Received: 22 December 2016 | Revised: 24 September 2017 | Accepted: 6 November 2017

DOI: 10.1002/rob.21771

REGULAR ARTICLE

WILEY

Detection of unanticipated faults for autonomous underwater vehicles using online topic models

Ben-Yair Raanan[1] | **James Bellingham**[2] | **Yanwu Zhang**[1] | **Mathieu Kemp**[1] | **Brian Kieft**[1] | **Hanumant Singh**[3] | **Yogesh Girdhar**[2]

[1]Monterey Bay Aquarium Research Institute, 7700 Sandholdt Road, Moss Landing, California, 95039

[2]Woods Hole Oceanographic Institution, 86 Water St, Woods Hole, Massachusetts, 02543

[3]Northeastern University, 360 Huntington Ave., Boston, Massachusetts, 02115

Correspondence
Ben-Yair Raanan, Monterey Bay Aquarium Research Institute, 7700 Sandholdt Road, Moss Landing, California 95039.
Email: byraanan@mbari.org

Abstract

For robots to succeed in complex missions, they must be reliable in the face of subsystem failures and environmental challenges. In this paper, we focus on autonomous underwater vehicle (AUV) autonomy as it pertains to self-perception and health monitoring, and we argue that automatic classification of state-sensor data represents an important enabling capability. We apply an online Bayesian nonparametric topic modeling technique to AUV sensor data in order to automatically characterize its performance patterns, then demonstrate how in combination with operator-supplied semantic labels these patterns can be used for fault detection and diagnosis by means of a nearest-neighbor classifier. The method is evaluated using data collected by the Monterey Bay Aquarium Research Institute's *Tethys* long-range AUV in three separate field deployments. Our results show that the proposed method is able to accurately identify and characterize patterns that correspond to various states of the AUV, and classify faults at a high rate of correct detection with a very low false detection rate.

KEYWORDS

autonomous underwater vehicle (AUV), autonomy, fault detection and diagnosis, topic modeling

1 | INTRODUCTION

As the capabilities of autonomous underwater vehicles (AUVs) improve, the tasks they perform become more complex and require longer endurance and higher reliability. Current generation AUVs are limited in their ability to diagnose faults[1] in hardware/software components and detect unforeseen events, such as unexpected interactions with the surrounding environment. In principle, AUVs equipped with the ability to diagnose faults and reason about mitigation actions could improve their survivability and increase the value of individual deployments by replanning their mission in response to faults.[1] However, in practice system-level fault protection architectures implemented onboard most AUVs employ a rule-based emergency abort system that is triggered by specific events, such as critical subsystems becoming unresponsive, or the vehicle exceeding its maximum depth limit. This approach is expedient, but since the developers rarely have complete knowledge of the vehicle's state and context, it is error-prone and generalizes poorly to the unexpected.

The long-term goal of our project is to give the vehicle the ability to mitigate problems autonomously by developing an onboard fault protection system that responds automatically to a wide range of performance anomalies, including the unexpected. Here, we focus on fault detection and diagnosis, and we argue that many of the limitations mentioned above can be alleviated by adopting a data-driven approach: (1) user-specified conditions and thresholds that define operational normality are replaced by general characteristics of classes that are inferred from data, and (2) faults are automatically identified as distinct classes.

Data-driven modeling techniques are increasingly prevalent in the domain of autonomous mobile robots. This domain presents fundamental modeling challenges due to its open-ended nature—the environments in which autonomous robots operate and often the systems themselves change over time, and these changes introduce new operational modes and failures. Existing data-driven fault detection methods seem too rigid in this regard; in particular, methods that rely on annotated datasets and are incapable of growing structurally as more data become available are incompatible with practical AUV operations, where the possibility of observing new performance modes must be considered. Hence, there is a need for automated modeling techniques that are not only capable of characterizing the system's performance

patterns accurately, but that can also adapt their complexity to incorporate new nominal and fault modes as they emerge.

In this paper, we extend the application of an online Bayesian nonparametric[2] (BNP) topic modeling technique based on Latent Dirichlet allocation (LDA[2,3]) to the problem of fault diagnosis in AUV vertical plane flight. BNP topic models have been shown to be effective in identifying patterns in unstructured datasets and building models whose structure grows and adapts to data.[4] These models do not require prior annotation or labeling of the dataset—the patterns emerge from the natural structure of the data.

Our proposed approach is to use the BNP technique to build a model of the vehicle's performance, including faults, directly from training datasets gathered in previous AUV operations, and then use this model for online fault detection and diagnosis by means of a nearest-neighbor classifier based on the Kullback-Leibler (KL) divergence measure.[5] The principal features of the method are that it accepts data from multiple domains, it does not require prior labeling of the dataset, and it automatically infers the number of classes present in the data. Moreover, the method allows the complexity of the model to continue and grow as more data accumulate, making the incorporation of new modes of operation straightforward. Although it is demonstrated by an AUV in the paper, the method applies to any autonomous vehicle. Our results suggest that the proposed framework is capable of automatically extracting meaningful performance patterns directly from AUV field data with no *a priori* knowledge and that distinct patterns relate to the various control policies executed onboard the AUV as well as to particular fault modes.

The paper is organized as follows: Section II describes related work. In Section III we introduce the topic modeling framework and its adaptation for modeling AUV sensor data, and we present our approach for fault detection and diagnosis based on the topic model's outputs. In Section IV we apply the method to state-sensor data collected by the Monterey Bay Aquarium Research Institute's *Tethys*-class long-range AUV (LRAUV), and we demonstrate its ability to classify distinct performance patterns and diagnose faulty states. We summarize and discuss our results in Section V, and we conclude in Section VI.

2 | RELATED WORK

Existing work on fault detection and diagnosis for underwater robotic systems can be divided into three main approaches: (1) rule-based, (2) model-based, and (3) data-driven.

2.1 | Rule-based

As mentioned above, automatic fault diagnosis has traditionally been performed using rule-based systems that target precise signatures (e.g., using thresholds) to identify the symptoms of a fault. Although rule-based systems are intuitive and easy to implement, their detection capabilities are limited to previously encountered faults and potential contingencies anticipated by developers—if a new fault that endangers the vehicle is observed during operations, additional rules will often be added. This results in a fault protection system that is complex and difficult to maintain, lacks flexibility, and relies on the quality and completeness of expert knowledge.

2.2 | Model-based

Model-based diagnosis has been successfully applied in a number of domains.[6] These methods are generally based on linear approximations of the system's dynamics, and they require models to be built for both nominal and faulty states; diagnosis proceeds by comparing model output with observed behavior and using various techniques to explain any discrepancies. A survey of fault-detection strategies using onboard unmanned underwater vehicles has been presented by Antonelli.[7] Many of these strategies are model-based but tend to be restricted to subsystems, e.g., the thrusters.[8] Another approach is consistency-based diagnosis,[9] which has led to the development of Livingstone,[5,10] a widely deployed system-level diagnosis engine.[11–13] A limitation of Livingstone is that it does not support numeric representations of variables. One way to overcome this is to use particle filters.[14,15] Although model-based approaches produce powerful tools to detect and identify faults, their design relies heavily on expert knowledge and therefore requires significant resources to develop and implement onboard complex systems.

2.3 | Data-driven

Data-driven approaches leverage statistical methods to identify patterns in data generated by the system and use them to classify nominal and faulty states.[16] An important dichotomy in data-driven fault detection distinguishes between *supervised* and *unsupervised* learning methods. In the supervised approach, a classifier is trained using annotated data containing both nominal and faulty conditions, and is then used to diagnose faults in data that have not yet been labeled. In the unsupervised approach, the data are not labeled or only include examples of nominal performance, and the broad goal is to find patterns and structure within the data and classify them into groups (clusters).

Data-driven techniques for fault detection and diagnosis in AUVs constitute a broad field of research and include implementations of artificial neural networks (ANNs),[4,15,17] support vector machines (SVMs),[10,11] and Bayesian belief networks (BBNs).[13,18] Nearly all of these implementations use supervised techniques and make a strong assumption that annotated data containing all fault types are available for training. However, in practice such data are typically absent and very expensive to produce.

In this paper, we have chosen to focus on the use of unsupervised learning algorithms that impose as few *a priori* assumptions about the data as possible. The framework presented here is based on latent Dirichlet allocation (LDA),[2] a probabilistic topic model used to discover patterns in an unstructured collection of discrete data. Although LDA was originally developed for semantic analysis of text documents, it has since been applied in the domain of robotics to model context in a humanoid robot,[19] for activity analysis,[8] and autonomous exploration.[20]

Following Blei, Griffiths, Jordan, & Tenenbaum,[21] we use a BNP extension to LDA to enable the topic model to automatically adapt its complexity and infer the number of groups, or clusters, present

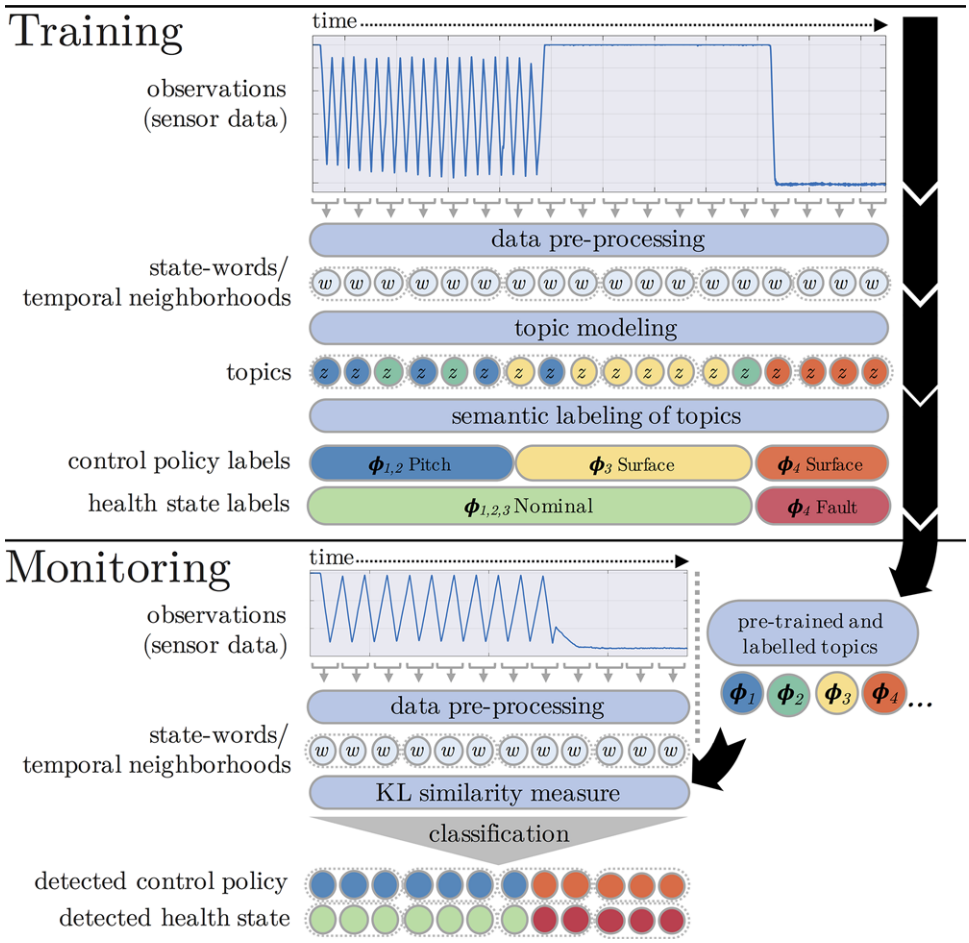

FIGURE 1 Overview of our proposed technique for fault detection using probabilistic topic models. Training (offline): given a collection of training datasets containing observations of state-sensor data, we process the data to extract discrete features (state-words) and group them in temporal neighborhoods (see Section 3.2). We apply the BNP topic modeling algorithm to learn the AUV's performance patterns (topics; Section 3.3) and compute estimates of the model's uncertainty to identify anomalous observations (Section 3.4). Finally, we inspect the trained model and ascribe semantic meaning to the topics (Section 3.5). Monitoring (online): given new incoming observations, we extract state-words and group them in temporal neighborhoods. We compute the similarity between each temporal neighborhood and the topics learned from the training datasets using KL, and we classify the temporal neighborhoods using the semantic labels associated with the most similar topic (i.e., the nearest-neighbor; Section 3.6)

in a dataset. We have chosen to use this method over other algorithms that can infer the number of clusters from the data, such as spectral clustering[22] and affinity propagation,[14] mainly because it is a fully Bayesian generative probabilistic model. As such, the method offers an inherent uncertainty criterion for estimating the clustering quality of the model and its ability to generalize to unseen data. Furthermore, similar BNP methods have been shown to produce meaningful results when used for automatic classification of seafloor imagery[17] and chemical sensor data[23] collected by AUVs. Of particular relevance to this study is the work by Girdhar *et al.*,[3] which introduced an online variant of BNP-LDA, for automatic scene characterization and anomaly detection in image and video data collected in unstructured underwater environments.

3 | APPROACH

In this section, we introduce the BNP topic modeling framework and its adaptation for modeling AUV sensor data, and we present our approach for online fault detection and diagnosis. An overview of the proposed framework is shown in Figure 1. As shown, the approach is divided in two stages: (1) training, where we apply the BNP topic modeling technique to AUV sensor data gathered in previous operations in order to build a model of the vehicle's performance, and (2) monitoring, where we use the learned model for online fault detection and diagnosis by means of a nearest-neighbor classifier based on the KL divergence measure.

3.1 | Latent Dirichlet allocation (LDA)

LDA[2] is a generative mixed-membership model originally used for semantic analysis of text corpora. The basic assumption made in LDA is that each group of observations (documents) is generated from a random mixture of latent components (topics)—each topic is a distribution over the collection's vocabulary. Formally, given a collection of D documents composed from a vocabulary V, the LDA generative process is as follows:

1. For each topic $k = 1, \ldots, K$:
   a. $\phi_k \sim Dirichlet(\beta)$.
2. For each document $d \in D$:
   a. $\theta_d \sim Dirichlet(\alpha)$.
   b. For each word $w_i \in d$:
      I $z_i \sim Discrete(\theta_d)$,
      II $w_i \sim Discrete(\phi_{z_i})$,

where $x \sim Y$ indicates that random variable x is sampled from distribution Y, and α and β are the hyperparameters for the Dirichlet priors from which the discrete distributions are sampled. Each word w_i is a discrete element from a fixed vocabulary indexed by $\{1, \ldots, V\}$, while each z_i represents the topic responsible for generating the word instance w_i, and it is indexed by $\{1, \ldots, K\}$. Each θ_d is a document-specific distribution over topics (it can be seen as a low-dimensional representation of the dth document), and ϕ_k specifies the distribution of the kth topic over the vocabulary words.

The LDA generative process results in the joint probability distribution:

$$\mathrm{P}(\boldsymbol{w}, z, \theta, \phi|\alpha, \beta) = \mathrm{P}(\phi|\beta)\,\mathrm{P}(\theta|\alpha)\,\mathrm{P}(z|\theta)\,\mathrm{P}\left(\boldsymbol{w}|\phi_z\right), \tag{1}$$

where the variables z, θ, and ϕ are unknown (latent). To learn them, LDA reverses the generative process by expressing the conditional posterior distribution of the latent variables given the observed data:

$$\mathrm{P}(z, \theta, \phi|\boldsymbol{w}, \alpha, \beta) = \frac{\mathrm{P}(\theta, \phi, z, \boldsymbol{w}|\alpha, \beta)}{\mathrm{P}(\boldsymbol{w}|\alpha,\beta)}. \tag{2}$$

Approximate inference techniques such as variational inference[2] or collapsed Gibbs sampling[12] are then used to resolve the posterior.

3.2 | Data preprocessing

Extending the topic modeling framework to AUV sensor data requires that the general idea of a textual word be replaced by discrete features we refer to as *state-words*. To generate a vocabulary of state-words, we discretize each of the N signals, $\boldsymbol{S} = \{s_n\}_{n=1}^{N}$, used to describe the AUV's state into m_n nonoverlapping bins,[3] and we concatenate them into a vocabulary of size $V = \sum_{n=1}^{N} m_n$. To extract state-words from a given signal s_n, we map each element of s_n to its closest corresponding state-word in the vocabulary (Fig. 2). When no measurement is available for a given sensor (missing data), no word is generated. This process can be viewed as a transformation of a time-series made of heterogeneous data (e.g., numeric, Boolean, or text) to a common domain space.

3.3 | Online Bayesian nonparametric topic modeling

Observations made by robotic systems are generally continuous in nature, and so the descriptors used to compute the topic labels must account for temporal dependencies that may exist between the data. Following Girdhar *et al.*,[3,20] we address this issue by generalizing the idea of a document to a temporal cell and computing the topic labels for a state-word within a cell in the context of its neighboring cells. Given a sequence of observations of the AUV's state, we extract state-words

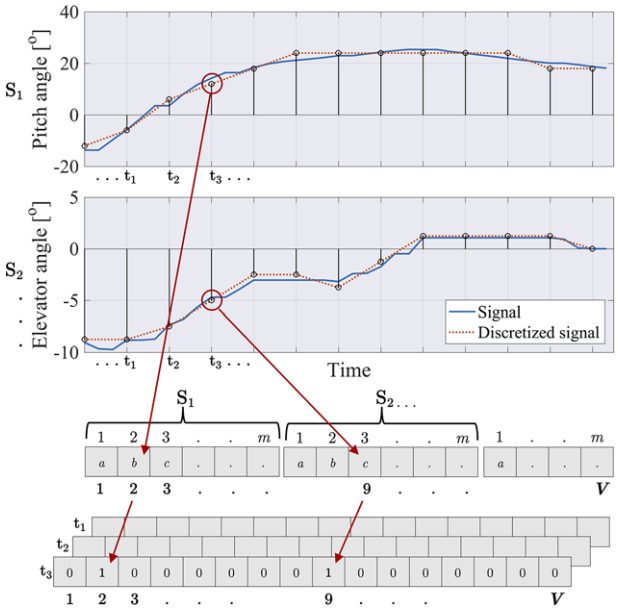

FIGURE 2 An illustration of the state-word extraction process. (a) Each signal (s) used to describe the AUV's state is discretized using m nonoverlapping bins. (b) The bins are concatenated into a vocabulary of size V. (c) Each discretized element of s is mapped to its closest corresponding state-word in the vocabulary, and the word count is incremented

$\boldsymbol{w}$, each with a corresponding time-step t. Similar to Girdhar *et al.*,[3] we model the likelihood of the observed data in terms of the latent topic label variables z, which denote the underlying state of the vehicle:

$$\mathrm{P}(w|t) = \sum_{k \in K_{active}} \mathrm{P}(w|z = k)\,\mathrm{P}(z = k|t). \tag{3}$$

Here the distribution over vocabulary words $\phi_k \equiv \mathrm{P}(w|z = k)$ models the appearance of the topic k, as it is shared across all temporal coordinates. The second part of the equation $\theta_t \equiv \mathrm{P}(z = k|t)$ models the distribution of the topic labels within the temporal neighborhood of time-step t.

We make no *a priori* assumptions about the number of latent topics. Instead, we adopt a BNP approach and assume that there is an infinite number of them, but only a finite number is needed to explain the observed data. We use a method similar to the Chinese Restaurant Process (CRP[22]) to learn the active[4] topic labels K_{active} directly from the data and specify a CRP prior γ over the infinite groupings to control the growth of the number of labels so as to favor the lowest number that can adequately explain the data.[3,21] The algorithm models whether an observation is best explained by an existing topic or by a new, previously unseen topic, thus allowing the model to grow automatically with the size and complexity of the data.

Finally, we use the online collapsed Gibbs sampler proposed by Girdhar and Dudek,[20] which divides computational resources equally between computing the posterior topic distribution of recent observations and updating topic labels for older ones. Consequently, the algorithm works to maintain the model at a nearly converged state at any given time.

3.4 | Uncertainty estimation and novelty detection

During the training phase, we monitor the uncertainty in the topic model's predictions by computing the per-word *perplexity* score for each time-step. The per-word perplexity score for a set of state-words observed at a time-step t is defined as

$$\text{perplexity}\,(t) = \exp\left(-\frac{\sum_i^{W_t} \log \mathrm{P}\,(w_i|t)}{W_t}\right), \tag{4}$$

where W_t is the number of state-words in time-step t, w_i refers to the ith state-word, and the term $\mathrm{P}(\omega_i|t)$ is computed using Eq. (3). Observations that contain prevalent state-words that have been associated by the model with a topic in previous observations (i.e., "learned") produce a low perplexity score, whereas observations that contain rare or previously unobserved state-words that are poorly represented within the model produce a high perplexity score. Thus, we use the perplexity score not only to measure convergence and the overall quality of the topic model, but also to identify novel or anomalous information to which the topic model was not previously exposed.

3.5 | Semantic labeling of topics

Topics derived from a sequence of observations of the AUV's state represent the latent processes that are responsible for *generating* those states. As such, these topics should correspond to the control policies or behaviors that are executed onboard the AUV, and capture the dynamic relationship between these control policies and the AUV's performance. In this respect, the topic modeling framework can be used to generate a model of the AUV's performance directly from training data.

We apply the BNP topic modeling algorithm to a collection of training datasets to learn the performance patterns that correspond to nominal states of the AUV, as well as to specific faults. Once the training process is complete, we analyze the trained topic-model and ascribe semantic meaning to each of the learned topics. This is a necessary step that allows us to use the trained topic-model for classification. The correspondence between a learned topic and a class (e.g., a control policy or a fault) can be determined qualitatively, or quantitatively if the dataset is annotated. We provide a mathematically rigorous method for evaluating the correspondence between a topic and a class for the latter case.

Given a series of operator-supplied class labels corresponding to each time-step, we compute the marginal probability distribution that defines the topic label proportions for that class:

$$\mathrm{P}\,(z = k|class) = \sum_{t \in T_{class}} \frac{\mathrm{P}\,(z = k|t)}{|t_{class}|}, \tag{5}$$

where T_{class} is the index of all time steps belonging to that class, and $\mathrm{P}(z = k|t)$ is the topic label distribution of each time step t. We then use Bayes' rule to reverse $\mathrm{P}(z = k|class)$, and we compute the conditional probability

$$\mathrm{P}\,(class|z = k) = \frac{\mathrm{P}\,(z = k|class)\,\mathrm{P}\,(class)}{P\,(z = k)}, \tag{6}$$

which defines the probability of the class given the topic label. We define $\mathrm{P}(class)$ to be $|T_{class}|/|T|$, where T is the total number of time steps, and we calculate $\mathrm{P}(z = k)$ using Eq. (5) and substituting T_{class} with T.

3.6 | Online fault detection and diagnosis

We hypothesize that a topic model trained on previously observed examples of nominal performance and faults can be used to compute a robust estimate of the vehicle's state in new, previously unseen observations. Given a trained and semantically labeled topic-model Φ, we monitor the health of the system online by measuring the similarity between the learned topic distributions $\phi_k \in \Phi$ and the distribution of state-words extracted from each incoming observation over the defined vocabulary V. If a distribution of state-words from a given observation is most similar to a topic ϕ_k that corresponds to a faulty state, then a fault is identified.

We use the symmetrized Kullback-Leibler (KL) divergence to measure the similarity between two distributions p and q:

$$KL\,(p, q) = \frac{1}{2}\left[D\,(p, q) + D\,(q, p)\right], \tag{7}$$

where

$$D\,(P, Q) = \sum_{i=1}^{V} P_i \log_2 \frac{P_i}{Q_i}. \tag{8}$$

The most relevant topic is the one that minimizes KL (i.e., the nearest-neighbor). A similar approach has also been used in other studies for facial recognition,[18] audio classification of bird species,[2] and event recognition in video.[23]

4 | EXPERIMENTAL RESULTS

We conducted experiments using the Monterey Bay Aquarium Research Institute's *Tethys*-class LRAUVs (Fig. 3).[1,6] Three datasets were collected separately in 2013, 2015, and 2016. The 2013 and 2015 datasets include examples of nominal performance of the LRAUV as well as failures (critical faults) of the vehicle's mass-shifting system that caused the vehicle to collide with the seabed. The 2016 dataset was chosen because it includes examples of nominal performance of the LRAUV in various states and a new control policy.

Evaluation was done in two steps: first, we used the 2013 data as a training set for the topic-model. We evaluated the model's performance during the training phase using the perplexity measure [Eq. (4)], and we evaluated the correspondence between the output topics and the executed control policies and faults using our proposed method for topic labeling [Eqs. (5) and (6)]. Then, we used the 2015 and 2016 datasets as test sets to evaluate the classification performance of the trained topic-model on unseen data. We used the 2015 test set to evaluate the method's ability to accurately classify a fault, and the 2016 test set to evaluate classification accuracy on a fault-free control.

Table 1 lists the state-sensor signals and data-products that were used as inputs to the model.

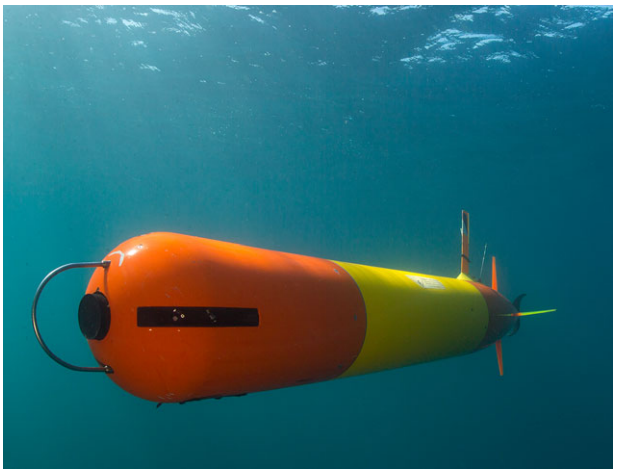

FIGURE 3 The *Tethys* LRAUV is 2.3 m long and 0.3 m in diameter. The vehicle is controlled by a propeller, a traditional elevator and rudder control surfaces, a variable buoyancy system (VBS), and an actuated mass-shifter

4.1 | Training dataset

We post processed state-sensor data and onboard data-products that were collected by the LRAUV during a scientific field campaign in Monterey Bay, California, between September 9 and 14, 2013. During the initial part of the deployment, the vehicle correctly executed a series of vertical profiles using four control policies [Fig. 4(a)]: Float-on-surface (purple), Pitch (yo-yo trajectory; blue), Surface (ascend to surface; orange), and Depth (hover at depth; green). At 22:22 UTC, as the vehicle was descending on a yo-yo dive, a rupture of the mass-shifter lead-screw caused the battery-mass to shift all the way forward. As a result, the AUV, now extremely nose-heavy, was unable to correct its downward attitude and collided with the bottom. At 23:15 UTC, the vehicle's software[9] identified the problem as a "failure to ascend" fault and triggered the AUV's safety behaviors. However, these actions failed to bring the vehicle to the surface, and so the LRAUV remained on the bottom for 27 h and was eventually located on the beach near Rio Del Mar, California, 8 km away from its last reported position.

We processed this dataset using the BNP topic modeling framework: we extracted state-words $(V = 356)$ from the dataset, which included 62,920 observations, and we ran the algorithm to compute topic distributions for each time step. We determined the value of the hyperparameters by running the model with a range of choices $\alpha \in \{0.01, 0.1, 1, 5\}$, $\beta \in \{0.01, 0.1, 1, 5\}$, and $\gamma \in \{1e-6, 1e-5, 1e-4\}$, and we selected the combination that minimized the average perplexity score [Eq. (4)]. After the model was trained, we evaluated the correspondence between the learned topics and the control policies [Eqs. (5) and (6)] using a time-series of the control policies that were logged onboard the LRAUV [line color in Fig. 4(a)], and we evaluated the topics' correspondence with the fault using an operator-labeled fault record of the deployment [red shading in Fig. 4(a)].

We achieved the best model performance with $\alpha = 0.1$, $\beta = 5$, and $\gamma = 1e-5$ as the Dirichlet and CRP hyperparameters. Figure 4(b) shows the distribution of topic labels for each time step (θ_t) and illustrates how topics change over time within the model; the height of each band reflects the topic proportion $P(z = k|t)$. We find that the executed control policies are consistently aligned with distinct topics, and that unique topics are assigned to the deployment segments where the AUV has bottomed (topics 7 and 8).

Figure 4(c) shows the perplexity scores computed for each time step [Eq. (4)]. As shown, most observations are represented by the model with high certainty (low perplexity) reflecting good overall convergence of the topic-model. The highest perplexity scoring, excluding the initial "burn-in" period, coincides with the bottoming incident (22:22 UTC) and reflects the model's exposure to the new faulty state. In the time-steps that follow the fault, the perplexity score tapers off

TABLE 1 AUV state-sensor signals and data products

Numerical			Boolean	
Signal	**Range***	**Comment**	**Signal**	**Comment**
Depth rate (m/s)	(– 2, 2)		Drop weight dropped	
Surge velocity (m/s)	(– 3, 3)		Buoyancy full	
Heave velocity (m/s)	(– 1, 1)		Surface depth	Depth = 0 m
Roll angle (deg)	(– 90, 90)		Stop envelope	Safety metric
Pitch angle (deg)	(– 90, 90)		YoYo envelope	Safety metric
Roll rate (deg/s)	(– 2, 2)		Going to surface	Safety metric
Pitch rate (deg/s)	(– 2, 2)			
Stern plane angle (deg)	(– 15, 15)			
Rudder plane angle (deg)	(– 15, 15)			
Thruster power (W)	(0, 35)			
ΔMass position (mm)	(– 25, 25)	From default pos.		
ΔBuoyancy position (ml)	(– 400, 400)	From neutral pos.		
ΔPitch angle (deg)	(– 50, 50)	From commanded		
ΔDepth (m)	(0, 225)	From commanded		

*Quantization interval centers are N equally spaced values between (a, b), where N = 25 for all numerical signals.

FIGURE 4 (a) Time series of vehicle depth (2013 dataset); line color indicates the executed control policy and the red shaded background indicates the bottoming fault. The LRAUV system identified the "failure to ascend" fault approximately 50 min after the vehicle had bottomed (red triangle). (b) A stacked plot showing the distribution of topic labels for each time step *t*, computed using the BNP topic model. The learned topics exactly match the various control policies, and unique topics are assigned to the deployment segments where the AUV has bottomed (topics 7 and 8). (c) Time series of per-word perplexity scores. The highest perplexity scoring coincides with the bottoming fault (22:22 UTC) and reflects the model's exposure to the new fault

as the model "learns" the new performance patterns that are associated with the bottoming fault. Other high perplexity events (spikes) observed during the deployment segments where the AUV performed nominally are associated with the yo-yo transition phases (topic 3) and surfacing events (topic 5) of which there are relatively few examples throughout the dataset.

Figure 5 presents a simplified two-dimensional view of the topic model. We encode the topics as circles, with areas proportional to the relative prevalence of each topic, $P(z = k)$. The distances between the circles reflect the intertopic differences computed using the KL similarity measure [Eq. (7)], subject to principal component analysis (PCA) dimensionality reduction.[19]

In Figure 5(a), the pie-chart slices in each circle reflect the relative probability of each control policy P(control policy|z=k), computed using Eq. (6). Topics 4 and 6 correspond to the Depth and Float-on-surface control policies (respectively) with high probability. Topic 5 mostly corresponds to the Surface control policy. Topics 1, 2, and 3 correspond to the Pitch control policy, which commands the LRAUV while profiling the water column. A closer examination of the time-series revealed that these topics, in fact, correspond to the downward, upward, and transition phases of the yo-yo trajectory.

In Figure 5(b), the pie-chart slices reflect the relative probability of the AUV's health P(health state|z=k). Topics 1–6 correspond to nominal performance of the LRAUV. Topics 7 and 8 correspond to the fault, and essentially characterize the underlying performance patterns that correspond to the AUV's state during the bottoming incident. The transition from topic 7 to topic 8 being most dominant [Fig. 4(b)] coincides with the detection of the fault by the AUV's system [23:15 UTC; red triangle in Fig. 4(a)], which

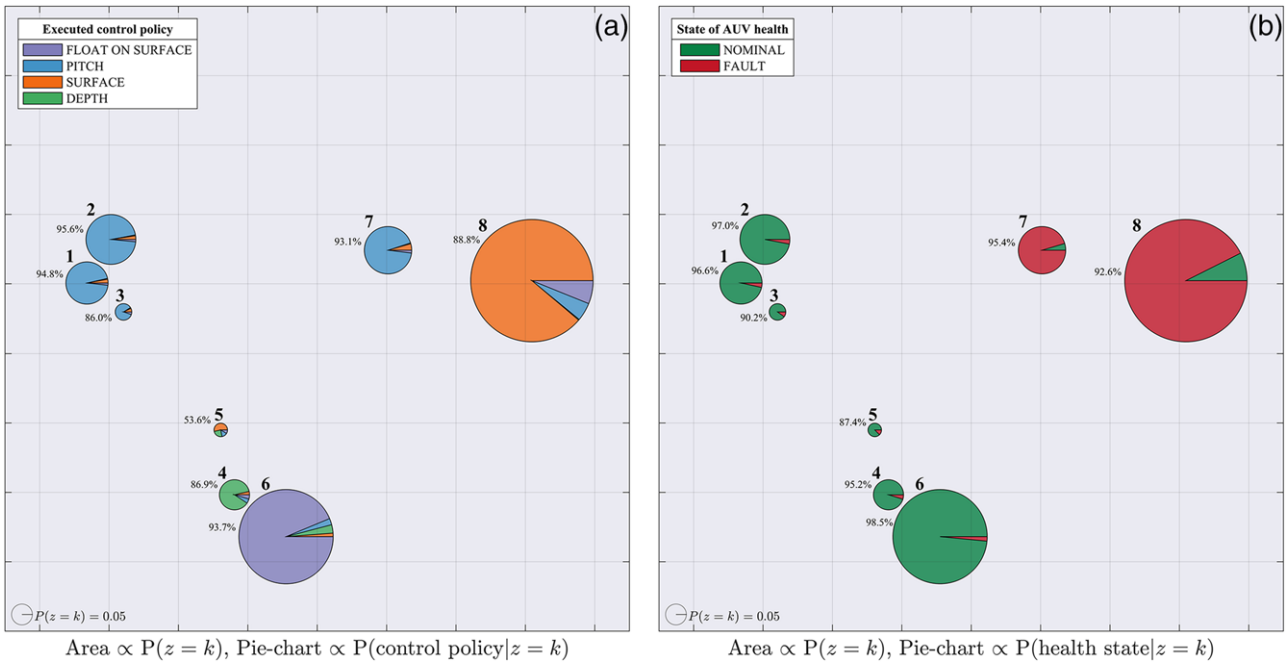

FIGURE 5 2D representation of the topic model. The areas of the circles are proportional to the relative prevalence of each topic, P($z = k$). The locations of the circles and the distances between them reflect how similar topics are to one another. Similarities are calculated using the KL similarity measure [Eq. (7)] and reduced to 2D using PCA. The PCA recomposition to 2D preserved 97% of the variance. (a) The pie-chart slices in each circle are proportional to the relative probability of the control policies, P(control policy|z=k), computed using the proposed method for topic labeling [Eqs. (5) and (6)]. In (b) the pie-chart slices are proportional to the relative probability of the AUV's health, P(health state|z=k). Topics 7 and 8 are associated with the bottoming fault with high probability

TABLE 2 Semantic labeling of topics. The semantic labels assigned to each topic are shown in bold text

	P(health state|topic)		P(control policy|topic)			
	Nominal	Fault	Float on sur.	Pitch	Surface	Depth
Topic 1	**0.97**	0.03	0.02	**0.95**	0.03	0.01
Topic 2	**0.97**	0.03	0.01	**0.96**	0.02	0.01
Topic 3	**0.90**	0.10	0.05	**0.86**	0.07	0.02
Topic 4	**0.95**	0.05	0.05	0.05	0.04	**0.87**
Topic 5	**0.87**	0.13	0.09	0.13	**0.54**	0.24
Topic 6	**0.98**	0.02	**0.93**	0.02	0.01	0.03
Topic 7	0.05	**0.95**	0.02	**0.93**	0.04	0.01
Topic 8	0.07	**0.93**	0.06	0.05	**0.89**	0.00

terminated the mission (Pitch) and triggered the LRAUV's safety behaviors (Surface).

Table 2 summarizes the computed conditional probabilities, P(control policy|z=k) and P(health state|z=k), and shows the semantic labels assigned to each topic.

4.2 | Test datasets

The first test dataset was collected by LRAUV along the coast of Año Nuevo, California, between September 15 and 16, 2015. Similar to the 2013 dataset, the test set contained a fault in the mass-shifting system that caused the LRAUV to bottom and led to temporary loss of the vehicle (red shading in Fig. 6(a)). The fault was triggered by an erroneous software configuration that caused the internal mass-shifter to repeatedly overload and eventually disabled it. Unlike the 2013 incident, the LRAUV's onboard fault-detection system detected the fault immediately and triggered the safety behaviors at 02:46 UTC. The fault prevented the LRAUV from adjusting its trim and eventually caused it to collide with the bottom.

The second test dataset was collected by LRAUV during a scientific field campaign in Monterey Bay, California, between February 3 and 4, 2016. The 2016 test set did not include any failures; instead, it exposed the classifier to a variety of control policies that were executed correctly by the LRAUV (Fig. 7(a)) and included an additional control policy, i.e., Depth-rate (magenta; Fig. 7), that was not part of the 2013 training set. The Depth-rate control policy is used to execute vertical profiles in hover mode (i.e., using only the VBS[24]) and is functionally most similar to the Depth control policy.

We extracted state-words from the 2015 and 2016 test datasets, which included 75,920 and 146,305 observations (respectively), and used Eq. (7) to compute KL similarities between the state-words extracted from each observation and the topic-word distributions Φ learned from the 2013 data. Then, we labeled each time step according to its nearest-neighboring topic, and we validated the classification results against the time-series of executed control policies and the fault-record that were obtained from the vehicle's log files. For comparison, we repeated the procedure with the 2013 training dataset to attain an "in-sample" classification accuracy estimate.

Table 3 summarizes the classification accuracies obtained for the test and training datasets using the proposed KL-based nearest-neighbor classifier. In the first test dataset (2015), the classifier

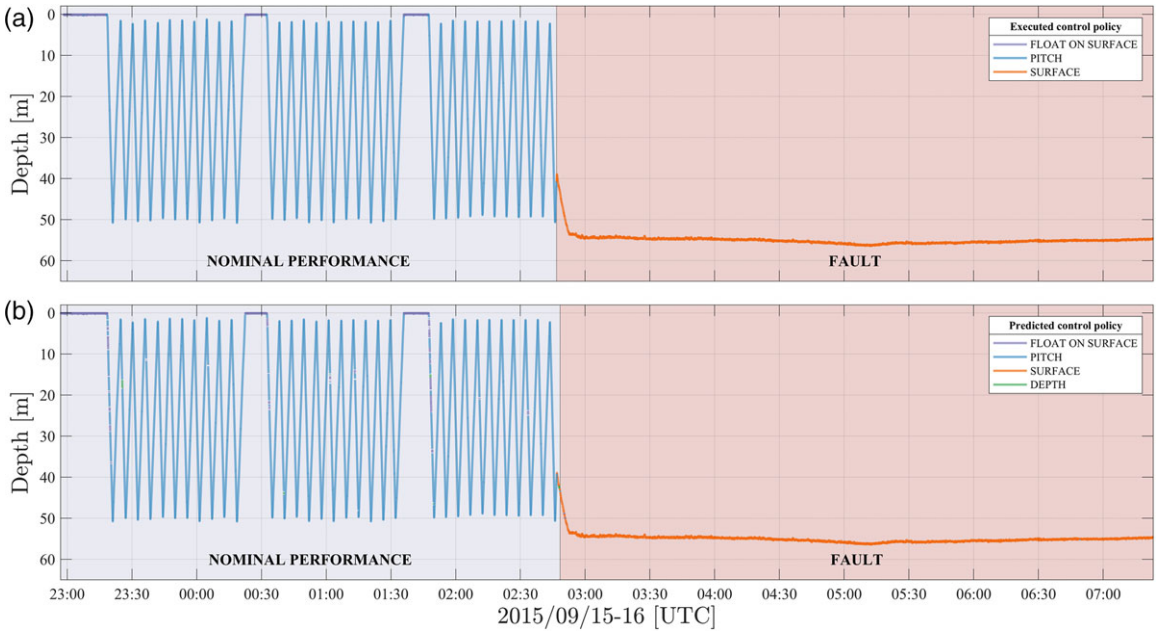

FIGURE 6 (a) Time series of vehicle depth (2015 dataset); line color indicates the executed control policy, and the red-shaded background indicates the fault. Initially, the LRAUV correctly executed a series of vertical yo-yo dives using the pitch (blue) and float on surface (purple) control policies. At 02:46 UTC the LRAUV detected an overload fault in its internal mass-shifter and immediately triggered the emergency safety behaviors (red shading). (b) Time series of vehicle depth with the control policy and fault records (line color and red shading, respectively) that were reconstructed from the classification results

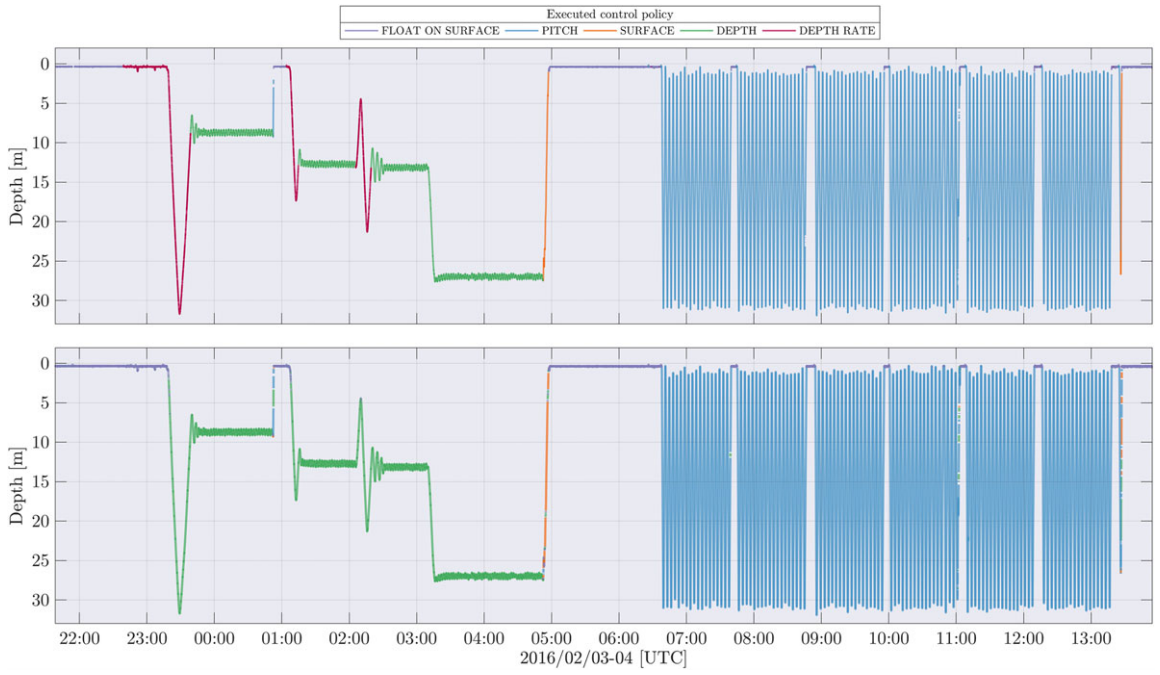

FIGURE 7 (a) Time series of vehicle depth (2016 dataset); line color indicates the executed control policy. The LRAUV correctly executed a series of hover dives using the Depth Rate (magenta), Depth (green), and Surface (orange) control policies, followed by a series of vertical yo-yo dives using the Pitch (blue) control policy. The Depth Rate control policy was not used in the 2013 training set. (b) Time series of vehicle depth with the control policy record that was reconstructed from the classification results. The deployment segments where the Depth-rate control policy was executed were classified as a mixture of Depth and Float-on-surface control policies

accurately classified the state of the AUV's health in 99.5% of observations and predicted the executed control policy correctly in 99.8% of observations (on average). More importantly, the classifier detected the bottoming fault with no false positives. The classifier identified the bottoming fault at 02:48:23 UTC, 1.65 min after the LRAUV's onboard fault-detection system identified the overload fault in the mass-shifting system, and approximately 3.8 min before the AUV collided with the sea floor. For reference, Figure 6 shows a comparison between the original 2015 test dataset [Fig. 6(a)] and the control policy and fault records (line color and red shading,

TABLE 3 Summary of KL nearest-neighbor classification accuracies. TPR: true positive ratio; FPR: false positive ratio

Dataset	Class		Accuracy (%)	TPR (%)	FPR (%)
Test set (2015)	Health state	Nominal	99.49	100.0	0.94
		Fault	99.49	99.06	**0.00**
	Control policy	Surface	99.88	99.79	0.02
		Depth	–	–	–
		Pitch	99.39	98.53	0.10
		Float on sur.	98.99	99.04	1.02
Test set (2016)	Health state	Nominal	100.0	100.0	0.00
		Fault	–	–	–
	Control policy	Surface	99.69	52.94	0.01
		Depth	93.29	100.0	9.36
		Pitch	99.23	98.27	0.17
		Float on sur.	94.02	93.79	5.91
Training set (2013)	Health state	Nominal	99.96	100.0	0.06
		Fault	99.96	99.94	**0.00**
	Control policy	Surface	99.55	99.31	0.35
		Depth	99.28	92.57	0.13
		Pitch	99.40	98.92	0.31
		Float on sur.	99.11	99.64	1.06

respectively) that were reconstructed from the classification results [Fig. 6(b)].

In the second test set (2016), the classifier accurately classified the state of the system's health in 100% of observations (no false positives) and predicted the executed control policy correctly in 95.5% of observations (on average). The deployment segments where the Depth-rate control policy was executed were classified as a mixture of the Depth control policy (during vertical profile dives) and the Float-on-surface control policy (when the vehicle was on the surface). Figure 7 shows the depth record of the 2016 dataset with the control policy record (line color), which were reconstructed from the classification results along with the original dataset.

In the training dataset (2013), the proposed classifier accurately classified the state of the system's health in 99.96% of observations, with no false positives, and predicted the executed control policy correctly in 99.3% of observations (on average). The classifier identified the bottoming fault at 22:24:08 UTC, nearly 51 min before the LRAUV's onboard fault-detection system, and approximately 0.4 min (25 s) before the AUV had bottomed.

5 | DISCUSSION

We extended a BNP topic modeling framework to automatically identify and characterize AUV performance patterns directly from state-sensor data, and we applied a KL-based nearest-neighbor classifier for online fault detection and health monitoring of an AUV. We evaluated the framework using datasets collected by the *Tethys* LRAUV in three separate field deployments, two of which included faults that led to temporary loss of the vehicle. We used the first dataset to train the topic-model, and the other two to evaluate classification performance on unseen data.

We found a strong correspondence between the topics and the control policies and fault records indicating that the method is capable of accurately characterizing the performance patterns that correspond to the various states of the AUV. During the training phase, the BNP approach ensures that the model adapts automatically to the size and complexity of the data, and the computed perplexity scores indicate exposure to novel or anomalous information. We have found that in combination with operator-supplied semantic labels, the topic-based representation can be used as a reference for classifying between nominal AUV performance and specific faults, and that the learned information generalizes well to new observations through the use of KL-based similarity functions. The method produced a high rate of correct detection with a very low false-detection rate.

We have found that in most cases, the initial exposure of the model to a new state triggered an abrupt increase in perplexity, which subsequently decreased as the model "learned" the new performance pattern. In contrast, the deployment segments that included yo-yo inflections (topic 3) consistently produced high perplexity scores despite the fact that they occurred repeatedly throughout the training set [spikes in Figure 4(c)]. The reason for the difference is that a yo-yo inflection requires, among other things, a reversal of vehicle pitch and of the elevator angle and of the pitch-rate; as a result, yo-yo inflections occupy a larger chunk of state-space volume than, say, holding depth or climbing at constant pitch, which consists of small exploration around a steady-state, and so they take longer to learn. Figure 4(c) shows that the yo-yo inflection perplexity does in fact trend down, suggesting that it will eventually vanish with more training examples. We also point out

that this "perplexity overshoot" phenomenon is specific to the training phases, and has little impact on the classification performance of the system—which is, in fact, high.

We have shown that the topics learned during a training mission provided good classification during subsequent test missions. One interesting question this raises is whether the topics learned on one vehicle are stable to changes in vehicle configuration or to environmental variability. To address this, we trained a new topic model using the 2015 dataset and compared the output topics to the ones learned from the 2013 training dataset. We found that although the deployments were performed at different locations and used different vehicle configurations, the topics extracted from the two sets were in fact very similar, providing some encouraging indication of robustness. This similarity also suggests that a scheme using historically learned topics as the starting point of a large-disturbance learning procedure could be effective.

The ability of the framework to learn the performance patterns using a single training set is particularly relevant for AUVs, where unanticipated faults slowly emerge over time and where the availability of labeled training data is limited. The framework's ability to learn new fault-models based on a small number of examples could conceivably enable the developers to maximize the information gain from rare events. The topic-based representation also offers an efficient way to add new information to the AUV's system without increasing the complexity of the autonomy software.

The model identifies fault *states*, rather than determining which specific subsystem is subject to failure. This type of situational diagnostic is particularly useful for unanticipated fault detection, as was the case in the 2013 bottoming incident where a failure of the mass-shifter went undetected by the onboard fault-detection system, but was easily identified in postprocessing by the proposed technique. If this fault information had been available to the vehicle, bottom impact could have been prevented by shutting off the thruster and inflating the variable buoyancy system.

6 | CONCLUSION

We applied the proposed framework to characterize the performance patterns of an AUV and to detect and diagnose faults. We trained the topic model and evaluated the classification performance using datasets collected in three separate field deployments.

Our results demonstrate that the framework was able to automatically characterize patterns that relate to vertical plane performance of the AUV, and to classify faults with a high probability of detection and a low false-detection rate. A key feature of the framework is that it does not rely on expert knowledge, but instead learns the relationship between the executed control policy and the vehicle's performance directly from the data. Although it was demonstrated by an AUV in this paper, the framework is applicable to any autonomous vehicle.

Our ongoing efforts are to compare the performance of the proposed framework to other existing methods. We are interested in the development of a health monitoring architecture that is capable of learning performance topic models online and that leverages the topic-based representation of the system's state to inform autonomous replanning and automatic selection of mitigation actions in response to failures.

ACKNOWLEDGMENTS

We are grateful for support from the Office of Naval Research (ONR grant N00014-14-1-0199) and the David and Lucile Packard Foundation. The authors thank Brett Hobson, M. Jordan Stanway, Jon Erickson, Denis Klimov, Ed Mellinger, and Carlos Rueda for LRAUV operations and for insightful discussions.

ENDNOTES

[1] We define a fault as a deviation from expected behavior.

[2] Here "nonparametric" implies that the number of classes is open-ended.

[3] In this work, we use equal-width-binning; however, any binning approach is valid.

[4] A label k is active if there is at least one observation assigned to it.

ORCID

Ben-Yair Raanan http://orcid.org/0000-0001-5585-495X

REFERENCES

1. Bellingham JG, Rajan K. Robotics in remote and hostile environments. *Science*. 2007;318(5853):1098–1102.
2. Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. *J Machine Learning Res*. 2003;3(Jan):993–1022.
3. Gershman SJ, Blei DM. A tutorial on Bayesian nonparametric models. *J Math Psychol*. 2012;56(1):1–12.
4. Healey AJ. A neural network approach to failure diagnostics for underwater vehicles. In: *Proceedings of the 1992 Symposium on Autonomous Underwater Vehicle Technology, 1992. AUV '92*. IEEE; 1992:131–134.
5. Kullback S, Leibler RA. On information and sufficiency. *Ann Math Statist*. 1951;22(1):79–86.
6. Hobson BW, Bellingham JG, Kieft B, McEwen R, Godin M, Zhang Y. Tethys-class long range AUVs—Extending the endurance of propeller-driven cruising AUVs from days to weeks. In *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*. IEEE; 2012:1–8.
7. Antonelli G. A survey of fault detection/tolerance strategies for AUVs and ROVs. *Fault Diagnosis and Fault Tolerance for Mechatronic Systems: Recent Advances*. 2003; 109–127.
8. Duckworth P, Al-Omari M, Charles J, Hogg DC, Cohn AG. Latent Dirichlet allocation for unsupervised activity analysis on an autonomous mobile robot,. In *AAAI*; 2017:3819–3826.
9. Kieft B, Bellingham J, Godin M, Hobson B, Hoover T, McEwen R, & Mellinger E. Fault detection and failure prevention on the Tethys long-range autonomous underwater vehicle. In *17th International Unmanned, Untethered Submersible Technology Conference, Portsmouth, NH*, 2011.
10. Williams BC, Nayak PP. A model-based approach to reactive self-configuring systems. In *Proceedings of the National Conference on Artificial Intelligence*; 1996:971–978.
11. Antonelli G, Caccavale F, Sansone C, Villani L. Fault diagnosis for AUVs using support vector machines. In *2004 IEEE International Con-*

ference on Robotics and Automation, 2004. Proceedings, ICRA '04 (Vol. 5); 2004:4486–4491. http://doi.org/10.1109/ROBOT.2004.1302424

12. Griffiths TL, Steyvers M. Finding scientific topics. *Proc Natl Acad Sci (USA)*. 2004;101:5228–5235.
13. Madsen AL, Kjærulff UB, Kalwa J, Perrier M, Sotelo MA. Applications of probabilistic graphical models to diagnosis and control of autonomous vehicles. In *20th Conference on Uncertainty in Artificial Intelligence*, 2004.
14. de Freitas N, Dearden R, Hutter F, Morales-Menendez R, Mutch J, Poole D. Diagnosis by a waiter and a Mars explorer. *Proc IEEE* 2004;92(3):455–468. http://doi.org/10.1109/JPROC.2003.823157
15. Narasimhan S, Dearden R, Benazera E. Combining particle filters and consistency-based approaches for monitoring and diagnosis of stochastic hybrid systems, In *15th International Workshop on Principles of Diagnosis*. 2004.
16. Aldrich C, Auret L. *Unsupervised Process Monitoring and Fault Diagnosis with Machine Learning Methods*. London: Springer-Verlag; 2013.
17. Steinberg D, Friedman A, Pizarro O, Williams SB. A Bayesian nonparametric approach to clustering data from underwater robotic surveys. In *International Symposium on Robotics Research, Flagstaff, AZ*. Citeseer; 2011.
18. Shakhnarovich G, Fisher JW, Darrell T. Face recognition from long-term observations. In *7th European Conference on Computer Vision*. Springer Science + Business Media; 2002:851–865. http://doi.org/10.1007/3-540-47977-5_56
19. Celikkanat H, Orhan G, Pugeault N, Guerin F, Şahin E, Kalkan S. Learning and using context on a humanoid robot using latent dirichlet allocation. In *2014 Joint IEEE International Conferences on Development and Learning and Epigenetic Robotics (ICDL-Epirob)*. IEEE; 2014:201–207.
20. Girdhar Y, Dudek G. Gibbs sampling strategies for semantic perception of streaming video data, 2015. *CoRR, abs/1509.0*. Retrieved from http://arxiv.org/abs/1509.03242
21. Blei DM, Griffiths TL, Jordan MI. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *J ACM*. 2010;57(2):7. http://doi.org/10.1145/1667053.1667056
22. Teh YW, Jordan MI. Hierarchical Bayesian nonparametric models with applications. *Bayesian Nonparametrics* 2010;1.
23. Jakuba MV, Steinberg D, Kinsey JC, Yoerger DR, Camilli R, Pizarro O, & Williams SB. Toward automatic classification of chemical sensor data from autonomous underwater vehicles. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*; 2011:4722–4727. http://doi.org/10.1109/IROS.2011.6095158
24. Zhang Y, Kieft B, McEwen R, Stanway J, Bellingham J, Ryan J, Hobson B, Pargett D, Birch J & Scholin C. Tracking and sampling of a phytoplankton patch by an autonomous underwater vehicle in drifting mode. In *OCEANS 2015-MTS/IEEE Washington*. IEEE; 2015:1–5.
25. Bajwa A, Sweet A. The livingstone model of a main propulsion system. In *Aerospace Conference, 2003. Proceedings* (Vol. 2); IEEE; 2003:2869–2876. http://doi.org/10.1109/AERO.2003.1235498
26. Bellingham JG, Zhang Y, Kerwin JE et al. Efficient propulsion for the Tethys long-range autonomous underwater vehicle. In *2010 IEEE/OES Autonomous Underwater Vehicles* (pp. 1–7). IEEE; 2010.
27. Blei DM, Griffiths TL, Jordan MI, Tenenbaum JB. Hierarchical topic models and the nested Chinese restaurant process. In *Advances in Neural Information Processing Systems*; 2004:17–24.
28. Briggs F, Raich R, Fern XZ. Audio classification of bird species: A statistical manifold approach. In *2009 Ninth IEEE International Conference on Data Mining*. IEEE; 2009:51–60.
29. Chuang J, Ramage D, Manning C, Heer J. Interpretation and trust: Designing model-driven visualizations for text analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM; 2012:443–452.
30. Ferreira B, Matos A, Cruz N. Automatic reconfiguration and control of the MARES AUV in the presence of a thruster fault. In *OCEANS 2011 IEEE - Spain*. IEEE; 2011:1–8.
31. Frey BJ, Dueck D. Clustering by passing messages between data points. *Science* 2007;315(5814):972–976.
32. Girdhar Y, Cho W, Campbell M, Pineda J, Clarke E, Singh H. Anomaly detection in unstructured environments using Bayesian nonparametric scene modeling. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016:2651–2656.
33. Girdhar Y, Giguère P, Dudek G. Autonomous adaptive exploration using realtime online spatiotemporal topic modeling. *Int J Robotics Res*. 2013, 278364913507325.
34. Hayden S, Sweet A, Shulman S. Lessons learned in the Livingstone 2 on Earth observing one flight experiment. In *AIAA 1st Intelligent Systems Tech. Conf., Am. Inst. Aeronautics and Astronautics*. American Institute of Aeronautics and Astronautics; 2004.
35. Hjort NL, Holmes C, Müller P, Walker SG. *Bayesian nonparametrics* (Vol. 28). Cambridge University Press; 2010.
36. Isermann R. Model-based fault-detection and diagnosis—Status and applications. *Ann Rev Control*. 2005;29(1):71–85.
37. Khokhar S, Saleemi I, Shah M. Similarity invariant classification of events by KL divergence minimization. In *2011 International Conference on Computer Vision*. IEEE; 2011:1903–1910.
38. Kleer, Williams BC. Diagnosing multiple faults. *Artificial Intell*. 1987;32(1):130–197.
39. Kurien J, Nayak PP. Back to the future for consistency-based trajectory tracking. In *AAAI/IAAI*; 2000:370–377.
40. Miguelanez E, Patron P, Brown KE, Petillot YR, Lane DM. Semantic knowledge-based framework to improve the situation awareness of autonomous underwater vehicles. *IEEE Transactions on Knowledge and Data Engineering* 2011;23(5):759–773.
41. Ranganathan N, Patel MI, Sathyamurthy R. An intelligent system for failure detection and control in an autonomous underwater vehicle. In *IEEE Transactions on Systems, Man, and Cybernetics–Part A: Systems and Humans* (Vol. 31); 2001:762–767. http://doi.org/10.1109/3468.983434
42. Shi C, Zhang R, Yang G. Fault diagnosis of AUV based on Bayesian networks. In *First International Multi-Symposiums on Computer and Computational Sciences, 2006. IMSCCS '06* (Vol. 2); 2006:339–343. http://doi.org/10.1109/IMSCCS.2006.224
43. Sun Y, Li Y, Zhang G, Zhang Y, Wu H. Actuator fault diagnosis of autonomous underwater vehicle based on improved Elman neural network. *J Central South Univ*. 2016;23(4):808–816. http://doi.org/10.1007/s11771-016-3127-8
44. Von Luxburg U. A tutorial on spectral clustering. *Statistics Comput*. 2007;17(4):395–416.
45. Zhang M, Wu J, Chu Z. Multi-fault diagnosis for autonomous underwater vehicle based on fuzzy weighted support vector domain description. *China Ocean Eng*. 2014;28(5):599–616. http://doi.org/10.1007/s13344-014-0048-x

How to cite this article: Raanan B-Y, Bellingham J, Zhang Y, et al. Detection of unanticipated faults for autonomous underwater vehicles using online topic models. *J Field Robotics*. 2018;35:705–716. https://doi.org/10.1002/rob.21771