Navy Messaging Analytics to Improve Readiness, Response and Preparedness

Raheleh Dilmaghani*, Kyle Turco†

*NIWC-PAC, San Diego

raheleh.dilmaghani@navy.mil

†PEO C4I, San Diego

kyle.turco@navy.mil

Abstract—There are tens of thousands of official messages exchanged within the Navy’s messaging system daily. The messages carry a tremendous amount of data that is not fully processed nor analyzed. If we could enhance extracting the value of these data, and automate, we could improve speed, readiness, performance, maintenance, outage scheduling, and much more. Although messages follow a prescribed format, it is not uniform across the message types and data entry is not enforced and prone to human interpretation or error in containing all relevant information and correct categorization or labeling. Often there is relevant information in other messages not directly linked or referenced, but if inferred could enhance the observer’s perspective and response to the situation. By analyzing the massive data stored within Navy tactical messages, we could learn about fleet readiness without asking a particular unit or group, “are you ready?” Better yet, we may be able to predict future readiness based on past schedule and performance data if correlated correctly to learn patterns that indicate future mission success. In this paper, we present results from a pilot project where we show the significance of processing available messages for relevant information by not limiting the message types that seem to be the right category or to those with an explicit keyword. This process helps to detect relevant information and create meaning correlations, which could contribute to more informed decision making or to provide the Commander information to make decisions proactively instead of reacting to negative outcomes like casualties or mishaps. Extracting relevant information in turn can contribute to more informed decision making in saving lives, reduced sustainment costs, or repair costs.

Index Terms—Navy Tactical messages, readiness, analytics

I. INTRODUCTION

Data analytics is not a new term, however, applying advanced analytics techniques in the domain of tactical and Navy is nothing like what has been explored in business and industry applications. Analytics is the discovery of meaningful pattern and information in data to enhance decision making and response. Companies frequently and commonly apply analytics to data they can collect from individual profiles to increase the benefit, and to improve business performance. Automating analytics can reduce operational costs by optimally allocating resources where and when they are needed. There are a large number of messages exchanged within Navy messaging daily, weekly, monthly, and discovering knowledge within is challenging due to data volume, variety, and velocity. There are different types and categories within messages. For example CASREP is a message type that can be analyzed

to reduce duration of in-service or maintenance of essential equipment by developing learning techniques, information retrieval, automatic concept mapping, etc. Another message type of interest is COMSPOT which is used to report actual or forecast outage. Within a message type, there are different categories to reflect on the severity of the problem and the urgency of attention needed to process. Processing the large amount of data requires scalable processing and automation to reduce human-made errors. Additionally the process is time consuming, and inefficient for human. Data mining techniques can predict future outage or enhance planning resources, response and maintenance.

In this paper, we present our research and results from a pilot project that demonstrates the significance and applicability of mining Navy messages to improve response and readiness and to guide future research questions that lead the selection of techniques for the best results. This research benefits Navy *3rd Fleet*, and SURFPAC in addition to Navy Messaging program of record.

The contribution of this paper is summarized as follows:

- to share research on proof of concept to understand data
- to demonstrate the value of the information that is implicit to data to enhance operational readiness
- to increase awareness in need of inferring knowledge from data not explicitly marked as such
- to improve operational readiness and response via comprehensive analytics techniques
- share existing research and to provide a direction toward a more complete implementation
- to automate processing and analytics for scale, speed and accuracy

Organization of the rest of the paper is as following: Related works are presented in Section II. Section III presents the problem statement, its significance and uniqueness in III-A followed by a description of Navy messages data and the unique challenge that is required to prepare data for mining in section III-B. Section IV presents a brief discussion on different machine learning (ML) algorithms we consider in this project. Section V presents a description of the pilot project, the objective, modeling, and compares the results of three different algorithms used to detect the *redline*. Section VI shares the lessons we learned through this project and Section

VII presents case studies and research questions for future work. Section VIII concludes the paper.

II. RELATED WORKS

With the ever increasing volume of data, the need for high performance data mining algorithms has become a necessity. This becomes more valuable when processing hundreds and thousands of Navy messages that collectively reveal more information than they do individually. Hence, additional pre-processing and training must be in-place to enable extracting relevant information from the pool of messages. Existing analysis techniques consist of two main phases; looking for certain message categories and performing partial analysis on those individual Navy messages. There is little related work in the literature. Looking at the aggregated data is novel in the sense of improving response and preparedness by extracting relevant information out of all the data available.

In [1], the authors present a dynamic distributed data structure, DD-Rtree, which preserves spatial locality while distributing data across compute nodes in a shared environment. Other Spatial data mining algorithms like Dbscan, Optics, and Slink use static spatial indexing structures like k-d-tree, quad-tree, and grid files where they need to scan the entire dataset to determine the partitioning coordinates. Considering the increasing volume of Navy messages, the necessity of a dynamic distributed data structure for efficient data mining and analytics is clear. Static spatial indexing will not be able to process and produce relevant information fast enough, limiting its applicability to tactical messages.

The challenge of mining very large database is an interdisciplinary research problem that includes data mining techniques, parallel processing, relational database, and storage. Navy tactical messaging analytics is a challenging task that requires context, domain knowledge, processing resources, and storage. In [2], the authors present a data mining system, RFDM (RHadoop-based Fuzzy Data Mining) that supports fuzzy data mining process with reduced cost. Scalable data mining techniques are required to process large volume of data to extract information from messages that may have not been tagged appropriately and yet contain relevant information to improve decision making and readiness.

In [3], the authors present their research on the use of a real-time enterprise data mining tool that along with the operator knowledge can be used to analyze data to improve Naval aircraft maintenance deficiencies. Their solution is able to handle massive combat data in real time to extract useful information and to assist decision makers with increased level of relevant information. Capturing operators knowledge and input will improve the context of the messages and mining results. This is a challenging task that can contribute to future continuation of this work.

Responding to an emergency, a critical maintenance, or a crisis, typically involves different entities who may or may not have higher authority information. In [4], authors present their work on the role of feedback among different entities. In [5], the authors present a modeling technique to illustrate how use

of a supervisory control solution on a discrete event system model can improve communication at the organizational level to improve decision making and response. In this work, the authors present challenges and lessons learned from real-life exercises they participated. Applying lessons learned as a feedback into the data mining techniques will improve learning and contribute to enhancing future results, response, and planning based on historical data and similar experience.

In [6], authors present their approach to rapidly develop a computational model for emerging domains where there is not a lot of domain-specific resources available. Their use case is to develop an evaluation technique quickly for the COVID pandemic. These types of approaches heavily rely on evaluation techniques that can be rapidly adopted even before a lot of data becomes available to support an urgent development. Considering the dynamic and unique nature of each platform and the diversity of Navy messaging types and categories, we can benefit from a rapid computational model which can adapt to a large volume of data in a short time.

In [7], the authors present LOLITA (Large-scale, Object-based, Linguistic Interactor, Translator, and Analyzer) system which is a natural language processor with a core functionality on which a number of different applications can be built. LOLITA is designed as a general purpose Natural Language Processing (NLP) system, developed at the University of Durham since 1986. The system is designed to provide NLP capabilities to support many applications in multiple domains by providing a core platform upon which different applications can be built. LOLITA core platform provides two main functionalities of analysis that converts text to a logical representation of its meaning, and generation that expresses information represented in the logical form as text. LOLITA is designed as a system to bring its knowledge of specific domains where and when applicable. The future work of the pilot project presented in this paper will benefit tremendously from an enhanced NLP capability in converting text to a meaningful representation of the information.

In [8], the authors present their work on the use of computational linguistics methods to process unstructured text, and to automatically extract meaningful information in order to provide users with a variety of information access tools appropriate to their specific needs. A natural language processing based, domain-independent information extraction framework is needed to convert the semantic content of unstructured texts into structured knowledge representations. This domain-independent and automatic semantic analysis of text, similar to human capability, has been beyond the computational capabilities of NLP. The novel information extraction framework described in this paper has been shown to be useful in populating knowledge bases for automated reasoning applications, for question-answering when users have very specific information needs, and for populating a visual browser for when users' needs are vague.

In [9], the author presents a comprehensive summary of information extraction from [10], with trends in information extraction, automated knowledge acquisition, and bridging

the gap between information extraction and understanding the story. These two paradigms present a tradeoff and the challenge between scalability and depth that can together leave to a broad in-depth NLP system with understanding the big picture. Understanding and analyzing Navy messages to infer and enhance relevant information extraction will contribute to better planning, readiness, and response.

In [11], the authors present a new language representation model called BERT (Bidirectional Encoder Representations from Transformers) that pre-trains deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications. A similar language representation model, can be used to adopt the data mining technique to a similar category or a sub-set of data. For example, a data mining technique valid for afloat platforms, can be modified with one additional output layer to create a model for a specific category of afloat platforms.

In [12], the authors present their research results from a user case study where they investigated users' preferences in question-answering systems. These results can impact user interface design. They found out that users prefer a paragraph-sized chunk of text over just an exact phrase as an answer to their questions. Additionally, they found out that the users generally prefer answers embedded in context, regardless of the perceived reliability of the source documents. When users research a topic, increasing the amount of text returned to users significantly decreases the number of queries that they pose to the system, suggesting that users utilize supporting text to answer related questions. Being able to produce just-enough relevant information with the goal of not overwhelming the user who is already pressed to report or overwhelming a commander who needs to make a decision in fractions of a minute, is of high interest and can increase usability of the solution.

Extracting the valuable information in large volume of Navy messages that follow a prescribed format but it is not uniform across categories and types, makes the problem unique and challenging.

III. MINING NAVY MESSAGING

Processing Navy messages with significant information is challenging due to volume, variety, velocity, and the fact that they are not precisely following the guideline and the template.

A. Problem Statement and Significance

Why does mining Navy messages matter and how is this application different? Navy messages carry a tremendous amount of information that may not be fully processed or analyzed. It is important to note this challenge cannot be solved by simply indexing messages or searching the given data within the messages for a discrete answer. The answer to questions that could help a commander make timely critical

decisions may not be in a given message or may not be directly or properly tagged as relevant information. Individual questions or queries, that may ask the right question, rely on the responses from the individual crafting the message and what he or she knows or could gather in a short time. Many times, the right question may not even be asked or known until there is little or no time to ask it. All the preceding situations lead to time consuming, error-prone, and inefficient data calls to collect answers that likely are incomplete or irrelevant once transmitted and processed by the requestor.

A simple use case illustrates this situation, the case involves a Commander needing to know if a particular unit is ready and capable of performing an emergent mission. No one knows when to ask until the very moment and by the time the question is asked or identified as relevant or irrelevant, it may be just too late. However, the Commander likely understands the set of missions that could arise in a given operating area in the current climate. What if one could define and refine those mission sets continuously and in turn an algorithm could calculate the ready and capable answer for each mission by unit based on all available data, past and present? The Commander could have the answers in near real-time and make timely and relevant decisions. This scenario is exactly what advanced machine learning efforts should be designed to solve.

While the technology is maturing and we have only begun to explore its potential impact to our ability to Command and Control forces, it is not only about technology. There is an equally significant shift required in how we generate, store and disseminate data that must take place. This is an end-to-end overhaul of the Command and Control process that will be enabled by machine speed answers to nearly any question whose answer can be derived from the U.S Navy's data ocean, if we had one. Fundamentally we have to shift our thinking and stop asking a question hoping to be given a timely discrete answer, and instead ask all of our questions all of the time and accept continuously refined answers inferred from all available data, to include a predicted outcome based on the potential choices made. The operational relevance of this capability cannot be overstated, it is safe to say that regardless of scale or location, most past and all future conflict's outcomes will be primarily driven by information; those that have it in real-time and influence what information is available to an adversary will succeed. Those who are stuck conducting data calls or combing through megabytes of message traffic, will not.

B. Semantics of Nav Messages Data

Before we get started, we needed to study the data to learn more of the following:

- what tactical messages are keys in impacting decision making?
- what section of those messages carry critical information and impact those decisions?
- what are the examples of words that if exist together in a message, carry a meaningful flag?

- what are the components or issues that a ship can never sail without those?
- in the example of "Bonhomme Richard", were there information that if noticed early, would prevent the damage?
- in the case above, was there a message type that should have been considered yet not urgent or red flagged?
- what parts of messages or types of different messages together can be considered to predict or to prevent?

These are the questions that show the scope of large volume of data that can benefit from automation. No human is able to process such volumes of data. Advanced analytics and machine learning techniques will bring out the information which is critical for decision making, readiness and response. There are many questions that would help fleet and decision makers to be able to predict, plan, and respond. As an example we were looking at a use case on what the indicators of a ship crossing *redline* are. There are certain message types and categories which explicitly contain the word *redline* and hence easy to detect. However based on several conversations with stakeholders, we learned that there are other types and categories which may not contain the keyword *redline*, yet unable to sail because of a mechanical problem. These messages need to be captured in the analysis. The ultimate objective is to be able to predict whether a ship will cross the *redline* tomorrow or will it be ready for a safe deployment tomorrow.

A *redline* occurs when a ship does not meet a threshold essential for operational, deployment, supply or other mission critical requirements. When a ship crosses a *redline*, it will not be operational or fully mission capable which impacts the vessel's ability to be deployed and ultimately impacts the readiness across the Fleet. Crossing *redline* is a good use case to study as occasionally the guidelines to indicate the state is not precisely followed and if the very specific phrases, such as *redline crossed*, are not included, it may simply go unnoticed. Needless to state the significance and the high risk that such ignorance can lead to. There are different causes that may lead to crossing *redline*, from ammunition, fuel, communications, logistics, and other ship components, in which the severity, the mitigation, and ultimately planning and response vary. It is critical that such information properly detected, processed and evaluated to help answer the question for future readiness and response.

IV. MACHINE LEARNING ALGORITHMS

A. Logistic Regression (LR) Model

Logistic Regression is a type of supervised machine learning used for classification tasks. In this use case, the model takes the inputs and classifies whether the ship is ready or not ready. Logistic regression provides a binary classification, however the underlying statistics are how probable one class, class one, is versus the other, class zero. This enables us to understand the probability of a ship being ready to deployed based on past data. For this use case, the data has been divided to a training set and testing set to assess how the model performs.

If the wording in the message suggested that a *redline* was crossed, then that message would be labeled with a one, to indicate a *crossed redline*, or otherwise, labeled with a zero to indicate no *redline* crossed. Upon completion of the labeling, the supervised learning model was trained to predict a ship crossing the *redline*.

Several editorial modifications were made to the messages to prepare it for modeling. The next processing step was to eliminate any mention of the words *redline* or crossed so that they would not skew the results. We did not want the model to just score messages based on presence or absence of that word or phrase, rather to inspect the context around the phrase that indicates an issue related to *redline*. There are many varieties of the phrase *redline crossed* to account for, such as *crosses redline*, *redline crossed*, *redline crossing*, and more that we used the regular expression to account for such phrasing.

We used similar regular expressions to remove emails, all non-alphabetic characters, any length one strings, common email tags such as ref, rel, subj, and month abbreviations. The standard practice for NLP is to lemmatize the text, then convert the messages to vectors with a form of embedding or vectorization (usually term frequency, inverse document frequency or TF-IDF). Additional features were integrated to the purely text-based TF-IDF features since the data set had metadata associated with each message. Altogether, these additional features appended 31 extra columns on to the feature vector for each message. Table I present the statistics regarding the size of the training and test sets:

In general, logistic regression performs better when the number of noise variables is less than or equal to the number of explanatory variables and random forest has a higher true or false positive rate as the number of explanatory variables increases in a dataset.

If the goal is to make a prediction or describe data, logistic regression is often a poor choice. Limitations of logistic regression are high reliance on a proper presentation of the data, meaning that logistic regression is not a useful tool unless we have already identified all the important independent variables. Since its outcome is discrete, LR can only predict a categorical outcome.

B. Random Forest (RF)

Decision trees are usually superior to logistics regression as they are easier to interpret, and to predict better. Decision tree performs faster and as well as logistics regression or even neural networks. Decision trees are expensive in terms of sample size. Each time it splits data using a predictor, the remaining sample size reduces and eventually there is not enough data to identify further predictors. By contrast, logistic regression looks at simultaneous effects of all predictors so can perform much better with smaller sample size.

Random Forest aggregates the results of a collection of decision trees and they are popular as they can process data as is and reduces variance by training on different samples of data.

TABLE I
THE SIZE OF DATA SET

type	TF-IDF	Additional	Total
Features	13398	31	13329
-	Train	Test	Total
Observations	8437	2554	10991

C. Support Vector Machines (SVM)

Support Vector Machine is widely regarded as one of the best text classification algorithms with high accuracy. This technique relies on the concept of *distance* between different point to maximize the margins. SVM models perform better in document classification where there are a lot of features and in any document (message here), only a small fraction of these features have a considerable value greater than zero.

D. Other Algorithms

Another common way of analyzing data may be using transformer networks. This technique usually starts with a pre-trained question answering model trained on a large dataset and then fine tuning the model on the target dataset. For example you may have a model for vehicles. Later on you are interested in trucks which is a subset of vehicles and hence transform modeling can be used instead of developing a model from scratch. There are open source codes to build an open domain question answering system.

Zero shot training refers to when you train on vehicles data set for example and then you test your model on truck dataset. The bigger the dataset the modeler is trained with, the more relevancy the results have. A key factor in impacting the response is how much weight to put on the layers or previous input.

We considered using BERT, conceptually simple and empirically powerful, [13] and [14], a state-of-the-art neural network model that can be used for transfer learning. BERT can be used to directly score the similarity between any two documents without directly creating a document embedding. In this mode, one could score all pairs of documents, then use k-nearest neighbors to perform classification. This is significantly more computationally intensive and would benefit from a GPU. One should also fine-tune BERT by retraining the top layer of the neural network for this specific data set or task, which would require a GPU. It is unlikely BERT would justify the extra effort since this type of unusually formatted text with much domain-specific vocabulary is not amenable to transfer learning.

V. EXPERIMENTATION

A. Project Description

A *redline* occurs when a ship fails to meet an objective threshold essential for operational, supply, or other mission critical requirements. When a ship crosses a *redline*, it may no longer be operational or fully mission capable, impacting the vessel's ability to deploy and ultimately impacting readiness across the Fleet. Through engagement with Tactical Messaging

stakeholders, the Third Fleet, at the end of July 2020, the team learned how important *redline* is to the Fleet, and that some ships do not follow the guidelines for tactical messaging by failing to include specific phrases, such as *redline crossed*, when a *redline* has occurred. There are also many different types of *redline*, from ammunition, fuel, communications, logistics, and other ship systems, which typically are communicated through CASREP message types. Personnel related *redline*, such as medical, are usually covered in other message types.

The objective of our pilot project is to process the incoming messages and to flag the ones that have a high probability of crossing a *redline* without explicit mention of the term. These messages require immediate attention, and they will otherwise fail to get the attention they need, which may have severe impact on the vessel's ability to deploy, on the mission readiness, on lives, and cost.

There are two types of learning in general, supervised learning and unsupervised learning. Unsupervised learning is a type of machine learning that is used to infer information from data without labeled response. An example is topic modeling, where the data is clustered into topics based on discrete word frequencies and corpus statistics. The caveat to unsupervised learning is that one cannot guarantee the model is inferring information from the given dataset as there is no feedback. The significance of feedback in collaboration and crisis response is discussed in [4]. We are interested in supervised learning where an input is mapped to an output based on labeling the training data. In this case study, there is no labeling available and hence the data needs to be processed to develop relevant labels and ultimately to automatically create those labels.

- group ships by class
- establish the mean values for the input parameters such as the number of CASREP messages, categories, the percentage of fuel, the mechanical problem, the number of sick staff, etc.
- define the label, ready or not ready as output and create labels
- define the margin to indicate readiness, for example two standard deviations away from the mean

The above process creates the labels for messages. After producing labels, a machine learning algorithm can be trained to analyze how the inputs impact the outputs.

B. Assumptions and Model Description

The team's working environment consisted of a single laptop with no GPU. The team installed Ubuntu 20.04 onto the development machine along with an offline install of

the necessary Data Science, NLP, and Machine Learning Python packages required to leverage various ML and AI algorithms and preprocessing capabilities. The team primarily worked with Python 3.7 through Jupyter Notebooks in the development environment.

Our goal was to create a model that can classify CASREPs as crossing the *redline* when the word does not appear in the message. We created a training data set which consisted of 14,000 CASREPs dated from March to July of 2020. After eliminating duplicated messages, we were left with about 11,000 CASREPs on which we built and tested a Natural Language Processing (NLP) model. One use case for such a model is to triage incoming messages and flag ones that have a high probability of crossing a *redline* according to our model but lack the term *redline*. These messages require immediate attention, and they might otherwise fail to get the attention they need, which can have a negative impact on the vessel's ability to deploy or be mission capable. The Third Fleet estimated that about half the ships do not adhere to tactical message guidelines when it pertains to the *redline*. We noted that out of 149 total ships that appeared in this data set, only 65 of them ever used the term *redline*. It is possible that the other 84 ships never encountered a *redline* situation, but that seems unlikely, so this validates the Third Fleet's intuition. The team's working environment consisted of a single blank SIPR development machine with no GPU inside the OASIS lab at NAVWAR. The team installed Ubuntu 20.04 onto the development machine along with an offline install of the necessary Data Science, NLP, and Machine Learning (ML) Python packages required to leverage various ML and AI algorithms and preprocessing capabilities. The team primarily worked with Python 3.7 through Jupyter Notebooks in our development environment.

C. Model Parameters and Results

We selected three standard models to perform classification: Random Forests (RF), Support Vector Machines (SVM), and Logistic Regression (LR). We mainly used the default parameters since tuning did not yield much improvement. The one parameter which helped slightly was added regularization, which correspond to parameter C in SVM and LR, where smaller values mean stronger regularization. We found $C=0.5$ to be optimal.

The `RandomForestClassifier()` from `sklearn.ensemble`, `LogisticRegression(c=0.5)` from `sklearn.linear-model`, and `SVC(probability=True, C=0.5)` from `sklearn.svm` have been leveraged.

Table II present the results from the three models with and without the additional features (AUC is area under the ROC curve):

The operational use case for this kind of model is to triage incoming CASREPs to highlight the ones that are likely *redline*, but are not explicitly noted as such. Many of the test observations that the model labels as false positives are actually true positives that were not labeled as such since they did not mention *redline*. The accuracy of the model is the most

important at low (1-5%) False Positive Rates (FPR). The ROC curves for the models showed a clear separation there; in that low range of FPR, logistic regression had a true positive rate around 45 %, which was significantly better than SVM or RF.

Feature importance offers some insight into how the models work. Here we present a brief selection of the most important features for two models; a more in-depth study is available in the classified Jupyter notebooks. RF and LR have built-in methods for feature importance, so we chose to use those models for this analysis; feature importance can be derived from the SVM model as well, but it requires added functionality. For RF the feature importance is calculated automatically and stored.

The most important feature for RF is the word *waiver* which fits our intuition because when a ship is in *redline* it needs to request a waiver to continue, so many *redline* messages also contain the word *waiver*.

The feature importance for LR can be derived from the magnitude of its coefficients which are stored. The single most important feature under this metric is the additional variable *CATEGORY* which also make sense because *redline* messages almost always have CASREP category 3 or 4.

VI. LESSONS LEARNED

Processing Navy messages with significant information is challenging. Considerable amount of pre-processing is needed to make data ready for analytics. Approximately half of the users do not adhere to the exact guidelines when an incident happens and hence processing of all data with relevant information becomes more challenging. The case presented in this paper demonstrates the importance of inferring information and running advanced analytics on Navy messages. *waiver* is important to be considered along with the *redline*.

There are many dimension that can place a ship on a *redline* status. Leveraging machine learning algorithms can assist in identifying and extracting those conditions. Understanding and analyzing a ship's mechanical and supply status, maintenance schedule, and requirements can assist in improving readiness and planning. An automated mechanism will speed up the response and reduces the probability of human-made errors. This provides foundation for an *Alexa* for the ship to return real-time information to assist with decision making.

Additional questions that were explored initially in this pilot project, include number of a certain category message onboard a platform. Another question to look into was whether the ship reported any ordnance shortfalls in their most recent relevant category message. Analyzing aggregated data would help to answer questions such as whether the ship can perform the expected number of days with normal speed and operation without refueling. More advanced techniques can help to improve analysis results by analyzing past messages and incidents to improve information relevancy.

This pilot project helped to identify high priority questions that can be processed using advanced machine learning techniques to infer valuable relevant information which cannot be found by indexing or looking at individual messages. It

TABLE II
MODELS COMPARISON

Model	Time (Sec)	AUC (Text only)	AUC (Add. features)
RF	0.9	0.7	0.85
LR	0.15	0.73	0.92
SVM	10	0.68	0.91

illustrated how the workflow can be improved to remove redundancies and to extract correlated information. We looked at an objective threshold that requires a clear definition. There are many aspects that could impact a ship's readiness status in operations, maintenance, personnel, logistics, security, etc. and leveraging advanced machine learning techniques can provide immense value extracting such information.

VII. FUTURE WORK

Capturing and analyzing all relevant information within the large dataset improves readiness and response. Leveraging machine learning techniques to assist in analyzing multiple issues, a large number of messages with different formats can lower the risk of error and to continue on normal operation. In addition to many unique challenges of Navy messages analytics, there are quite a few research questions to be considered in future modeling including but not limited to the following:

- Can the model learn conditional property?
- Which model works better on smaller training data?
- Which model is suitable when we have correlated information in data?
- Do we want to model the decision boundary between the classes or model the actual distribution of each class?
- Can the model learn joint probability based on the data?
- Which algorithm is best for text classification?
- Which model learns conditional probability?
- What happens when we give correlated features in a model?
- Which models works very well even on less training data?
- Which model will take less time to get trained?
- Which model will take less time to predict output?
- Which model fails to work well if we give a lot of features?
- What happens when training data is biased over one class in a Model?
- Which model is more sensitive to outliers?
- Will you be able to fill out the missing values in a dataset with these models?

There are many possibilities of word combinations that together may reveal additional relevant information within the context of messages.

Furthermore, data processing techniques can enhance extracting the results from a large data set that may contain typographical errors or alternate choice of words that may contain pertinent information such as more precise problem description, what actions have been taken to mitigate, the updates to the original message, a list of certain types and categories of messages on a platform, and more.

VIII. CONCLUSIONS

Redline is one of many important characteristics across the Navy and a better understanding of when, how, and why *redline* occurs can prove beneficial to readiness and other operational requirements. This pilot project focused on investigating whether the ships have crossed the *redline*, and presented the analysis and pipeline that can be expanded to include real time processing to flag the ships that are in *redline* and yet not explicitly noted.

This work is being further expanded to evaluate other types of messages and to automate the analysis to increase efficiency and to reduce error. Extracting relevant information can help a commander to make critical decisions with related data, as a result of aggregated large amount of data to infer the relevant information from all data. Ultimately this work can be extended to include processing data from additional external resources that carry valuable information.

ACKNOWLEDGMENT

The authors acknowledge contribution of Mr. Tyler Lewis, Mr. Spencer Rodrigues, Ryan Lance, and other team members at Booz Allen Hamilton throughout a short pilot project. The authors would like to thank Mr. Michael Smallwood and the entire messaging team for their support on providing the data to the performer.

REFERENCES

- [1] J. S. Challa, P. Goyal, S. Nikhil, A. Mangla, S. Balasubramaniam, and N. Goyal, "Dd-rtree: A dynamic distributed data structure for efficient data distribution among cluster nodes for spatial data mining algorithms," vol. 20, Feb. 2016, pp. 27–36. [Online]. Available: <https://ieeexplore.ieee.org/document/7840586>
- [2] P. Sun, L. Xu, and H. Fan, "R haadoop-based fuzzy data mining: Architecture, design and system implementation," vol. 20, Feb. 2016, pp. 1–52. [Online]. Available: <https://ieeexplore.ieee.org/document/7509796>
- [3] H. W. Lin, J. Chen, and Y. Song, "Warship combat decision support system based on data mining technology," vol. 166, 2020, pp. 248–251. [Online]. Available: <https://doi.org/10.1016/j.procs.2020.02.105>
- [4] B. Hudgens and A. B. Bordetsky, "Feedback models for collaboration and trust in crisis response networks," 2009, pp. 1–8. [Online]. Available: <https://ieeexplore.ieee.org/document/4755454>
- [5] R. Dilmaghani and R. Rao, "Supervisory decision making in emergency response application," 2013.
- [6] R. Tang, R. Nogueira, E. Zhang, N. Gupta, P. Cam, K. Cho, and L. J., "Rapidly bootstrapping a question answering dataset for covid-19," 2020. [Online]. Available: <https://www.semanticscholar.org/paper/Rapidly-Bootstrapping-a-Question-Answering-Dataset-Tang-Nogueira/0f995b05821b58b02e914422b56fba615d0e8d7f>
- [7] R. Morgan, R. Garigliano, P. Callaghan, S. Poria, M. Smith, A. Urbanowicz, R. Collingham, M. Costantino, C. Cooper, and the LOLITA group, "Description of the lolita system as used in muc-6, in sixth message understanding conference (muc-6)," 1995.
- [8] O. N. P. and W. Paik, "The chronological information extraction system (chess)," 1998.

- [9] R. Ellen, "Information extraction as a stepping stone toward story understanding, in understanding language understanding: Computational models of reading," 1997, pp. 1–8.
- [10] A. Ram and K. Moorman, "In understanding language understanding: Computational models of reading."
- [11] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019, p. 4171–4186.
- [12] J. Lin, D. Quan, V. Sinha, K. Bakshi, D. Huynh, B. Katz, and D. Karger, "What makes a good answer? the role of context in question answering," 2003, pp. 25–32.
- [13] C. McCormick, "Question answering with a fine-tuned bert," 2020. [Online]. Available: <https://mccormickml.com/2020/03/10/question-answering-with-a-fine-tuned-BERT/>
- [14] R. Horev, "Bert explained: State of the art language model for nlp," 2020. [Online]. Available: <https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>