6.UAP: A Collaborative Filtering Mechanism for Individal Movement Through Andorra

Joseph Cunningham *
Faculty Supervisor: Kent Larsen, Principle Research Scientist †
Direct Supervisor: Naichun Chen, Master of Science in Architecture ‡
jmcunnin@mit.edu

May 2016

Abstract

Recommender systems have been used in a variety of applications with the intent of matching users with a preferred product or outcome. Here, we present an application of the concepts of recommender system to the domain of predicting optimal movement patterns. Utilizing cell-tower data records (CDR's) from the nation of Andorra, we present a set of features derived from these data. We the present three methods of combining these features: a naive distance metric, a learned distance metric, and a consensus metric based on feature clustering. We conclude with an analysis of the efficacy of these systems.

1 Introduction

Increasingly, recommender systems have been adapted for use in a variety of academic and industrial applications. Generally, these systems aim to generate meaningful recommendations for a user based on either user-specific or item-specific profile attributes. Recommender systems can be classified as one of three types of systems: collaborative-filtering, content-based, or hybrid systems. A *collaborative-filtering* system infers meaningful recommendations for users by grouping users together based on user histories. Alternatively, a *content-based* system generates recommendations based on

*Depts. of Computer Science and Mathematics, Massachusetts Institute of Technology
†Changing Places Group, Media Lab, Massachusetts Institute of Technology
‡Changing Places Group, Media Lab, Massachusetts Institute of Technology

grouping together content by specific features. Finally, a *hybrid* model combines both collaborative-filtering and content-based systems metrics to generate recommendations. [5]

Recently, the Andorran government has approached the Changing Places group in the Media Lab to better utilize the information at their disposal. Andorra is a small (approx. 468 km^2), mountainous nation in Central Europe located between France and Spain [4]. The economy of Andorra is primarily centered around tourism, retail, and finance, with approximately three-quarters of the nation's $3.163B economy coming from these three sectors [4]. To this end, here we present a recommender system derived from cell-tower data records (CDR), recorded across Andorra, as seen in Figure 1. Specifically, we generate a hybrid recommender system with the goal of generating meaningful recommendations for a preferred next location.

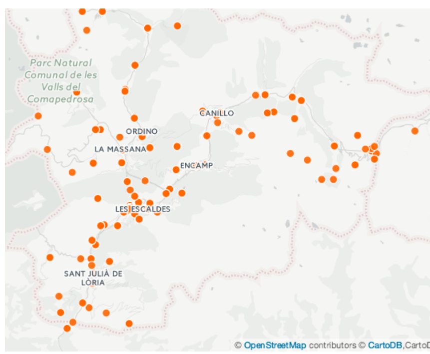


Figure 1: The Cell-Towers in Andorra

In the remainder of this report, we first discuss the information available from the CDR dataset. Next, we discuss the features and implementations of the separate recommender systems. Finally, we conclude with a discussion of the results of comparing the various recommender systems and future work.

2 Problem

As described above, the data available for this project is derived from the cell-tower data records (CDR) from Andorra. These data are broken into groups as follows:

User Information	identifier (sender/receiver), nationality (sender)
Call Length	start time, end time
Location	cell tower (sender and receiver)
Device Identification	IMEI, IMSI, and TAC Codes [1]

These data, however, are limited with respect to previous research in recommender systems. Specifically, many papers rely on data sets that have features such as specific geographic traces of individual movement or specific user response matrices [7–9]. Our data, however, provide limited ability to develop such individual user traces. Further, there is little information to develop a meaningful link between users in our case and sentiment expressed in other media. Our data, however, does provide a basis for developing inferences of these specific characteristics utilized in other datasets. To this end, we present three different approaches to combing our data in the following section.

3 Methods

As described above, our recommender system is derived from a non-traditional set of data. As such, we first must define the set of interactions that we provide recommendations on. Next, we will define the set of weak classifiers we have generated from the above dataset in Section 3.1. Further, we define the clustering method that we use to generate groups of similar regions for recommendations in Section 3.2. Finally, we will conclude with the three methods we use to combine the different sets of data in Section 3.3.

Traditionally, recommender systems interact with an event structured between a user and a product. Specifically, geographic recommender systems are oriented around tracking specific users. A recommender system that falls within the collaborative filtering model might generate a user's GPS trace to understand preference patterns. Similarly, a system that follows a content-based approach would generate a sentiment-based approach to understanding locations. Deconstructing these patterns, we roughly break the problem of recommendation into two sources of information: context-dependent and user-dependent sources of information. Simply, any content-based recommender system is attempting to understand the current context of a user. Similarly, any collaborative-filtering system is simply trying to understand similar users to the current user.

[1] These codes are universal codes assigned by manufacturers identifying their products. Of particular interest are the TAC codes which identify the type of phone the user has.

Based on this deconstruction, we must define exactly what an interaction is within the context of recommendations. Here, we define an interaction to be uniquely defined by three features: a *person*, a *time*, and a *place*. Therefore, the analog of context-dependent features is a characterization of the cell-tower regions provided in the dataset. Similarly, the analog of user-dependent sources of information are simply the ways that users interact with different cell-tower regions throughout Andorra. Using this definition of an interaction, we now present the set of features derived from the dataset based on these data source types.

3.1 Features

As described above, any set of features can be broken into either context-dependent or user-dependent features. Here, we develop four features which can each be characterized by a pair-wise distance metric comparing regions.

3.1.1 Context-Dependent

Context-dependent features depend on characterizing cell-tower regions based on the overall composition of the region. Here we present two such features: the nationality of a user and the phone cost of a user. The nationality of a user provides an understanding of the preferences of tourists of different nationality. Alternatively, the phone cost metric provides an inference on the relative wealth of the region.

Nationality: From the data, we derive a length 193 vector representing the count of individuals from each country of origin for each region. Given these vectors for two regions, A and B, we define the distance between the two regions as:

$$\text{dist}(A, B) = ||A - B||$$

where $A - B$ is the difference between the vectors and $||\quad||$ denotes the vector norm.

Phone Cost: As an analog of wealth, we derive the cost of the phone of a user of the region from the TAC data. Given the phone cost of an individual i, $c(i)$, the distance between two regions A and B is defined as follows:

$$\text{dist}(A, B) = |\frac{1}{|A|}(\sum_{i \in A} c(i)) - \frac{1}{|B|}(\sum_{j \in B} c(j))|$$

3.1.2 User-Dependent

The second set of features is based on understanding how a user interacts with the different regions across Andorra. From the data, we derive two such features related to the connectivity of the individual: where the individual moves during a given day and the regions that the individual calls during the course of a day. These metrics provide a way of connecting different regions based on users that share a common preference for these regions.

Movement: We define the movement feature of two regions, A and B, as the Jaccard index of the two regions:

$$\text{dist}(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Connectivity: Similar to our definition of Nationality, we develop the connectivity as a vector of regions called, and define the distance as the normalized difference of the vectors.

$$\text{dist}(A, B) = ||A - B||$$

3.2 Clustering

Because the distance metrics we use are not-necessarily Euclidean, we are restricted to pairwise clustering functions. For this purpose, we choose to use the Unweighted Pair Group Method with Arithmetic Mean (UPGMA) algorithm. This algorithm agglomeratively clusters the input towers by combining two clusters, A and B, that, given a distance function $d(x, y)$, minimize the function:

$$f(A, B) = \frac{1}{|A||B|} \sum_{x \in A} \sum_{y \in B} d(x, y)$$

Generally, this function takes a base-up approach to clustering that, initially, considers all of the elements as clusters and iteratively joins the clusters with the smallest average distance between them.

3.2.1 Clustering Characterization

Generally, UPGMA is utilized to generate a dendrogram for applications that wish to study the historical relationship between a set of clusters. This approach, however, is sub-optimal for generating meaningful recommendations. Therefore, we cluster dependant on optimizing the following objective function:

$$\text{minimize: } \mathrm{CHAR}(C) = \frac{1}{|C|}\Big(\sum_{C_i \in C} \frac{1}{|C_i|} \sum_{x,y \in C_i} d(x,y)\Big) + \frac{|C|}{\sum\limits_{C_i \in C} |C_i|} \tag{1}$$

This function sums the average intra-cluster distance and the general number of clusters and aims to optimize these two features. We optimize this objective function utilizing Monte Carlo techniques with the following algorithm.

Algorithm 1 Optimal Cluster Finding

procedure FindClustering(D)
    Bounds = 1000 Random Avg. Distance Bounds
    best, best_char = $None, None$
    for bound **in** Bounds **do**
        cl = CLUSTER($D, bound$)
        if CHAR(cl) < best_char **then**
            best, best_char = cl, CHAR(cl)
    return best

3.3 Distance Metrics

The above features provide specific pairwise distance function that relate different cell-tower locales across Andorra based on a specific characterization. These characterizations, however, measure different characteristics. Therefore, we must derive a means to combine these different metrics for a final clustering. We present here three such techniques for generating a final clustering: a naive-distance metric, a trained distance metric, and a consensus method for combining pre-clustered information.

3.3.1 Naive Metric

As a first approach to providing recommendations, we first generate a distance matrix, D, that is simply the average distance generated by each distance metric. First, consider the four distance features generated by each feature where $M(i,j)$ represents the movement distance between regions i and j and $N(i,j), P(i,j), C(i,j)$, similarly, represent the inter-regional distance for nationality, phone cost, and connectivity, respectively. Then, we define the distance matrix D as follows:

$$D_{i,j} = \frac{1}{4}(M(i,j) + N(i,j) + P(i,j) + C(i,j))$$

Utilizing this distance matrix as input, we cluster as in Section 3.2. Note, the matrix D is symmetric as each individual distance metric is symmetric.

3.3.2 Trained Distance Metric

Next, we pose the problem of defining a new distance function as an optimization problem. Specifically, we construct the distance matrix as the following function, where $M(i,j), N(i,j), P(i,j)$, and $C(i,j)$ are defined as in the previous section:

$$D_{i,j} = \frac{1}{(\alpha+\beta+\gamma+\delta)}(\alpha M(i,j) + \beta N(i,j) + \gamma P(i,j) + \delta C(i,j))$$

We restrict $\alpha, \beta, \gamma, \delta$ to the interval $[0,1)$. From this distance function, we then solve for variable α, β, γ, and δ using a hill-climbing algorithm to optimize each parameter. Generally, we randomly generate the four variables, and then optimize for each variable iteratively until our characteristic function for clusterings converges. This algorithm is described in the following algorithm.

Algorithm 2 Hill-Climbing Algorithm

procedure FindTrainedBound(M,N,P,C)
    $\alpha, \beta, \gamma, \delta = rand(0,1)$
    $D_{init} = \frac{1}{\alpha+\beta+\gamma+\delta}((\alpha M + \beta N + \gamma P + \delta C)$
    best, best_char = FINDCLUSTERING(D_{init}), CHAR($best$)
    $\Delta_{char} = \infty$
    while $\Delta_{char} > .05$ **do**
        old_char = best_char
        for param **in** $[\alpha, \beta, \gamma, \delta]$ **do**
            new = FINDCLUSTERING(D_{new})
            ▷ D_{new} generated randomly with new param
            if best_char > CHAR(new) **then**
                best_char, best = CHAR(new), new
        Δ_{char} = |old_char − best_char|
    return best

3.3.3 Consensus Partitions

The third distance metric utilizes a slightly different approach than the previous two techniques. Specifically, the previous two approaches generate clusters directly from the features described in Section 3.1. A separate technique for combining these different feature would be to perform an initial set of clusterings by feature. The method of combining different partitions of a dataset has taken many different forms [1–3]. To generate a clustering based on previous clustering, we must develop a consensus function that combines the different clusterings into one. Here, we utilize the Cluster-based Similarity Partitioning Algorithm (CSPA). This algorithm makes use of clustering algorithms twice.

First, we cluster based on each individual features. Next, we generate a similarity for a clustering based on a feature, F, we develop a similarity matrix, $S(F)$:

$$S(F)_{i,j} \begin{cases} 1 & \text{if } \exists F_i \in F \text{ s.t. } x, y \in F_i, \\ 0 & \text{if } \forall F_i \in F \text{ s.t. } x, y \notin F_i, \end{cases}$$

Then, for each feature, we generate a combined similarity matrix: $S(M)+S(N)+S(P)+S(C)$. Next, we take the inverse of each member of the matrix to find the total dissimilarity (and set it equal to one if there is not similarity). Finally, we again cluster based on this algorithm using the same clustering algorithm as in Section 3.2.

4 Experiments

4.1 Initial Comparison of Metrics

As an initial step in the characterization of our data, we cluster each individual distance metric using the clustering technique described in Section 3.2. We then compare the different clusterings by taking the normalized size of the largest intersection over a clustering. Specifically, consider the clustering, A. We define the similarity between the clustering A and a second clustering B as the function:

$$\text{CLUST\_SIM}(A, B) = \frac{1}{|A|} \sum_{A_i \in A} \frac{\sup_{B_j \in B} |A_i \cap B_j|}{|A_i|} \tag{2}$$

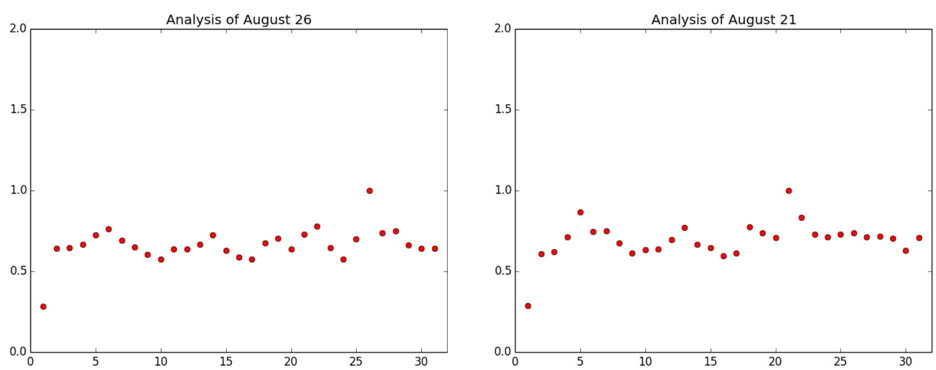
Note, we will utilize this function as a comparison of different clusterings again. Note, this distance function is normalized by the size of the first clustering, and, therefore, is not symmetric. The results of our initial clustering compared to each other clustering is as follows.

	Jacc.	Conn.	Nat.	Phone
Jacc.	-	0.645 ± 0.244	0.439 ± 0.245	0.427 ± 0.238
Conn.	0.692 ± 0.210	-	0.423 ± 0.239	0.396 ± 0.220
Nat.	0.482 ± 0.256	0.425 ± 0.229	-	0.475 ± 0.216
Phone	0.384 ± 0.216	0.363 ± 0.192	0.442 ± 0.254	-

These results provide insight into the different features. Specifically, it seems that there is some over between features, but each feature does contain some uniqueness. The features with the largest overlap seem to be the Jaccard and Connectivity features which do detail similar features. From these similarities, we now move to an analysis of each combination technique over the course of a month.

4.2 Periodicity

The first set of experiments run on the clusters developed from the three different distance metrics is a pairwise comparison of the different clusters using the same comparison metric show in Equation 2 above. With respect to the naive distance metric, comparing each individual day's clusters with all of the other clusters presents an interesting pattern. Specifically, each daily clustering shows a periodic character across the length of the month.

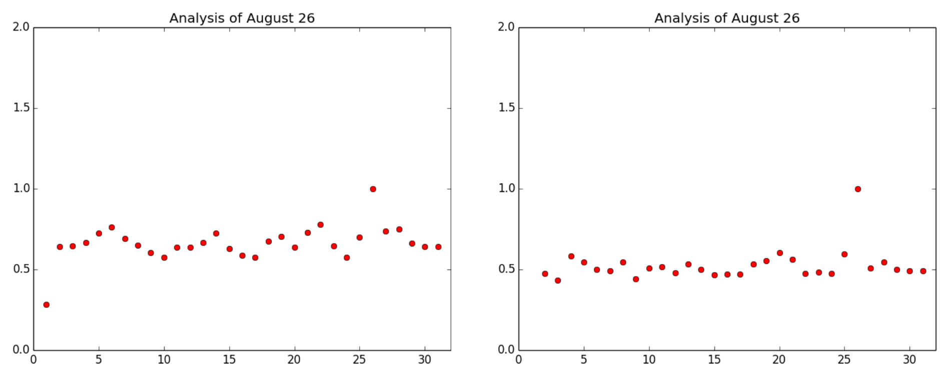


(a) A Weekday In August   (b) A Weekend In August

Figure 2: The Naive Distance Metric

Now, clearly these figures indicate that the naive metric is incorporating some value that is significantly altering the day-to-day patterns that are affecting the composition of the metric as a whole. The question then becomes whether these patterns are truly important in the overall patterns of users moving throughout Andorra. Here, we turn to the other two metrics to understand whether these periodic patterns are truly important differences.

First, we consider the case of the trained distance metric. Structurally, this metric is similar to the naive metric in that it is the weighted sum of the individual features. This feature differs in that we learn the weighting of each individual feature. A side-by-side comparison of these two distance metrics, as seen in Figure 3, provides a very different picture from the original comparison. Although there still seems to be some trends in the trained distance metrics, the total month comparison is much smoother overall. Another notable difference is that there seems to be a weaker agreement between clusters in the trained metric versus the untrained metric.



(a) Naive Metric

(b) The Trained Metric

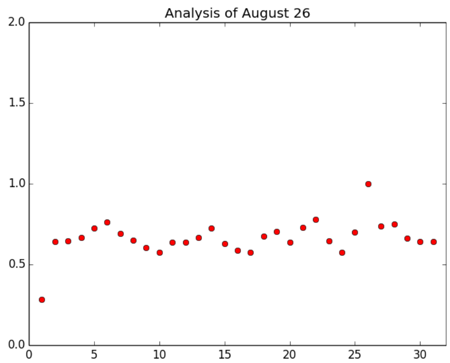
Figure 3: Comparison of the Naive and Trained Distance Metrics

Next, we turn to a similar comparison between the naive comparison and the consensus model. Similarly, there are notable differences between the naive and consensus models, as seen in Figure 4. Although the consensus metric does seem to generate specific trends across the data, the trends do not seem to be as periodic as the trends in the naive metric.

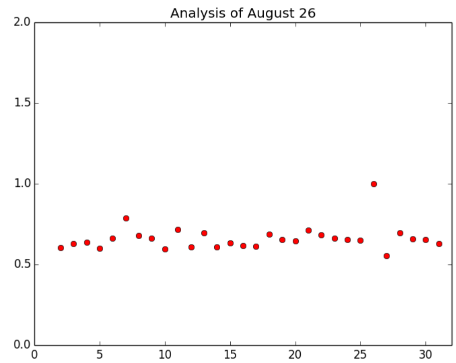
As a measure of the consistency of these trends, the following table provides the mean distance and standard deviation between the day and the rest of the days.

Naive	Trained	Consensus
0.7186 ± 0.0738	0.4671 ± 0.1012	0.6587 ± 0.0717

The above measures of mean and standard deviation of each metric provides



(a) Naive Metric



(b) The Consensus Metric

Figure 4: Comparison of the Naive and Trained Distance Metrics

some insight into the exact spread of the given set of data. Specifically, the similarity between different clusterings is, generally, as observed in the images. The naive metric provides the most similarity between each day, and the trained data set provides the least similarity between days. This implies that the trained data set provides a more selective measure of the features that are important as opposed to unimportant. Further, as could be expected, the consensus metric provides some intermediate between these two measures of clustering. Next, we move to a direct comparison between the classes of clusterings.

4.3 Comparison of Methods

As a final benchmark of the relative performance of our features, we next consider a direct comparison between the different clusterings we have generated above. Specifically, using the same clustering comparison algorithm, we generate a pairwise comparison by day utilizing Equation 2 and then generate the statistics below.

	Naive	Consensus	Trained
Naive	-	0.5300 ± 0.0934	0.5198 ± 0.0958
Consensus	0.4627 ± 0.0907	-	0.6064 ± 0.0880
Trained	0.3595 ± 0.0808	0.4113 ± 0.1131	-

These statistics provide interesting insights into the similarities between the similarities and differences in measures. Specifically, it seems that the trained and naive recommendation systems are providing very different clusterings of the data. Further, the trained algorithm seems to be the most similar to both of the other recommender system, and so seems to be splitting the difference.

4.4 Applying to Recommendations

Having compared the respective methods, it is now time to actually apply the methods that we have generated to the problem of recommendation itself. Recommenders, generally, are hard systems to benchmark as they often involve a high-degree of sentiment [6]. Here, we aim to generate a meaningful understanding of our system based on the following metric.

Generally, we aim to understand how well our system predicts movement in the regions. Therefore, we present the following measure of precision, as described by Bao, et.al. [7].

Specifically, consider the set of movements, X, as the set of changes between any two regions. Define the matrix, M, by setting each element $M_{i,j}$ to be the sum of movements in X where one of the endpoints is i and one is j. The precision of our metric on a clustering C, $\text{PRECISION}(X, M, C)$, is defined as:

$$\text{PRECISION}(X, M, C) = \frac{1}{|X|} \sum_{C_i \in C} \sum_{i \in C_i} \sum_{j \in C_i, j \neq i} M_{i,j}$$

The precision of our function will provide an approximation of whether or not we are understanding the underlying patterns present in the data we are modeling.

4.4.1 Comparison Against Different Months

Next, we compare the system derived from the months of August against four separate months: July 2014, October 2014, and June 2015. In particular, the following figures represent the average precision of the system per day with the standard deviation.

Month	Naive	Concurrence	Trained
July 2014	0.1101 ± 0.0476	0.2008 ± 0.0604	0.2113 ± 0.0520
Oct. 2014	0.1031 ± 0.0411	0.2132 ± 0.0636	0.2102 ± 0.0378
June 2015	0.0688 ± 0.0326	0.1444 ± 0.0540	0.1392 ± 0.0392

While our precision is certainly not perfect, we note two definite results of the figure. First, all three of the given systems are performing much better than a random choice across the months (given that there are ≈ 140 towers, a randomly chosen tower would have $\approx \frac{1}{140}$ chance of correctly choosing the next tower). Further, given the particularly stringent choice of precision metric, our system is performing very well. In particular, the concurrence and trained systems, which generally double the efficacy of the naive system, are correctly predicting where an individual will move next about twenty percent of the time. These results are equivalent to the results cited in Bao et.al. of a general collaborative filtering system that has information well-suited for recommendation [7]. Another significant result of our work is that the consensus and trained metrics are relatively similar in predictive accuracy. Returning to Section 4.3, these metrics have approximately 50% agreement. Further analysis of these differences requires understanding the structure of the clusters resulting from these different clusters.

4.4.2 Comparison of the Different Metrics

Given the above performance, we now turn to an inspection of the performance of the different metrics. As a first pass, a visualization of the naive metric provides a clustering that divides Andorra as in Figure **??**. Note, this mapping only maps the three largest clusters, there are also multiple smaller clusters in the clustering. Further, each color is mapped on an intensity scale mapping how many of the towers in the region are in the cluster.

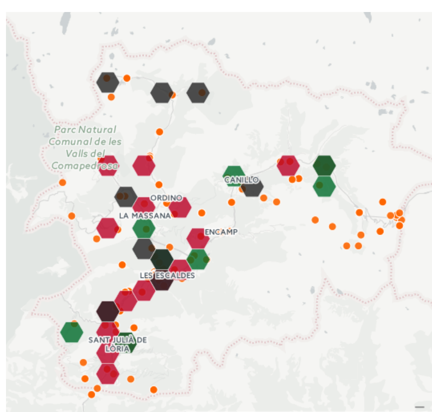
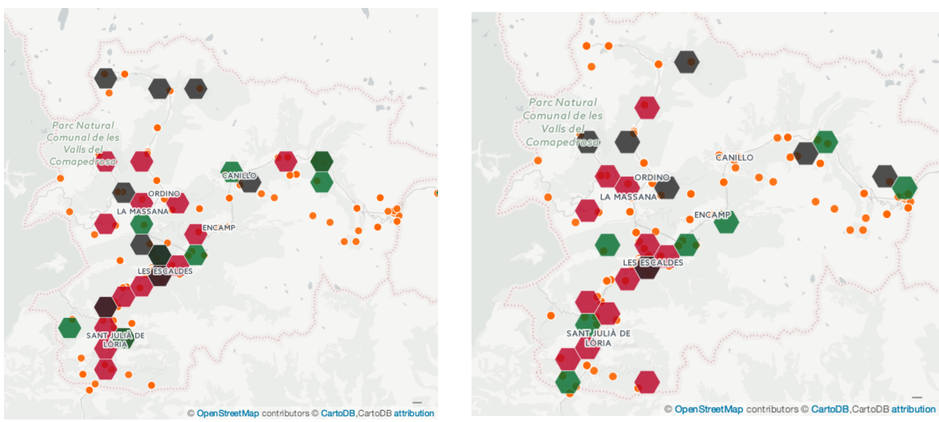


Figure 5: A Clustering of August 18 with the Naive Algorithm

This figure provides an initial idea of the relative size of the different clusterings. Specifically, there exists one large clustering that connects much of the urban center of Andorra. Further, there exist two smaller clusters that are interspersed throughout Andorra. Of note, many of the naive clusterings have a much more defined partition of the different regions present.

The above figure, however, represents the algorithm that proves least effective in predicting the next step. Next, we discuss a comparison between the trained algorithm and the untrained algorithm. Recall, the trained algorithm is similar structurally to the naive algorithm with the largest difference being that we train the algorithm to learn the correct weightings of the features. A comparison of the clusterings on August 18 are provided in Figure 6



(a) Naive Metric on August 18

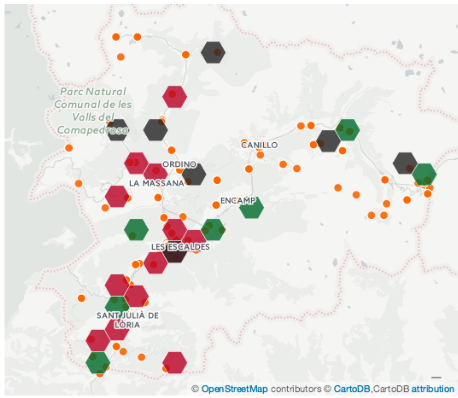
(b) Trained Metric on August 18

Figure 6: Clustering w/ Naive and Trained Distance Metrics

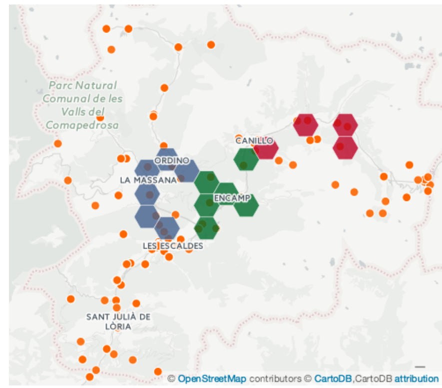
From this figure, we see that the trained distance metric, as might be expected, follows similar patterns to the naive distance metric. Interestingly, the re-weighting of the distance matrices leads to a less geographically imposed separation of the different regions. After the analysis above, however, this seems to imply that the above clustering provides a better overall understanding of the movement patterns than are present in the original clustering.

Next, we move to the clustering provided by the consensus approach. As noted above, this method provides similar predictive power to the trained distance metric. A visual comparison is provided in Figure 7.

The consensus metric seemingly provides a significantly different pattern in clustering. In particular, the consensus metric picks up a specifically regional characteristic that is not necessarily mirrored in the results of the



(a) Trained Metric on August 18



(b) Consensus Metric on August 18

Figure 7: Clustering w/ Trained and Consensus Distance Metrics

trained or naive metrics. Therefore, the similarity in results between the consensus and trained metrics provides interesting opportunities for combinations. Recall, we only plot the three largest clusterings available.

5 Discussion

From above, the different techniques that we have developed present an several interesting differences. This section will be restricted to a discussion of the trained and consensus metrics, as the trained metric is largely a completely superior metric to the naive metric.

In comparison, the trained and consensus metrics seem to be accessing very different underlying patterns to make predictions. Specifically, the consensus metric is accessing patterns that are fundamentally related to specific regional characteristics. In particular, the characteristics are most noticeable in specific regions surrounding cities and are less so the further one travels from an urban area. Conversely, the trained metric is accessing categories that are not as related to specific geographic ideas, but rather a learned understanding of the different metrics available to the individual. Notably, although these metrics are seemingly accessing very different patterns in the underlying features, the metrics achieve similar predictive accuracy on the next step for an individual in a region. As described below, future work should involve seeking to reach a fuller understanding of how these different metrics interact, and how these metrics can be combined to improve predictive accuracy.

6 Future Work

Future work on the recommender system largely falls into the category of developing further features to characterize the regions. Of note, we discuss a few of the possible features/types of features that would benefit the system. Additionally, we include a description of further work needed for further analysis of the system.

Sentiment: Much work in collaborative-filtering systems is dependent on characterizing the sentiments of users in relation to specific items. In the context of our problem, this is very difficult in that linking the identifying information provided in the CDR records to any individual would be somewhat of a difficult task. One way of circumventing this problem, however, would be to mine a medium like Twitter for sentiment in relation to specific locations. This is limited in application both by information sparsity (number of tweets) and the connection between different regions. A model that describes this phenomena, however, would be powerful in improving the user-dependant features of the recommender system.

Area Characterization: A further improvement of the recommender system would integrate characterizations of the area based on the different businesses/attractions in the area. Similarly, a major limitation to this characterization is relating the impact of attractions to each other and to individuals. For example, consider an individual that is skiing. An inference on their movement might assume that they would move from skiing to an event like a restaurant. Generalizing this type of interaction would likely require domain-specific knowledge that would possibly not prove robust for general applications.

Analysis: Further work should be carried out to more formally consider the impact of different variations between months. Specifically, the system currently performs relatively well regardless of month. However, a further consideration of how the different months are similar and different would improve the analysis of how to modify the system based on seasons.

Day-to-Day A final piece of work that should be explored is understanding how the different days fit together. Currently, we utilize the different clustering very naively to predict how the patterns seen in different months transfer to new months. A more sophisticated approach might improve the overall predictive capacity by combining the clusterings across months in a way that is informed by different movement patterns based on day and time.

Combining Clusterings: As noted above, the consensus and trained metrics perform relatively similarly, but seem to model different phenomena in the data. Generating a model that combines these data would be an interesting task moving forward.

References

[1] F. Wang, J. Sun, S. Ebadollahi *Integrating Distance Metrics Learned from Multiple Experts and its Application in Patient Similarity Assessment* SIAM-SDM, 2011.

[2] Z. Lu, Y. Peng, J. Xiao *From Comparing Clusterings to Combining Clusterings* AAAI, 2008.

[3] A. Topchy, A. Jain, W. Punch *Clustering Ensembles: Models of Consensus and Weak Partitions* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005.

[4] The CIA World Factbook: Andorra, *The World Factbook*. Web Access, February 11, 2016.

[5] P. Melville, V. Sindhwani *Recommender Systems* IBM T.J. Watson Research Center.

[6] M. Ge, C. Delgado-Battenfeld, D. Jannach *Beyond Accuracy: Evaluating Recommender Systems by Coverage and Serendipity* Recommender Systems-ACM, 2010.

[7] J. Bao, Y. Zheng, M. Mokbel *Location-Based and Preference-Aware Using Sparse Geo-Social Networking Data* ACM-SIGSPATIAL, 2012.

[8] V.W. Zheng, Y. Zheng, Q. Yang *Collaborative Location and Activity Recommendations with GPS History Data* ACM, 2012.

[9] Y. Zheng, L. Zhang, X. Xie, W.Y. Ma *Mining Interesting Locations and Travel Sequences from GPS Trajectories* ACM, 2009.