# Towards a Predictive Model of Individual Belief Dynamics

Jianghui Li (jli159@syr.edu), Joshua Introne (jeintron@syr.edu)

Syracuse University
School of Information Studies

https://c4-lab.github.io

## Introduction

Although the related bodies of research on persuasion, attitude change, and opinion dynamics are well explored, we lack compelling data-driven theories that can predict how belief are likely to change "in the wild." Social media has offered an unprecedented window into the dynamics of collective opinion formation and belief change and may help us to generate such a theory. In this work, we track the dynamics of peoples' expressed beliefs across three different topic areas and show that we can use this information to build a general predictive model of individual belief dynamics. We find that the degree to which people express "conventional" beliefs is predictive of their overall dynamicity.

## Method

Process Data → Construct Belief Landscapes

Identify Belief Clusters → Identify Belief Attractors

Trace Users' Belief Trajectories → Classify Users by Type of Dynamism

Build Features on Cluster Topology → Predicting Belief Dynamics

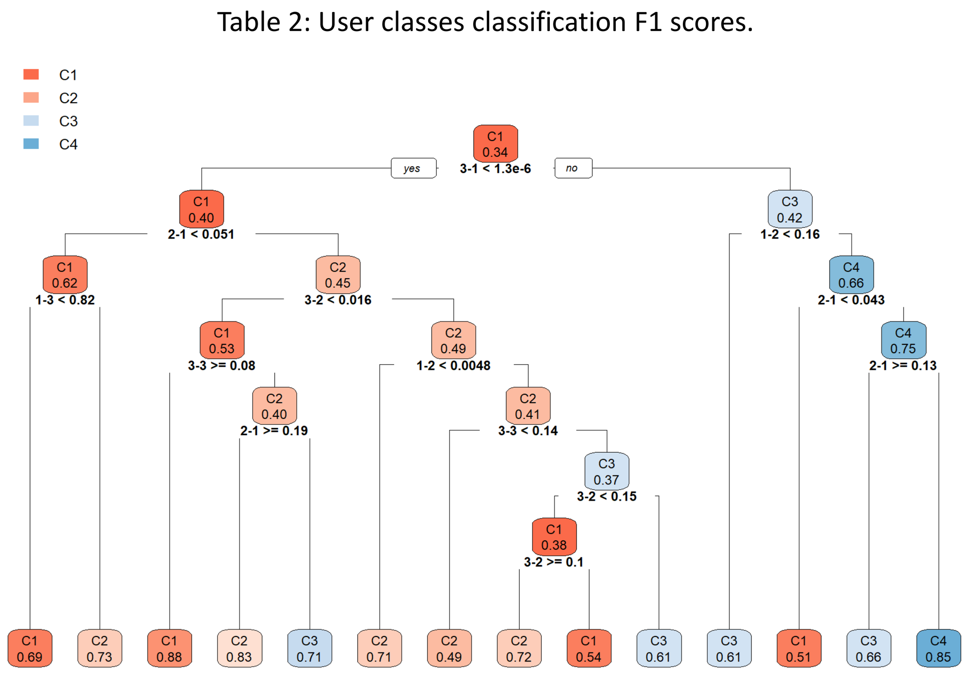
## Predicting User Classes

| User Class | AI (Mastodon) | Climate Change (Twitter) | K-pop & BLM (Twitter) |
|---|---|---|---|
| Class 1 (Stable) | 400 | 3659 | 796 |
| Class 2 (Changer) | 86 | 140 | 76 |
| Class 3 (Bistable) | 82 | 386 | 774 |
| Class 4 (Dynamic) | 36 | 5 | 983 |

Table 1: Summary of user count for each class. Stable user (never changed home location), changer (changed from one home to another), bistable user (changing between two homes), and dynamic user (appeared in more than two homes).

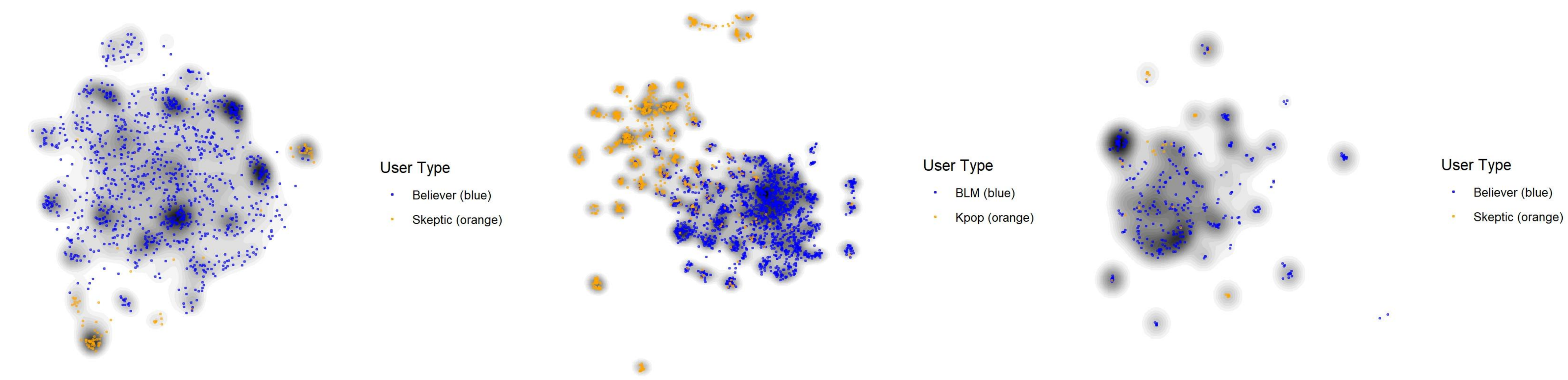
| Classifier | User Class | AI F1 Scores | Climate Change F1 Scores | K-pop & BLM F1 Scores |
|---|---|---|---|---|
| Decision Tree | Stable | 0.66 | 0.70 | 0.68 |
| | Changer | 0.68 | 0.81 | 0.75 |
| | Bistable | 0.76 | 0.77 | 0.55 |
| | Dynamic | 0.87 | 0.92 | 0.69 |
| Random Forest | Stable | **0.96** | **0.92** | 0.93 |
| | Changer | 0.97 | **0.97** | 0.95 |
| | Bistable | **0.99** | **0.94** | 0.90 |
| | Dynamic | **0.99** | **0.99** | 0.93 |
| K-Nearest Neighbors | Stable | 0.94 | 0.89 | **0.95** |
| | Changer | **0.97** | 0.96 | **0.97** |
| | Bistable | 0.98 | 0.94 | **0.94** |
| | Dynamic | 0.99 | 0.97 | **0.95** |

Table 2: User classes classification F1 scores.

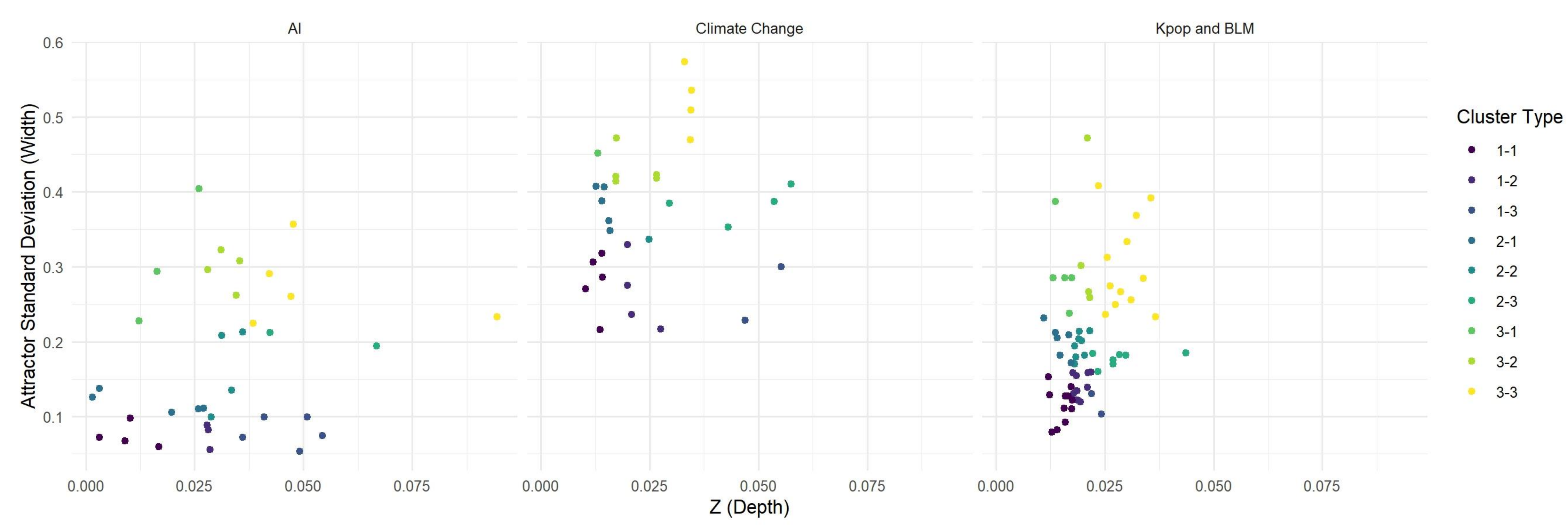


Sample Decision Tree: Each node is labeled as width-depth, with quantiles arranged from low to high, e.g.: Width: small(1), med(2), large(3); Depth: shallow(1), med(2), deep(3). We found that shallow, wide attractors -> increasing dynamism.

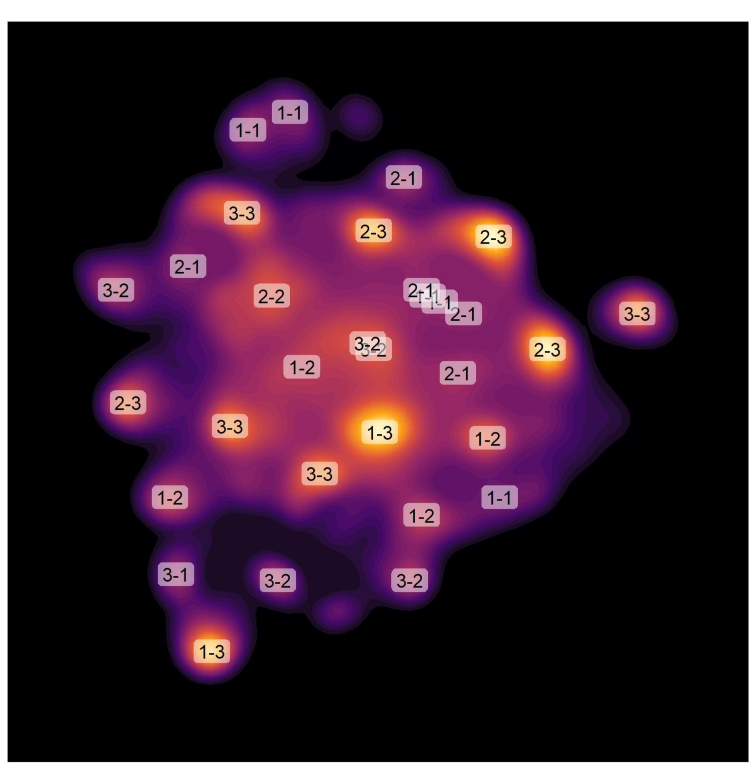
## Belief Landscapes




Steps to constructed belief landscapes include **extract belief-liked statements -> map beliefs into an embedding space -> cluster beliefs into a 2D landscape**. Three belief landscapes are shown. Climate change landscape (left). The dots on the landscape represent a user at a time (only 1% are shown). K-pop and BLM landscape (middle) with 1% data. AI belief landscape (right) with 5% data. The landscape contours are visualized using 2D kernel density estimation.

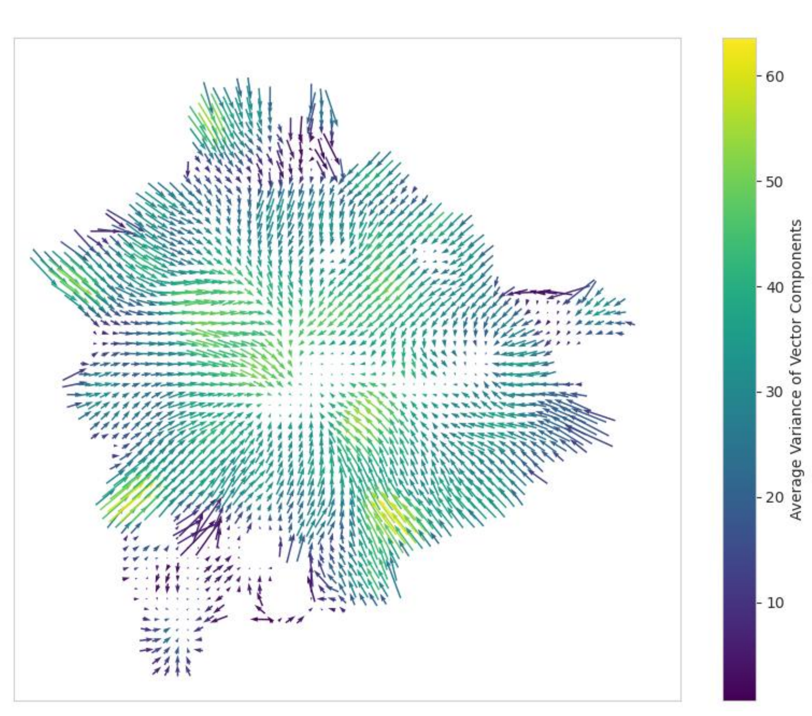


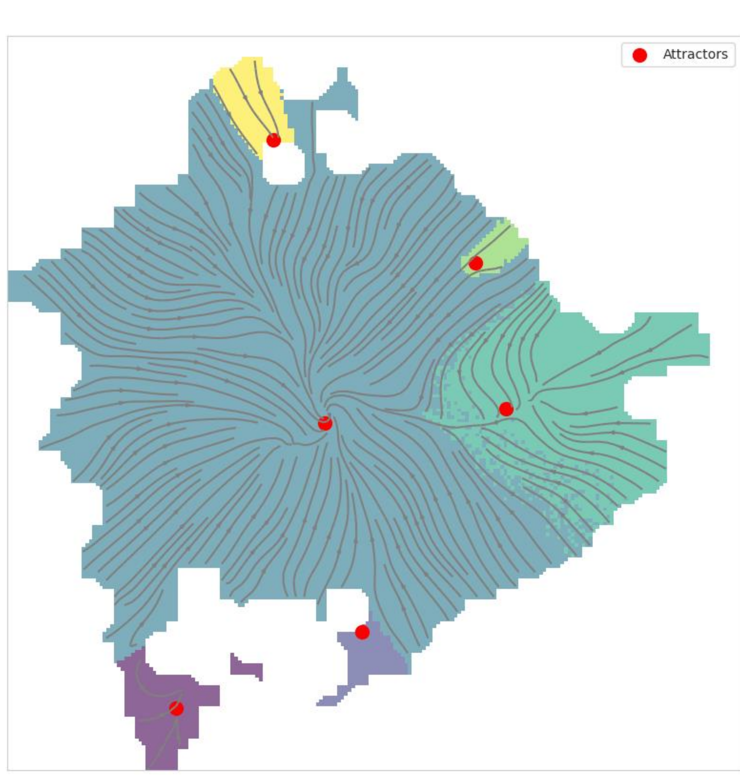

We perform kernel density estimation to obtain centers of the belief clusters, then assign data points to the nearest cluster centers. We then assign belief clusters into classes based on their width (Standard deviation of observed Euclidean distances from cluster center) and depth (Z-score from Kernel density estimation). Discretize width and depth into three classes each (9 classes in total) defined by dual quantiles (3 quantiles for each; '1' for low, '2' for medium, and '3' for high) of their width (number to the left) and depth (number to the right).



This is an example of climate change belief landscape with belief cluster classes.

To get belief attractors, the first step is to build a vector field from movement data across a grid by calculating transitions between coordinates for each user. A Bayesian update method is then employed to calculate the mean vectors(direction and magnitude of movement) and their variances for each grid cell.




The next step is to extract attractors by running walks over the vector field until stopping points are detected using Euler's method. The step size were adjusted to find dominant attractors while screen out the smaller spurious ones. After generating trajectories, the stopping points are clustered using DBSCAN to identify attractors and assign data points to the nearest attractor.

The final step is to smooth the attractor boundaries and fill in holes. This involves morphological operations, contour simplifications, concave hull smoothing, and spline fitting. The result is then mapped back to the belief landscape data.


## Performance as a Function of Posting Volume and Amount of Data Observed