A Data Mining Framework for Analysing Geospatial-Temporal Data

Raunak Sarbajna*, Dr. Sujing Wang†
Department of Computer Science, Lamar University
Beaumont
Email: *rsarbajna@lamar.edu, †swang3@lamar.edu

***Abstract*—The purpose of this study is to detect spatiotemporal changes within sequential maps and generate automated storytelling features. Change analysis models are essential in understanding larger patterns and trends in multifaceted, time-series geographic data. All polygons under consideration are closed. spatial, georeferenced sets. The change detection is done through three primary set operations: union, intersection and erase. We initially generate polygons, by creating convex hulls from point data. Then, we calculate area of each individual polygon within each map layer. We then execute a union operation and calculate area. The union layer now contains the original areas of both layers and the areas of the overlapping polygons - we now need to query them properly to prepare for calculating the change percentage and tabulating intersection. To outline the polygon, we examine several different methods: (1) we find features common to either of the layers but not both, essentially performing a symmetrical difference, (2) we erase the larger of the polygons from the smaller, thus retaining only the growth, and do vice-verse for shrinking, (3) we perform simple intersection and then invert selection to get changed regions. We then use fitness and interesting-ness functions to determine whether or not a particular statistic is notable enough. All operations are performed using the ArcGIS/ArcPy toolkit. Our sample data for this process were shapefiles of drought intensity and impact from the North American Drought Portal and Twitter emotion spatial cluster measurements created through sentiment analysis.**

***Index Terms*—geoinformatics, change analysis, data science, data mining, sentiment analysis, storytelling**

I. Introduction

Analyzing change in spatial data is critical for many applications including developing early warning systems that monitor environmental conditions, detecting political unrest and crime monitoring.

Change analysis models are essential in understanding larger patterns and trends in multifaceted, time-series geographic data. The purpose of this study is to detect spatiotemporal changes in land use within sequential (time-series) maps. Changes in land use can be categorized by the complex interaction of structural and behavioral factors associated with technological capacity, demand, and social relations that affect both environmental capacity and the demand, along with the nature of the environment of interest.

The goal of this research project is to detect and analyze how the patterns of features change over time and space in spatiotemporal land use datasets. All polygons under consideration are closed spatial georeferenced sets, rather than raster imagery.

Our approach provides a change monitoring framework which creates a change graph that captures the changes in spatial land uses clusters and a change summarization framework that creates specific change summaries based on the change graph based on the change story types.

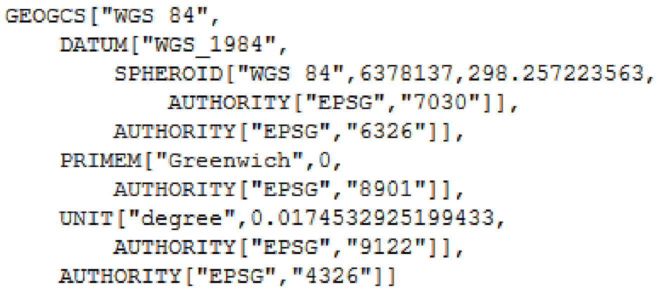
A. Data Sources

There are two different datasets under consideration here. First is the Spatiotemporal Drought Datasets from the North American Portal: https://www1.ncdc.noaa.gov/pub/data/nidis/shapefiles/.

The dataset was constructed using ”Geotagged Twitter posts from the United States” [9] and the tool used for creating the emotion clusters was the K2 framework [7]. The raw dataset contains the timestamp, longitude, latitude and the text of each tweet, which are then processed and tokenized.

According to K2, the best package for the emotion score was the Valence Aware Dictionary and Sentiment Reasoner (VADER) system [10], whose analyser parses the tokenized text and checks within a lexicon for words with strong sentiment. The final score is created from the weighted average of all sentimental words and lies within the range [-1, 1], as is required for our Aconcagua implementation.

The spatial reference for the Drought shapefiles can be seen in figure 1, and for the twitter data can be seen in figure 2.



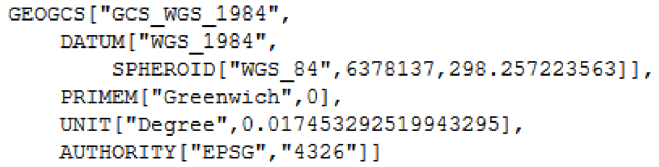
```
GEOGCS["WGS 84",
    DATUM["WGS_1984",
        SPHEROID["WGS 84",6378137,298.257223563,
            AUTHORITY["EPSG","7030"]],
        AUTHORITY["EPSG","6326"]],
    PRIMEM["Greenwich",0,
        AUTHORITY["EPSG","8901"]],
    UNIT["degree",0.0174532925199433,
        AUTHORITY["EPSG","9122"]],
    AUTHORITY["EPSG","4326"]]
```

Fig. 1. Spatial Reference for Drought data

The layer definition for the data sets can be seen in tables 1 and 2.

II. Motivation, Related Work

A survey of the classical change detection algorithms can be found in the Lu et al. [3] paper and tells us that the integrated



```
GEOGCS["GCS_WGS_1984",
    DATUM["WGS_1984",
        SPHEROID["WGS_84",6378137,298.257223563]],
    PRIMEM["Greenwich",0],
    UNIT["Degree",0.017453292519943295],
    AUTHORITY["EPSG","4326"]]
```

Fig. 2. Spatial Reference for Drought data

TABLE I
LAYER SPECIFICATION FOR DROUGHT DATA

Name	Type	Width	Precision
long	Real	24	15
lat	Real	24	15
id	Integer	9	0
dnstyTh	Real	24	15
avgScor	Real	24	15
numTwts	Integer	9	0
stdDev	Real	24	15
batchNm	Integer	9	0
geoData	String	80	0

GIS and remote sensing approaches yield the best results. However, they are very sensitive to registration accuracies between images. Thus, images must be properly orthorectified and georeferenced, especially because the changes in the emotion polygons are so subtle. This assumes the emotions are to be treated as just another feature in the map, like any other category.

Since our data is primarily in an urban environment, with all the grid like rigidity that entails, it is a good idea to look at change detection algorithms optimized for urban environments. One of the hardest aspects to measure is to distinguish between change and no-change, as well as different kinds of change. Comparing image differencing, image regression, tasseled-cap transformation and chi square transformation, Ridd and Liu [3] find image differencing to be the most consistent, with a sustained overall accuracy of >80%.

It is useful to have a programming-oriented study comparing several of the change detection algorithms using MATLAB, rather than pure application-oriented comparison, in order to have a benchmark. Minu and Shetty [5] analyzed image differencing, image ratioing, change vector analysis, tasseled cap transformation and principal component analysis for efficiency and effectiveness. Although their area of study was not urban but a variety of land use/ land cover, change vector analysis gave the best overall accuracy. We also studied two novel methods that are recent developments and are showing promising results: Neighborhood Correlation Image and Comprehensive Change Detection Method, both of which are optimized for remote sensing imagery but can be adapted to vectorized maps without loss of generality.

TABLE II
LAYER SPECIFICATION FOR TWITTER DATA

Name	Type	Width	Precision
FIPS_ADMIN	String	4	0
GMI_ADMIN	String	7	0
ADMIN_NAME	String	42	0
FIPS_CNTRY	String	2	0
GMI_CNTRY	String	3	0
CNTRY_NAME	String	40	0
POP_ADMIN	Integer	9	0
TYPE_ENG	String	26	0
TYPE_LOC	String	50	0
SQKM	Real	16	2
SQMI	Real	16	2
COLOR_MAP	String	2	0

The change detection model using Neighborhood Correlation Image (NCI) logic works because of the obvious fact that the same geographic area (e.g., a 3x3 pixel window) on two dates of imagery will tend to be highly correlated if little change has occurred, and uncorrelated when change occurs [1]. Computing the piecewise correlation between two data sets demonstrates that NCIs contain change information and that NCIs may be powerful tools for change detection.

A high-performance remote sensing method called Comprehensive Change Detection Method (CCDM) integrates spectral-based change detection algorithms and a novel change model called Zone, which extracts change information from two Landsat image pairs [2]. This can be easily modified to work on the Twitter-based emotional grading maps. This method is simple, easy to operate, widely applicable, and capable of capturing anthropogenic changes like our area of interest.

III. METHODOLOGY

A. Emotion Spatial Clusters from Twitter

We are implementing and improving upon the change analysis framework called Aconcagua [6]. The system expects an input of emotion polygons annotated with emotion assessment scores, with +1 representing a very high positive emotion and -1 representing a very high negative emotion. While this method does lead to inconsistencies in locations due to georeferencing inaccuracies, we find that the 8 10m precision [8] works well within city limits.

There are several ways of using the numerous point data we have obtained from this step into an actual polygonal map:

1) Creating closed contour lines for contour lines that lie on the boundary of the observation area.
2) Creating a convex hull from points with similar scores.

We elaborate on creating convex hulls in the following section.

B. Point Data Sources

Our initial approach to this problem was to store all shapefiles in a postgres database with a GIS addon and perform operations in python. We used psycopg2 and osgeo libraries to import, process and visualize maps. However, this lead to many problems with interconversions between georeferencing schemes, while converting from WKT geometry to PostGIS geography.

We start with basic point data, which contains latitude/-longitude, along with metadata identifying value of interest, whether that is drought level or emotion value. We insert the

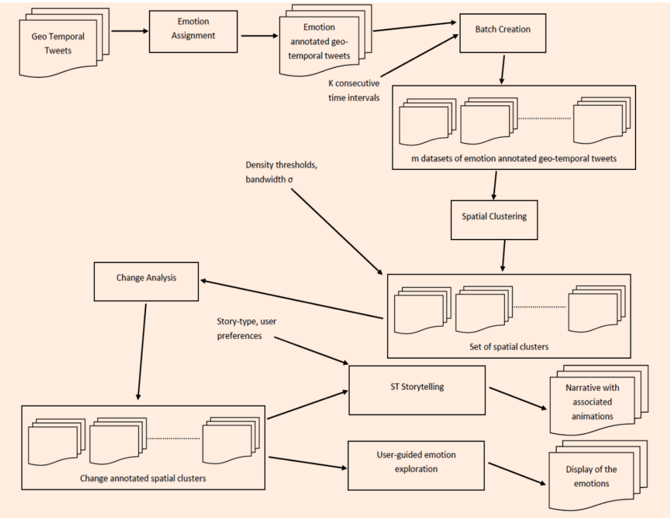
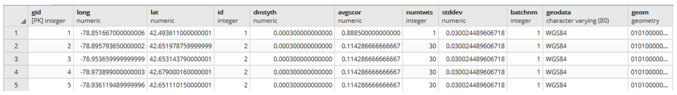

Geo Temporal Tweets
Emotion Assignment
Emotion annotated geo-temporal tweets
Batch Creation
K consecutive time intervals
m datasets of emotion annotated geo-temporal tweets
Density thresholds, bandwidth σ
Spatial Clustering
Change Analysis
Set of spatial clusters
Story-type, user preferences
ST Storytelling
Narrative with associated animations
Change annotated spatial clusters
User-guided emotion exploration
Display of the emotions

Fig. 3. The K2 Framework

contents of the shapefile into a PostGreSQL database using the shp2pgsql toolkit that comes along with the PostGIS extension. The results can be seen in Figure 4.



	gid [PK] integer	long numeric	lat numeric	id integer	dnstyth numeric	avgscor numeric	numtwts integer	stddev numeric	batchnm integer	geodata character varying (80)	geom geometry
1	1	-78.851667000000006	42.493611000000001	1	0.000300000000000	0.888500000000000	1	0.030024489606718	1	WGS84	010100000...
2	2	-78.895793650000002	42.651978759999999	2	0.000300000000000	0.114286666666667	30	0.030024489606718	1	WGS84	010100000...
3	3	-78.953659999999999	42.653143790000001	2	0.000300000000000	0.114286666666667	30	0.030024489606718	1	WGS84	010100000...
4	4	-78.973899000000003	42.679000160000001	2	0.000300000000000	0.114286666666667	30	0.030024489606718	1	WGS84	010100000...
5	5	-78.936119489999996	42.651110150000001	2	0.000300000000000	0.114286666666667	30	0.030024489606718	1	WGS84	010100000...

Fig. 4. Twitter shapefile data inserted into Postgres db

As our framework relies on using polygonal data, we cannot use this. So we begin by creating a convex hull of points and inserting it into a new table, while preserving related meta-data. Due to engine limitations, we need to be careful and insert only those convex hulls that are polygons specifically, not points or lines.

1) Our query for this purpose was:

```
INSERT INTO public.june1poly (avgscor,
numtwts, geodata, id, batchnm, geom)

(SELECT d.avgscor, d.numtwts, d.geodata,
d.id, d.batchnm, ST_ConvexHull
(ST_Collect(d.geom))
FROM public.”2014−06−01 ” AS d
GROUP BY (d.id, d.avgscor, d.numtwts,
d.geodata, d.batchnm)

HAVING ST_GeometryType(ST_ConvexHull
(ST_Collect(d.geom))) = ’ST_Polygon’)
```

2) Then we insert the centroid of each polygon into the table using the query:

```
UPDATE public.june1poly
SET centroid=ST_Centroid(geom)
```

3) We repeat this process for every shapefile needed.

Next we run our change predicates, which include:

1) $S - Continuing(c, m) \leftrightarrow Agreement(c, m) \geq 0.8$
2) $B - Continuing(c, b) \leftrightarrow Oap(c, b) \geq 0.8$
3) $Growing(c, m) \leftrightarrow Contaiverlnment(c, m) \geq 0.9$
4) $Shrinking(c, m) \leftrightarrow Growing(m, c)$
5) $Disappearing(c) \leftrightarrow \exists i(belong - to(c, i)$
6) $Novel(c) \leftrightarrow \exists i(belong - to(c, i) and (i = 1 or not(B - Continuing(c, i - 1))$
7) $Shifting$

Which are defined as:

- $Agreement(c, m) = (area(c \cap m))/(area(c \cup m))$
- $Overlap(s, f) = area(p \cap (p_1 \cap \ldots \cap p_m))/area(p)$

We will demonstrate three of these predicates here.

1) To detect polygons that are increasing in size, we check for similar IDs, intersection and then the rate of overlap. We initially check whether the polygons intersect at all before querying for amount of overlap. This leads to faster processing as it discards the many combinations where the polygons don’t touch each other. Our query is structured as:

```
SELECT DISTINCT j2.∗
FROM public.june1poly j1,
 public.june2poly j2
WHERE ST_INTERSECTS(j1.geom, j2.geom)
AND
(ST_AREA(ST_INTERSECTION(j2.geom, j1))
/st_area(j2.geom)) > .85
```

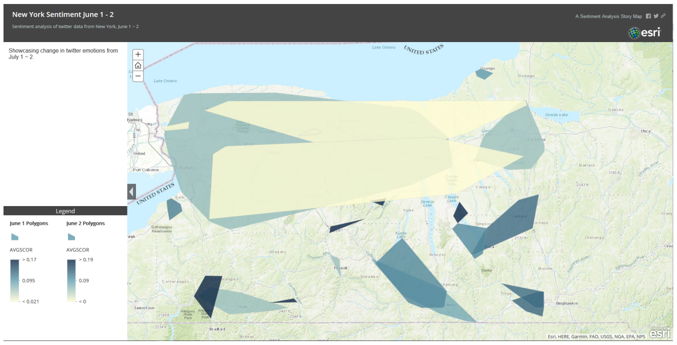
2) To detect polygons that are shrinking in size, we check for similar IDs, and lower rates of overlap. This can be modified based on need.

```
SELECT DISTINCT j2.∗
FROM public.june1poly j1,
 public.june2poly j2
WHERE ST_INTERSECTS(j1.geom, j2.geom)
AND
(ST_AREA(ST_INTERSECTION(j2.geom, j1))
/st_area(j2.geom)) < .25
```

3) To detect polygons that have shifted we compare their centroids and check if they have moved over 75km. Remember these polygons are created through a convex hull of points, which cannot ensure the centroid will lie within the polygon itself. Which is why we are taking a sufficiently large bounding value for the polygon.

```
SELECT ST_Distance_Spheroid
(j1.centroid, j2.centroid,
 ’SPHEROID[”WGS 84”,6378137,298.25]’),
  j1.id FROM public.june1poly j1,
  public.june2poly j2
WHERE j1.id = j2.id AND
 ST_Distance_Spheroid(j1.centroid,
j2.centroid,
 ’SPHEROID[”WGS 84”,6378137,298.25]’)>75000;
```

A sample of what these generated maps look like can be found in Figure 5


New York Sentiment June 1 - 2

Sentiment analysis of twitter data from New York, June 1 ~ 2

Showcasing change in twitter emotions from July 1 ~ 2.

Legend

June 1 Polygons — AVGSCOR: > 0.17, 0.095, < 0.021

June 2 Polygons — AVGSCOR: > 0.19, 0.09, < 0

Fig. 5. Polygons generated from Sentiment analysis

C. Polygonal Map Data Sources

Our next, more successful approach was done through three primary set operations: union, intersection and erase. We calculate area of each individual polygon within each map layer. We then execute a union operation and calculate area. The union layer now contains the original areas of both layers and the areas of the overlapping polygons - we now need to query them properly to prepare for calculating the change percentage and tabulating intersection.

To outline the polygon, we examine several different methods:

1) We find features common to either of the layers but not both, essentially performing a symmetrical difference
2) We erase the larger of the polygons from the smaller, thus retaining only the growth, and do vice-verse for shrinking
3) We perform simple intersection and then invert selection to get changed regions.

Our approach then combined several techniques:

1) **Data Pre Processing** This involves curation of datasets with obvious georeferencing errors. This would preferable be done by minimizing the root mean square error. We initially.
2) **Parametrization of polygons** Calculate shape and area parameters for each individual polygon with each map layer.s
3) **Analysis through Symmetrical Difference** Extract features common to either of the layers
4) **Polygon Union Computation** Union sequential layers to contain the original areas of both layers and the areas of overlapping polygons
5) **Polygon Erase Operation** Erase larger polygons from smaller (or vice-versa) for detection of growth/shrinking
6) **Polygon Intersection/Invert** Selecting and then labelling the changed regions

Our original data source for the drought sets were defined by the USDA as seen in Figure 6.

IV. Change Storytelling

In order to able to tell a coherent spatio-temporal data story from the change analysis output, we need to be able to pick out resultant change polygons that have seem to have significant impact on the dataset as a whole. We propose using an interestingness function to look for these.[11]

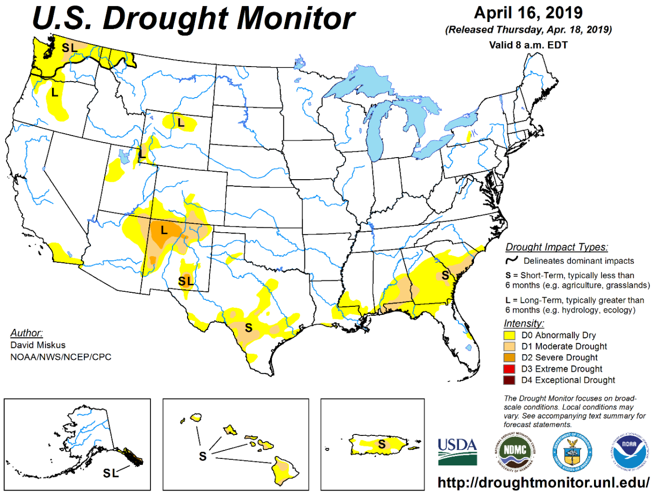

U.S. Drought Monitor
April 16, 2019
(Released Thursday, Apr. 18, 2019)
Valid 8 a.m. EDT
SL
L
L
L
L
SL
S
S
S
SL
S
S
Author:
David Miskus
NOAA/NWS/NCEP/CPC
Drought Impact Types:
Delineates dominant impacts
S = Short-Term, typically less than 6 months (e.g. agriculture, grasslands)
L = Long-Term, typically greater than 6 months (e.g. hydrology, ecology)
Intensity:
D0 Abnormally Dry
D1 Moderate Drought
D2 Severe Drought
D3 Extreme Drought
D4 Exceptional Drought
The Drought Monitor focuses on broad-scale conditions. Local conditions may vary. See accompanying text summary for forecast statements.
USDA
NDMC
NOAA
http://droughtmonitor.unl.edu/

Fig. 6. United Sates Drought Monitor Data

We choose a set of change polygons SCP. These not only contains those objects but also their associated characteristics. For example, a SCP could contain a set of spatial clusters with polygon, their average drought score, total area, centroid coordinates and other summaries for each spatial cluster. We define the function as:

$$f : SCP \; \rightarrow \; [0 \, , \infty)$$

We define a threshold ω, which ensures that a narrative will only be generated an an object $p \in SCP$ such that $f(p) \geq \omega$. Example parameters for ω include

- $Max\left(\frac{area(Polygon \;\; P_i)}{\sum_1^i area(Polygon \;\; P_i)}\right)$
- $Max\,(Percentage \;\; Change \;\; in \;\; Polygon)$
- Largest shift in polygon centroids

The threshold parameters need to be finely tuned so as to not exclude those polygons who fall through exceptions. Once we have a suitable selection of polygons and have chose a threshold value, we can create a summary narrative.

V. Results

The original polygons for the drought datasets, before our analysis was started, can be found in Figure 7

We focus on two specific regions to highlight. First, figure 8 shows the regions where areas of drought grew in California following the wildfires in 2017. We noticed patterns of increasing drought surrounding the regions that were burnt down, with especially large period upstream from rivers. Our procedure worked well dealing with the large number of small polygons in the region that occur due to isolated wildfires. However, we had difficult dealing with convex polygons that intersected with each other multiple times.

Next, we inspect the region in Texas after Hurricane Harvey. Figure 9 shows the regions where drought affected regions

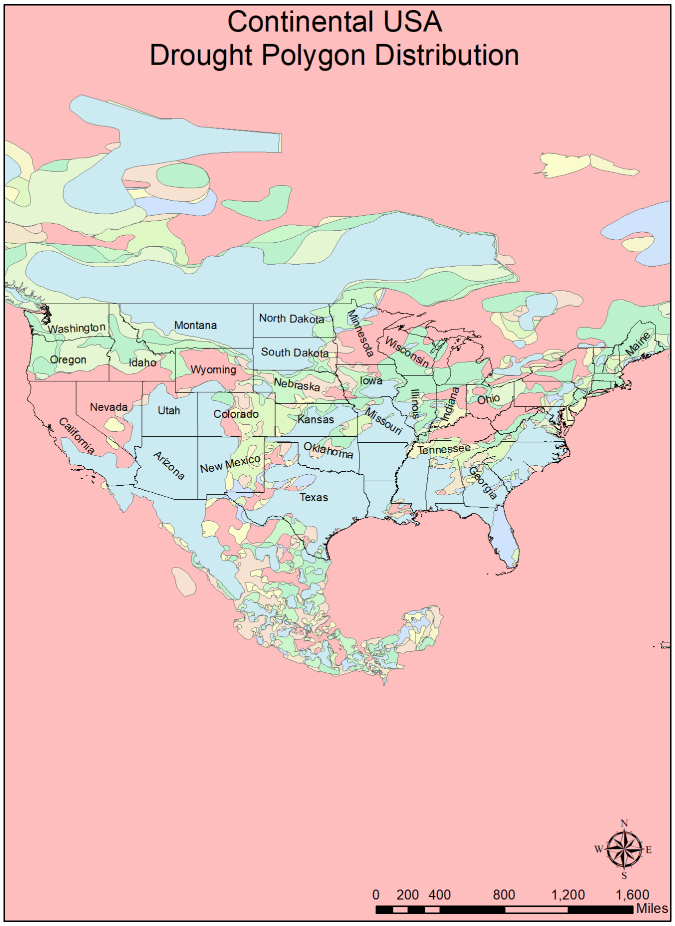


Fig. 7. Continental USA Original Drought polygon Distribution for 2017

increased following the disaster and Figure 10 shows the regions where they decreased.

We observe that drought affected regions decrease at a high rate around the South Eastern Texas and Louisiana region, which follows common logic. However, there is no clear relation between drought prone regions and river basins close to the coast. We believe that high amount industrial regions create a micro climate that affects the water content entering the soil.

According to our results, the drought prone regions increased substantially around the West Texas region both during and after Hurricane Harvey, with a larger increase as winter came around. However, this could be an artefact of the fact that the original dataset had very few polygons in the Texas area, which leads to broader conclusions. Our technique is still severely dependent on the resolution of the input imagery.

The results from the analysing the twitter emotion maps can be seen in Figure 11.

VI. Conclusion

Our experimental studies show that our change detection and analysis framework can successfully detect changes in both polygonal and point-set land use spatiotemporal datasets.

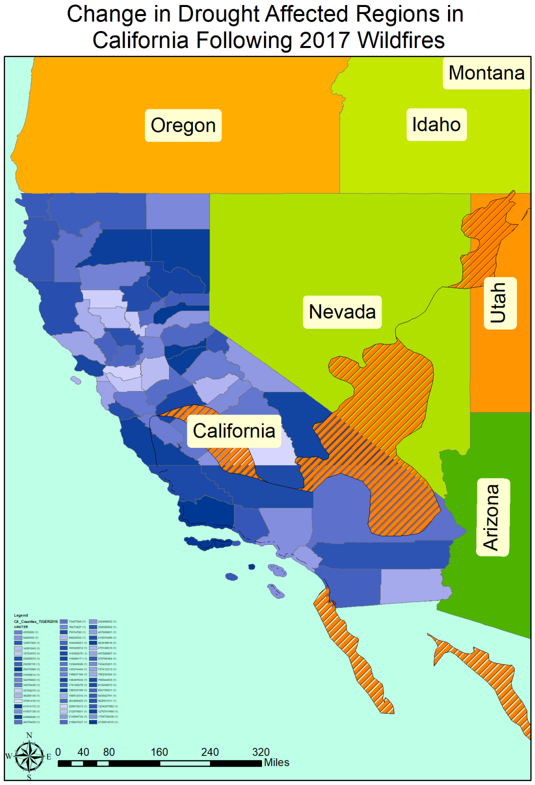


Fig. 8. Change in drought affected regions in California following 2017 wildfires.

VII. Future Work

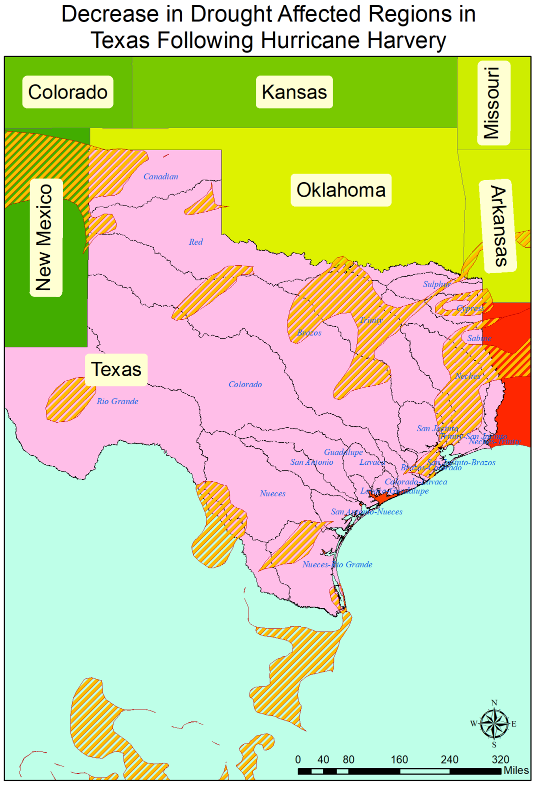
There is a lot work left in dealing with irregularly shaped convex polygons that appear in poorly georeferenced real world data. We believe this is because the sensitivity of the geographic operations in PostGIS or ArcGIS. We are currently working on extending our framework to deal with edge cases like those.

We are also looking towards consolidating and extending some of the change predicates, especially trying to tune them for fine changes.

We will work on the creating change summaries and stories, as detailed in Section IV, from the data in future.

Acknowledgment

The authors wish to thank Dr. Sujing Wang for her advice and critical guidance during bottlenecks in the project. We would also like to thank Dr. Stefan Andrei for his support.



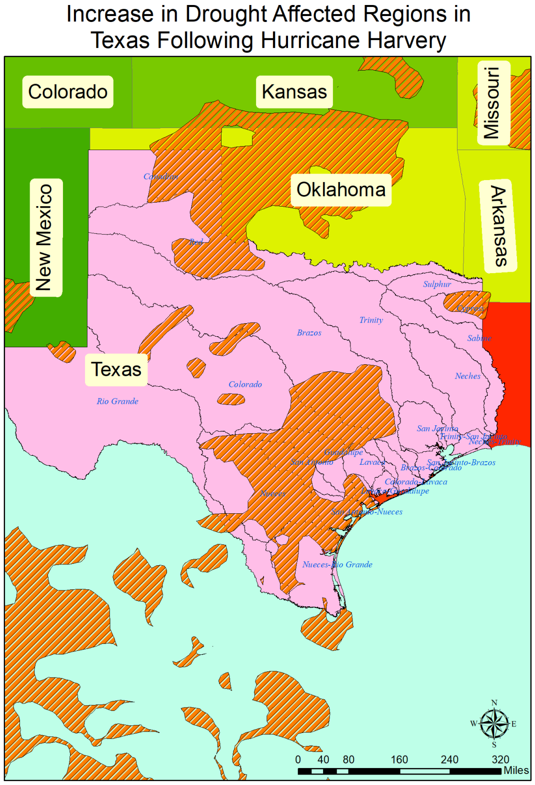


Fig. 9. Decrease in Drought Affected Regions in Texas following Hurricane Harvey

Fig. 10. Increase in Drought Affected Regions in Texas following Hurricane Harvey

References

[1] J. Im and J. Jensen. 2005. A change detection model based on neighborhood correlation image analysis and decision tree classification. Remote Sensing of Environment 99, 3 (2005), 326–340. DOI:http://dx.doi.org/10.1016/j.rse.2005.09.008

[2] Suming Jin, Limin Yang, Patrick Danielson, Collin Homer, Joyce Fry, and George Xian. 2013. A comprehensive change detection method for updating the National Land Cover Database to circa 2011. Remote Sensing of Environment 132 (May 2013), 159–175. DOI:http://dx.doi.org/10.1016/j.rse.2013.01.012

[3] D. Lu, P. Mausel, E. Brondízio, and E. Moran. 2004. Change detection techniques. International Journal of Remote Sensing 25, 12 (June 2004), 2365–2401. DOI:http://dx.doi.org/10.1080/0143116031000139863

[4] Merrill K. Ridd and Jiajun Liu. 1998. A Comparison of Four Algorithms for Change Detection in an Urban Environment. Remote Sensing of Environment 63, 2 (1998), 95–100. DOI:http://dx.doi.org/10.1016/s0034-4257(97)00112-0

[5] S. Minu and Amba Shetty. 2015. A Comparative Study of Image Change Detection Algorithms in MATLAB. Aquatic Procedia 4 (March 2015), 1366–1373. DOI:http://dx.doi.org/10.1016/j.aqpro.2015.02.177

[6] K. Elgarroussi, S. Wang, R. Banerjee, and C. F. Eick, "Aconcagua: A Novel Spatiotemporal Emotion Change Analysis Framework," in Proceedings of the 2Nd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery, New York, NY, USA, 2018, pp. 54–61.

[7] R. Banerjee, K. Elgarroussi, S. Wang, A. Talari, Y. Zhang, and C. F. Eick, "K2: A Novel Data Analysis Framework to Understand US Emotions in Space and Time," Int. J. Semantic Computing, vol. 13, no. 01, pp. 111–133, Mar. 2019.

[8] Geomenke, "How Accurate is the GPS on my Smart Phone? (Part 2)," Community Health Maps, 07-Jul-2014. .

[9] W. Z.-M.-L. I. for the S. S. Person, "Geotagged Twitter posts from the United States: A tweet collection to investigate representativeness," Jan. 2016.

[10] C. J. Hutto, VADER Sentiment Analysis. VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social .. 2019.

[11] X. Huang, A. An, and N. Cercone, "Comparison of Interestingness Functions for Learning Web Usage Patterns," in Proceedings of the Eleventh International Conference on Information and Knowledge Management, New York, NY, USA, 2002, pp. 617–620.

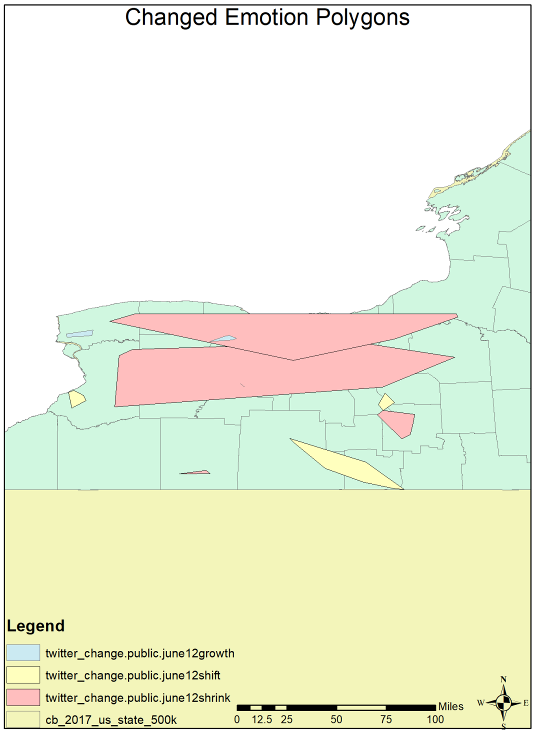

Changed Emotion Polygons
Legend
twitter_change.public.june12growth
twitter_change.public.june12shift
twitter_change.public.june12shrink
cb_2017_us_state_500k
0
12.5
25
50
75
100
Miles
N
W
E
S

Fig. 11. Change in emotion Polygons in New York State