Markov Logic Networks for context integration and situation assessment in maritime domain

Lauro Snidaro*, Ingrid Visentini†, Karna Bryan† and Gian Luca Foresti*
* University of Udine, Department of Mathematics and Computer Science
Via Margreth 3, 3100 Udine, Italy
Email: lauro.snidaro@uniud.it
† NATO Undersea Research Centre
Viale San Bartolomeo 400, 19126 La Spezia, Italy
Email: visentini@nurc.nato.int

***Abstract*—The detection of anomalies has a critical role in situational assessment. In this paper, we break down the concept of anomaly in the maritime domain into different levels and relate them to the JDL fusion model. We also show how uncertain context knowledge can be encoded through Markov Logic Networks (MLNs) which offer a convenient framework leveraging both the expressive power of first order logic and the probabilistic uncertainty management of Markov networks. Every formula in the knowledge base is assigned a weight indicating its confidence. Different types of knowledge with associated uncertainty can therefore be fused together within MLNs and on-line inference can be performed as input data is processed by the system, and the formulas are grounded in the knowledge base. Promising examples are demonstrated on a sample set of rules for maritime event and anomaly detection.**

I. Introduction

Maritime event recognition, threat assessment and, generally, situational awareness (SA) have recently become hot topics in the research community, considering the increasing interest for a timely, updated, comprehensive and accurate situation picture for threat detection, crime prevention, security of citizens, anti-terrorism countermeasures, as well as disaster relief management.

Up to now, low-level sensor-based data is the main source of information to understand the evolving scenario and to raise an alarm; in particular, maritime surveillance relies on Automatic Identification System (AIS), coastal radars, Synthetic-Aperture Radar (SAR) imagery, and other sensors, to form a picture on which the operator can recognize complex patterns and make decisions [1], [2]. Low-level context injection approaches can be found in the literature, as in [3], where low-level contextual information is encoded for improving tracking performance, High-level information, in the form of context or domain expert knowledge, is instead poorly integrated, if not absent, in current fusion systems for SA, even if the richness and completeness of this contextual information is extremely useful to properly interpret the available stream of raw sensor data. In fact, the main goal of a reasoning engine or probabilistic inference system is to associate a posterior probability distribution to a set of queries [4], given some observed evidence. The incorporation of abductive/inductive and deductive inferencing is a vital element in an automated fusion system, and it represents a fundamental base for situational awareness. How this involvement can be obtained, on both theoretical and applicative levels, is a crucial point, and is subject of ongoing research. Expert systems were one of the first knowledge representation tools [5]. Though simple, intuitive and easy to code, they present many shortcomings, mainly because they are rigid structures that do not handle uncertainty.

Dealing with uncertainty is one of the most desirable characteristics for a fusion system, as uncertain data affects decisions and the quality of the estimates [6]. Probability theory provides a way to overcome and represent the uncertainty introduced from ignorance of the observed world; on this side, Bayesian Networks (BN) and Hidden Markov Models (HMM) are extensively used in surveillance domain (see [7] for a recent survey). In [8] BN have been used for assesing the threat probability obtained by the combination of five types of anomalies or abnormal behaviours in a maritime scenario. Despite being so largely used, they have strong disadvantages, including the fact that they allow reasoning about the same fixed number of attributes, as their nature is essentially propositional: the set of random variables is fixed and finite, and each has a limited domain [4]. As a result, their application to complex problems lacks flexibility. Ontologies are another popular means to encode knowledge and represent relationship among entities [9], [10]. They are effective tools for representing taxonomies and relations but not rule-knowledge. Anomaly detectors or event recognition systems are presented in [11], [12], [1], [13], [8], but no uncertainty modelling is explicitly provided.

A much more powerful tool is offered by first-order logic (FOL), which, in contrast to propositional logic, is expressive enough to represent complex environment in a concise way. A combination between FOL and graphical models (Markov Networks), that fuses statistical and logical reasoning, are the Markov Logic Networks (MLN) [14]. MLN are a new promising technique that are constituted of a knowledge base of first-order formulas with associated weights. The main idea is that, if the knowledge base is violated by an interpretation (which provides grounding to the formulas and assigns truth values to the predicates), then that *world* is less probable, but not impossible. Already applied to video surveillance

systems for event detection [15], we show their potential as a powerful fusion tool for SA systems which can be effectively used to combine observations coming from multiple heterogeneous (hard and soft [16]) sources of information, integrate contextual and a priori knowledge in a maritime scenario.

II. Anomalies, events and JDL levels

State-of-the-art situation assessment systems (e.g. an automatic surveillance system [17]) are able to deal with vast amounts of data and information also of a heterogeneous kind. Their goal is to provide a constantly updated situational picture about the observed environment or set of entities to an operator in order to facilitate human decision making. Updating the current system representation of the situation is generally performed by acquiring, through sensors or other sources of information, new observations which provide a possibly incomplete and uncertain view. More specifically, following the JDL model [18] terminology, an observation could be more or less refined, or processed, and could thus provide input to the system at different levels. Very briefly, data from raw signals from sensors are considered Level 0, features (e.g. position, class, identity, etc.) of detected targets are Level 1, relations among entities are considered Level 2, which eventually conveys the state of the observed environment: actions, interactions and intentions of detected entities. Level 3 deals with the future impacts that may result from the present situation as depicted by Level 2. Level 4 is dedicated to the fusion process itself and the ways to optimize the performance of the system according to mission goals. In the following, the term *level* will be used as per this JDL terminology.

A. Events

Building a situational picture requires a system to be able to assess the current state of the observed environment. More specifically for security purposes, the system should be able to detect and recognize events. An event can be considered a "significant occurrence" and can be subdivided in *simple* and *complex*. Simple events can be considered as the variation of a quantity or state, while complex events are made of component events. An event modelling framework is presented in [19], where a counterpiracy example is presented with the intent of facilitating the decision making process, with a graphical representation of events, but no automated reasoner is associated with this representation.

1) Simple Events: A *simple event*, also called primitive or atomic in the literature, is any significant variation of input data, at any level, discernible by the system. Therefore, variations of input signals (Level 0), of a target's state (e.g. speed, direction, etc. , Level 1), or of a target's relation with other entities (Level 2), are all examples of simple events. They can be directly observable or not (e.g. inferable from other indicators), but the common property is that they cannot be further decomposed into simpler constituting events. Within the framework described in Section III, a simple event will be described by a predicate.

2) Complex Events: Complex events are composed of a combination of two or more component events (simple or complex) that can be arbitrarily combined through logic operators ($\wedge$, $\vee$, $\neg$) according to domain knowledge. Complex events can also be triggered by a specific time-ordered sequence of component events, or be just an unordered collection depending on contextual information. It should be noted here that a complex event can be composed of a heterogeneous combination of events generated by data at different levels. This is further discussed in Section II-E. In this work, complex events will again be described by predicates, the difference being that they cannot be directly observable and should therefore only be inferred from component events.

B. Anomalies

Input data, whichever its nature, can be expected to assume certain values e.g. by taking into account past values up to current time t and predicting next ones at time $t + 1$ as done by filtering algorithms. This holds for both numerical and non-numerical data, such as labels. Even considering fluctuations due to process or measurement noise, whenever the expected input falls beyond a certain *threshold* then it can be considered unexpected or anomalous thus raising an exception. An example could be a vehicle or vessel detected exceeding a certain speed or the position of a target crossing into a forbidden area. Thresholds, provided by domain experts or learned automatically by the system from data, are therefore used to immediately spot an anomalous condition. However, anomalies provide no notion whatsoever on the meaning of the exceptional input. Just like in programming languages, whenever an exception is thrown, it should be handled properly. In a Situation Assessment system, the knowledge base is consulted to infer a possible conclusion from the anomalous condition.

Anomalies can therefore be considered critical events to which the system is required to respond. In the specific case of situation assessment, anomaly or abnormal event detection is the primary goal of the system. A significant taxonomy of anomalies in maritime domain can be found in [11].

C. Explicit event modelling

Another distinction can be made on the way events are modelled in the system: explicitly or implicitly. In the former case the system has a complete description of what an anomalous event is. The detection of such patterns can identify a potentially dangerous situation that the system should try to identify for prevention, or at least reaction. Events are defined explicitly in the sense that they are encoded directly in the system exploiting expert and contextual knowledge. Historically, explicit modelling is used in expert systems [20] where the knowledge base assumes the form of a set of rules that fire upon the realization of the preconditions.

A common issue associated with this approach is that it can detect only known patterns. In other words, the operator has to manually specify all the events of interest or anomalies. Another aspect is the lack of flexibility in defining the knowledge base. Rules are generally imposed as hard

constraints: the preconditions have to be fully satisfied for the rules to be activated. It should be noted that this problem is not mitigated by the introduction of fuzzy definitions of the preconditions. Fuzzy sets allow to convert numerical input to labels corresponding to data intervals. This does not really account for a proper encoding of a degree of uncertainty in the rules as instead will be the case for the framework adopted here (Section III).

D. Implicit event modelling

Within this event modelling paradigm patterns of activities are learned automatically by the system in order to detect the most common (and therefore hopefully "normal") ones. This is therefore an unsupervised approach where an anomaly is defined as any behaviour differing from learned models and that automatically detected as a deviation from common patterns of activity. Therefore, while explicit models directly encode expert knowledge and anomalies in the knowledge base, here the system automatically builds and maintains models of "normality".

This approach, generally implemented by machine learning techniques, has the advantage of not requiring an extensive manual definition of all possible anomalous conditions as in the previous case. It is also adaptable to varying patterns in the observed environments thus allowing a continuous update of the models through learning algorithms. The downside is that no expert or contextual knowledge can be directly injected into the system. This is particularly true in the case of complex events. While relations between simple events can be learned, there is no way of specifying a well-known anomaly to domain experts without combining this approach with explicit modelling.

E. Levels and events

"High (low)-level events" or "High (low)-level anomalies" are something often informally discussed in the scientific community, but, to our knowledge, never really formalized. This is particularly true in communities (e.g. video surveillance) other than Fusion, where a reference data or process model is lacking. Following the description given in the above sections taking into account JDL levels, our position here is that whenever the system detects any appreciable variation of input data at any level, a corresponding event is generated.

It is not true however, that this event must be flagged as an anomalous situation, and must necessarily be transmitted up through the levels following increasing processing and refinement steps.

III. Markov Logic Networks

We here provide essential background notions of Markov Logic Networks, but the reader is advised to refer to [14] for further details. Markov Logic Networks (MLN) are a powerful tool for combining logical and probabilistic reasoning. While a knowledge base (KB) of logic formulas can be satisfied only by those worlds (truth values of atomic formulas) in which it is true, a MLN relaxes this hard constraint by associating a probability value to the worlds that do not fully satisfy the KB. Therefore, the fewer formulas a given world violates the more probable it is [14].

An MLN is then a set L of pairs (F_i, w_i) where F_i is a first order logic formula and w_i its corresponding real-valued weight. The set of all formulas F_i in L constitutes the KB while the weight w_i associated to each F_i reflects how strongly the constraint imposed by the formula is to be respected. This directly impacts the probability assignment: worlds which satisfy a high weight formula are going to be much more probable than those that do not.

A Markov Logic Network L together with a finite set of constants C defines a Markov network $M_{L,C}$ that models the joint distribution of the set of random (binary) variables $X = (X_1, X_2, ..., X_n) \in \mathcal{X}$. Each variable of X is a ground atom (predicate whose arguments contain no variables) and $\mathcal{X}$ is the set of all possible *worlds*, that is the set of all possible truth value assignments of n binary variables. Clearly, $|\mathcal{X}| = 2^n$ where $|.|$ is the cardinality operator. The network is built as follows:

- $M_{L,C}$ contains one (binary) node for each possible ground atom given L and C
- An edge between two nodes indicates that the corresponding ground atoms appear together in at least one grounding of one formula in L. Ground atoms belonging to the same formula are connected to each other thus forming cliques.
- A feature f_i is associated for each possible grounding of a formula F_i in L. Each f_i assumes value 1 if the corresponding ground formula is true and 0 otherwise.

The probability distribution over X taking values $x \in \mathcal{X}$ specified by $M_{L,C}$ is given by:

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_{i=1}^{|L|} w_i n_i(x)\right) \tag{1}$$

where $|L|$ indicates the cardinality of L, thus counting the number of formulas of the knowledge base, and $n_i(x)$ is the number of true groundings of F_i in the world x.

$$Z = \sum_{x' \in \mathcal{X}} \exp\left(\sum_{i=1}^{|L|} w_i n_i(x')\right) \tag{2}$$

is a normalizing factor often called *partition function*.

According to the definitions given in Section II, a MLN provides an explicit way of encoding knowledge. However, both rule weights and the rules themselves can be be learned from data [14]. These capabilities make MLNs a powerful tool that combines the benefits of both implicit and explicit modelling.

IV. Knowledge representation

The creation of a knowledge base (KB) implies the use of a representation formalism to capture and code the Subject Matter Expert's (SME) knowledge into formulas.

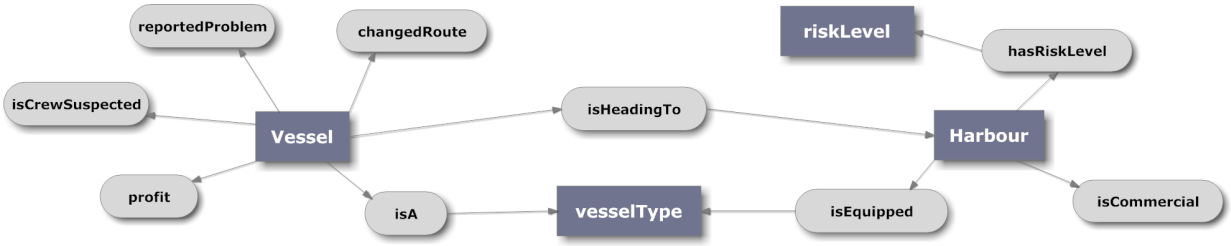

Fig. 1. Entities and relations of the proposed maritime example scenario.

A. Maritime scenario

We start defining a knowledge base that will model our domain, with entities and their relationship to depict a very simple scenario in which a cargo ship heads toward a harbour. A general ontology of the example is illustrated in Figure 1.

We create a predicate $isA(v, vtype)$ for describing that a certain vessel ($vessel(v)$) is of type $vtype = \{Cargo, FishingBoat, Tanker, Pleasure\}$. We introduce also the predicate that states that cargo ships are motivated by $profit(v)$. Another predicate tells us that the vessel is heading ($isHeadingTo(v, h)$) toward a harbour h ($harbour(h)$), that can be commercial or not ($isCommercial(h)$) and capable or not to handle a particular type of cargo ($isEquipped(h, vtype)$).

A simple low-level anomaly detector cannot classify a cargo entering in a small harbour as an anomalous event. However, additional information provided by context can help to raise a flag. Context can be represented by other facts such that

- The ship has changed direction without any apparent motivation. It is known that fishing vessels can change the estimated port of arrival, as they may want to sell their goods for a better deal in another location. A cargo, as well, can have many reasons for which the new destination can be appropriate, from an engine failure to environmental or geopolitical conditions, from better prices for the goods is carrying to human factors. A deviation from a routine behaviour can be a response to external factors and not necessarily a threat indicator.
- The ship is heading toward a port that can not handle its content, that is not classified as commercial harbour and does not have the appropriate equipment to handle a cargo. As a cargo ship is motivated by profit, this situation raises an anomaly. This data is provided *a priori*, as it describes the port and its facilities.
- Members of the crew have a criminal record or are suspected.
- The harbour is classified as sensitive target or high-risk zone, as it is a restricted area, or it is built, for instance, close to a chemical plant or to sensitive objectives.

The sensitivity of the harbour is defined as $riskLevel(h, hlevel)$, while other predicates define the fact that the ship changed route ($changedRoute(v)$) and thus destination port, maybe because of a reported problem, as an engine failure, ($reportedProblem(v)$), or hosts a suspected crew member ($isCrewSuspected(v)$).

The domain knowledge is specified in Table I, where the higher the weight the more confident the statement. The weights are expressed as fractions of the maximum weight ω which expresses a hard constraint [15]. A subject matter expert can help hand coding the rules that are reasonable for the domain. We state that a cargo ship or a fishing boat are motivated by profit (2.), that implies that the vessel heads toward a harbour equipped to handle its content (3.). If the harbour is not commercial, is must not be equipped with facilities to handle cargo ships or fishing boats (5.). As general rules, a suspected crew member, a sudden change of destination or a reported issue alone are mild anomalies that alert the operator to pay attention to the ship (6.). If combined (7.), the probability of an alarm rises. If a ship is heading toward a commercial harbour, it is because of profit (8.). Anomalies occur when a cargo ship is heading toward a non commercial port (9.), or a cargo has changed route, even if the crew is not suspicious (10.). From theses rules, we can infer that if a harbour is considered commercial, it is equipped to handle either cargos and fishing boats and if a ship is heading toward it, the ship is motivated by profit with strong confidence. An alarm implies (with low confidence) that the ship will not make profit (13.) Other complex events refer to a cargo ship heading toward a high-risk harbour with a crew member that is not clear (11.), or toward a harbour that is not properly equipped (12.).

V. Grounding and Results

This section aims to describe contextual and observed evidences that are used to ground the network, and to provide an example to clarify the Markov Logic Networks application to a maritime domain. Also, we want to demonstrate that MLN work in presence of *partial* evidence as well.

A. Contextual information

Some contextual information, for instance the type of harbour (commercial or passengers), its risk level (high, medium or low), or the fact that it is equipped to handle the content of a cargo ship, must be provided as a priori information. In general, a static entity and the associated resources or characteristics can be described a priori by a human operator, and this knowledge can be updated in time when some of these features may vary. On the contrary, evidence about moving or non-static objects is created on-the-fly, and it is not permanent

TABLE I
KNOWLEDGE BASE IN FOL WITH ASSOCIATED WEIGHTS

Rule	Weight
1. $isA(v, Cargo) \wedge isHeadingTo(v, h) \Rightarrow harbour(h)$	$1/5\ \omega$
2. $isA(v, Cargo) \vee isA(v, FishingBoat) \Rightarrow profit(v)$	$4/5\ \omega$
3. $isA(v, vtype) \Rightarrow isHeadingTo(v, h) \wedge isEquipped(h, vtype) \wedge profit(v)$	$4/5\ \omega$
4. $isEquipped(h, Cargo) \Rightarrow harbour(h)$	ω
5. $\neg isCommercial(h) \Rightarrow \neg isEquipped(h, Cargo) \wedge \neg isEquipped(h, FishingBoat)$	$4/5\ \omega$
6. $isCrewSuspected(v) \vee changedRoute(v) \vee reportedProblem(v) \Rightarrow alarm(v)$	$4/5\ \omega$
7. $isCrewSuspected(v) \wedge changedRoute(v) \wedge reportedProblem(v) \Rightarrow alarm(v)$	$5/6\ \omega$
8. $isCommercial(h) \wedge isHeadingTo(v, h) \Rightarrow profit(v)$	$3/5\ \omega$
9. $isA(v, Cargo) \wedge isHeadingTo(v, h) \wedge \neg isCommercial(h) \Rightarrow alarm(v)$	ω
10. $isA(v, Cargo) \wedge \neg isCrewSuspected(v) \wedge changedRoute(v) \Rightarrow alarm(v)$	$2/5\ \omega$
11. $isA(v, Cargo) \wedge isHeadingTo(v, h) \wedge riskLevel(h, High) \wedge isCrewSuspected(v) \Rightarrow alarm(v)$	$4/5\ \omega$
12. $isA(v, Cargo) \wedge isHeadingTo(v, h) \wedge \neg isEquipped(h, Cargo) \Rightarrow alarm(v)$	$4/5\ \omega$
13. $alarm(v) \Rightarrow \neg profit(v)$	$2/5\ \omega$

TABLE II
CONTEXTUAL INFORMATION PROVIDED A PRIORI

$harbour(Harbour_1)$
$riskLevel(Harbour_1, Low)$
$isEquipped(Harbour_1, Cargo)$
$isEquipped(Harbour_1, Fishing)$
$isCommercial(Harbour_1)$
$harbour(Harbour_2)$
$riskLevel(Harbour_2, High)$
$isEquipped(Harbour_2, Cargo)$
$isEquipped(Harbour_2, Fishing)$
$isCommercial(Harbour_2)$
$harbour(Harbour_3)$
$riskLevel(Harbour_3, Low)$
$\neg isEquipped(Harbour_3, vtype)$
$\neg isCommercial(Harbour_3)$
$harbour(Harbour_4)$
$riskLevel(Harbour_4, High)$
$\neg isEquipped(Harbour_4, Cargo)$
$isEquipped(Harbour_4, Fishing)$
$\neg isCommercial(Harbour_4)$

as it can vary over time. For this reason, we must distinguish between *given evidence*, i.e. contextual (static) information, and *observed evidence* that refers to a specific vessel of interest in a certain instant of time.

In our examples, we specified four types of harbours with different characteristics:

- $Harbour_1$, defined as a commercial harbour, equipped to handle cargo and fishing ships.
- $Harbour_2$, defined as a commercial harbour which is classified as high risk, as there is a chemical plant nearby.
- $Harbour_3$, described as a non commercial harbour which has low risk.
- $Harbour_4$, a passenger harbour that is a sensitive area.

It is important that this information is the most complete as possible, to depict with fidelity the scenario and its entities.

B. Observed evidence

We provide observable evidence (derived from sensory data) and ground the MLN predicates in Table III. Five entities are involved in this example:

- $Cargo_1$, a cargo ship that delivers its content to $Harbour_1$ harbour. It represents a condition of normalcy.
- $Cargo_2$, a cargo ship that heads toward $Harbour_4$, that is a non commercial harbour.
- $Cargo_3$, a ship that hosts a suspected crew member and is going toward $Harbour_2$ harbour after a route change. $Harbour_2$ is considered a commercial harbour but classified as high-risk.
- $Cargo_4$, it is a cargo heading toward a non commercial harbour after changing route. No information on crew members is available. This is an interesting case to demonstrate how the reasoning engine works also when parts of evidence are missing.
- $Fishing_1$, that is a fishing ship that is heading toward a non commercial harbour.
- $Fishing_2$, that is a fishing vessel heading toward a commercial harbour (high-risk); this case is not classified as suspicious.

C. Results

We tested our scenario using Alchemy[1] and probCog[2]. Due to the complexity of the domain, instead of exact inference (that is computationally #P-hard) we used MCMC (Gibbs sampling) with 5000 steps. For our experiments we empirically set the knowledge base weights as in Table I. When a large database of known patterns is available, Alchemy allows us to learn the weights from data.

The results of Table IV are obtained from the queries $P(alarm(v)|M_{L,C})$ and $P(profit(v)|M_{L,C})$ representing the probability for the predicates $alarm$ and $profit$ being true for a given vessel v, where $M_{L,C}$ is the Markov Network created groundings the set formulas L shown in Table I, C is the set of constants as defined in Section V-B, and contextual and sensory evidence is provided according to Tables II and III respectively.

As expected, $Cargo_2$, $Cargo_3$ and $Fishing_2$ raised an anomaly, while $Cargo_4$ has a suspicious behaviour but with a medium-high confidence, that a human operator can interpret as a possible alert. $Cargo_1$ and $Fishing_2$ have a very low probability of violating normalcy conditions. The profit, on another side, is inversely proportional to the alarm probability,

[1] http://alchemy.cs.washington.edu/

[2] http://wwwbeetz.informatik.tu-muenchen.de/probcog-wiki/index.php

TABLE III
EVIDENCE EXTRACTED FROM SENSORY DATA

$isA(Cargo_1, Cargo)$
$isHeadingTo(Cargo_1, Harbour_1)$
$\neg isCrewSuspected(Cargo_1)$
$\neg reportedProblem(Cargo_1)$
$\neg changedRoute(Cargo_1)$
$isA(Cargo_2, Cargo)$
$isHeadingTo(Cargo_2, Harbour_4)$
$\neg isCrewSuspected(Cargo_2)$
$\neg reportedProblem(Cargo_2)$
$\neg changedRoute(Cargo_2)$
$isA(Cargo_3, Cargo)$
$isHeadingTo(Cargo_3, Harbour_2)$
$isCrewSuspected(Cargo_3)$
$\neg reportedProblem(Cargo_3)$
$changedRoute(Cargo_3)$
$isA(Cargo_4, Cargo)$
$isHeadingTo(Cargo_4, Harbour_3)$
$\neg reportedProblem(Cargo_4)$
$changedRoute(Cargo_4)$
$isA(Fishing_1, FishingBoat)$
$isHeadingTo(Fishing_1, Harbour_3)$
$\neg isCrewSuspected(Fishing_1)$
$\neg reportedProblem(Fishing_1)$
$\neg changedRoute(Fishing_1)$
$isA(Fishing_2, FishingBoat)$
$isHeadingTo(Fishing_2, Harbour_2)$
$\neg isCrewSuspected(Fishing_2)$
$\neg reportedProblem(Fishing_2)$
$\neg changedRoute(Fishing_2)$

TABLE IV
ALARM PROBABILITY ASSOCIATED TO EVIDENCE OF TABLE III

Ship	Alarm probability	Profit probability
$Cargo_1$	0.001	0.998
$Cargo_2$	0.993	0.002
$Cargo_3$	1.000	0.001
$Cargo_4$	0.733	0.261
$Fishing_1$	1.000	0.001
$Fishing_2$	0.001	1.000

as a suspicious vessel is flagged as not business-oriented when showing an anomalous behaviour.

Although being a preliminary study on the application of Markov Logic Networks in maritime domain, the results are promising and encouraging further developments on more complex or routine scenarios.

VI. CONCLUSIONS

In this paper we examined the concept of anomaly in the maritime domain from the point of view of the JDL fusion model. Events and anomalies are fundamental concepts to build a situational picture about the observed environment or set of entities to facilitate human decision making.

Furthermore we presented the Markov Logic Networks as an efficient and robust tool that leverages both the expressive power of first order logic and the probabilistic uncertainty management of Markov networks. In our example, observed (incomplete) evidence is fed into an on-line inference engine that allows reasoning under uncertainty. The set of formulas in the knowledge base is grounded with the empirical evidence, and reasoning is performed exploiting high-level contextual information. In our case, context is represented by formulas previously provided by a SME, and by a priori contextual evidence that is used to ground the MLN.

REFERENCES

[1] B. Ristic, B. La Scala, M. Morelande, and N. Gordon, "Statistical analysis of motion patterns in ais data: Anomaly detection and motion prediction," in *Proceedings of the 11th International Conference on Information Fusion*, July 2008.

[2] C. Carthel, S. Coraluppi, and P. Grignan, "Multisensor tracking and fusion for maritime surveillance," in *Proceedings of the 10th International Conference on Information Fusion*, July 2007.

[3] I. Visentini and L. Snidaro, "Integration of contextual information for tracking refinement," in *14th International Conference on Information Fusion*, Chicago, Illinois, 2011.

[4] S. J. Russell and P. Norvig:, *Artificial Intelligence - A Modern Approach (3rd ed.)*. Pearson Education, 2010, vol. I-XVIII.

[5] J. Roy, "Rule-based expert system for maritime anomaly detection," in *Proceedings of the 12th International Conference on Information Fusion*, July 2009.

[6] L. Snidaro, I. Visentini, and G. Foresti, "Fusing multiple video sensors for surveillance," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 8, no. 1, pp. 7:1–7:18, Feb. 2012.

[7] A. Patcha and J. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Comput. Netw.*, vol. 51, pp. 3448–3470, August 2007.

[8] R. Lane, D. Nevell, S. Hayward, and T. Beaney, "Maritime anomaly detection and threat assessment," in *Proceedings of the 13th International Conference on Information Fusion*, july 2010.

[9] J. Garcia, J. Gomez-Romero, M. Patricio, J. Molina, and G. Rogova, "On the representation and exploitation of context knowledge in a harbor surveillance scenario," in *Proceedings of the 14th International Conference on Information Fusion*, July 2011.

[10] R. Carvalho, R. Haberlin, P. Costa, K. Laskey, and K. Chang, "Modeling a probabilistic ontology for maritime domain awareness," in *Proceedings of the 14th International Conference on Information Fusion*, July 2011.

[11] J. Roy, "Anomaly detection in the maritime domain," in *Optics and Photonics in Global Homeland Security IV. Proceedings of the SPIE*, vol. 6945, 2008, pp. 69 450W–69 450W–14.

[12] M. Riveiro, G. Falkman, T. Ziemke, and T. Kronhamn, "Reasoning about anomalies: a study of the analytical process of detecting and identifying anomalous behavior in maritime traffic data," *Proceedings of SPIE Defense, Security, and Sensing 2009*, vol. SPIE Volume 7346, 73460A, 1317 April 2009 2009.

[13] C. Brax and L. Niklasson, "Enhanced situational awareness in the maritime domain : An agent-based approach for situation management," in *Intelligent Sensing, Situation Management, Impact Assessment, and Cyber-Sensing : Proceedings of SPIE Defense, Security, and Sensing 2009*. SPIE, 2009.

[14] M. Richardson and P. Domingos, "Markov logic netwoks," *Machine Learning*, vol. 62, pp. 107–136, 2006.

[15] S. D. Tran and L. S. Davis, "Event modeling and recognition using markov logic networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer Berlin / Heidelberg, 2008, vol. 5303, pp. 610–623.

[16] D. Hall, M. McNeese, J. Llinas, and T. Mullen, "A framework for dynamic hard/soft fusion," in *In Proceedings of the 11th International Conference on Information Fusion*, Cologne, Germany, June-July 2008.

[17] L. Snidaro, I. Visentini, and G. Foresti, *Studies in Computational Intelligence*. Springer-Verlag, 2011, vol. 336/2011, ch. Data fusion in modern surveillance, pp. 1–21.

[18] J.Llinas, C. L. Bowman, G. L. Rogova, A. N. Steinberg, E. L. Waltz, and F. E. White, "Revisiting the JDL data fusion model II," in *Proceedings of the Seventh International Conference on Information Fusion*, vol. II, Stockholm, Sweden, June 2004, pp. 1218–1230.

[19] W. R. van Hage, V. Malaise, R. H. Segers, L. Hollink, and G. Schreiber, "Design and use of the simple event model (sem)," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 9, no. 2, 2011.

[20] L. Snidaro, M. Belluz, and G. Foresti, "Domain knowledge for surveillance applications," in *Proceedings of the Tenth International Conference on Information Fusion*, Quebec City, Canada, July, 9-12 2007.