

How Machine Learning is Improving U.S. Navy Customer Support

Michael Powell, Ph.D.,¹ Jamison A. Rotz,¹ and Kevin D. O'Malley, Ph.D.²

¹ Manada Technology, LLC, Harrisburg, PA USA

² Naval Information Warfare Center, Pacific

powell@manadatechnology.com, rotz@manadatechnology.com, kevin.o'malley@navy.mil

Abstract

The U.S. Navy is successfully using natural language processing (NLP) and common machine-learning (ML) algorithms to categorize and automatically route plain text support requests at a Navy fleet support center. The algorithms enhance routine IT support tasks with automation and reduce the workload of service desk agents. The ML pipeline works in a five-step process. First, an archive of documents is created from various sources, including standard operating procedure (SOP) memos, frequently asked questions (FAQs), knowledge articles, Wikipedia articles, encyclopedia articles, previously closed support requests, and other relevant documents. Next, a library of words and phrases is generated from the archive. Then, this library is used to vectorize an incoming support request, producing a term frequency inverse document frequency (TF-IDF) vector. Following, the TF-IDF vector is used to compute similarity scores between the support request and the documents in the previously-created archive. Finally, the similarity scores are processed by support vector machine (SVM) classifiers to categorize and route the incoming support request to the correct support provider. This algorithm was deployed at a U.S. Navy customer support center as part of a pilot study, where it decreased the amount of time agents spend on tickets by 35%; the amount of time required to assign tickets by 74%; and the amount of time to close tickets by 60%. Our internal tests show that, with an error rate of 2%, a 35% reduction in ticket volume could be achieved by fully deploying these algorithms.

Introduction

Nowhere has artificial intelligence (AI) and machine learning (ML) been more disruptive and influential than in the U.S. Navy. From unmanned aerial vehicles over the sea, to sophisticated cyber defenses on our ships, to unmanned submersibles under the sea, ML has made our nation safer, our defenses stronger, and kept our soldiers and sailors out of harm's way. ML, however, is also being used in many aspects of military support far beyond the battlefield. One of

these areas is in the realm of military customer support centers. Advancements in ML and NLP are reducing the cost of labor and saving the DoD precious taxpayer dollars.

The US Navy has relied on 24/7 customer support centers for decades to assist sailors with a wide range of problems, including aircraft maintenance, IT support, shipboard equipment, and logistics. These customer support centers collectively process more than 1,000 requests each day, the majority of which are submitted as text via email. In the past, customer support agents were typically required to read and analyze the text for routing to the correct support provider for resolution. Robust NLP algorithms, however, are now assisting support request center personnel by reading incoming requests, categorizing them according to the requesting sailors' needs, and automatically assigning the requests to the correct support provider organization. The algorithms are also responding to queries with approved solutions to a wide range of naval issues.

The U.S. Navy's NAVY 311 program, operating in the program executive office for enterprise information systems, developed a capability that improved upon current information technology (IT) help desk and customer relationship management (CRM), specifically in parsing and analyzing help desk communications, reports, and logs. The program operates a 24/7/365 customer support center primarily for Navy sailors seeking help to common non-tactical problems when underway and ashore. The resulting predictive algorithm is being used to employ technical enhancements to the military IT support and the CRM domain.

Background

The U.S. Navy operates the NAVY 311 program as the CRM component of the Navy's Distance Support capability. The program's support center is a single point of customer service entry into the Naval ashore infrastructure and network of fleet support providers. Through NAVY 311, the

fleet, sailors, military families, and civilians can get on-demand information assistance for non-emergency, non-tactical issues. This gateway to comprehensive support assists with issues ranging from shipboard systems and equipment, information technology, technical data, personnel (e.g., career, manpower, training), supply and logistics (e.g., requisition follow-ups and household goods), and installations and facilities (e.g., environmental, public works, community support). The network of service centers provides customers with a common support experience because the entire community of providers all abide by the same business rules and minimum service levels.

The demand for effective and efficient support services at the Navy service desks grew tremendously as technologies continued to advance. The advancements required frequent integration for the service desks to keep pace with customer demand. Customer service applications in the Navy, however, still required manual input and coding into CRM systems to correctly deal with the needs of sailors seeking help from ships and sites around the world.

The commercial world has faced the same difficulties. In private industry, customer support is such an integral part of success that many companies have customer service departments with dedicated representatives/agents to provide support and help customers resolve issues they encounter (Molino, 2018). Today, technologies such as NLP and ML are helping customer service centers to better understand how to respond to customer feedback and questions at support centers. Customer service is a differentiating capability for companies, but it faces significant challenges due to the growing individualization and connectivity of products, the increasing complexity of knowledge that service employees need to deal with, and steady cost pressure (Koehler et. al., 2017). Interestingly, the most significant performance improvements at service centers are when humans and machines work together. Through such collaborative intelligence, humans and AI actively enhance each other's complementary strengths: the leadership, teamwork, creativity, and social skills of the former, and the speed, scalability, and quantitative capabilities of the latter (Wilson, 2018).

Similar to commercial customer support centers, the Navy needed an improved service center capability. The Navy was looking for tools that could search content by meaning and context; and also search through different media, including email, telephone, chat, and browsing history to guide and enhance search results for the fleet. For example, recent years have seen growing interest in conversational agents, such as chatbots, which are a very good fit for automated customer support because the domain in which they need to operate is narrow (Hardalov, 2018). The new capability also needed to promote search results for emergent and common issues when several other users entered similar search queries, predicting instantaneous, recent, and widespread trends in user searches.

Research in the field shows ML being used extensively in commercial customer support. Gupta, Gilbert, and Di Fabrizio (2013) developed a ML classifier to identify emotional emails sent by customers and suggested that the classifier can be used to automatically route these emails to specialized representatives. Also, Sarikaya, Hinton, and Ramabhadran (2011), Xu and Sarikaya (2013), and Gupta et.al. (2006) built models to detect intent and to extract named entities for call classification and routing for NLP and speech recognition systems. Many of these systems were investigated to compare pros and cons and determine utility in a military support setting.

Many options existed in the marketplace from which to choose. Current state-of-the-art technologies address customer support needs with ML cloud services. Google, as just one example, developed many pre-trained models in the form of APIs to handle the ML needs of customer support (Google, 2019). Google offered APIs such as Cloud Vision API, Cloud Speech API, Cloud Jobs API, Cloud Translation API, Cloud Video Intelligence API, and the Cloud Natural Language API. These APIs include many technologies that can help companies better understand how to respond to customer feedback and questions, and ultimately improve their customer service (Forest, 2017). The APIs, however, only provided general tools to begin an approach to address the Navy's requirements. The tools were limited to analyzing stored state data sets within online searchable databases. The products were generally an online response that could not be consumed as a service by the Navy 311 CRM system.

Although these APIs and others proved useful for many customer support centers in the private sector, the Navy had unique requirements, such as encryption, classification, and an existing worldwide CRM environment embedded with the warfighter. A specific implementation was needed that outperformed results from current government agency business processes and ad hoc methods in use at the time.

IBM Watson is a cloud-based deep learning system that processes natural language and other data resources to provide predictive analytics (High, 2012). While IBM Watson is an impressive enterprise-grade automation-focused AI system, in the CRM space it is focused more on pre-packaged solutions for answering basic customer questions than on specific implementations for performing advanced processing. IBM Watson does provide some speech-processing modules; however, they are focused on customer interaction and basic system tasks, like profile management.

BMC Remedy offers some basic automation tools, including routine and recurring network tasks, enterprise job scheduling, and batch processing. Remedy's TrueSight Orchestration program offers workflow coordination to automate tasks across applications; however, their focus is on pre-built content that is not specific to any individual support center's needs (BMC, 2019). Remedy plans to offer

more automation tools in the future and does not offer customer interaction protocols.

Genesys offers some predictive routing capabilities, automation of routine tasks, and chatbots that are designed to initiate conversations with customers. However, this is only offered as a part of their complete CRM package. Genesys's AI tools would require costly system and software migrations to implement.

Upon assessing the competition in this space, it became clear that none of these offerings provide a comprehensive solution that addresses all the capabilities in a fully accredited environment: These competing solutions are not capable of performing the complex tasks required to support Navy contact center business rules. Our modular, cloud-based solution will offer real-time performance, flexibility to integrate with logically and physically separated systems, and the ability to upgrade individual model components over time as technology evolves. As the call center landscape changes, it is quite likely that the Navy will choose to implement new or different CRM, telephone, and customer interaction software in the future. It is unlikely that these changes will take place across the board at Navy call centers and helpdesks in a consistent manner at the same time. Our solution provides a forward-looking, state-of-the-art system that will consolidate institutional knowledge and provide AI and automation on a platform that can remain constant and relevant as the Navy customer support community evolves.

After extensive research and testing, the NAVY 311 program's solution was to build a custom system using SVMs and a cloud ML engine. A deep-learning approach was shown to be more computationally expensive in the cloud environment, unacceptably lengthy in updating the training set, and underperformed the customized model of SVM algorithms. The results were similar to other previous studies, such as Spens and Lindgren (2018) who show that SVMs outperformed neural networks in many parameters in a customer support environment.

The initiative proceeded by training a prototype with examples of content, including classifiers and solutions, and then pushed the algorithm into testing and production. The NAVY 311 program completed necessary engineering, system integration, packaging, and testing to field the capability into Navy customer support centers.

Methodology

The NAVY 311 program developed a supervised machine learning system using SVM algorithms that both improved the customer experience and saved the Navy significant funding for support centers. The ML system was inspired by the Explicit Semantic Analysis (ESA) method of Grubilovich and Markovitch (2009). The ML system performs

the following five operations: 1) Builds an archive of documents 2) Mines this archive to develop a library of words and phrases, 3) Vectorizes support requests via TF-IDF, 4) Transforms TF-IDF vectors into vectors of document similarity scores, and 5) Processes these vectors of similarity scores to predict the support request's categorization.

Generate an Archive

The first step in the data pipeline was to generate a corpus by gathering technical and non-technical documents. The documents in the corpus, or archive, covered a wide range of concepts and ideas relevant to the support requests that are to be categorized. After being generated, this archive can be referenced by the algorithm when predicting support request categorizations.

The algorithm builds this archive primarily from two main sources: Wikipedia articles and previously-categorized support requests. Wikipedia articles are included because they cover a wide range of subjects that are featured in support requests. For example, a support request asking for a password reset would have a lot in common with the Wikipedia articles for "password" and "technical support." Other document sources for this archive include technical manuals, support center website FAQs, and standard operating procedures.

Develop a Library of Terms

The previous step of this algorithm generated an archive of documents. The second step mines that archive to develop a library of words and phrases. This library must be comprehensive and must contain words and phrases that are commonly present in support requests.

This library was developed by reading every document in the archive, and adding every word and phrase to the library that met the following two conditions: 1) The word or phrase occurs in typically at least 1% of documents in the archive, and 2) The number of occurrences of the word across all documents is equal to or exceeding the number of documents. The former rule ensures that any given term has a broad enough reach across documents, while the latter ensures that the word is common enough. For this application, this process generated a library that has approximately 300,000 words and phrases.

Vectorize Support Requests

The third step of the algorithm is to transform support requests into numeric vector representations. In this step, a support request is cross-referenced with the previously developed library of words and phrases to create a term frequency inverse document frequency (TF-IDF) vector. A TF-IDF vector is a vector that states the frequency of each word or phrase in the support request (without reference to the relative locations of those words and phrases, i.e. a bag

of words approach), and weights those occurrences according to their relative frequencies in the previously discussed archive of documents. This process automatically weights common “stop” words (e.g. “a,” “the”) minimally while emphasizing domain and problem-specific terms (e.g. “password”, “drone”) (Salton & Buckley, 1988). Typically, most entries in TF-IDF vectors are zero because most support requests contain only a small fraction of the words and phrases present in the library. At the end of this step, the text of a support request has been transformed into a numeric representation.

Compute Similarity Scores

The fourth step of this algorithm is to compare the support request to each document in the previously generated archive. Each document comparison results in a single similarity score, and is computed as the correlation coefficient between the support request’s TF-IDF vector and the document’s TF-IDF vector.

This process is scalable because new documents are only compared to documents that have been saved in the archive, and NOT to all documents that have previously been processed by this algorithm. (E.g. If we previously constructed an archive of 3,000 documents, then each new document that is being processed by this pipeline would be converted into 3,000 individual similarity scores.)

Similarity scores are a proven method for text classification (Dalal & Zaveri, 2011; Hu, et al., 2008; Li, et al., 2011). This step transforms the information-sparse (i.e. most values are zero) TF-IDF vector into an information-dense similarity-score vector.

Predict Categorizations

The final step of this algorithm is to predict the support request’s categorization from its vector of similarity scores. The prediction is accomplished with SVM classifiers, which are considered by many to be one of the most powerful “black box” learning algorithms. SVMs have quickly become one of the most widely used learning algorithms today, and are seen by many as the most successful current text classification method (Ghareb, Bakar, & Hamdan, 2016; Leopold & Kindermann, 2002).

Our SVM classifiers process the vector of similarity scores (constructed above) and return two pieces of information: The predicted category to which the support request belongs, and the confidence of its prediction (which is derived from the distance to the hyperplane boundary). In practice, the system will decline to make a prediction if the confidence level is < 95%. Confidence values need to be mapped from hyperplane distances to confidence intervals. This is accomplished by running the trained classifiers on a representative subset of support requests and observing the accuracy of the system with different hyperplane distance

thresholds, where predictions are only made if the thresholds are exceeded.

Because SVM classifiers are ML systems, they need to be trained on labeled representative data to make accurate predictions. The classifiers used by this algorithm were trained with support requests that had previously been categorized by support request center personnel.

Our ML system was configured as an ensemble of SVM classifiers. When using our classifiers, we often had to predict multiple values for a single document (e.g. a document’s Assigned to Group and Source of Support (SOS) Group). To accomplish this, we would train one ensemble of SVM classifiers to predict each value.

The classifiers we used were improved from publicly-available libraries by modifying the hyperplane boundary decision functions, customizing the training epochs, and optimizing inner-product computations.

Results

We performed five studies on our algorithms: 1) Deployment at NAVY 311 as a pilot study, 2) Accuracy when populating fields, 3) Potential for Automatically Routing Support Requests, 4) Comparison with Deep Learning Algorithms, and 5) Post-Processing Results and Data Scrubbing.

Deployment at NAVY 311

Manada Technology LLC developed the proposed algorithm and implemented it at NAVY 311. The algorithm was deployed as a Java web server application, took up 99 MB of space, and categorized each support request in 10 ms. The algorithm could then be accessed by copying the text of a support request into an online web form. The predictions would then be formatted as a JSON and posted to BMC Remedy through custom backdoor code.

NAVY 311 ran a pilot study to evaluate the algorithm’s performance. This pilot study ran from March 14, 2017 through August 24, 2017. Five Navy helpdesks were chosen to take part in this study. These helpdesks cover a wide range of issues, including (among others) IT support, purchasing, parts, and fleet reports.

Through the five months of this study, the predictive algorithm processed a small, randomly-selected subset of tickets submitted via email to these five Navy helpdesks. The algorithm processed each of these support request tickets and automatically populate all fields on the ticket for which it could make a high-confidence prediction (confidence \geq 95%). (Fields with low-confidence predictions were left blank for manual population.) Service desk agents then had to either sign off on each automatically-populated field, or manually change the values.

Throughout this study, three metrics were recorded: The median touch time (i.e. the cumulative total amount of time

	Model	Manual	Efficiency Gains
Touch Time	17 mins	26 mins	35%
Time to Assign	4.5 mins	17 mins	74%
Time to Close	144 hrs	362 hrs	60%

Table I: When our predictive algorithms were deployed at the NAVY 311 service desk, efficiency gains of 35%-74% were observed.

all agents spent on any given ticket), the median time to assignment (i.e. the amount of time an agent spent on a ticket before it was first assigned to a helpdesk), and the median time to close (i.e. the amount of time between when a ticket was submitted and when the ticket was closed). These metrics were recorded for support request tickets that were processed by the predictive algorithm and for tickets that were processed by agents alone.

At the conclusion of the study, over 16,000 support requests were submitted to these five Navy helpdesks via email (and were thus included in this study). Approximately 10% of these support requests were processed by the predictive algorithm, and the remaining were processed manually by service desk agents. The resulting metrics are listed in Table I. This table shows that using the predictive algorithm saved work time (and therefore money), got tickets assigned faster, and accelerated support request resolution.

Accuracy Test

In the above pilot study, there was no methodology in place to record the accuracy of the predictive algorithm: NAVY 311 did NOT record how often the predictive algorithm's populations were changed by service desk agents. To gauge the accuracy of our algorithm, we ran a separate study where we trained our algorithms on the three months leading up to the NAVY 311 pilot study, and evaluated its performance on all support requests populated by the predictive algorithm in the NAVY 311 pilot study. (This was done entirely separately from the above-discussed pilot study.) Through this methodology, by using the 95% confidence interval threshold, we estimate that 69.9% of fields were populated with 94.5% accuracy. Population rates by field for other thresholds are shown in Figure 1.

From this figure, it can be seen that the predictive algorithm performed better on fields related to routing support request tickets (e.g. Assigned to Group, SOS Group Name) than it did on fields related to ticket descriptions (e.g. Subject Level 4, Subject ID Latest Tier 3). To explain this discrepancy, we posit that fields that are related to routing are populated with a higher consistency than fields that are subjective and/or description-based.

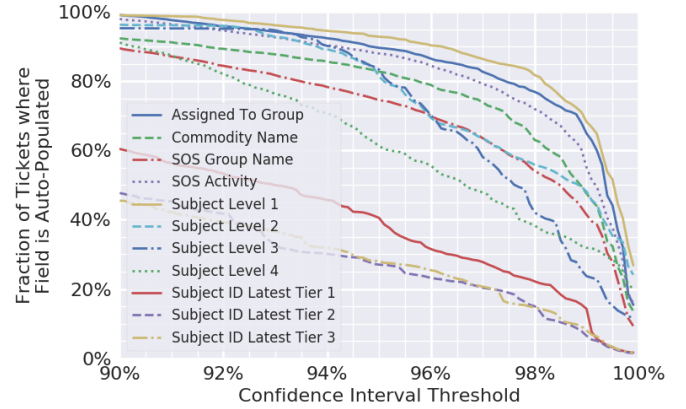


Figure 1: Results of the predictive algorithm when categorizing and routing support request tickets.

Automatic Routings

The previous two studies concerned themselves with augmenting an agent's performance by suggesting field populations to be manually reviewed. This is one of two use cases for AI in call centers that we considered. The other use case is for AI to independently perform some of the work of service desk agents without manual review.

When approaching this use case, we determined which fields were relevant to routing tickets, and which fields were descriptive items for categorizing and logging tickets. We identified five fields that were relevant to routing: Assigned To Group, Commodity Name, SOS Group, SOS Activity, and Subject Level 1. Then, to study the use case of automatic routings, we re-analyzed the results of the previously-discussed Accuracy Test. By setting a common threshold across all five of these fields, and only populating support requests where all these fields were predicted with confidences meeting or exceeding the threshold, we obtained the results shown in Figure 2

This figure shows that 35% of support requests can have all fields relevant to their routings auto-populated with 98% accuracy. This presents a compelling case for auto-routing support request tickets: More than one-third of tickets can have all routing-related fields auto-populated with a 2% error rate.

Comparison to Deep Learning Systems

To gauge the accuracy of our machine learning algorithm (an ensemble of modified SVM classifiers) relative to industry-standard ML systems, we implemented over 100 deep-learning artificial neural networks from Google's TensorFlow library to compare to our predictive algorithm. These networks had three to seven layers, consisted of 300 to 10,000 neurons per layer, used 0% to 25% dropout, and were trained with two to eight epochs. Also implemented was Py-

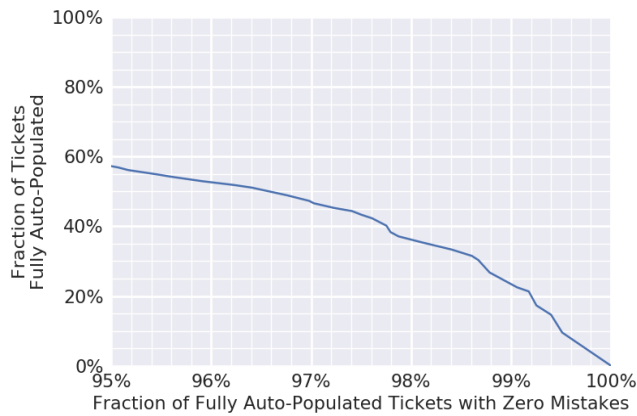


Figure 2: Results of the predictive algorithm when jointly populating and then automatically routing support requests.

thon's default SVM classifier from their Scikit-Learn library. The previously discussed NLP algorithms were used to pre-process, quantize, and vectorize the support requests. The ML systems were then used to classify the support requests into their respective categories and classes. The results of this comparison are shown in Figure 3, which demonstrate that our SVM-based approach performs comparably to deep-learning-based approaches.

Each of the 100 deep-learning systems discussed above took between fifteen minutes and six hours to run. Our classifier, however, took only nine minutes to run. The accelerated performance enabled daily updates to the predictive algorithm (or even more frequently), which allowed for more rapid alerts to changing practices and emergent issues. Additionally, multiple models can be built quickly, enabling development of better and more accurate classifiers in the future.

Post-Processing Results and Data Scrubbing

Our predictive algorithms perform optimally when predicting where a support request ticket should be resolved. The algorithms do not perform optimally, however, on the other fields, such as the detailed subject level fields. On investigation, the team discovered that this is because there is a lot of variability in those other fields: For example, in a certain Subject Level field, many tickets are marked General even when a more specific value could have been chosen for that field. The team developed some data regularization algorithms that used template-matching, value-mapping, and other data-scrubbing techniques to normalize and standardize ticket populations. By post-processing the algorithm's

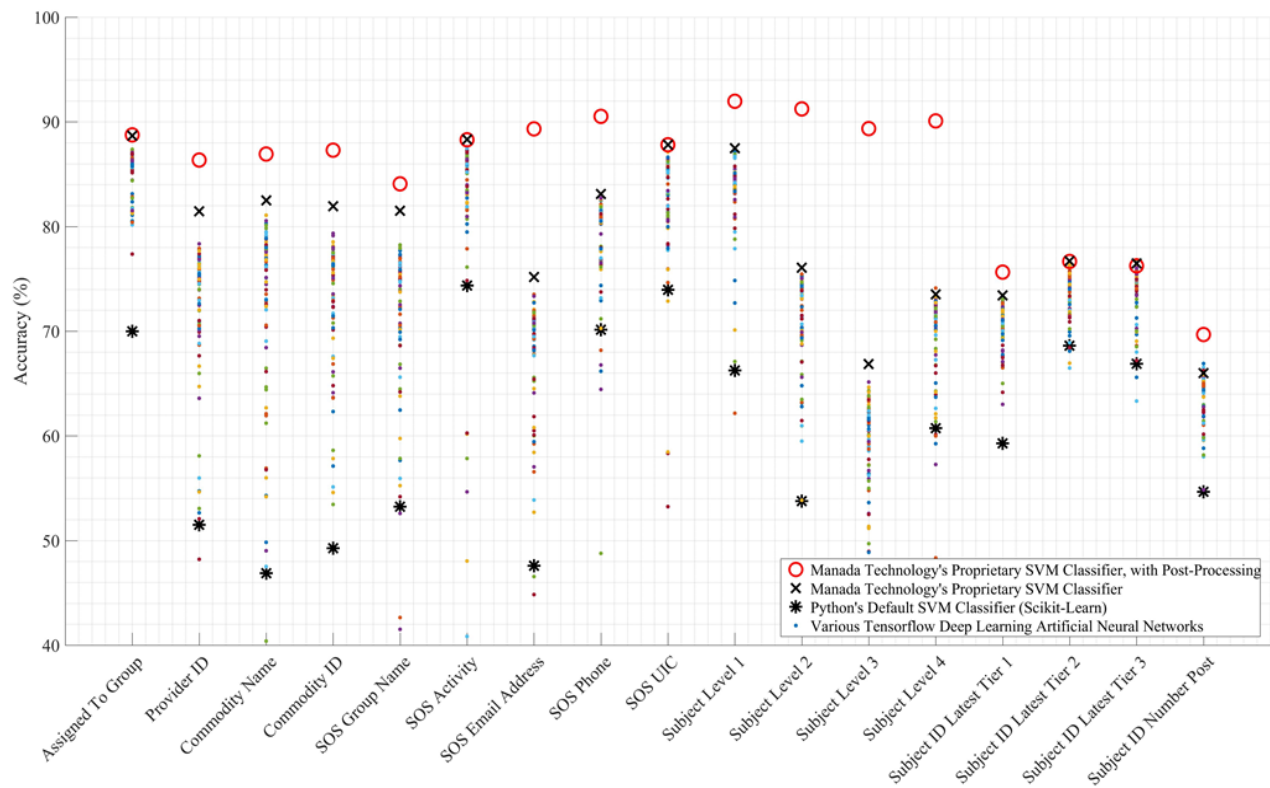


Figure 3: Classifier accuracy relative to industry-standard machine-learning systems. Also shown are the results after post-processing classifier predictions.

classifications with these routines, we obtained the performance shown in Figure 3, which shows a vast improvement over the standard model.

Conclusion

The results significantly improved current Navy help desk and customer relationship management (CRM) technology, specifically in parsing and analyzing help desk communications, reports, and logs. The new tool set was used to employ technical enhancements to the Navy's customer relationship management field.

The predictive algorithm showed significant gains in efficiency at NAVY 311 in terms of the amount of time spent on each ticket, the amount of time required to assign a ticket, and the amount of time required to close a ticket. Internal tests showed that our algorithms correctly predicted assignments for approximately 35% of support requests with approximately 98% accuracy. Furthermore, because this algorithm can be packaged as a single jar file, it can be integrated with currently-deployed CRM systems. By integrating with currently-deployed systems in the military, the solution offers modular integrations that minimize staff retraining by using technical implementations that are already in place. The predictive algorithm differs significantly from most commercially-available AI systems, which require full CRM systems and product suites deployed to use their algorithms.

We plan to deploy these algorithms as a core part of the Navy's customer support workflow. As a military application, requirements for security, information assurance, and cyber defense demand a detailed and lengthy accreditation process. We have been planning deployment schedules and use cases that fulfill all contractual and security requirements since the completion of the pilot study, and are optimistically planning on deploying improved versions of these algorithms and services within the next two years.

In addition to routing support requests, we are investigating using these core algorithms to power an interactive correspondence engine where a user describes their problems and the engine suggests a solution. The engine would use our core algorithms by being trained on support requests that are labeled with applicable solutions, and then predicting what solution (if any) is applicable to a new support request. With this added functionality, we can predict solutions that are delivered to customers as interactive prompts, which can help customer self-service their issues.

Limitations

Our algorithms are configured to predict support request categorizations and routings based upon historical standards. As business processes change, our algorithms should be able to detect and adapt to those changes. However, sudden

changes may cause incorrect routings until new workflows are firmly established. Additionally, some support requests can be "correctly" populated with multiple values for a given field; however, in historical archives, only one value is listed in each field. Because of this discrepancy, our algorithms may predict relatively low confidence on highly variable fields.

Recommendations for Military Application

The tools are ML algorithms that can be recalibrated to categorize any type of text-based document into any type of category. Furthermore, the algorithms can process data to self-train, making improvements that will be tested and either kept or discarded by the system automatically. The system is also capable of tuning by expert system operators and then tested and deployed. Therefore, the algorithm has a myriad of applications. For example, the algorithm could be trained to read emails as the messages are sent to alert security personnel if sensitive or classified information is being forwarded to an unauthorized source. The algorithm could also be used to predict upcoming spikes in Navy resources that will require extended labor demands or expanded electronic resources. Predictive analytics could be employed to predict future levels of customer satisfaction based on the past impact of various contributory variables. The NLP tools could be used to crawl the web for upcoming Security Technical Implementation Guides (STIG) or STIGs that will likely impact military operations. The capability is built to improve upon its own activities, improving over time.

The NAVY 311 program plans to integrate the product into the wider Navy support center community of over 100 independent service desks that serve a wide array of capabilities, such as personnel, records management, aviation maintenance, and IT networks. The deployment plan includes projections of return on investment in cost and reduced labor time if these proven customer support solutions are employed.

References

- BMC. (2019). TrueSight Orhestration: Reduce costs and errors through IT process automation. Retrieved from <https://www.bmc.com/it-solutions/truesight-orchestration.html>.
- Dalal, M. K. & Zaveri, M. A. (2011). Automatic text classification: A technical review. *International Journal of Computer Applications* 28.2 37-40.
- Forrest, C. (2017). How to use machine learning to improve customer service. Retrieved from <https://www.techrepublic.com/article/how-to-use-machine-learning-to-improve-customer-service>.
- Gabrilovich, E., & Markovitch, S. (2009). Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research*, 34, 443-498.
- Ghareb, A., Bakar, A. & Hamdan, A. (2016). Hybrid feature selection based on enhanced genetic algorithm for text categorization. *Expert Systems with Applications* 49. 31-47.
- Google (2019). AI and Machine Learning Products. Retrieved from <https://cloud.google.com/products/ai/>
- Gupta, N. K., Gilbert, M., & Di Fabbriozio, G. (2013). Emotion Detection in Email Customer Care. *Computational Intelligence* 29, 3. 489– 505.
- Gupta, N. K., Tür, G., Hakkani-Tür, D., Bangalore, S., Riccardi, G., & Gilbert, M. (2006). The AT&T spoken language understanding system. *IEEE Transactions on Audio, Speech & Language Processing* 14, 1, 213–222.
- Hardalov M., Koychev I., & Nakov P. (2018). Towards Automated Customer Support. In: Agre G., van Genabith J., Declerck T. (eds) Artificial Intelligence: Methodology, Systems, and Applications. AIMS 2018. Lecture Notes in Computer Science, vol 11089. Springer, Cham
- High, R. (2012). The era of cognitive systems: An inside look at IBM Watson and how it works. *IBM Corporation, Redbooks*.
- Hu, J., Fang, L., Cao, Y., Zeng, H., Li, H., Yang, Q., & Chen, Z. (2008). Enhancing text clustering by leveraging Wikipedia semantics. *Proceedings of the 31st Annual International ACM SIGIR conference on Research and Development in Information Retrieval. ACM*.
- Koehler, J., Fux, E., Herzog, F.A., Lötscher, D., Waelti, K., Imoberdorf, R., & Budke, D. (2017). Towards Intelligent Process Support for Customer Service Desks: Extracting Problem Descriptions from Noisy and Multi-lingual Texts. *Business Process Management Workshops*.
- Leopold, E. & Kindermann, J. (2002). Text categorization with support vector machines: How to represent texts in input space. *Machine Learning* 46.1-3. 423-444.
- Li, Z., Xiong, Z., Zhang, Y., Liu, C., & Li, K. (2011). Fast text categorization using concise semantic analysis. *Pattern Recognition Letters* 32.3. 441-448.
- Molino, P., Zheng, H., & Wang, Y.C. (2018). Cota: Improving the speed and accuracy of customer support through ranking and deep networks. *Proceedings of the 24th ACM SIGKDD* dl.acm.org
- Salton, G. & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management* 24.5, 513-523.
- Spens, H. & Lindgren, J. (2018). Using cloud services and machine learning to improve customer support : Study the applicability of the method on voice data (Dissertation). Retrieved from <http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-340639>.
- Sarikaya, R., Hinton, G.E., & Ramabhadran, B. (2011). Deep belief nets for natural language call-routing. In 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 5680–5683
- Wilson, H.J. & Daugherty, P.R. (2018). Collaborative intelligence: Humans and AI are joining forces. *Harvard Business Review*, July-August.
- Xu, P. & Sarikaya, R. (2013). Convolutional neural network based triangular CRF for joint intent detection and slot filling. In 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, December 8-12, 2013. IEEE, 78–83 3–19.