DT-GIS System for Tactical Pattern Exploration in Asymmetric Conflicts[1]

Xing Wang, Jason Lin, and Jyh-Charn Liu

Texas A&M University, College Station, TX, {xingwang, senyalin, liu}@cse.tamu.edu

Stephen George

U.S. Department of Defense, Washington D.C., ticom.dev@gmail.com

Abstract: In asymmetric conflicts, the aggressors (typically insurgents) tend to prefer locations with a good observability, to support attack functions, and good concealment, to protect their positions. Ongoing research has shown that statistical machine learning (SML) techniques are effective in training classifiers to predict high-risk locations based on features derived from terrain, visibility measures, and the local population under constrained experiment environment. But it is unclear if these prediction techniques are ready for deployment in the real world. One important issue is the interpretability, the ability for a human to understand and incorporate the SML output into decision-making and action. To overcome this shortcoming, we offer a decision tree-based geographic information system (DT-GIS). The decision tree (DT) classification engine presents its classifiers as human-understandable rules. The GIS system illustrates the locations in its environment overlaid with DT-based predictions that help humans verify the results from the DT system. We find that the primary features of the decision tree for Afghanistan are the presence of the local population, but at this scale the decision rules only identify likely conflict regions and cannot discriminate inside the region. We propose a human-in-the-loop interface that allows a user to define the region of interest (ROI) and explore the characteristics of attack locations in the ROI only. Results show that DT for ROI is more likely focus on visibility and terrain features. Quantitative evaluation based on a ROC curve and a trade-off analysis between sensitivity and specificity are presented. The visualization ability of the system is demonstrated as a powerful tool for system checking and feature engineering.

Topic: 3 Data, Information and Knowledge; 4 Experimentation, Metrics and Analysis

Keywords: Intelligence, surveillance, target acquisition and reconnaissance; Decision Tree; Geographical Information System; Decision-Making.

1. INTRODUCTION

Intelligence, surveillance, target acquisition and reconnaissance (ISTAR) is the practice that links several battlefield functions together in order to improve a commander's situational awareness and consequently their decision-making [1]. In the era of asymmetric conflicts, the ISTAR practice is crucial for defense, allowing a commander to recognize threats, and useful for offense, where disparate functions provide a layered view of the battle space. In the case of asymmetric conflict attacks, although the locations for the attack may vary, there are essential elements that are difficult to remove, mask or conceal. These essential elements can be mined or extracted from available data and captured as tactical patterns by exploring historical attacks. As a source of intelligence, these patterns play an important role in the countermeasures design. In [2], we propose the MECH (Monitor, Emplacement, Control in a Halo) model to describe the environment of potential attack locations, specifically improvised explosive device (IED) and direct fire (DF) and conduct prediction analysis using feature derived from MECH. Accurate prediction performance is achieved under constrained experiment settings, but additional effort is needed to successfully and safely integrate a statistical pattern mining system into ISTAR practice.

[1] This work was supported in part by an ONR grant N00014-12-1-0531 and a National Defense Science and Engineering Graduate (NDSEG) fellowship. Opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsors. Corresponding author is J.C. Liu.

In this work, we focus on the interpretability issue of statistical machine learning (SML). Most SML toolkits work as a black box with numerical operation inside. They accept the feature vector of a sample and give a prediction with an associated confidence level. This approach does not satisfy the goal of using SML as an information source to control the process. In this paper, we want to transform the inner state and operation inside the toolkit into human understandable knowledge and check whether it could be helpful in assisting ISTAR practice. In this work, we adopt a decision tree algorithm to construct rules that discriminate attack locations from non-attack locations. A geographical information system (GIS) combined with DT engine is used to visualize the internal data structure of the patterns discovered by the DT classifiers. A total of 62 features from geomorphometry, visibility and population data are adopted to describe candidate attack locations [2]. In addition, we introduce human-in-the-loop (HITL) operations that support iterative pattern exploration by constraining the region of interest (ROI).

Overall, this research offers three contributions: First, we offer interpretable patterns describing asymmetric conflict events as the main goal of our statistical machine learning. Traditional measures such as the accuracy or receiver operating characteristic curve are secondary indicators. Through step-by-step illustration, the user gains deeper insight into class ontologies, data sampling and rule representation of the data-driven pattern extraction. Second, we adopt an HITL framework that enables analysts to iteratively redefine the region of interest. By selecting a region of interest, we can get refined rule set that provides a more localized view of conflict event tactics and the terrain selected for attacks. Often, this tends to be strongly related to terrain and visibility. Finally, the mapping module of DT-GIS system provides an analyst with a rich set of visual information including roads, building, vegetation, farms, mountains, etc. By inspecting the attack location through DT-GIS, the analysts may be inspired to design more features and are able to more easily understand the limitations of the existing feature sets.

The paper is organized as follows: Section 2 presents related work on the ISR and SML field. Section 3 introduces the feature derived from MECH model. Section 4 describes the Decision Tree Algorithm. Section 5 presents the system design. Section 6 presents the experiment design, results, case study, and analysis. .

2. RELATED WORKS

Asymmetric conflict (AC) has been widely studied and discussed for decades. There are books that describe AC from an attacker's point of view, like Che Guevera's famous treatise on guerrilla warfare [3], and from a defender's point of view, like the the U.S. Army/Marine Corps Counterinsurgency Field Manual [4] . However, most literature in this field contains qualitative descriptions without the associated quantitative reasoning. Three categories, composed primarily of newer publications, address the study of AC using a quantitative approach.

The first category is coarse heat-map work. Shakarian adopted the SCARE-S2 [5] system to find high-value targets, but builds maps with region-level resolution. Eck, et al. [6] use crime-driven hot spots and heat maps to visualize crime data but rely on high-precision geographic, economic and historical criminal data. Both of these works reliably reduce the size of the region where surveillance is required but do not help locating the exact location of key emplacement and support elements of an AC attack.

A second category focuses on modeling the problem as time-series point process as in [7]. This approach produces interesting findings about general trends at the regional level but requires significant datasets, "a few dozens per year" per region, to estimate parameters with some degree of stability. Between the assumption of temporal invariance (tactics never change) and homogenization of attack types (all attack types are considered together and equally), point process approaches will likely require significant maturation before they are ready to support situational awareness at sub-region levels. Regardless, time-series properties are not the topic in this paper, although we may pursue this direction later.

The final category consists of statistical machine learning-based methods. A detailed elaboration on statistical machine learning can be found in [8]. In our previous work [2][9], statistical machine learning is adopted for prediction analysis with high accuracy. However, a significant shortcoming is that none of the algorithms used —support vector machines, k-Nearest Neighbors, and discriminant analysis— output explicit rules that a human can understand. Situational awareness might be enhanced if underlying rules and decisions related to AC could be synthesized from analysis of attack-related data. Another important characteristic of our problem is unbalanced data sets. Even in the most contested areas, the total count of historical AC event locations is tiny when compared to the total count of potential sites in an entire road network in a region or country. More elaboration of this problem can be found in the work related to downsampling and evaluation criteria [10][11].

3. OVERVIEW OF FEATURES

Table 1 lists a total of 62 features currently proposed for use in the MECH model [2]. These are grossly divided into features extracted from the viewshed of a potential Emplacement site, G_1 and G_2; geomorphometric features that describe the area at or near an Emplacement site, G_3; and features that capture the distance to nearby populated areas.

Feature set G_1 is composed of two features related to viewshed, collected using four different windows, with intervisibility based on elevations provided by the ASTER Global Digital Elevation Map (DEM). The first feature, visibility index, measures the total area within the collection window that has intervisibility with the Emplacement site. Larger viewsheds tend to expose a target to an attacker for longer periods of time. An alternative interpretation is that the target is likely to have to move a greater distance before reaching cover. The second feature, shape complexity, estimates the dispersion or texture of the viewshed. More complicated or textured viewsheds offer both attackers and target greater possibilities for concealment.

Feature set G_2 consists of five features extracted from sparse viewshed, a viewshed estimation technique that summarizes intervisibility based on a set of evenly spaced radials originating at the Emplacement site. Local openness, built on work by Yokoyama [12], estimates the openness or exposure of the Emplacement site. Longest and shortest radial lengths are used to measure the distance to the farthest and nearest invisible regions. Planimetric area estimates the total extent of the main contiguous portion of the viewshed and rugosity estimates the roughness of the same area. Three-dimensional shape complexity uses radials to estimate the roughness of the viewshed.

Conventional geomorphometric features [13] of the Emplacement site and its environs are captured in feature set G_3. These are calculated from elevation data provided by the DEM and consist of four features measured at the site itself —elevation, slope, convexity and texture— and two features that describe the embedding of the site in local terrain — elevation range and roughness.

The final feature set, G_4, estimates the distance from the Emplacement site to populated areas of different sizes. The socio-cultural feature is based on the notion that attackers need some level of support in order to operate, often in the areas of lodging, supplies, and access to long distance communications.

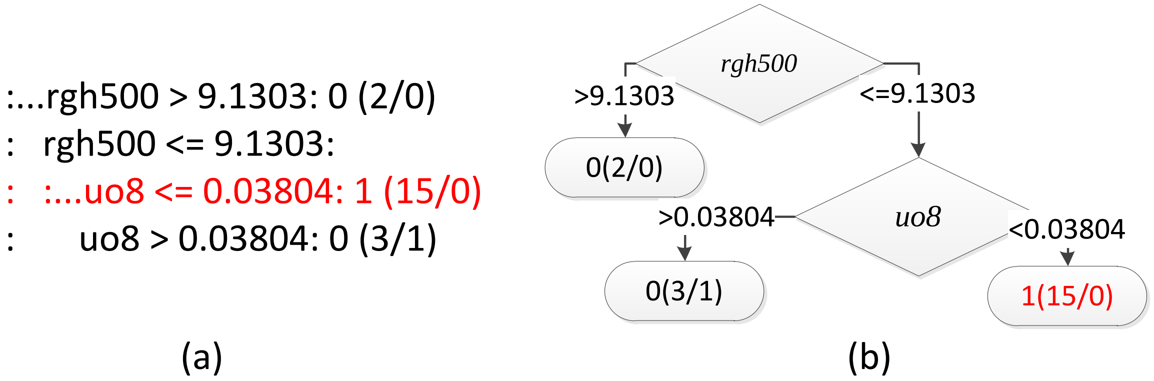
Table 1. Features and relevant behavior

G_1: Viewshed from the E location (window radius 100-350, 350, 500, 1K meters)	
Visibility Index (4)	A larger viewshed exposes a target to attackers for a longer period.
Shape Complexity (4)	A complex visible region is more difficult to assess and defend.
G_2: Sparse viewshed from the E location, # of directions (4,8,16,32,64)	
Local openness (5)	More open areas tend to offer the attacker a larger Emplacement while the target has less cover.
Distance to invisible region (min/mean/max) (15)	Nearby invisible regions offer cover for an attacker; Far invisible regions are good for Monitor functions
Planimetric area (5)	Sparse viewshed version of visibility index.
Rugosity (5)	Sparse viewshed version of roughness within the visible range.
Shape Complexity 3D (5)	Shape complexity based on sparse viewshed.
G_3: General Terrain Features (window radius 50, 100, 350, 500,1000 meters)	
Elevation (1), Slope (1), Convexity(1) and Texture(1), Elevation range (5), Roughness (5)	
G_4: Population related features (threshold of 1, 1K, 10K, 50K, 100K people)	
Minimum distance to the city of at least certain size (5)	Populated areas offer support for attackers.

With the exception of feature set G_4, all of these features are derived from estimates of elevation within the DEM. This reliance on the DEM alone potentially injects errors into the features. For example, intervisibility calculations only consider elevation change in their calculation. However, vegetation and buildings may significantly shorten actual sight lines. Map resolution is another problem. The DEM provides elevations at 30 meter intervals (~30X~30 meter pixels or 900 meters^2). However, this resolution probably masks characteristics of the terrain that are significant to an attacker. (By way of comparison, two standard basketball courts would fit into a single DEM pixel at this resolution.) This resolution may also mask significant interruptions to line of site that occupy less than one pixel. Roadside berms and smaller wadis are two examples of common features that might be too small to affect pixel-sized elevation estimates.

4. DECISION TREE TRAINING, PREDICTION, AND INTERPRETATION

The Decision Tree (DT) learning algorithm is a supervised classification system which uses a divide-and-conquer computing approach to solve complex statistical decision making problems. One common implementation of the decision tree construction algorithm is based on criteria of entropy minimization. Smaller entropy implies less mixed samples in each subgroup and therefore better abilities to separate data from different classes. The algorithm first searches for a feature f and its corresponding threshold T that can best separate data into subsets with the largest *information gain*, which is defined as the difference between the entropies of the dataset before and after the separation [14]. In the meantime, the data set D received at this new node is divided into two subsets $D \rightarrow \{D_T, D\backslash D_T\}$, where D_T represents the subset of data with value of feature f larger than T, and $D\backslash D_T$ the rest of data that do not meet the condition. The selected feature, together with its threshold are assembled into a decision node N. Both D_T and $D\backslash D_T$ may be subject to further separation to generate child node of N, based on different features and corresponding thresholds, and the process repeats iteratively until a predetermined termination condition is met. A typical rule is as follows: the subset mostly contains data of one class or the size of the subset is less than a threshold. When the descent stops and a leaf node is generated with the class label as the majority class label of the data in current subset. By traversing back from leaf to the root, we can get a rule that can be used to determine whether a sample belongs to type t.



```
:...rgh500 > 9.1303: 0 (2/0)
:   rgh500 <= 9.1303:
:   :...uo8 <= 0.03804: 1 (15/0)
:       uo8 > 0.03804: 0 (3/1)
```

Figure 1 Example of a decision tree generated by the C5.0 package

Figure 1.a shows the text output generated by C5.0 decision tree package[2] in one experiment near N34.451363 E68.7792 on Kabul-Behsud Hwy that attempts to separate attack and non-attack locations. The textual output (Figure 1.a) can be difficult to interpret, so we transform it into tree-view in Figure 1.b. In the building of this decision tree, the feature 'rgh500' with a threshold 9.1303 was found to separate the data into subsets with the largest information gain. For the subset of samples with 'rgh500>9.1303', there are two non-attack locations and no attack location. This samples in this subset are mostly non-attack locations, indicating that any sample that satisfies the condition 'rgh500>9.1303' is likely to be a non-attack location. This outcome produces a leaf node annotated with “0 (2/0)”. The text indicates the class label (“0”), the number of samples labeled with this class (“2”) and the number of samples labeled with some other class (“0”). The other subset of samples, 'rgh500<=9.1303', contained 16 attack locations and three non-attack locations. The algorithm iterates by selecting a new feature/thresholds that divide this subset into more refined subsets. The feature 'uo8' with a threshold of 0.03804 is selected. The dataset is separated into two subsets. One subset contains three non-attack locations and one attack location, and the other contains 15 attack locations. The decision tree building process stops here because there is not enough data for the first subset and there is no need to split the second subset.

When there are many features and a large data set, the decision tree can be large and hard to understand. One solution is to focus only on the leaf nodes that cover a large number of samples with low entropy (high discriminant ability). For these leaf nodes, decision rules can be generated by traversing the tree from leaf to root and using the condition for each decision node to form a conjunction clause. This rule can used to predict whether a location is suitable for attack. The conditions for the red leaf node in above tree are listed below:

- rgh500<=9.1303
- uo8<0.03804

[2] https://www.rulequest.com/see5-info.html

This decision rule consists of two conditions. The first condition requires that standard derivation of elevation of locations with 500 meters of the emplacement location to be less than 9.1303. The second condition requires the slopes from emplacement location to the first invisible locations in 8 discrete directions to be small. Taken together, these conditions suggest that the terrain near an Emplacement site tends to be moderately textured and generally flat.

5. SYSTEM DESIGN

Traditionally, a machine learning system receives the samples as a feature vector and feeds the data into pre-processing and optimization algorithms. In this process, users face several problems:

(1) The structure of the training data is not apparent.

(2) Only information contained in the feature vector is made available. Other information is lost. The example in Section 4 benefits significantly when it is anchored on a real map.

(3) They cannot understand how training data contributes to the final classifier.

(4) They cannot understand how a new sample is classified or what factors contributed to its classification.

(5) They lose the ability to adjust or tune the classifier. This negates the real world value of the experience of the users and prevents them from testing hypotheses based on intuition or inference.

Our system provides several features to overcome these problems. To overcome problems 1 and 2, DT-GIS can show the relevant events on a digital map, Google Map for these experiments. This allows users to gain context from the actual locations on the ground. Conceivably, the insights gained may identify bugs in the feature engineering or the need for new features. To overcome the problems 3 and 4, DT-GIS adopts the decision tree as its machine learning engine in order to provide human-understandable rules. The relationships between decision rules and samples can be easily visualized and explored by clicking the decision tree/rules. For problem 5, DT-GIS supports a human-in-the-loop (HITL) exploration procedure that enables the user to re-define the region of interest at each stage. This ability to selectively include or exclude regions of the map based on interest, intuition, or intelligence (ROI) provides the user with a mechanism to explore alternative hypotheses by constraining the study area.

5.1 System Overview

Figure 2 shows the architecture of our system, which consists of two main parts: a user interface and a backend data center and web service. The data center is an offline system responsible for feature generation. It is responsible for transforming raw data into feature descriptions for each location. All the raw data and extracted features are stored in a relational PostgreSQL database for fast query and retrieval. The web service integrates the database with the decision tree engine to connect the data flow for classifier generation and provide data based on geographical or feature based filters.

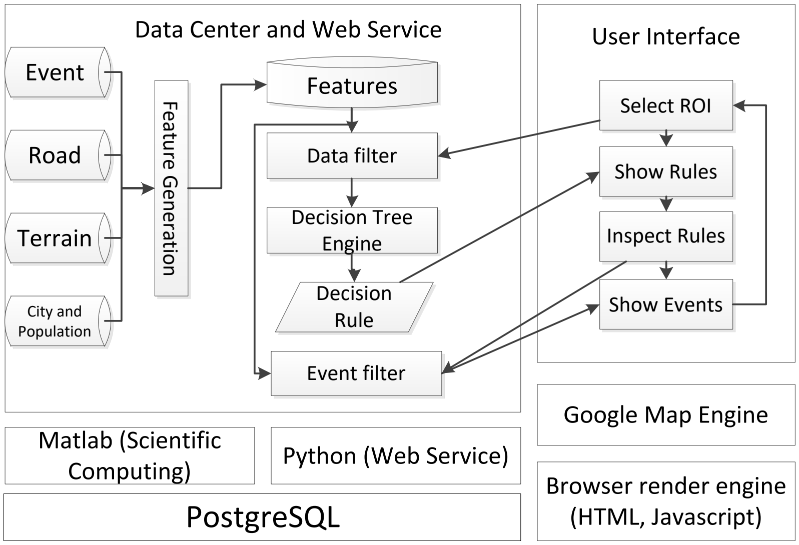
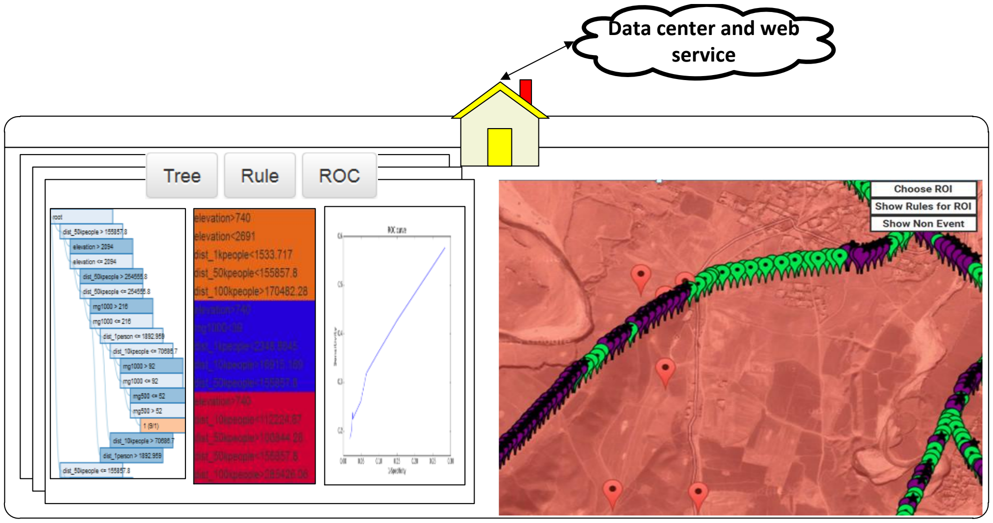


Figure 2 Architecture of iterative pattern exploration based on the decision tree

The main element of the user interface is shown in Figure 3. There are three elements in the UI. The main element is a Google map control unit that supports normal map operation such as zooming in/out and showing map/satellite view. Based on the API provided by Google Map, API v3[3], markers and polylines can be added to the map to illustrate certain information. The second control unit is on the right-top of the map. It allows the user to specify the region of interest and trigger the training of the classifier. Once trained, the user can conduct predictive analysis with the trained classifier to predict whether other potential Emplacement sites are suitable for attacks (a purple marker indicates suitable and a green marker as not suitable). The third UI element shows details of the decision tree classifier and is found on the left of the user interface. It has three panels that simultaneously show the classifier in tree-view, the classifier in rules-view, and the prediction performance in a receiver operating characteristic (ROC) graph. By clicking on the node in the decision tree or rule, the event samples satisfying them will show up as red markers.



elevation>740
elevation<2691
dist_1kpeople<1533.717
dist_50kpeople<155857.8
dist_100kpeople>170482.28
elevation>740
[illegible]
[illegible]
[illegible]
[illegible]
elevation>740
[illegible]
[illegible]
[illegible]
[illegible]

Figure 3 User Interface

5.2 DT training framework for AC location classification

Statistical machine learning is a data-driven process. Here, we use a decision tree as a supervised classification method with the framework shown in Figure 4. This approach includes data collection, down-sampling, classifier training and evaluation and follows a systematic progression. Three underlying assumptions affect the output of the DT-based framework.

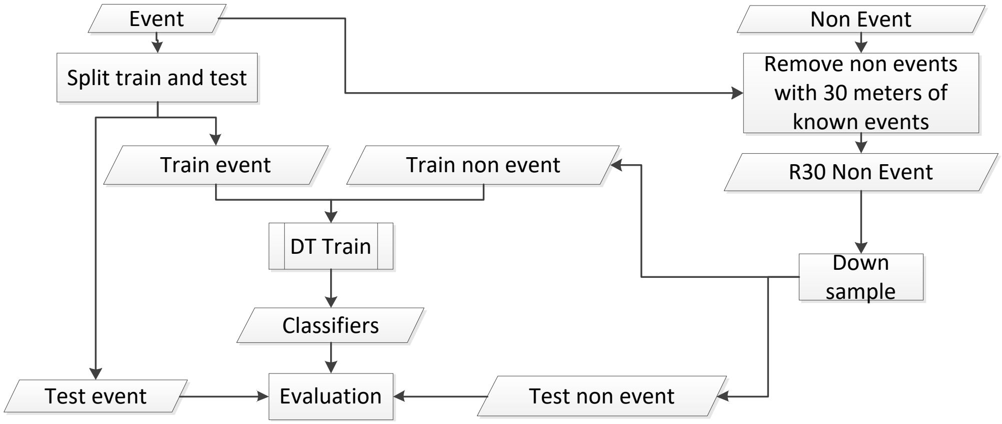


Figure 4 Decision Tree Training and Evaluation Framework

[3] https://developers.google.com/maps/documentation/javascript/

First, we note that the class labels in the training data are inaccurate, at least in the case of the non-event class. The explanation lies in the ontology of the classes we intend to identify in the data. The classifier is expected to label locations as suitable for attack (SA) or not (NSA). However, it is non-trivial to determine whether or not a location is suitable for an attack. We claim with high confidence that most locations with event history are suitable for an attack because they were previously used for an attack. But for non-event locations, there are three possibilities: (1) the location will never be suitable for an attack, (2) an attack occurred at this location but was not recorded in the available data, or (3) the location is suitable for an attack but no attack has happened yet. In this paper, we treat all locations without event history as 'not suitable for attack'. Additionally, since the resolution of the DEM used for terrain and visibility analysis is approximately 30 meters, two locations separated by less than 30 meters are treated as the same. Samples in the NSA class within 30 meters of any sample in the SA class are removed.

Second, the data available for training is very unbalanced. There are a lot more samples for the NSA class than the SA class. If we simply feed unbalanced data into the decision tree algorithm, the classifier will tend to overemphasize the NSA class. This downgrades accuracy. However, when we downsample the NSA class, the quality of the decision rules will also be affected. In this work, we set the ratio of NSA samples to SA samples as 10 unless otherwise specified. Further theoretical research and experiments are required to analyze the effects of the non-event/event ratio.

Finally, a test set must be identified to evaluate the classifier. For the test dataset, we do not need to down-sample the non-event samples since more confident evaluation results could be generated based on a larger test dataset. However, for computation concerns, we downsample the non-event class set to be the same size as the event class set in the test data set. In order to ensure stability, each experiment is repeated ten times. Random down-sampling of the non-event data is repeated in each experiment.

6. EXPERIMENT AND PERFORMANCE EVALUATION

6.1 Dataset and evaluation criteria

In order to validate our system and the decision tree algorithms, we use a data set consisting of direct fire (DF) attacks provided by ISAF-NATO Regional Command South Civilian Integration Team. There are 16610 DF samples collected in Afghanistan between February 1, 2011 and August 23, 2012. Non-event data samples are randomly sampled from road data, which is collected and maintained by the Afghanistan Information Management Service. There is a huge unbalance between the non-event samples and DF attacks (3,413,240 : 16,610), and we have not imported non-event locations that are away from roads. Given this imbalance, we adopt the ROC to evaluate the performance as proposed in [15]. The ROC curve [16] illustrates the performance of binary classifiers system under different parameter settings by plotting the true positive rate (sensitivity) against the false positive rate (1-specificity). In an ideal circumstance, both sensitivity (true positive) and specificity (false negative) are maximized. But in the real situation, increases in sensitivity occur at the cost of a decrease in specificity. The final discriminant threshold should be decided based on the final utility function with different weights for sensitivity and specificity.

6.2 ROC Performance

We show prediction performance along the AH1 highway using Kandahar-Ghazni-Kabul as an example. The results are achieved by using different NSA:SA ratios as shown in Figure 5 and Table 2. From the table, we can find that the sensitivity decrease with the ratio and the specificity increase with the ratio. The largest sensitivity is about 0.576, which means there are more than 40% of attack locations are not recognized as high-risk locations. The low sensitivity might be due to the shift of pattern. Specificity is relatively high and increases with the NSA:SA ratio. This is evidence that the machine learning algorithm overemphasizes classes with a larger sample size. There is a trade-off between the sensitivity and specificity. If the sensitivity is low, many attacks will not be detected; if the specificity is low, many false alarms will show and cost lots of resources. The recommended solution is to identify a trade-off that balances these two impacts.

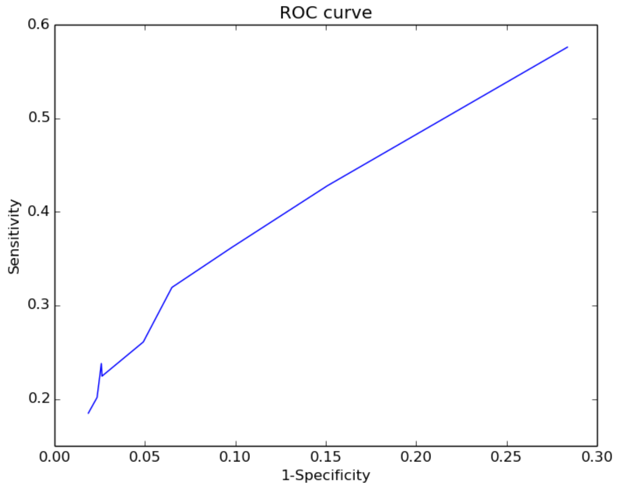


Figure 5 ROC for Kandahar-Ghazni-Kabul Hwy

Table 2 The detail values for ROC curve

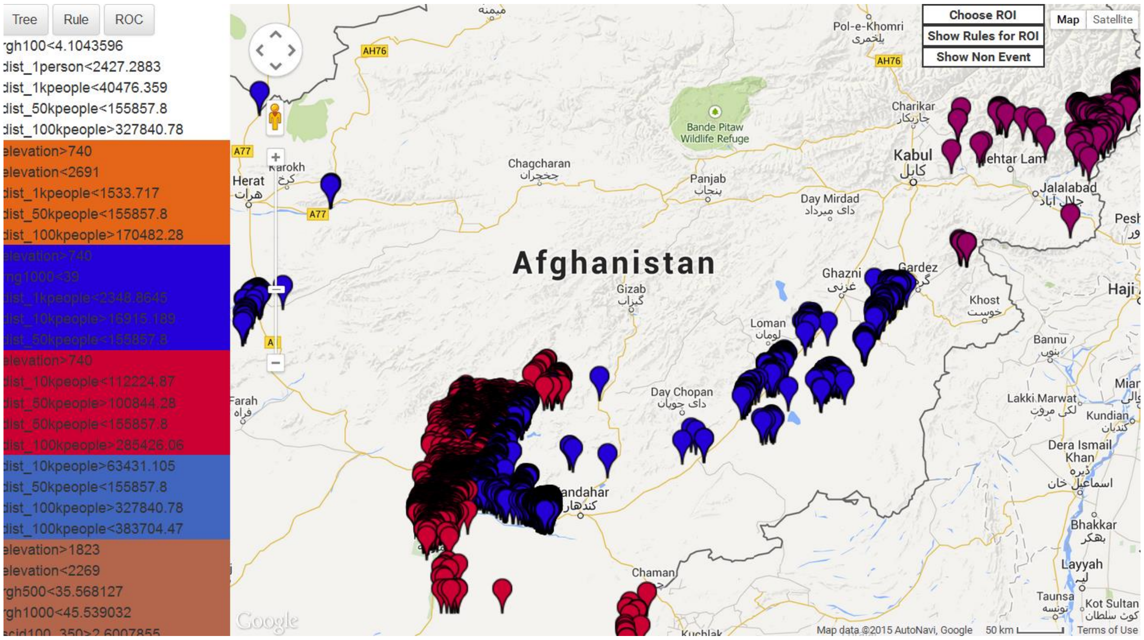
Ratio	1	2	3	4	5	6	7	8	9
Sen	0.576	0.428	0.362	0.319	0.261	0.224	0.238	0.202	0.185
1-Spc	0.284	0.151	0.098	0.065	0.049	0.026	0.026	0.024	0.019

6.3 Interpretability Explanation

In this section, we present a case that illustrates the human-in-the-loop (HITL) exploration process. A decision tree is first built using data from all of Afghanistan. By inspecting the resulting rules, we find that most rules are only indicator of high-risk regions and fail to provide useful tactical-level information. The analyst proceeds to choose an ROI that covers samples for one rule in the decision tree for the all of Afghanistan. More tactically relevant rules are mined out. At last, we summarize the limitations of this existing solution as observed during the exploration process.

6.3.1 The decisions rule for the whole Afghanistan

In this study, we first train a decision tree classifier trained with samples from all of Afghanistan. Figure 6 and Figure 7 show the rule-view and tree-view of the decision tree. The most dominant nodes in the decision tree are the population features that describe the distance to the nearest city with a population of a certain size. The samples that satisfy each rule concentrated in a region or several regions as shown in Figure 6. The events satisfying rule with red background mainly concentrate in the west of Kandahar. The events satisfying rule with a purple background mainly concentrate in MehtarLam, east of Kabul. The events satisfying rule with a blue background corresponds to segmented regions, but mainly on Highway AH1.



rgh100<4.1043596
dist_1person<2427.2883
dist_1kpeople<40476.359
dist_50kpeople<155857.8
dist_100kpeople>327840.78
elevation>740
elevation<2691
dist_1kpeople<1533.717
dist_50kpeople<155857.8
dist_100kpeople>170482.28
elevation>740
rng1000<39
dist_1kpeople<2348.8645
dist_10kpeople>16915.189
dist_50kpeople<155857.8
elevation>740
dist_10kpeople<112224.87
dist_50kpeople>100844.28
dist_50kpeople<155857.8
dist_100kpeople>285426.06
dist_10kpeople>63431.105
dist_50kpeople<155857.8
dist_100kpeople>327840.78
dist_100kpeople<383704.47
elevation>1823
elevation<2269
rgh500<35.568127
rgh1000<45.539032
scid100_350>2.6007855

Figure 6 User Interface and Rule View

Another example is the decision rule DR_{may}, which corresponds to the leaf node marked with a red star★ in the left column of Figure 7, which marks a cluster of attacks located 15 km north of Maymana, capital of Faryab province. From this case study, we can draw some qualitative summary on the nature of attacks based on the DT-GIS analytics in Table 3, where the conditions for this particular decision rule are listed in the descending order of their significance:

Table 3 Conditions and tactical interpretation of decision rule DR_{may}

Rule conditions (quantitative)	Tactical interpretation (qualitative)
dist_50kpeople > 155857.8	away from larger cities
elevation <= 2094	relatively low elevation grounds
dist_50kpeople <= 254555.8	not too far from larger cities
rng1000 <= 92	maximum elevation change within 1 km < 92 meters
dist_1person <= 1892.959	less than 2 km from the nearest populated area
dist_10kpeople <= 70686.7	less than 70 km from the nearest small town
rng500 > 52	maximum elevation change within 500 m >52 meters

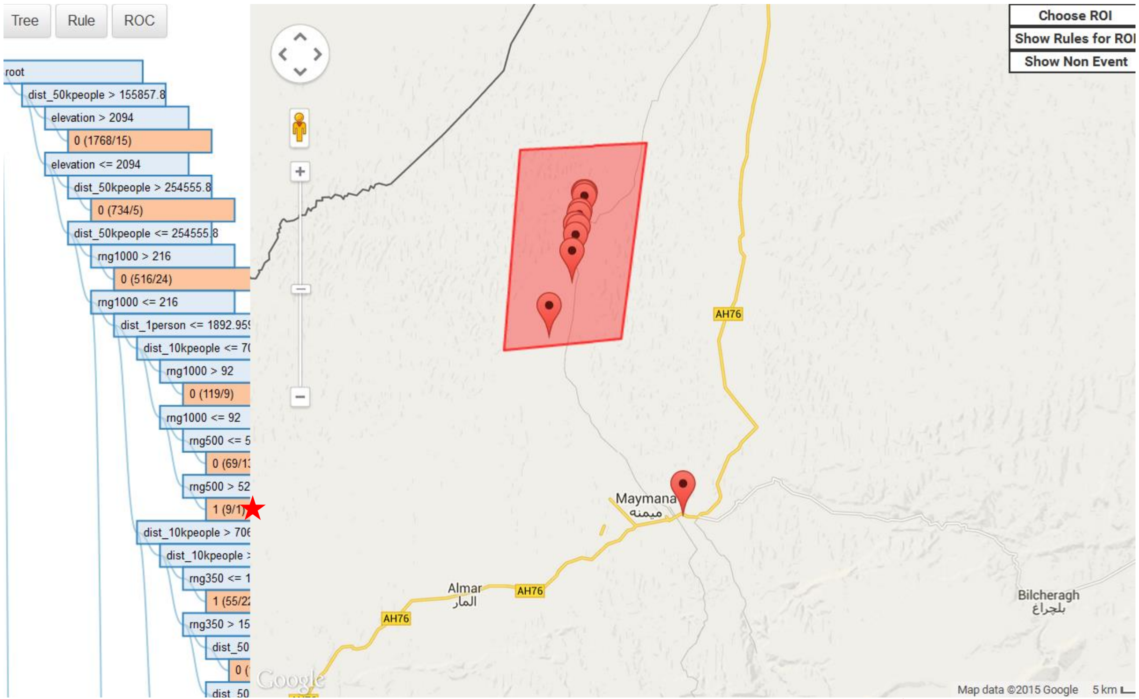


Figure 7 User Interface and Decision Tree View for the decision rule DR_{may}, which is marked as★the left column, and events satisfying rule DR_{may} are shown as markers in the right side map.

The rank of the conditions is equal to the depth of the corresponding decision node on the decision tree. From the mining outcome shown in Table x, the top three most relevant conditions are only related to geographic information (distances to populations and elevation), but it offers little tactical information. The main reason is that the non-attack locations are randomly distributed along roads across the whole map, i.e., Afghanistan, while the attack locations tend to concentrate. To overcome this deficiency, we can specify a *region of interest* (ROI) to reduce the effect of unrelated regions in selected study area through an interactive, human in the loop process as follows.

6.3.2 Rules Mining based on samples for one rule

Enabled by the HITL interaction design, an analyst can specify the ROI as the region north of Maymana (highlighted on google map in Figure 7) after clicking 'Choose ROI' in the right top panel. Then we can get a major rule marked with a red star on the decision tree in Figure 8. Unlike the rules generated from the global data set (all of Afghanistan), the rule set for this region emphasize visibility and terrain features over population-related features. Condition 'longrad16 <= 494.2' means that the attacker tends to select areas where the longest sight line from a potential Emplacement is less than 500 meters. Conditions 'rgh100 <= 11.374' and 'rgh500 > 6.0728' means the roughness of the terrain surrounding a potential Emplacement is relatively high but not extreme. Roughness is an indicator of texture. A value of zero indicates a perfectly flat surface and the increasing value indicate terrain that is increasing uneven, rough or rugged. To perform predictive analysis, the analyst can apply this rule to candidate locations. The red markers representing events satisfying the rule in the selected ROI will show up after clicking on the leaf node marked by a red star. By clicking the 'Show Non Event' in the panel on right top corner, non-attack locations will also show up. The purple marks with stars are NSA (non-attack) locations that should be considered as high risk because they satisfy all rules in the rule set learned from historical data. The green locations have a lower risk of being attack, at least using tactics similar to those in the historical event set. This function is useful for visualizing high-risk areas and providing support information for other tasks.

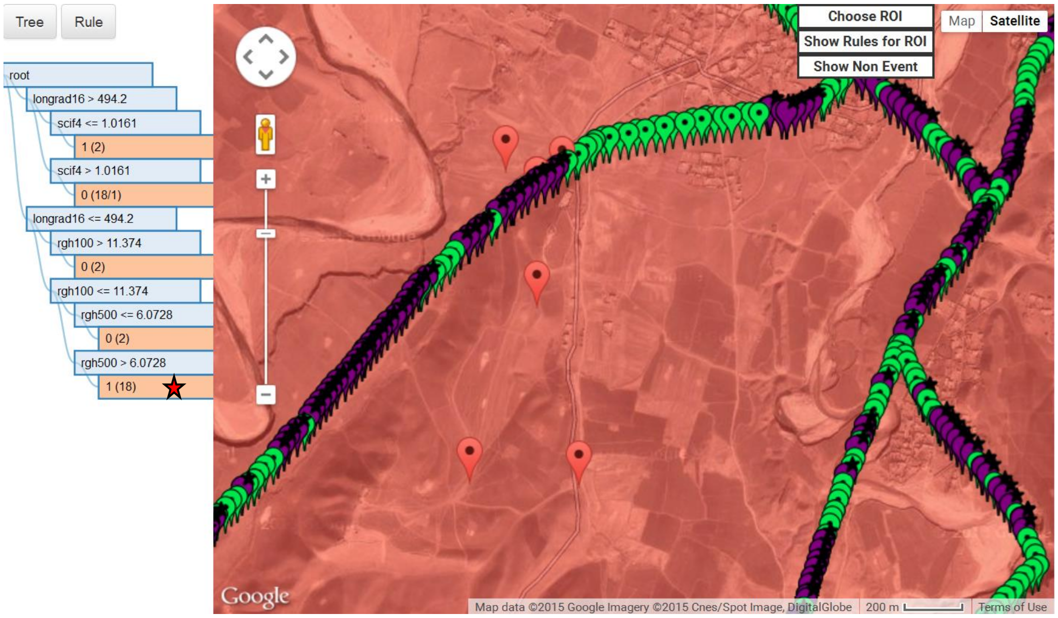


Figure 8 Refined analysis by constraining ROI

6.3.3 *DT-GIS System as a system debugging and feature engineering tool*

By going through this process using a map as external information, we discovered limitations of our workflow, especially in the areas of data collection, feature design and power of prediction.

- As shown in Figure 8, we find that our roadway map is incomplete. There are attack locations that occurred on clearly visible roads that are not in our roads database. This discrepancy may skew sample collection and performance evaluation. A possible reflection of this is the high number of road locations predicted as attack locations. It is worthwhile to explore the difference between real event locations and possible false alarms.
- As shown in Figure 9, the visibility analysis based on DEM may be invalid. For an attack in towns or cities, the visibility analysis will be affected because of missing of building information. In essence, a building acts as a disruption to line-of-sight. However, our resolution is only 30 meters, which is much larger than normal buildings. Similar disruptions occur in forested land, places where crops or undergrowth obstructs the view, and rugged terrain where berms, hills or washes are smaller that the DEM resolution.

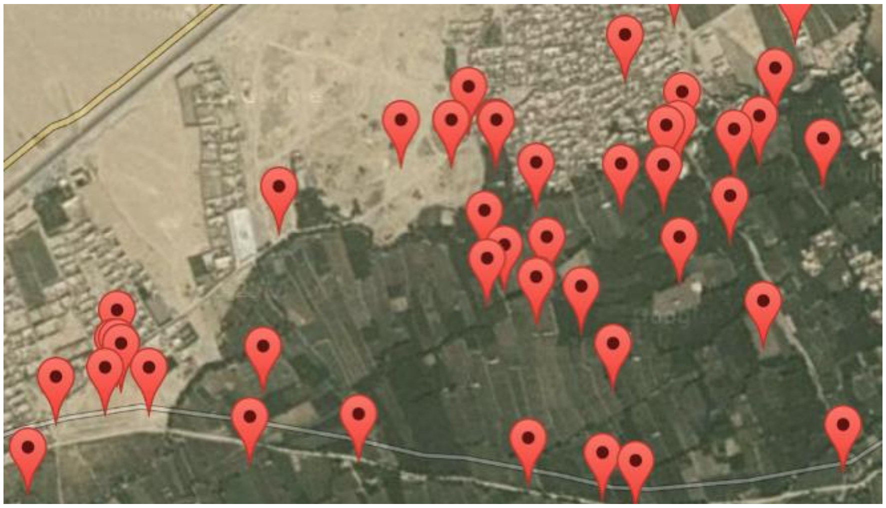


Figure 9 Attacks in populated and vegetation area

In order to overcome these limitations, our system could benefit from new information source. For NSA sample collection, more accurate road data or roads recognized from the satellite images could help complete the non-event locations. Troop movement information might be helpful in constraining non-event locations to places where the soldiers have gone. Second, high-resolution DEM, vegetation, and building information could help improve visibility analysis. Finally, we can transform more qualitative tactics that describe the interaction between two parts into quantitative feature representations.

7. CONCLUSION

In this work, we designed the DT-GIS system with a friendly user interface to explore patterns using a DT classifier. We show that it is possible to present the classifier output as human-understandable rules rather than treating statistical machine learning as a black box. This implies the possibility of integrating SML as an assistant or analytical support tool in an ISTAR system. We also enabled rich interaction between the analyst and the system to help understand how the SML toolkit is functioning. Analysts can access the SML training process by showing rule-related events, understand the prediction process by predicting non-attack locations, and understand the importance of data collection by constraining the ROI enable by the HITL framework.

Through these exploration processes, we also discovered limitations of the system with regards to class ontology, data collection, feature transformation, and evaluation measurement. In future work, time information as scales ranging from time-of-day to daily/weekly/monthly trends should be considered to support pattern shift detection and to refine and improve attack prediction.

REFERENCES

[1] "Intelligence, surveillance, target acquisition, and reconnaissance," [Online]. Available: http://en.wikipedia.org/wiki/Intelligence,_surveillance,_target_acquisition,_and_reconnaissance. [Accessed 27 Jan 2015].

[2] S. George, X. Wang, and J. Liu, "MECH: A Model for Predictive Analysis of Human Choices in Asymmetric Conflicts," in 2015 International Social Computing, Behavioral Modeling and Prediction Conference (SBP15), Washington D.C., 2015.

[3] E. C. Guevara, Guerrilla Warfare, CreateSpace Independent Publishing Platform, 2013.

[4] S. Sewall, J. Nagl, D. Petraeus and J. Amos, The U.S. Army/Marine Corps Counterinsurgency Field Manual, University of Chicago Press, 2008.

[5] P. Shakarian, M. Nagel, B. Schuetzle and V. Subrahmanian, "Abductive inference for combat: using SCARE-S2 to find high-value targets in Afghanistan," in Twenty-Third Innovative Applications of Artificial Intelligence Conference, San Francisco, 2011.

[6] J. Eck, S. Chainey, J. Cameron and M. Leitner, Mapping Crime: Understanding Hot Spots, 2005.

[7] A. Zammit-Mangion, M. Dewar, V. Kadirkamanathan and G. Sanguinetti, "Point Process modelling of the Afghan War Diary," in PNAS, 2012.

[8] T. Hastie, R. Tibshirani and J. Friedman, The elements of statistical learning: Data mining, Inference, and Prediction, Springer Series in Statistics, 2009.

[9] X. Wang, S. George, J. Lin, and J. Liu, "Quantifying Tactical Risk: A Framework for statistical classification using MECH," in 2015 International Social Computing, Behavioral Modeling and Prediction Conference (SBP15), Washington D.C., 2015.

[10] N. V. Chawla, "C4.5 and imbalanced data sets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure," in ICML'03 Workshop on Class Imbalances, 2003.

[11] N. V. Chawla, N. Japkowicz and A. Kotcz, "Editorial: special issue on learning from imbalanced data sets," ACM SIGKDD Explorations Newsletter - Special issue on learning from imbalanced datasets, pp. 1-6, June 2004.

[12] R. Yokoyama, M. Shirasawa and R. Pike, "Visualizing topography by openness: A new application of image processing to digital elevation models," Photogrammetric Engineering and Remote Sensing, pp. 251-266, 2002.

[13] T. Hengl and H. Reuter, Geomorphometry: Concepts, Software, Applications, Elsevier, 2009.

[14] J. Quinlan, "Induction of Decision Tree," Machine Learning, pp. 81-106, 1 3 1986.

[15] P. Foster and F. Tom, "Robust Classification for Imprecise Environments," Machine Learning, pp. 203-231, March 2001.

[16] T. Fawcett, "An introduction to ROC analysis," Pattern Recognition Letters, pp. 861-874, June 2006.