Identification of the safest path using spatio-temporal analysis
Motivation

- In today's society criminal activities are on the rise
- We intend to come up with a way by which one can ensure that he travels from one place to the other by the safest route possible
- Governments all over the world are spending millions trying to curb this menace
Approach

News Article
  ↓
Classification
  ↓
Identification of Location and Date
  ↓
Temporal Analysis
  ↓
Dijkstra’s Algorithm for safest path

Police Record
Classification of articles

• We use the Latent Semantic Analysis[1] for classifying articles.
• LSA is essentially creating a vector representing a document.
  • Construct a term-document matrix of the corpus.
• Single Value Decomposition (SVD) is then employed to reduce the dimensionality of the matrix.
• The LSA helps in grouping words with similar topics together.
• Classification using k-nearest neighbors with respect to cosine distances of the document vectors.
Identification of Location

- Statistical NER methods not well-suited to the dynamic nature of news as noted by Stokes et.al [2]
- We use fuzzy geotagging [3] to resolve the bootstrapping problem associated with the traditional method
- In fuzzy geotagging a toponym recognition system first finds the toponyms $T$ in an article $a$. 
• Given a news article, we tag each word with its part of speech, using the POS tagger, and collect all word phrases consisting of proper nouns.
• We also apply NER to the article, and collect all phrases tagged as locations.
• For resolving the POS tags we use a number of heuristic rules.
• Database of geographic locations, is then used to associate each $t \in T$ with the set of all possible interpretations $R_t$
• For each $t$ and $r \in R_t$, a weight $w_r$ is assigned to $r$ using default sense heuristics
# Heuristic Rules

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{H}_1$ Dateline</td>
<td>Resolve dateline toponyms using: $\mathcal{H}_4, \mathcal{H}_5, \mathcal{H}_6$. Resolve other toponyms geographically proximate to resolved dateline.</td>
<td>LONDON, Ont. – A police...巴黎, TX (AP) – New...</td>
</tr>
<tr>
<td>$\mathcal{H}_2$ Relative Geog.</td>
<td>Resolve anchor toponym using: $\mathcal{H}_1, \mathcal{H}_4, \mathcal{H}_5, \mathcal{H}_6$. Resolve other toponyms proximate to defined geographic point or region.</td>
<td>...4 miles east of Athens, Texas. ...lives just outside of Lewistown...</td>
</tr>
<tr>
<td>$\mathcal{H}_3$ Comma Group</td>
<td>Resolve toponym group using: $\mathcal{H}_6, \mathcal{H}_7$, Geographic Proximity.</td>
<td>...California, Texas and Pennsylvania. ...priority in Jordan, Minn., ... (news source dependent)</td>
</tr>
<tr>
<td>$\mathcal{H}_4$ Loc/Container</td>
<td>Resolve toponym pairs with a hierarchical containment relationship.</td>
<td>...issues with China, knowing... (article dependent)</td>
</tr>
<tr>
<td>$\mathcal{H}_5$ Local Lexicon</td>
<td>Resolve toponyms geographically proximate to local lexicon centroid.</td>
<td></td>
</tr>
<tr>
<td>$\mathcal{H}_6$ Global Lexicon</td>
<td>Resolve toponyms found in a curated list of globally-known places.</td>
<td></td>
</tr>
<tr>
<td>$\mathcal{H}_7$ One Sense</td>
<td>Resolve toponyms sharing names with earlier resolved toponyms.</td>
<td></td>
</tr>
</tbody>
</table>

Source: M.D Liebermann et. al
Pseudo-Code

Algorithm 1 Infer an intended audience’s local lexicon.

Input: Set of articles $A$, Maximum diameter $D_{max}$, Minimum lexicon size $S_{min}$

Output: Local lexicon $L$, or $\emptyset$ if none

1: procedure INFERLOCALLEXICON($A$, $D_{max}$, $S_{min}$)
2:   $G \leftarrow \emptyset$
3:   $L \leftarrow \emptyset$
4:   for all $a \in A$ do
5:     $G \leftarrow G \cup FUZZYGEOTAG(a)$
6:   end for
7:   $G \leftarrow ORDERBYWEIGHT(G)$
8:   for $i \in \{1 \ldots |G|\}$ do
9:     $H \leftarrow CONVEXHULL(L \cup G_i)$
10:    if DIAMETER($H$) > $D_{max}$ then
11:       break
12:   end if
13:   $L \leftarrow L \cup G_i$
14: end for
15: if $|L| < S_{min}$ then
16:   $L \leftarrow \emptyset$
17: end if
18: return $L$
19: end procedure

Source: M.D Liebermann et. al
Temporal Analysis

- Extract the date of the news article/FIR through crawling
- We will use a hybridization of artificial neural networks and ARIMA models for time series forecasting[4].
- In an ARIMA (p, d, q) model, the future value of a variable is assumed to be a linear function of several past observations and random errors.
  \[ \phi(B)\nabla^d(y_t - \mu) = \theta(B)a_t \]
- The parameters are estimated such that an overall measure of errors is minimized
• The time series is considered as function of a linear and a nonlinear component. Thus, $y_t = f(L_t, N_t)$

• After performing ARIMA model at the first stage we assume that the residuals will contain a non-linear relationship.

• A multilayer perceptron is used to model the non-linear component existing in the residuals
\[ N_{1t} = f^1(e_{t-1}, \ldots, e_{t-n}) \]
\[ N_{2t} = f^2(z_{t-1}, \ldots, z_{t-n}) \]
\[ N_t = f(N_{1t}, N_{2t}) \]

where \( f^1, f^2, f \) are the nonlinear functions determined by the neural network.

\[ y_t = f(N_{1t}, \tilde{L}_t, N_{2t} t) = f(e_{t-1}, \ldots, e_{t-n_1}, \tilde{L}_t, z_{t-1}, \ldots, z_{t-m_1}) \]

- We will use simple Dijkstra’s algorithm to find the “safest path” based on weights by temporal analysis
Dataset

1. Crime records have been extracted from the Delhi Police Website [5]
2. News articles (both crime and non crime) have been extracted from the Times Of India, Hindu etc. Website using a crawler.
Result and validation

• The validation will be a three fold procedure
  1. The accuracy for classification of an article as a crime/non-crime
  2. Accuracy with which the location can be correctly specified on ACE 2005 dataset
  3. Least Square residual for temporal analysis
Future Work

• Use actual road paths for mapping crime
• Include more sources of information for crime hotspot identification
References


5. http://delhipolice.serverpeople.com

Questions/Suggestions