Machine Learning Based Sound Speed Prediction for Underwater Networking Applications

Ambrin B. Riaz Ahmed, Mohamed Younis and Miguel Hernandez De Leon
Department of Computer Science and Electrical Engineering
University of Maryland, Baltimore County,
Baltimore, Maryland, USA,
ambrinr1@umbc.edu

***Abstract*— Underwater acoustic networks operate in an inhomogeneous and dynamic environment, which makes it difficult to model the propagation path of signals. In essence acoustic signals experience reflection and refraction due to sound speed variation, based on many parameters such as salinity, temperature, and depth. To enable modeling of signal propagation, the sound speed profile (SSP) has to be accurately estimated. The most famous SSP equation has been proposed by Mackenzie and has been widely used among others like the Coppens' and UNESCO equations. The drawback of these equations is that they yield different accuracy levels for various setups. They are also valid only for certain limits of salinity, depth and temperature. Moreover, the SSP estimation method should suit both deep and shallow water environments. In this paper, we use machine learning algorithms to predict sound speed in both deep and shallow waters and compare our results with data collected from acoustic tomography measurements. For training we have considered sound speed measurements across various oceans like Pacific Ocean, Arctic Ocean, Indian Ocean, etc. Our results show that our model achieves 99.99% accuracy and outperforms Leroy and Mackenzie equations.**

***Keywords*— Sound speed, acoustic communication, Underwater networks, Machine learning.**

I. INTRODUCTION

Despite the popularity of radios in terrestrial based wireless systems, RF signals get absorbed in water and acoustics are deemed the most viable means for establishing underwater communication links. The underwater environment is quite challenging and is characterized by its inhomogeneous and dynamically changing properties; these characteristics affect the propagation of acoustic signals and complicates the operation of underwater networking applications. For example, in an acoustic underwater network (AUN), omni-directional transmissions complicate medium access control and increase power consumption, and hence directional transmissions are favored. For directional transmissions to be effective, the propagation path of signals should be accurately predicted [1]-[3]. However, as illustrated in Fig. 1, an acoustic signal is subject to refraction due to changes in the sound speed experienced along its propagation path. Thus, to establish robust acoustic communication links in underwater setup, it is necessary to employ an accurate sound speed profile (SSP) to predict the signal propagation path and appropriately steer the transmitter antenna to reach the targeted receiver. Although in short range communications, the sound speed may be assumed to be constant, for long range communications, the sound speed variations have significant impact on the propagation path of acoustic signals and could degrade the link quality [3][4]. Furthermore, when there is a variation of sound speed with respect to depth, acoustic signals suffer from refraction and in extreme cases shadow zones could be created where little or no signal propagation could take place. This could result in communication failure.

For applications like navigation systems for underwater robotic vehicles and for underwater acoustic positioning systems (UAPS), highly accurate estimation of sound speed is necessary; otherwise it could lead to position errors. The results of [5][6] reveal that the acceptable position error to operate floats, gliders and unmanned underwater vehicles is 0.01% to 0.04% of the reachable range of surface-based beacon sources. In [7], the authors present a method to determine sound speed using signal propagation characteristics and show that their method, combined with multi-lateration, enhances localization accuracy. Nonetheless, for underwater tracking applications, which rely on range measurements, even a significantly smaller percentage error in effective sound velocity estimates could lead to serious errors in range measurements [8]. Furthermore, for applications like Internet of Underwater Things (IoUT), it is crucial for connectivity to be robust and for the bit error rate to be minimal in order to avoid intermittent reachability issues [9]. Hence, there is a need for highly accurate sound speed measurement for the aforementioned applications.

Sound speed is traditionally calculated using various

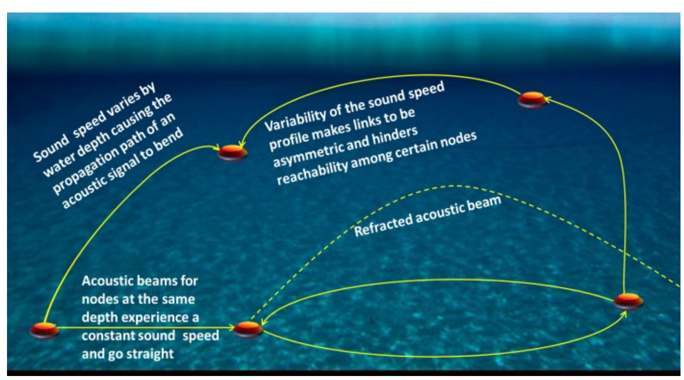

Fig. 1. Sound speed variations in underwater environment affects the propagation path of acoustic signals where the direct link between two nodes does not follow a straight line. A node often ought to pursue transmission angles in order to reach another node at different depth. In fact, the acoustic signal may refract due to sound variations.

equations like the Mackenzie, Coppens, UNESCO and Del Grosso's or NRL II equation [10]-[12]. However, these equations have limitations with respect to depth, temperature and salinity as shown below in Tables 1 and 2. The Mackenzie and Coppens equations (proposed in 1981) use depth, temperature and salinity values to compute sound speeds. The UNESCO and Del Grosso equations use pressure (instead of depth), temperature and salinity to measure sound speed [12]. However, these equations do not cover all oceans (deep water, shallow water or high salinity oceans) and based on the boundary conditions of applications, one need to choose suitable equations for calculating sound speed [13].

Leroy et al. proposed a different equation to calculate sound speed across all seas [14]. However, such an equation only factors in salinities from pure water up to 42% and fails to address other salinities. Hence, the applicability of Leroy's equation is restricted to certain seas. To better capture the effect of environment dynamics, in-situ measurements are aggregated to infer the sound speed within the deployment area. For example, sound speed is first measured by a few randomly deployed nodes and such information is then used to estimate the sound speed profile of a larger area [13]. However, such distributed in-situ methodology imposes a major computational burden on the underwater nodes and does not yield the level of accuracy expected for UAPS applications. Another method has been proposed in [15] to determine sound speed using surface measurements and satellite measurements. Obtaining these measurements would be cost intensive and this method has been tested only for the Central Arabian Sea.

Table 1: Validity ranges of sound speed for various depths [10][12].

Equation	**Depth (m)**	**Temperature (°C)**	**Salinity (%)**
Mackenzie	0 to 8000	-2 to 30	30 to 40
Coppens	0 to 4000	0 to 35	0 to 45

Table 2: Validity ranges of sound speed based on pressure [14].

Equation	**Pressure (bar)**	**Temperature (°C)**	**Salinity (%)**
UNESCO	0 to 1000	0 to 40	0 to 40
Del Grosso	0 to 980.66	0 to 30	30 to 40

Hence, there is a need for a technique to predict accurate sound speed across all seas, including salinity ranges of over 200% [14], and it should also be applicable to all underwater applications. To the best of our knowledge, there are no equations or methods to accurately predict sound speed for all salinities, depth, temperature or pressure ranges without limitations. This paper promotes a novel approach that employs machine learning to accurately model the sound speed and overcome the limited data availability. We exploit regression algorithms which could work efficiently with limited datasets and measure their performance. We use sound speed values obtained from XBT and Sonar measurements as training dataset. The predicted SSP values in our model have been found to lie within ± 0.05 m/s of the in-situ measured SSP values. Machine learning algorithms learn better with more training data. In practice, as more XBT and Sonar measurements become available, they could be integrated and the regression algorithms would be able to yield even better SSP estimates.

The paper is organized as follows. Section II reviews related work. Section III explains the machine learning approach used and briefly explains how regression algorithms work. Section IV analyzes data and results of sound speed predicted using regression algorithms with traditional equations and reports the performance and accuracy of our technique. Finally, Section V concludes the paper.

II. Background and Related Work

Given the contribution of the paper, we discuss related work under two categories: (i) adopting machine learning techniques for AUN and underwater target localization, and (ii) models for predicting sound speed based on various underwater conditions.

Use of machine learning in underwater environments: Machine learning could be considered as a "black-box" since it can work without any prior knowledge of the environment, yet it can provide instantaneous results that improve over time. This feature is particularly promising for underwater acoustics as the environment conditions are fluctuating and the use of machine learning would avoid costly measurements using ocean acoustic tomography methods [16]-[18]. Lefort et al. [16] have proposed using machine learning for source localization in underwater environments, e.g., determining the position of an acoustic transmitter in a network, or the source of the sound detected by a sensor. In [16], direct regressions have been employed for underwater source localization. Two regression models, namely, kernel regression and linear piecewise regression have been explored. Regression models only need a sample training dataset of the fluctuating environment, which is obtained from software simulations. Their experiments show that the source localization error is reduced by using machine learning algorithms. H. Niu et al. [19] have analyzed the performance of feed-forward neural network (FNN), support vector machine (SVM) and random forest (RF) for source localization and range estimation. The localization is studied as a supervised learning problem and the classification algorithms (FNN, SVM and RF) are found to perform well with a mean absolute percentage error of 2-3%. Better results have achieved by A. Horri [20] using convolutional neural networks.

A new data-driven technique has been proposed in [15] to determine SSP from surface parameters that are obtained using remote sensing measurements. Artificial neural networks (ANN) have been employed to determine SSP from hourly surface observations. Surface observations from mooring as well as satellites have been used for the analysis. The employed ANN algorithm is based on the multilayer perceptron model with two hidden layers utilizing the backpropagation algorithm. Comparing the SSP values calculated using ANN with in-situ measurements have revealed that 76% of the predicted values lie within ± 1 m/s, and 93% of predicted values lie within ± 2 m/s of the SSP values obtained from mooring. Such a data-driven method is based on the assumption that it is difficult to obtain in-situ measurements of temperature, depth and salinity profiles of the seawaters and has been tested only for the central part of the Arabian Sea. Unlike our approach, this work further involves obtaining surface and satellite measurements which are costly and time-intensive.

Underwater sound speed estimation: Conventionally, a number of approximate models are used for estimating sound speed in an underwater environment like the Mackenzie's, Coppens', UNESCO and Del Grosso's and NRL II equations. These models use field measurements to devise a polynomial that captures the effect of the various environment parameters, such as temperature and depth. However, these equations are not universal for all oceans and hence, based on application requirements the best candidate is chosen.

Although the UNESCO equation covers wider salinity ranges than the NRL II equation, its measurements are derived by comparison with pure water. Furthermore, the UNESCO equation gives incorrect results of sound speed under high pressure conditions and in regions where the acoustic signal propagates through greater depth, while the NRL II equation performs better in these conditions [14]. Due to the absence of precise measurements under pressure, there was no universal equation for sound speed calculation in low-saline oceans and pure water. Belogol'skii et al. [21] proposed an equation for measurement of sound speed in pure water under pressure with pressure estimation of up to 60 MPa and temperature ranges of 0.4 to 40°C. However, this equation requires complex computations and could be impractical for underwater nodes. Meanwhile, Leroy et al. [14] covers salinity ranges from pure water to salinities up to 42%, calculates sound speed by using depth, temperature, salinity and ocean latitude across all seas and oceans. The only exception is regions like the Mediterranean Sea, Red sea or Gulf of Mexico where salinities could exceed 200%. These regions have not been considered by Leroy et al. or others while calculating sound speed.

Popular sound speed equations use parameters like depth, temperature and salinity and hence are closely related to our machine learning model. Mackenzie's equation [10] for sound speed factors in depth, temperature and salinity is as follows:

$$C = 1448.96 + 4.591T - 5.304 \, x \, 10^{-2}T^2 + 2.374 \, x \, 10^{-4}T^3 + 1.340(S-35) + 1.630 \, x \, 10^{-2} \, D + 1.675 \, x \, 10^{-7}D^2 - 1.025 \, x \, 10^{-2} \, T(S-35) - 7.139 \, x \, 10^{-13} \, TD^3 \quad (1)$$

where T is temperature in Celsius, D is depth in meters and S is salinity in parts per thousand. Meanwhile the equation proposed by Leroy et al. [14] considering the ocean latitude is as follows:

$$C = 1402.5 + 5T - 5.44x10^{-2}T^2 + 21.x10^{-4}T^3 + 1.33S - 1.23x10^{-2}ST + 8.7x10^{-5}ST^2 + 1.56x10^{-2}Z + 2.55x10^{-7}Z^2 - 7.3x10^{-12}Z^3 + 1.2x10^{-6}Z(\phi - 45) - 9.5x10^{-13}TZ^3 + 3x10^{-7}T^2Z + 1.43x10^{-5}SZ \quad (2)$$

where T is temperature in Celsius, Z is depth in meters, S is salinity in % and ϕ is latitude in degrees.

Collected field measurements of sound speed have indicated that substantially more variations occur in the vertical axes, i.e., with depth, than on the horizontal axes [22]. Hence, most researchers consider a layered model to profile an acoustic region, corresponding to changes in depth. One such approach is the Distributed Real-time Oceanic Profiling approach (DROP) [4], which uses parabolic ranging and measured sound speed values of a few randomly deployed nodes to determine the sound speed profile of a larger area. In this method, the 3D environment is mapped to an equivalent 2D one and then a second order degree polynomial is applied to determine the path of the transmitted signal. The slope gradient of the polynomial is utilized to create a layered medium. DROP finally obtains an SSP by utilizing the locally determined sound speed values and the sound speed in each distinct layer (calculated from the signal refraction). Although DROP provides real-time estimation of SSP, it has high computational complexity.

We note that so far all existing equations for sound speed calculations are restricted to certain salinity, temperature, depth profiles and for varying applications suitable equations should be used. For various applications like IoUT and target recognition, we need precise values of sound speed. This paper fills the technical gap and promotes a novel machine learning based approach, where a model is trained using available sound speed measurements to predict SSP across all seas irrespective of their salinity, temperature, depth, pressure, or latitude profiles. In Section V, we show that our approach outperforms the popularly used sound speed equations.

III. Data-driven SSP Modeling

A learning problem in machine learning comprises a dataset on *n* samples and the algorithm tries to predict properties of unspecified data. Machine learning can be classified as either supervised or unsupervised learning. Supervised learning is subdivided into two groups, namely, classification and regression [23][24]. A classification problem is one where samples correspond to two or more classes and the objective is to factor in already-labeled data and estimate the class of unlabeled data. In other words, classification could be considered as a discrete form of supervised learning where the *n* samples of data have to be labeled as belonging to a particular class (category) from the available finite set of classes (categories). A regression problem is one where the required output is composed of one or more continuous variables. Since we are not trying to label but rather predict sound speed based on parameters like depth, temperature and salinity, the output has continuous values and thus sound speed profiling is a regression problem. We use two regression algorithms, specifically, Ridge and Ensemble Bagging Regression because we found through tests that these algorithms provide better results for our dataset. A brief overview of these algorithms is presented below.

A. Ensemble Bagging Regression

An ensemble is a linear combination of a model fitting or statistical learning methods rather than using one particular method. Let us consider the example of function estimation.

We are keen on finding a real valued function, $g: IR^d \to IR$, based on data $(X_1, Y_1) \dots \dots (X_n, Y_n)$ where X represents a d-dimensional predictor variable and Y represents a univariate response. Suppose a base approach is defined such that for every input data it produces an estimated function $\hat{g}(.)$. The base approach could be a regression tree, kernel estimator or other estimators. We can repeat the base approach several times by reweighting original data to generate corresponding estimates $\widehat{g_1}(.), \widehat{g_2}(.), \dots, \widehat{g_M}(.)$, where M is the estimate for the M^{th} reweighted input data set. An ensemble-based function estimate $g_{ens}(.)$ is then obtained as a linear combination of individual function estimates $\widehat{g_k}(.)$, as follows [25]:

$$\widehat{g_{ens}}(.) = \sum_{k=1}^{M} c_k \, \widehat{g_k}(.) \quad (3)$$

$\widehat{g_k}(.)$ is derived from the base approach using the reweighted k^{th} input data set. The value $c_k = \frac{1}{M}$, is the linear combination coefficient and is obtained by averaging the weights.

Boosting and Bagging are two common types of ensemble methods. Boosting involves creating samples and assigning initial weights to each sample as, $W_i = \frac{1}{N}$ where $i = 1,2 \dots N$. So the samples are trained on the original data. For every subsequent iteration, the sample weights are updated and then the learning algorithm is subjected to use the updated weights. The weights are increased for samples whose prediction was incorrect during the previous step and the weights are reduced for samples which gave accurate predictions in previous step. This process is repeated several times and the samples with worse predictions keep increasing their weights in each cycle. The final prediction is obtained by a majority vote or averaging of individual predictions. The sample with higher weight influences the final prediction [26]. Meanwhile, Bagging is an ensemble method used for enhancing unstable estimation and is popular because of its easy implementation. A bagging algorithm consists of the following steps [25]:

Step 1: Develop a bootstrap sample $(X_1^*, Y_1^*), \dots, (X_n^*, Y_n^*)$ by randomly drawing n times and by using replacement from data $(X_1, Y_1) \dots \dots, (X_n, Y_n)$

Step 2: Calculate the bootstrapped estimator from equation, $\widehat{g_*}(.) = h_n\big((X_1^*, Y_1^*), \dots, (X_n^*, Y_n^*)\big)(.)$

Step 3: Repeat steps 1 and 2 M times to obtain $\hat{g}^{*k}(.)$ where k=1, 2,…, M. The bagged estimator is given by,

$$\hat{g}_{Bag}(.) = \frac{1}{M} \sum_{k=1}^{M} \hat{g}^{*k}(.) \quad (4)$$

In summary, ensemble bagged regression consists of creating random bootstrap samples from the initial dataset which are used for training and hence several versions of the base predictor are developed. Every estimation for a new observation involves predicting with all the trained predictors. The bagged estimation is therefore the mean of all individual predictions.

B. Decision Trees

A tree structure grows upside down starting at the root and leads to splits or nodes. An observation starts at the root and proceeds through the nodes (splits) where a decision is made regarding the direction to continue depending on the value of the explanatory variables. The predicted response is given when eventually a terminal node is reached. Binary recursive partitioning is used to fit trees. The term binary implies that the root or parent node would always be divided into two equal child nodes. The term recursive means that the process repeats wherein each child node becomes parent and is in turn divided into two equal child nodes and so on. The recursion stops if the node is a terminal node. The process starts one explanatory variable to create a split. The variable and location of the split are determined to reduce the node impurity at that position. A split yields two child nodes that are in turn split using the same basis and the process continues until no more splits are possible or if the process is terminated based on a user-defined criterion [27].

Decision trees are categorized into classification and regression trees. The terminal nodes of a classification tree ascribe to the class which constitutes the plurality of cases in that node. The terminal nodes of a regression tree are attributed a value, which is the mean of cases in that node. Two impurity measures are commonly used for the regression trees, namely, least squares and least absolute deviations. The least squares measure aims at minimizing the sum of squared difference between the observations and mean at each node. The least absolute deviations measure strives to reduce the mean absolute deviation from the median within the node. This method is superior to least squares because it is not sensitive to irregularities [27].

IV. DATA AND RESULTS

Several open source machine learning libraries are available today like Tensorflow, Scikit-learn, Theano, Caffe and Torch which have equivalent competence and either one could be used to solve machine learning problems [18]. Given its popularity we have used the Scikit-learn library to validate our approach; the source code of our implementation can be found at https://github.com/ambrinherz/underwater_ml.

A. Dataset

In order to validate our approach, we have used 3 datasets. The first dataset is the XCTD and Sonar data collected from submarines in the Arctic Ocean for Scientific Ice Expeditions from 1999 to 2000 [28]. This dataset includes sound speed measured at 140 positions of varying latitude and longitude; each position calculates the sound speed for depths between 13m and 1000m. This Arctic Ocean dataset has 170,574 records, each includes —measurements of depth, salinity, temperature and sound speed. The second dataset represents the oceanographic data collected in the Molucca sea, Celebes sea, Philippine Sea and North Pacific Ocean from June 2010 to July 2010 [29]. This dataset includes XBT data of sound speed measured at 38 positions of varying latitude and longitude; each position calculates the sound speed for depths between 0m and 800m. This data set referred to as the Pacific Ocean dataset has 44,057 sound speed values. The third dataset represents the data collected in the Southern oceans including Indian and South Atlantic Ocean from 1998 to 2000 and is hereby referred to as the Southern Oceans dataset [30].

This dataset includes XCTD data of sound speed measured at 36 positions of varying latitude and longitude; each position calculates the sound speed for depths between 0m and 1000m.This dataset has a total of 34,360 sound speed values.

We have applied our machine learning based approach to predict the sound speed. About 70% of the dataset was used for training and the remaining 30% was used for testing. We have also studied the effect of the training dataset size on performance. We have experimented with several machine learning algorithms for regression like Elastic Net, Lasso, Ridge, Ensemble Random Forest etc. Table 3 shows the algorithm prediction score per feature for the worldwide dataset. The score value lies between 0 and 1, where a higher score reflects better prediction accuracy. The worldwide dataset refers to the combination of the Arctic, Southern and Pacific Ocean datasets.

Table 3: Algo. prediction score per feature for the worldwide dataset

No.	Algorithm	Sound Speed	Depth	Temperature	Salinity
1	Lasso	0.7194	0.0496	2.11E-9	2.89E-6
2	Ridge	0.8507	0.0791	0.3498	0.8462
3	Elastic Net	0.0031	0.0011	2.11e-9	2.89e-6
4	Bagging Regressor	0.9997	0.9980	0.9997	0.9904
5	Random Forest	0.9996	0.9982	0.9997	0.9904
6	Extra Trees	0.9996	0.9985	0.9996	0.9909
7	Ada Boost	0.9983	0.7665	0.9797	0.5152
8	Gradient Boosting	0.9999	0.9757	0.9989	0.9615

Table 3 shows the performance of different algorithms when predicting one feature while the others are known. For example, when depth, temperature and salinity are known, the Lasso algorithm can predict sound velocity with a prediction score of 0.7194. We have found that Bagging Regressor, Random forest, Gradient Boosting and Extra trees algorithms have similar scores for predicting depth, temperature, salinity or sound velocity. Hence, any of these algorithms would be an appropriate choice. We have picked Bagging Regressor for further consideration.

We have used data from the Southern Oceans, Arctic Ocean, Pacific Ocean and a combination of all these datasets called worldwide dataset to predict sound velocity using Bagging Regressor and Decision Tree Regressor algorithms. We also used different base estimator sizes for the Bagging Regressor to determine the best prediction accuracy for the Arctic Ocean and compared the results with Mackenzie's equation as shown in Table 4. We increased the size of the base estimator from 10 to 400 and found that after an estimator size of 100 the prediction accuracy flattens or remains constant. Further the prediction accuracy is about 99.98% for all the estimator sizes. Hence, we decided to use the default base estimator size of 10 for subsequent testing. We have used default parameters for the Bagging Regressor which includes the following:

- "base_esimator", which defaults to the Decision Tree Regressor,
- "n_estimators" indicates the number of base estimators included in the ensemble and has a default value of 10,
- "max_samples: default", which refers to the number of samples that are drawn from the dataset and is set to 1,
- "max_features:default", which specifies the number of features that are drawn from the dataset and are in turn used to train the estimator. This parameter is set to 1.

We have used default parameters for the Decision Tree Regressor, which comprises "criterion= mse", "splitter=best", "min_samples_split=2", "min_samples_leaf= 1", "max_depth=None", etc. The criterion "mse" refers to mean squared error which is used to measure the quality of a split. When the maximum depth of the tree is None, the nodes are expanded to reach all pure leaves or to reach all leaves holding less than "min_samples_split" samples. The default "min_samples_split" is 2 and reflects the minimum needed number of samples to split an internal node. Each leaf node has a minimum sample of one [24]. At every depth in the test dataset, we have predicted the sound velocity and determined the difference between the predicted value and actual sound speed. We have then averaged this difference to calculate the mean difference in sound velocity for various oceans. We have also evaluated the mean difference for the sound velocity calculated using the Mackenzie equation; the results are tabulated in Table 4.

Table 4: Sound velocity prediction using Mackenzie's equation and Bagging Regressor in the Arctic Ocean

Mean Difference (m/s)					
Mackenzie Equation	Base estimator size				
	10	50	100	200	300
9.77 (90.23%)	0.0176 (99.982%)	0.0167 (99.983%)	0.0148 (99.985)	0.0149 (99.985%)	0.015 (99.985%)

Table 5 shows that for the Arctic Ocean, the Mackenzie equation is 9.77 m/s (on average) away from the actual values of sound velocity while the Bagging Regressor and the Decision Tree are only 0.018 and 0.0134 m/s (on average) away from the actual sound velocity values, respectively. We have repeated the tests to compare with Leroy's equation; Table 6 shows the results. Note that the Decision Tree and Bagging Regressor results in Tables 5 and 6 are different since Leroy's equation uses latitude and longitude as a parameter for calculations. Hence, the dataset used for this test includes latitude and longitude for all considered oceans while applying Bagging Regressor, Decision Tree and Leroy's equation.

We observe from Tables 5 and 6 that the Bagging Regressor and Decision Tree results are much closer to the actual sound speed. For example, in the Arctic Ocean we see that the Mackenzie and Leroy equations are only 90% accurate while the Bagging Regressor and Decision Tree are 99.98% accurate. This affirms that machine learning algorithms perform much better than traditional equations for predicting sound velocity. Since the Bagging Regressor algorithm internally uses several decision trees, we have been eager to find the effect a single decision tree has on the results. Based on Tables 5 and 6, a single decision tree has similar performance to the Bagging Regressor and runs much faster as well. Therefore, for the remaining tests we used a single decision tree for predicting sound velocity.

Table 5: Sound velocity prediction using Mackenzie's equation and Bagging/Decision Tree.

No.	Algorithm	Mean Difference (m/s)			
		Arctic Ocean	Indian Ocean	North Pacific Ocean	Worldwide
1	Mackenzie Equation	9.77 (90.23%)	0.14 (99.86%)	0.12 (99.98%)	6.74 (93.26%)
2	Bagging Regressor	0.018 (99.98%)	0.02 (99.98%)	0.034 (99.96%)	0.021 (99.97%)
3	Decision Tree	0.0134 (99.98%)	0.0137 (99.98%)	0.0219 (99.97%)	0.015 (99.98%)

Table 6: Sound velocity prediction using Leroy equation and Bagging/Decision Tree

No.	Algorithm	Mean Difference (m/s)			
		Arctic Ocean	Indian Ocean	North Pacific Ocean	Worldwide
1	Leroy Equation	9.732 (90.26%)	0.1935 (99.8%)	0.1195 (99.88%)	6.715 (93.28%)
2	Bagging Regressor	0.0184 (99.98%)	0.0185 (99.98%)	0.034 (99.96%)	0.021 (99.97%)
3	Decision Tree	0.0134 (99.98%)	0.0116 (99.98%)	0.0224 (99.97%)	0.015 (99.98%)

B. Predicting unseen data

We have also evaluated the prediction accuracy when the dataset was shuffled and also when the algorithm has never seen the data before. For this purpose, we have applied a random shuffle method in python to the dataset, and then split it into 70% training subset and 30% testing subset. We have repeated this process for the Arctic Ocean, Southern Oceans and Pacific Ocean. The training subset for the Arctic Ocean, Pacific Ocean, and Southern Oceans include 119,402, 30,840 and 24,052 sound speed values, respectively. The corresponding sizes for the test sets are 51,172, 13,217 and 10,308, respectively. The worldwide dataset refers to the combination of the Arctic, Southern and Pacific Ocean datasets. We further compared the results of the decision tree with the Mackenzie equation and observed that the decision tree outperforms the Mackenzie equation by 10% or more. The results are tabulated in Table 7.

Table 7: Comparing Sound velocity prediction using Mackenzie equation and Decision Tree (with dataset shuffling)

No.	Algorithm	Mean Difference (m/s)			
		Arctic Ocean	Indian Ocean	North Pacific Ocean	Worldwide
1	Mackenzie Equation	9.678 (90.3%)	0.1398 (99.86%)	0.1202 (99.87%)	6.916 (93.08%)
3	Decision Tree	0.0408 (99.95%)	0.039 (99.96%)	0.0716 (99.93%)	0.0465 (99.95%)

We used the same approach for testing with Leroy's equation, where the latitude and longitude are factored in. Again, decision trees outperform Leroy's equation by 10% or more. The results are tabulated in Table 8. The fact that the Decision Tree is able to predict unseen data with high accuracy in such a variety of regions is a significant advantage for our machine learning based sound speed estimator.

Table 8: Sound velocity prediction using Leroy equation and Decision Tree (with dataset shuffling)

No.	Algorithm	Mean Difference (m/s)			
		Arctic Ocean	Indian Ocean	North Pacific Ocean	Worldwide
1	Leroy Equation	9.744 (90.25%)	0.1935 (99.8%)	0.1197 (99.88%)	7.139 (92.86%)
3	Decision Tree	0.0436 (99.95%)	0.0391 (99.96%)	0.0728 (99.93%)	0.05 (99.95%)

C. Future prediction based on historical data

In order to gauge the effectiveness of machine learning algorithms in predicting unknown sound speed values in areas based on historical measurements, we have used data collected from the Arctic Ocean during the years 1999 to 2000 on SCICEX expedition as training data (same Arctic Ocean dataset as in section IV. A). The testing dataset includes sound speed measured at 27 positions of varying latitude and longitude for the years 2001-2002 in the Arctic Ocean during the SCICEX expedition; each position calculates the sound speed for depths between 13m and 1000m. This test dataset has a total of 34,276 records, each includes measurements of depth, salinity, temperature and sound speed. We have found that machine learning algorithms are indeed successful in predicting sound speed accurately for the years 2001-2002 and the results match with the measured values on the SCICEX expedition for the years 2001-2002. The results are shown in Tables 9 and 10.

Table 9: Future Sound velocity prediction using Mackenzie equation and Machine learning algorithms

No.	Algorithm	Mean Difference (m/s)
1	Mackenzie Equation	41.4515 (58.54%)
2	Bagging Regressor	0.0678 (99.93%)
3	Single Decision Tree	0.0903 (99.9%)

Table 10: Future Sound velocity prediction using Leroy equation and Machine learning algorithms

No.	Algorithm	Mean Difference (m/s)
1	Leroy Equation	41.4145 (58.58%)
2	Bagging Regressor	0.12 (99.88%)
3	Single Decision Tree	0.1175 (99.88%)

We observe from Tables 9 and 10 that the Bagging Regressor and Decision Tree methods achieve 99.9% prediction accuracy while the Mackenzie and the Leory equations are only 58.5% accurate. This validates our approach and confirms that the machine learning algorithms are thus invaluable and enable us to accurately determine sound speeds of the region under test

(Arctic Ocean in this scenario) sometime in the future using the current datasets.

D. Measuring the effect of training data size on accuracy

To assess the effect of the training dataset size on the accuracy of the sound speed prediction, we have used the shuffled worldwide dataset from Section IV.B again. After shuffling the dataset, we split it into various sizes, 40-60, 50-50, 60-40, 70-30, etc. A size of 40-60 means the dataset was split so that 40% of the dataset would be used for training and 60% of the dataset would be used for testing. The results are tabulated in Table 11. We notice that as the training dataset size increases from 40% to 80%, there is an increase in the accuracy of the sound velocity prediction. When the training dataset size is 80%, the accuracy is 99.95% and hence, it is very close to the actual values of sound velocity.

Table 11: Sound speed prediction for different training dataset sizes while using the Decision Tree algorithm

Mean Difference (m/s)				
Dataset size (40-60)	**Dataset size (50-50)**	**Dataset size (60-40)**	**Dataset size (70-30)**	**Dataset size (80-20)**
0.0599 (99.94%)	0.0543 ((99.94%)	0.0503 (99.94%)	0.0465 (99.95%)	0.0457 (99.95%)

V. CONCLUSIONS

Sound speed variations significantly impact signal modeling for underwater acoustic communications. Hence, it is critical to determine precise values of sound speed in applications like target recognition and IoUT networks. Although generic equations are available and other techniques utilizing a layered model have been explored, there is no clear universal method for all seas and all salinities. We have proposed using machine learning and regression algorithms in particular to predict sound speed. Using published XBT and Sonar data, we have been able to predict sound speed with more accuracy than the reference equations. Specifically, using datasets from various oceans like the Southern Oceans, Pacific Ocean and Arctic Ocean, an accuracy of 99.9% has been achieved for the SSP prediction. Our approach is naturally adaptive; as more data sets become available, they may be further incorporated to enhance prediction accuracy. Our future work will evaluate the impact of our approach on propagation models, e.g., Bellhop.

Acknowledgement: This work is supported by the National Science Foundation, Contract #0000010465. The authors thank Dr. Akram Ahmed for sharing insight about underwater sound speed and Dr. Wassila Lalouni for help in the evaluation setup.

REFERENCE

[1] L. E. Empokpae and M. Younis, "Throughput analysis for shallow water communication utilizing directional antennas," *IEEE Journal on Selected Areas in Communications*, 30(5), pp.1006-1018., 2012.

[2] S. Ryecroft, A. Shaw, P. Fergus, P. Kot, K. Hashim, A. Moody and L. Conway, "A First Implementation of Underwater Communications in Raw Water Using the 433 MHz Frequency Combined with a Bowtie Antenna," *Sensors*, 19(8), pp. 1813, Apr. 2019.

[3] L. E. Emokpae and M. Younis, "Surface based underwater communications.," in *Proc. IEEE GLOBECOM*, 2010.

[4] A. Ahmed and M. Younis, "Distributed real-time sound speed profiling in underwater environments.," in *Proc. IEEE International Conference on Communications (ICC)*, June 2017.

[5] D. Sun, H. Li, C. Zheng and X. Li, “Sound velocity correction based on effective sound velocity for underwater acoustic positioning systems,” *Applied Acoustics*, Vol. 151, pp. 55-62, 2019.

[6] P. Mikhalevsky, B. Sperry, K. Woolfe, M. Dzieciuch and P. and Worcester, “Deep ocean long range underwater navigation,” *The Journal of the Acoustical Society of America*, 147(4), pp. 2365-2382., 2020.

[7] S. Misra and A. Ghosh, “The effects of variable sound speed on localization in Underwater Sensor Networks,” *Proc. Australasian Telecom. Networks and Applications Conference (ATNAC)*, 2011.

[8] Z. Zhu, S. -L. J. Hu and H. Li, “Kalman-based underwater tracking with unknown effective sound velocity,” *Proc. MTS/IEEE OCEANS Monterey*, Sept. 2016.

[9] R. Khalil, M. Babar, T. Jan and N. Saeed, “Towards the Internet of Underwater Things: Recent Developments and Future Challenges,” in *IEEE Consumer Elec. Magazine*, doi: 10.1109/MCE.2020.2988441.

[10]K. Mackenzie, “Nine-term equation for the sound speed in the oceans,” *J. Acoust. Soc. Am.*, 70(3), pp. 807-812, 1981.

[11]G. W. a. S. Zhu, “Speed of sound in seawater as a function of salinity, temperature and pressure,” *J. Acoust. Soc. Am.*, 97(3), pp. 1732-1736, 1995.

[12]A. Coppens, “Simple equations for the speed of sound in Neptunian waters,” *J. Acoust. Soc. Am.*, 69(3), pp. pp 862-863, 1981.

[13]A. Ahmed, “Efficient Topology Construction and Autonomous Networking for Underwater Acoustic Nodes,” *Ph.D. Dissertation*, CSEE Dept., University of Maryland Baltimore County, 2019.

[14]C. Leroy, S. P. Robinson, and M. J. Goldsmith, “A new equation for the accurate calculation of sound speed in all oceans,” *The Journal of the Acoustical Society of America*, 124(5), pp. 2774-2782, 2008.

[15]S. Jain and M. M. Ali, “Estimation of sound speed profiles using artificial neural networks,” *IEEE Geoscience and Remote Sensing Letters*, 3(4), pp. 467-470, 2006.

[16]R. Lefort, G. Real, A. Drémeau, “Direct regressions for underwater acoustic source localization in fluctuating ocean,” *Applied Acoustics*, Vol. 116, pp. 303-310, 2017.

[17]H. Yang, K. Lee, Y. Choo, and K. Kim, “Underwater Acoustic Research Trends with Machine Learning: Ocean Parameter Inversion Applications,” *J. Ocean Eng. Technol.*, 34(5), pp. 371-376, 2020.

[18]J. Choi, Y. Choo, and K. Lee, “Acoustic Classification of Surface and Underwater Vessels in the Ocean Using Supervised Machine Learning,” *Sensors*, 19(16), pp. 3492, 2019.

[19]H. Niu, E. Reeves, and P. Gerstoft, “Source localization in an ocean waveguide using supervised machine learning,” *The Journal of the Acoustical Society of America*, 142(3), pp. 1176-1188, 2017.

[20]A. Horri, “Underwater Localization in a Confined Space Using Acoustic Positioning and Machine Learning,” *MS. Thesis*, ECE Dept. the University of Windsor, Windsor, Ontario, Canada, 2021.

[21]V. A. Belogol’skii, S. S. Sekoyan, L. M. Samorukova, S. R. Stefanov, and V. I. Levtsov, “Pressure dependence of the sound velocity in distilled water,” *Meas. Tech.*, Vol. 42, pp. 406–413, 1999.

[22]J. K. Fulford, “Cluster of sound speed fields by an integral measure,” *Proc. OCEANS 2009, MTS/IEEE Biloxi* - Marine Technology for Our Future: Global and Local Challenges, Biloxi, MS, 2009, pp. 1-4.

[23]M. R. M. Talabis, R. McPherson, I. Miyamoto, J. L. Martin, D. Kaye, *Analytics Defined*, Book Chapter, Information Security Analytics Finding Security Insights, Patterns and Anomalies in Big Data, Syngress, 2015.

[24]https://scikit-learn.org/

[25]P. Bühlmann, “Bagging, boosting and ensemble methods,” *Handbook of Computational Statistics*, pp. 985-1022. Springer, 2012.

[26]T. Hastie, R. Tibshirani, and J. Friedma, *The elements of statistical learning: data mining, inference, and prediction, 2nd Ed*. Springer, 2009.

[27]M. Krzywinski, N. Altman, "Classification and regression trees," *Nat. Methods*, vol. 14, No. 3. pp. 757–758, 2017.

[28]https://www.nodc.noaa.gov/archive/arc0021/0000568/. [5 Nov 2020].

[29]https://www.nodc.noaa.gov/archive/arc0033/0068160. [5 Nov. 2020].

[30]https://www.nodc.noaa.gov/archive/arc0001/0000681/. [5 Nov 2020].