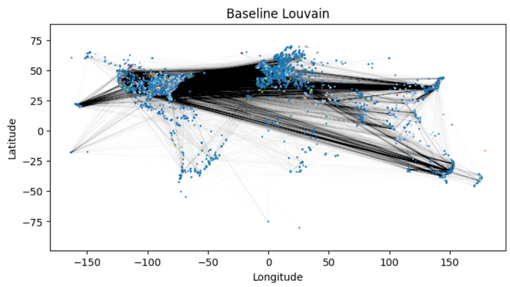
5.3.4 Visualization.

We visualize clustering outcomes based on user check-in coordinates to assess spatial coherence.

Baselines — Label Propagation and Louvain Figure 3 shows that both LP and Louvain, relying only on structural links, produce spatially fragmented clusters. Users in distant regions are often assigned the same label due to structural shortcuts.



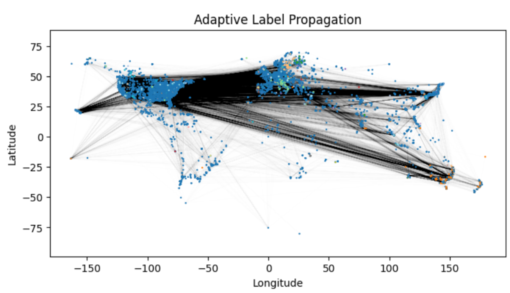
(a) Label Propagation (Worldwide)



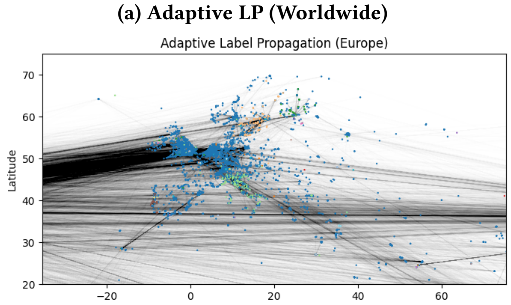
(b) Louvain (Worldwide)

Figure 3: Baseline Clustering Results

Adaptive Label Propagation In contrast, Figure 4 shows Adaptive LP yields spatially aligned clusters by using recent location data.



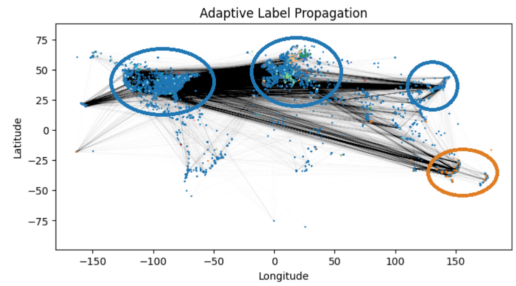
(a) Adaptive LP (Worldwide)



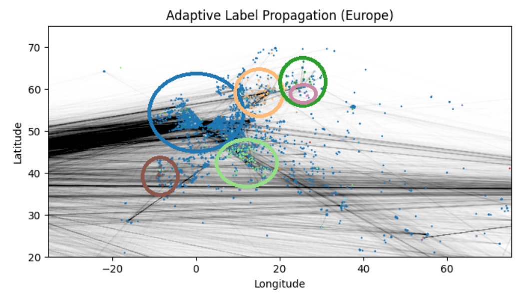
(b) Adaptive LP (Europe)

Figure 4: Clustering Results using Adaptive LP

Marked Clusters Figure 5 highlights compact, spatially coherent clusters found by ALP, further confirming its ability to form geographically meaningful communities.



(a) Adaptive LP (Worldwide, Marked)



(b) Adaptive LP (Europe, Marked)

Figure 5: Circled Clusters from Adaptive LP

6 Conclusion

We introduced a novel unsupervised graph clustering algorithm that extends classical Label Propagation by integrating recent user location data through an entropy-guided weighting scheme. By dynamically balancing topological and spatial similarity without training, labels, or features, our method produces geographically coherent clusters while maintaining competitive modularity and conductance on the Brightkite dataset. The adaptive α prioritizes spatial proximity during early, uncertain iterations and shifts toward structural signals as labels stabilize, making the algorithm lightweight, interpretable, and scalable. Its simplicity and efficiency make it suitable for real-time, resource-constrained applications such as urban mobility analysis, emergency response, and dynamic location-aware systems. Future work includes integrating temporal trajectory data, exploring alternative uncertainty quantifiers, and extending the method to dynamic and evolving social graphs.

References

[1] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, 10 (2008), P10008.

[2] Sheng Li and Yisen Wang. 2022. Structural entropy guided graph hierarchical pooling. *arXiv preprint arXiv:2206.13510* (2022).

[3] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. 2007. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E* 76, 3 (2007), 036106.

[4] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S Yu. 2021. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems* 32, 1 (2021), 4–24.