A Framework for Spatial Feature Selection and Scoping and its Application to Geo-Targeting

Ruth Huang Miller[#1], Chun-Sheng Chen[*2], Christoph F. Eick [*3], Abraham Bagherjeiran[+4‡]

[#]*Department of Mathematics and Computer Science, University of Detroit Mercy*
4001 W. McNichols Road, Detroit, MI 48221 USA
[1]millerru@udmercy.edu

[*]*Department of Computer Science, University of Houston*
4800 Calhoun Rd., Houston, TX 77004 USA
[2]lyons19@cs.uh.edu
[3]ceick@cs.uh.edu

[+]*ThinkersRUS*
Sunnyvale, CA USA
[4]abagher@thinkersr.us

Abstract— **Predicting if a particular user clicks on a particular ad is of critical importance for internet advertising. Associations between Internet ad performance data, such as number of clicks or Click Through Rate, CTR, and demographic data may be very weak on the global level, but strong at the regional level. Identifying regions with strong associations of a continuous performance attribute with geo-features can create valuable knowledge for geo-targeted advertising. In this paper, we present a novel framework for interestingness scoping to identify such regions and discuss how such interestingness hotspots can be used for geo-feature evaluation with the goal to develop more accurate prediction models for advertisers. We also present the ZIPS algorithm that takes initial seed zip codes and discovers interestingness hotspots/coldspots, and a geo-feature pre-selection algorithm which automatically finds promising geo-features and identifies initial seed zipcodes for the ZIPS algorithm. We applied our framework to a large number of geo-spatial data sets, combining data from a major ad network, demographic data from the 2000 Census, and binary feature data from other sources. Our experimental results demonstrate that creating geo-features can double CTR performance for an Ad.**

Keywords—**Spatial Data Mining, Region Discovery, Geo-Feature Selection, Contextual Advertising, Behavioral Targeting**

I. INTRODUCTION

Online advertising has become a very important component of our economy. Online ad revenues totalled $17 billion in 2007, 22 billon in 2008, and $25 billion in 2009 [23], and online advertising is growing significantly in 2010. Current estimates suggest a 20% growth rate in revenues for 2010 [18]. Predicting if a particular user clicks on a particular ad is therefore of critical importance for the success of online advertising companies.

Ad networks typically utilize a pay-per-click model for generating revenues for displayed ads; consequently, more clicks mean more revenue for the ad network. The primary objective of the ad network is to increase the *click through rate* (CTR) which is the ratio of the number of ad clicks to the number of ad impressions. In contrast, the primary objective of the advertiser is to increase the *conversion rate*, which is the ratio of the number of ad conversions to the number of ad impressions, where an ad conversion means the user has taken some action that has benefit for the advertiser and occurs after the user has clicked on the ad.

It has been pointed out in geographical literature that global, whole map statistics seldom provide useful insight and most relationships in spatial datasets are geographically regional, rather than global [13]. Goodchild observed [8] that "*efforts toward a scientific geography … had always assumed … that principles discovered in one part of the world should also be true in other parts. Idiographic geography had been disparaged as descriptive journalism, yielding nothing universal*." In general, regional knowledge has a scope that is the area on a map where this knowledge is valid—if it would be valid on the whole map it would be global knowledge. The Internet advertising community is already aware of the importance of regional knowledge, as it uses geo-targeting, delivering different ads to users based on their location.

The goal of this paper is to utilize regional knowledge in the area of internet advertising. It centers on a particular aspect of this user-ad-click prediction problem: how does the geo-location of a user impact the probability that this user clicks on a particular ad? In general, the paper presents computational frameworks that mine spatial patterns of clicking with the goal to create geo-features. *Geo-features* are a single binary spatial characteristic, or a set of binary spatial characteristics, that characterize a geographic region. The feature data value is set to 1 if the region, such as a zip-code, contains the geo-feature, and 0 otherwise. An automatic feature generation algorithm is presented that selects geo-features for use in finding interestingness hotspots for associations between a geo-feature and the performance attribute; interestingness hotspots are contiguous, spatial regions that show interesting associations of the performance attribute with the geo-features. There are millions of potential geo-features that one can choose from, and therefore pre-

[‡]This work was performed while Abraham Bagherjeiran was working forYahoo! Labs, in Santa Clara, CA.

selection of a relevant geo-feature is an important issue. Consequently, the paper additionally proposes a pre-selection algorithm which goes through a large set of geo-feature candidates to identify promising features for the next stage of the analysis. We will demonstrate that the so-obtained regional knowledge can be utilized to improve geo-targeted advertising in the identified regions.

The main contributions of this paper include: 1) it presents a unique, generic framework to find spatial interesting hotspots for a performance attribute with respect to a set of geo-features, 2) we discuss how performance attributes can be used as a pre-selection tool to identify promising geo-features, 3) novel algorithms are introduced that compute interesting hotspots regions, and 4) the presented framework and algorithms are evaluated in challenging case studies which associate geo-features with clicking behavior with the goal to obtain a more accurate model for Ad targeting.

The rest of this paper is organized as follows. Section 2 discusses related work. Section 3 presents our framework for identifying geographical interestingness hotspots, proposes a new interestingness scoping algorithm, and defines various interestingness functions that capture different aspects of the geo-features and performance attribute. It also presents our ZIPS algorithm for discovering hotspot/coldspot regions, and our geo-feature pre-selection algorithm for pre-seeding ZIPS. Section 4 presents experimental evaluation results. Section 5 presents our conclusions.

II. Related Work

Traditional data mining techniques focus on finding global patterns and statistics. For example, MLC++ [21] is a software package library of machine learning algorithms that can be used to build data mining applications. In addition, there are repositories [22] of machine learning algorithms and tools, and two comprehensive surveys of useful analysis algorithms [19], [20] that provide resources for building data mining tools.

However, global statistics seldom provide useful insight and most relationships in spatial datasets are geographically regional. Our approach, developed by the UH Data Mining and Machine Learning Group [4], centers on discovering regions and interesting associations which are valid in those regions. The framework operates in the continuous domain and views region discovery as a clustering problem in which an externally given fitness function is maximized [13], [14].

However, to date there is little work on using these data mining techniques on multiple spatial data sets to identify interestingness hotspots with respect to a performance attribute that could help identify generalized regional user profiles, which provide valuable input for selecting geo-targeted advertising in those regions.

A. *Hot Spot Discovery in Spatial Statistics*

In [2], the detection of hotspots using a variable resolution approach was investigated in order to minimize the effects of spatial superposition. The definition of hotspots was extended in [12] using circular zones for multiple variables. In [7], [14], a popular method to find hot spots in spatial datasets relying on the G* Statistic was proposed. The G* Statistic detects local pockets of spatial association. The value of G* depends upon an *a priori* given scale of the pockets and is calculated for each object individually. Visualizing the results of G* calculations graphically reveals hotspots/coldspots. However, such aggregates are not formally defined clusters, as the G* approach has no built-in clustering capabilities.

B. *Spatial Co-location Pattern Discovery*

Shekhar *et al*. discuss several interesting approaches to mine co-location patterns, which are subsets of Boolean spatial features whose instances are frequently located together in close proximity [15], [16], [17]. Huang *et al*. proposed co-location mining involving rare events [9]. In [10], Huang and Zhang explored the relations between clustering and co-location mining. Instead of clustering spatial objects, the features of spatial objects are clustered using a proximity function that is designed to find co-locations. However, it should be stressed that all the approaches mentioned above are restricted to categorical datasets and center on finding global co-location patterns, whose scope is the whole dataset. Localized association rule mining [1] takes a similar approach to ours, but it discovers association rules that hold in local clustered basket data, and is limited to non-spatial basket datasets.

III. A Framework for Identifying Geographical Interestingness Hotspots

Identifying geo-spatial location in the US has been made possible by the public availability of IP-address geo-location databases that associate each IP address with a global location, which are highly accurate and updated monthly. While geo-spatial location is a much more difficult problem in other parts of the world, techniques (mostly proprietary) have been developed by the major ad networks to alleviate this problem and significantly improve accuracy. Given that it is feasible, and relatively easy, to identify geo-spatial location of a user (at least to the user's ISP location), the question then becomes one of identifying spatial interestingness with respect to one or more geo-features and a performance attribute/metric.

A. *A Framework for Spatial Interestingness Scoping of a Performance attribute in Different Contexts*

We introduce computational techniques that compute interestingness hotspots with respect to particular associations of a performance attribute with a particular context. Interestingness hotspots are contiguous areas in space for which an interestingness function *i* assigns a reward $w \geq \theta$, indicating "news-worthy" regional associations between the performance attribute and context under which it is analyzed.

Our goal is to mine spatial patterns for a *performance attribute* in different contexts in a predefined space. A spatial pattern as far as this paper is concerned, is an association of a performance attribute with a single or a set of geo-features. The scope of a spatial pattern is a set of contiguous geographical regions for which the association is valid;

validity is assessed using interestingness functions. This is accomplished by estimating quantities of interest for the success of a business entity, such as average click-through rate, number of items sold (related to conversion rate), or average profits (based upon actual income less the actual cost for the advertisement). A context is a presence or absence of a specific geo-feature with respect to performance attributes that are analyzed. For example, we might be interested in finding interesting associations between click-through rate (the performance attribute) and one or more geo-features (the context under which the performance attribute is analyzed) that are assessed by evaluating correlations between them. Interestingness of associations is captured by interestingness functions that associate a reward with a region based on its interestingness.

More formally, we assume a spatial dataset O is given in which objects $o \in O$ are characterized by: the performance attribute p, a set of spatial attributes S, a set of binary attributes B, and a set of continuous attributes C.

$B \cup C$ defines the contexts under which the performance attribute p is analyzed. Moreover, we assume a spatial neighboring relationship N is given

$$N \subseteq O \times O$$

that describes which objects belonging to O are neighbors. N is usually computed using spatial attributes S of objects in O.

Finally, we assume that we have an interestingness measure

$$i: 2^O \rightarrow \{0\} \cup \Re^+$$

that assesses the interestingness of subsets of the objects in O by associating rewards with particular contexts. Moreover, we assume an interestingness threshold θ is given that defines which patterns are interesting. In general, *i* measures the interestingness of the performance attribute p, usually in some context[1].

$$X \subseteq B \cup C.$$

The goal of this research is to develop frameworks and algorithms that find interestingness hotspots $H \subseteq O$; H is an interestingness hotspot with respect to *i* if the following 2 conditions are met:

1. $i(H) \geq \theta$
2. H is contiguous with respect to N; that is, for each pair of objects (o,v) with $o,v \in H$, there has to be a path from o to v that traverses neighboring objects (w.r.t. N) belonging to H.

Interesting hotspots H are contiguous regions in space that are interesting ($i(H) \geq \theta$). Moreover, we call H a *global interestingness hotspot* if $i(O) \geq \theta$; otherwise, we call H a *regional interestingness hotspot* with respect to O.

B. *Interestingness Measures for Geo-feature Data Sets*

Interestingness hotspots are areas for which the interestingness function *i* assigns a reward $w \geq \theta$, indicating "news-worthy" regional associations between the performance attribute and the context under which it is analyzed.

In this section, different interestingness measures *i* will be introduced, some of which will be used later in Section 4 for the experimental results. The most simplistic interestingness measure we can think of is one that directly uses the value of the performance attribute p, which we call i_p in the following:

Let $H \subseteq O$

$i_p(H) = (\Sigma_{h \in H} h.p)/|H|$ where $|H|$ denotes cardinality of H

Another interestingness measure $i_{p|Y}$ analyzes the performance attribute within a set of binary contexts $Y \subseteq B$:

$$i_{p|Y}(H) = \left(\frac{\sum_{h \in H \wedge \forall y \in Y: h.y = true} h.p}{|\{h \mid h \in H \wedge \forall y \in Y : h.y = true\}|} \right)$$

Where i_p takes an average of the performance attribute for all objects in H, $i_{p|Y}$ restricts the average computation to objects in H for which binary attributes (such as geo-features) in Y are true. Moreover, when mining with respect to $i_{p|Y}$ we frequently require an additional support threshold *s*. If this approach is chosen, H is considered to be an interestingness hotspot with respect to $i_{p|Y}$ only if the following conditions are met:

1. $i(H) \geq \theta$
2. H is contiguous with respect to N; that is, for each pair of objects (o,v) with $o,v \in H$, there has to be a path from o to v that traverses neighboring objects (w.r.t. N) belonging to H.
3. $|\{h \mid h \in H \wedge \exists y \in Y \ h.y=true\}| \geq s$

C. *Computing Interestingness Hotspots*

This paper presents two algorithms related to finding Interestingness Hotspots: 1) the ZIPS algorithm for identifying regional hotspots/coldspots, and 2) an automatic feature pre-selection tool that identifies promising geo-features that should be further investigated by ZIPS.

Our algorithm for computing interestingness hotspots is called ZIPS (for Zip code Interestingness Proximity Selection). ZIPS, a generalization of MOSAIC [3] to handle polygon based regions, is an **agglomerative growing algorithm** that starts with a region $R=\{o\}$ containing a single object $o \in O$ and searches for interesting hotspots G_o by adding objects in the neighborhood of o to R.

Two groups of agglomerative growing algorithms can be distinguished:

a. Algorithms that try to obtain interestingness hotspots that maximize the size $|G_o|$ of the obtained region.
b. Algorithms that maximize the interestingness $i(G_o)$ of the obtained region.

A problem that both types of algorithms face is that they create a lot of overlapping interestingness hotspots which need to be post-processed to obtain a final set of hotspots. Post processing algorithms that can be used for this purpose have been proposed in [11].

[1] Occasionally, we just analyze the performance attribute itself, in which case $X=\emptyset$

```
Input:  interestingness features - fList, and dataset

featureSelection (fList):candidateFeatureList
{
   Set SelectedFeatureList := empty
   Set fsFeatureZips := empty
   Set inputFeatureList := fList
   Set candidateFeatureList := empty
   For each item c in inputFeatureList
      add c and –c to the candidateFeatureList
   For (i = 1; i<=candidateFeatureList.length; i++)
      Call createFeatureSetList(candidateFeatureList, 0,
                                 i, new int [i])
}
createFeatureSetList(cFeatureList, n, i, featureSet )
{
   if (i ==0)  {          //end IDDFS search
      For each zipcode z in DataSet {
         If (z contains the match to the features in featureSet )
            Add zipcode data to attributes with featureSet
         Else
            Add zipcode data to attributes without featureSet
      } //end for loop
      Calculate performance attribute and ratios for featureSet
      If ratio met the threshold {
         Add featureSet to selectFeatureList;
         Add zip codes with featureSet to fsSeedZip
      }   //end check threshold
   } // end outer if loop
   Else  //recursively search the space for more feature set {
      For (int k=n; k<= cFeatureList.length-i; k++) {
         //if meet the threshold continue search
         //else the rest of tree is pruned
         If (featureSet's subfeature sets met threshold)
            Call createFeatureSetList(cFeatureList, k+1,
                                      i-1, featureSet)
      } //end for
  } //end else
} //end function call
```

Fig. 1 Zip codes Pre-selection Algorithm.

1) *ZIPS Feature Pre-selection Algorithm:* The feature pre-selection algorithm, shown in Fig. 1, finds the initial seed regions automatically based on interestingness, thus eliminating the need for an expert to manually inspect them. The algorithm takes a set of geo-features as an input and identifies candidate feature sets using Iterative Deepening Depth-First Search (IDDFS). When a candidate feature set is identified, every zipcode's data is examined for the presence and absence of this feature set, and information is updated accordingly. Once all the zipcodes have been examined, the performance attributes are calculated for this feature set. If the interestingness threshold is met, then the feature set is saved into the selectedFeatureList and the zipcodes are saved as seeds for the ZIPS algorithm. A feature is pruned by the algorithm if its subfeatures are not part of the selected feature set. The complexity of the algorithm is $O((2*|features|)^{|features|}$. The seed list is then fed into the ZIPS algorithm. In addition, a Naive Bayesian algorithm could utilize the performance attributes to improve the predictions from the Yahoo! prediction algorithm with the information gained from the use of geo-features.

2) *ZIPS Regional Hotspot Identification Algorithm:* ZIPS, begins with a set of seed regions and grows each region by adding neighboring zip codes to it. The seed regions used as the initial set of regions is defined by the zip codes containing one or more geo-features, whose interestingness is above some threshold. ZIPS, which finds regions based upon zip codes, is an agglomerative clustering algorithm that operates on (zip code) polygon regions. Fig. 1 shows the pseudo code of the ZIPS algorithm. ZIPS computes zip code regions whose interestingness is above a user-defined threshold. Since the finest granularity of CTR and the census data is on the zip code level the algorithm walks through all the neighboring zip codes that share the boundary with the initial region. A neighboring zip code is merged into the initial region if the interestingness after merging is above the threshold. The algorithm stops if no more zip codes can be added. This is different from Kulldorff's algorithm [14], in that ours works with zip code based polygons, rather than growing circular regions from initial points. Also, our framework is more general in that it supports arbitrary interestingness functions.

```
INPUT:  AN INTERESTINGNESS FUNCTION F, a list of n initial zip
        regions zlist, interestingness threshold t
Set HotspotList := empty
Set NeighborList := empty
For each region z in zlist {
    If(F(z)>t) {
        Add (neighbor zip codes of z – Hotspots) and add
            to the NeighborList;
        While (size of NeighborList > 0) {
            Remove one zip code M from NeighborList;
            If (F(M+z) > t){
                Merge M to z;
            }
            Mark M as processed and add unprocessed
            neighbor zip codes of M to the NeighborList ;
        }
        Add z to HotspotList;
    }
}
Return HotspotList;
```

Fig. 2 ZIPS Hotspot Discovery Algorithm.

IV. Experimental Evaluation: Mining Geo-Spatial Patterns of Clicking Behavior

This section describes the results of our offline data mining experiments, which mine for regional and global spatial associations between performance attributes (like number of clicks and CTR) and promising geo-features.

A. Zip Code Regions

In the zip code data, the first digit, 0-9, designates the general area of the country, with numbers starting lower in the east and increasing as you move west. For example 0 covers Maine while 9 refers to California. The next two digits in a zip code refer to one of the 455 Sectional Center Facilities (SCFs) in the US. The last 2 digits identify the zip codes within a specific SCF.

B. Data Sets

Data from multiple sources were combined to obtain the data sets used in the experiments. The first source came from

TABLE 1. GEO-FEATURE IMPACT ON PERFORMANCE ATTRIBUTES

	CTR	Impression	Click
Feature: Starbucks			
Ratio T/F	1.563	2.041	3.189
Feature: Wholefood			
Ratio T/F	1.012	8.033	8.129
Feature: Target			
Ratio T/F	1.539	0.975	1.500
Feature: Walmart			
Ratio T/F	1.758	0.466	0.819
Feature: not high PhD and no Wholefoods			
Ratio T/F	1.725	0.399	0.688

TABLE 2. REGIONAL EFFECTS OF WHOLE FOOD MARKET

Region	% Diff w/o	% Diff w/
Entire US (Global)	-0.16%	1.03%
CA 90xxx-96xxx	-9.84%	46.89%
KY 40xxx-42xxx	1.12%	-30.47%
WA 98xxx	21.63%	-68.11%

Yahoo! click log data for the top 5 Yahoo! domains for rank 1 ad data. For this data, we set a reasonable minimum threshold of 1000 impressions, and 100 clicks. The original dataset

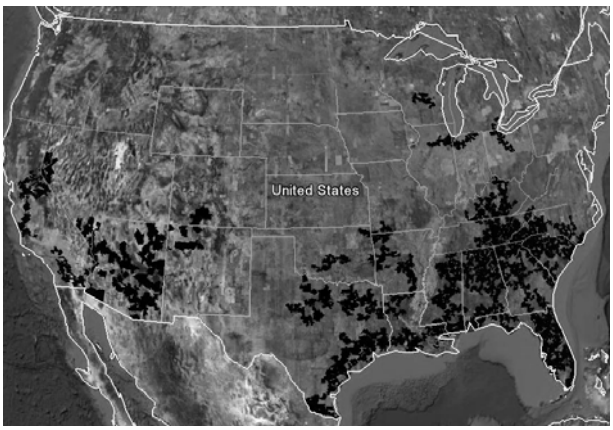

Fig. 3 Hotspot Regions with Starbucks geo-feature as used for ZIPS and minimum CTR threshold set to 4 times the average CTR

contained 38,000 zip codes, but after applying the thresholds the dataset was trimmed to 13,848 zip codes. The second source came from the US Census 2000 data. In this dataset,

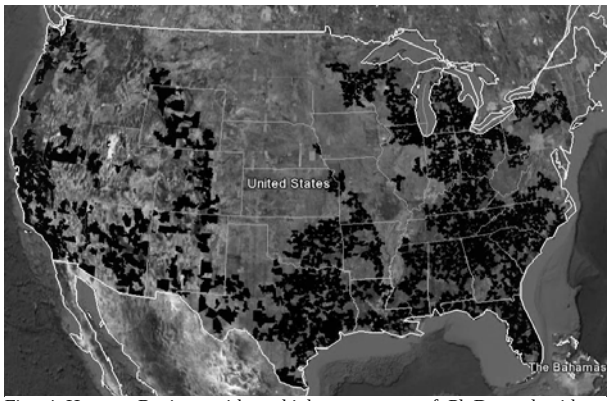

Fig. 4 Hotspot Regions without high percentage of Ph.D., and without Wholefoods market as geo-feature CTR threshold set 2 times the average

the race field contains one of the following: White, Black, American Indian, Asian, Hawaiian and Pacific Islander, Some Other Race, and 2 or More Races. The data contains both the population and percentage of each race in the zip code. The education field can be one of: Upto 12th Grade, Some College, BS, MS, Grad and Professional, and BS and Beyond. The education level data is limited to those who are 25 years of age or older, and is given as a percentage. The per capita income is a number that represents the average per capita income for that zip-code. For the third source, additional geo feature data were used to supplement these two datasets from sources such as: Starbucks.com, WholeFoods.com, Walmart.com, and Target.com. Datasets from this third set of sources were joined using the zip-codes of stores, obtaining a combined dataset that contains records for each of the 5 digit zip codes, with fields including: 1) the performance attributes *number of impressions*, *number of clicks*, and *CTR*, 2) other attributes such as per capita income, race, and education level, and the binary equivalent of these being 25% above the average rate and 3) binary geo-features such as presence of Starbucks, Whole Foods Market, Target, or Walmart stores.

C. *Identifying Interestingness Hotspots*

Interestingness was measured for all the US 5 digit zip codes with respect to performance attributes (*number of impressions*, *number of clicks*, and *CTR*), and geo–features (presence of Starbucks, presence of WholeFoods Market, and presence of airport with many nearby hotels). The zip codes containing the geo–feature that met the minimum interestingness threshold are used to seed ZIPS. Analysis was then performed using our hotspot identification algorithm, ZIPS, with respect to the performance attributes.

1) *Binary Regional Analysis:* Binary features need to be examined with respect to performance attributes, such as CTR, so that the interesting features can be further analyzed using the ZIPS algorithm. In Table 1, the presence of Starbucks, Whole Foods Markets, Target and Walmart binary features were studied against CTR. The values for impression, clicks and CTR are given as ratios of the actual values for those zip codes that have these geo-features to the actual values for those zip codes that do not have the geo-features. We have found some regional effect on both clicks and CTR when these geo-features are present. When a Starbucks or a Target store is located in a zip code, there is corresponding increase in all three performance attributes. This indicates that more users view ads in these zip codes, and are more likely to click on the ads they are presented.

The most interesting cases occur when either a Walmart or a WholeFoods Market is present. When a Walmart is present in a zip code, both #impressions and #clicks declines, but the CTR still increases. This suggests a loss of revenue stream, even though the CTR value increased. However, when a WholeFoods Market is present in a zip code, the CTR remains relatively flat. But, there is a corresponding increase in #impressions and #clicks, both of which grow at the same rate. Thus, there is an increased revenue stream in these zip codes,

even though the CTR is relatively unaffected. This supports the argument that CTR may not always be the most important performance attribute on which to base statistical analysis when regional user behavior.

Regional effect is also demonstrated in Table 2 showing regional effects for WholeFoods Markets. In California, zip-codes without a Whole Foods exhibit a 9.84% drop in CTR, while those with a Whole Foods exhibit a 46.89% increase. Washington State, on the other hand, shows a reverse in trend, with zip-codes without a Whole Foods exhibiting a 21.63% increase in CTR, and those without a Whole Foods exhibiting a 68.11% decrease in CTR. This example demonstrates the importance of geo-feature analysis for regional knowledge discovery .

2) *ZIPS Regional Analysis:* We applied our agglomerative regional discovery algorithm, ZIPS, to the pre-selected (features set, zip-code) pairs. The algorithm then discovered a set of regions that were interestingness hotspots with respect to the association of the feature set with the performance attribute. For example, the algorithm identified regional relationships between CTR and Starbucks locations. Fig. 3 shows the regions with improved CTR performance attribute. In Fig. 3, the ZIPS minimum CTR threshold for region discovery is set to 4 times the average zipcode CTR. The interestingness hotspot regions are identified in green. The ZIPS algorithm also found an interestingness hotspots between CTR and composite geo-feature of absence of high percentage of Ph.D. level education and absence of Wholefoods Markets, as shown in Fig.4.

V. Conclusions

We presented a novel framework for interestingness scoping to identify interesting regions with respect to geo-features, and performance attributes in a particular context, maximizing a plug-in user-defined interestingness function. Moreover, we presented a geo-feature pre-selection algorithm, and an agglomerative algorithm, called ZIPS, that finds such interesting hotspots by iteratively merging zip code areas. The pre-selection algorithm is capable to explore many thousands of possible geo-features and combinations automatically, identifying the most promising ones, while the ZIPS algorithm can automatically generate the regions that can benefit the most from those geo-features.

Given that the behavior of the population is the driving force determining CTR; it follows that those regional variations in population demographics would account for differences with respect to CTR in those regions. These results can, in turn, be utilized by the ad provider to improve its geo-targeted advertising in those regions.

In the experimental results, we identified promising geo-features and utilizing the selected zip codes to find strong regional associations to the performance attribute. We found strong association between regions with Starbuck, or Wholefoods increased Ad clicking behavior. Presence of Wholemarket can improve the CTR by 47% in Washington state while decrease by 68% in California. We also demonstrated that our algorithm can find composite geo-feature hotspots. In future work, a Naive Bayesian system could utilize this quantative information to improve the Yahoo! prediction algorithm.

References

[1] Aggarwal, C. C., Procopiuc, C., and Yu, P. S. Finding Localized Associations in Market Basket Data. IEEE *Trans. on Knowl. and Data Eng.* 14, 1, 2002, 51-62.

[2] Brimicombe, A.J. Cluster Detection in Point Event Data Having Tendency Towards Spatially Repetitive Events, *8th International Conference on GeoComputation*, 2005,

[3] Choo, J., Jiamthapthaksin, R., Sheng Chen, C., Celepcikay, O. U., Giusti, C., Eick, C. F. MOSAIC: A proximity graph approach for agglomerative clustering. *Proc. of the 9th Int. Conf. on Data Warehousing and Knowl. Discovery* DAWAK 2007, 231-240

[4] Data Mining and Machine Learning Group, University of Houston, http://www.tlc2.uh.edu/dmmlg.

[5] Ding, W., Jiamthapthaksin, R., Parmar, R., Jiang, D., Stepinski, T. F., and Eick, C. F., Towards region discovery in spatial datasets. *In Proceedings of the 12th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, 2008.

[6] Eick, C. F., Parmar, R., Ding, W., Stepinski, T. F., and Nicot, J. 2008. Finding regional co-location patterns for sets of continuous variables in spatial datasets. *In Proceedings of the 16th ACM SIGSPATIAL international Conference on Advances in Geographic information Systems*, 2008, 1-10.

[7] Getis, A., and Ord, J. K. Local Spatial Statistics: an Overview. *Spatial analysis: modeling in a GIS environment.* Cambridge, GeoInformation International, 261–277.

[8] Goodchild. M.F.. Twenty years of progress: GIScience, *JOURNAL OF SPATIAL INFORMATION SCIENCE*, Number 1, 2010 pp. 3–20

[9] Huang, Y., Pei, J., and Xiong, H. Mining Co-Location Patterns with Rare Events from Spatial Data Sets. *Geoinformatica* 2006, 239-260.

[10] Huang, Y. and Zhang, P.. On the Relationships between Clustering and Spatial Co-location Pattern Mining. *In Proceedings of the 18th IEEE international Conference on Tools with Artificial intelligence* 2006, ICTAI., PP513-522.

[11] Jiamthapthaksin,R., Eick, C. F. and Rinsurongkawong, V. An Architecture and Algorithms for Multi-Run Clustering, *in Proc. Computational Intelligence Symposium on Data Mining* CIDM09

[12] Kulldorff, M. Prospective Time Periodic Geographical Disease Surveillance Using a Scan Statistic. *Journal of the Royal Statistical Society Series A*, 2001, pp 61-72.

[13] Openshaw, S. Geographical data mining: Key design issues. In GeoComputation, 1999.

[14] Ord, J. K., and Getis, A. Local Spatial Autocorrelation Statistics: *Distributional Issues and an Application. Geographical Analysis*, 1995, 27(4), 286-306.

[15] Shekhar, S. and Huang, Y. Discovering Spatial Co-location Patterns: A Summary of Results. *In Proceedings of the 7th international Symposium on Advances in Spatial and Temporal Databases*, 2001. pp236-256.

[16] Xiong, H., Shekhar, S., Huang, Y., Kumar, V., Ma, X., and Yoo, J. S. A Framework for Discovering Co-location Patterns in Data Sets with Extended Spatial Objects. *In Proc. of SIAM Intl. Conf. on Data Mining* , SDM2004.

[17] Yoo, J.S., and Shekhar, S. A Join-less Approach for Mining Spatial Co-location Patterns. *IEEE Tran. on Knowledge and Data Eng,*. TKDE2006, 18

[18] http://investor.google.com/earnings.html.

[19] Guyon, I., and Elisseef, A. An Introduction to Variable and Feature Selection. Journal of Machine Learning Research, 3, 1157-1182, (2003).

[20] Liu, H., and Yu, L. Toward Integrating Feature Selection Algorithms for Classification and Clustering. IEEE Trans. On Knowledge Discovery and Data Engineering, 17, 491-502, 2005.

[21] MLC++ libraries. http://www.sgi.com/tech/mlc/index.html.

[22] ASU Feature Selection Repository. http://featureselection.asu.edu.

[23] D. Agarwal, B. –C. Chen, "Regression based latent Factor Models," *ACM SIGKDD Conference on Knowledge Discovery and data Mining*, pp. 19 – 28, 2009.