A Behavior Tree Cognitive Assistant System for Emergency Medical Services

Sile Shu*, Sarah Preum†, Haydon M. Pitchford, Ronald D. Williams*, John Stankovic†, Homa Alemzadeh*

*Electrical and Computer Engineering, †Computer Science, University of Virginia, Charlottesville, VA 22904
{ss5de, preum, hmp4yf, rdw, jas9f, ha4d}@virginia.edu

Abstract—This paper presents a cognitive assistant system for emergency medical services (EMS) that aims at improving situational awareness of the first responders by automated collection and analysis of data from the incident scene and providing suggestions to them. The proposed system relies on a Behavior Tree (BT) framework that combines the knowledge of EMS protocol guidelines with speech recognition, natural language processing, and machine learning methods to (i) extract critical information from responders’ conversations and verbalized observations, (ii) infer the incident context, and (iii) decide on safe and effective response interventions to perform. We use a data-set of 5302 real EMS call records from an urban, high volume regional ambulance agency in the U.S.A. to evaluate the responsiveness and cognitive ability of the system and assess the safety of the suggestions provided to the responder. The experimental results show that the developed cognitive assistant achieves an average top-3 accuracy of 89% in selecting the correct EMS protocols and an average F1-score of 73% in suggesting the protocol specific interventions, while providing transparency and evidence for the suggestions.

I. INTRODUCTION

First responders such as EMS personnel and firefighters need to initially assess and control the situation at the accident scene and provide both basic and advanced life support to the victims prior to transporting them to hospital. However, filtering, processing, and recording information with different levels of importance and confidence requires a significant amount of responders’ cognitive effort that could otherwise be utilized for emergency response.

Previous research [1], [2], [3] proposed use of assistive technologies to improve first responders’ situational awareness and decision making. For example, using wearable assistive agents for trauma documentation and management [2], [4], developing a portable communication framework for coordinating multiple agents (e.g., medical and communication devices, EMS vehicles) in distributed emergency response [5], [3], [6], simulating dynamic interactions between different human agents and potential digital agents in a hospital emergency environment using state-machine based models [7], real-time information extraction under noise for emergency response [1], and an information visualization agent that presents information gathered based on intent predicted using recent observations collected at emergency scene [8]. However, to the best of our knowledge, none of the existing research focuses on dynamically recommending situation-aware EMS protocol specific interventions for real-time emergency response decision support.

This paper presents design of a cognitive assistant system that analyzes speech data from the responders’ communications and observations at the scene, to infer the incident context, and suggest on the best response actions or interventions to perform based on standard EMS protocols. The proposed cognitive assistant can be implemented as a social robot or wearable virtual assistant, interacting with a team of first responders before, during, and after arrival to incident scene or during EMS training exercises. However, in this paper we only focus on developing the perception and cognition capabilities for such a robot. There are several challenges in design of a cognitive assistant for EMS:

- Emergency medical responders make decisions and provide interventions based on their training and knowledge of local EMS protocols. These protocols are developed and approved by a physician medical director to standardize medical care for all the responders and thus achieve excellent, consistent pre-hospital care for patients. To perform a similar task, a cognitive assistant system needs to be trained with the same knowledge and have the ability to process the information from the scene and make decisions in real-time.
- Despite limited availability of pre-collected EMS scenario datasets, most of this data is not properly labeled according to the EMS guidelines. Significant amount of manual effort and domain expertise are needed for labeling such data.
- At an incident scene, the speech data might be noisy or missing critical information needed for inference, which might affect the quality of decision making and intervention suggestion by the cognitive assistant.
- Many of the EMS protocol specific interventions are safety critical in nature (e.g., Fentanyl in pain management protocols or endotracheal intubation in respiratory distress protocols) and might cause serious consequences for the patient if mistakenly suggested by the system and performed by the responder.

To address these challenges, this paper adopts a Behavior Tree (BT) framework for real-time retrieval of the critical information from the scene and infer the correct EMS protocol specific interventions based on the retrieved information. The main contributions of the proposed framework can be summarized as follows:

- We develop a weakly supervised method for selection...
of the most appropriate EMS protocols to be followed based on the situations inferred from the scene and the knowledge of the EMS protocol guidelines. Our evaluation using 3657 labeled EMS records indicate that this method achieves an average top-3 accuracy of 89%.

- We present two kinds of methods for prediction and suggestion of the most effective interventions by the cognitive assistant: an unsupervised knowledge-driven method based on developing executable behavioral models of the EMS protocols using BTs and a supervised data-driven ML method based on learning models from historical EMS records. Our results show that ML methods achieve better accuracy (F1 score) in predicting correct interventions. However, the BT method provides more transparency and evidence for the suggested intervention and does not rely on the availability of labeled data.

- We introduce a method for assigning confidence for the protocol and intervention suggestions made by the BT model to reduce the risk of performing safety-critical interventions and prevent harm to patients. When considering the potential risks of performing incorrect interventions by responders, the suggestions provided by the BT model on average have at least 22% lower risk compared to the best performing supervised ML models.

II. PROBLEM FORMULATION

Our goal is to design a cognitive assistant system that can infer critical information about the situations at the accident scene, including physical status and medical history of the patients, from responders’ conversations and verbalized observations. This information are represented in the form of medical or EMS semantic concepts and are mapped into the standard EMS protocols to provide suggestions on the best interventions to perform. For example, the opioid overdose protocol (Figure 1b) indicates when the first responders observe that a patient is suffering from hypoxemia (i.e., patient’s SpO2 level is lower than the normal range), they need to provide supplemental oxygen to the patient.

Formally, we consider a set of standard emergency medical service protocols $P$. For each protocol $P_i$, we use a set of critical concepts (e.g., signs, symptoms, and medical history of patient) to model the conditions for which the protocol should be selected by the first responder to manage the emergency situation. We define $C$ as the set of all the concepts describing the protocol set $P$. We define $I$ as the set of all possible interventions recommended by the protocols in $P$. At any time $t$, we assume all the information verbalized by the first responder so far are included in a segment of speech data denoted as $S_t$. Then, we can summarize the problem as follows. At an arbitrary time $t$, the cognitive assistant needs to find the appropriate subset $I_j$ in the intervention set $I$ based on the knowledge of $P_i$ in the EMS protocol set $P$ according to a subset of $C$ extracted from the speech data $S_t$.

To solve this problem, we separate it into three consecutive sub-tasks:

1) Extract a subset of $C$ to represent the situation for an arbitrary time $t$ from the speech data $S_t$;
2) Rank the EMS protocols in $P$ and find a subset of EMS protocol $P_i$ in $P$ whose usage scenario is closest to what is described by the speech data $S_t$;
3) Find the intervention subset $I_j$ based on the knowledge of the selected protocol subset $P_i$.

In the following sections we present the methodologies to perform these three sub-tasks.

III. APPROACH

We propose a BT framework for implementing the natural language processing and cognitive inference by the cognitive assistant as illustrated in Figure 1a. Figure 1b shows an example of Overdose Opioid protocol sub-tree in the BT. The details are presented below.

A. Overall BT Framework

Behavior Trees are a mathematical model of plan execution used in robotics and intelligent agents, which first emerged from video game industry. Recent work has shown the potential of BTs as a flexible and interpretable data structure for representing medical processes and clinical practice guidelines in AI systems [9]. BTs can model the behavior of an intelligent agent as a directed rooted tree, presenting each sub-task as a leaf, and combine them into behaviors through nodes in a specific order [10]. A BT root generates a signal, called tick, periodically following a frequency $F$. Every node receiving the tick from its parent, starts execution and returns its status on achieving its goal as success or failure. There are two types of execution nodes: Action nodes that return success upon completion of certain action and Condition nodes that return success if a specific condition is met [11].

We choose BTs as an executable behavioral modeling framework for design of our cognitive system due to its modularity, high responsiveness, and the ability to learn and adapt using reinforcement learning. As shown in Figure 1a, in every tick, the sequential node “Root” ticks the execution of the different nodes of the cognitive assistance pipeline, to perform conversion of text from speech, gathering important concepts from the text, transforming the concepts into vector space, protocol selection, and protocol execution/intervention suggestion. The protocol execution and intervention suggestion is implemented as a parallel node with multiple children, concurrently executing multiple applicable protocols. Every protocol node is a sequential node, which sequentially ticks the condition and action nodes, respectively, implementing the conditions to satisfy for executing the protocol and the sub-tree of the protocol logic as defined by the EMS protocols. The details of the BT nodes implementing different components of the cognitive assistant are provided next.

B. Speech to Text Conversion

The purpose of this component is transferring the input speech data $S_t$ from first responder to text $T$. We apply the Google Speech API to perform speech to text conversion
on the audio streams collected from the accident scene. Our previous experiments have shown that Google Speech API provides the best results among other state-of-the-art speech recognition software [12]. As shown in Figure 1a, at every tick of the behavior tree, first the sub-task *Speech to Text Conversion* is executed to get the generated text from the incoming audio stream. Then the collected text is passed to the following components via blackboard, a typical component in BTs to store and transport data between the sub-trees and nodes. Upon completion of these steps, the *Speech to Text Conversion* sub-task will return success to its parent node.

C. Information Extraction

After retrieving text from the speech recognition component, the collected text is fed into the *Information Extraction* component. In this component, input text $T$ is represented by a concept set $E$, which is a subset of the whole concept set $C$, and consequently essential information for emergency medical services can be extracted, including patient’s physical condition and medical history, situations of the accident scene, and treatments performed by the first responders. The information extraction process consists of the following steps.

1) **UMLS Concept Extraction:** At this step, we apply MetaMap, a widely available tool for mapping biomedical text into the concepts in the Unified Medical Language System (UMLS) Metathesaurus [13], to extract the biomedical concepts from the text along with their negation condition, semantic type, and position information. Every single concept is assigned with a unique identifier in the UMLS, called Clinical Unique Identifier (CUI).

2) **Concept Filtering:** Our previous analysis of the outputs from MetaMap, showed that not all of the extracted concepts are useful for the decision making in EMS [1]. Thus, we compiled a set of EMS protocol specific concepts that are required by the EMS protocols or are frequently used by the medical responders. Each concept was then extended to an additional set of terms that share the same or similar meaning with the original concept and these terms are mapped into unique UMLS CUIs. The list of CUIs was generated by sending the original text as queries to UMLS online API and selecting the 25 most related CUIs (i.e., top 25 ranked in order of relevance). At the concept filtering stage, this extended list of CUIs ($C$) is used to filter the results from MetaMap and keep the concepts most relevant to the EMS protocols.

3) **Value Retrieval:** We find additional information related to the concepts identified in the text, e.g., for the extracted concept *pulse rate*, we are interested to also extract the value of the pulse rate. For retrieving the corresponding numeric values of specific concepts such as vitals (e.g., pulse rate, blood pressure, spo2) we find the closest number to the concept as their value via regular expression matching. We directly use the preferred names as the value of the abstract concepts (e.g. history of symptoms, quality of pain, past illness).

4) **Confidence Assignment:** We assign a confidence score to the extracted concepts from text to indicate the notion of uncertainty in our detected evidence from the scene due to non-perfect quality of speech recognition and concept annotation components. In our confidence calculation and assignment, we consider the confidence score for the recognized words by the Google Speech API [14] and the similarity score provided by the MetaMap API indicating the level of confidence in mapping between the input text and UMLS concepts [15]. By combining these two different confidence scores, we can have a score representing the overall confidence in the information collected from the conversations of emergency responders at the scene. Incorporating other factors contributing to uncertainty and lack of confidence (e.g., missing information, noisy speech) is the subject of future work.

The collected concept set $E$ is modeled as a dictionary with each element defined using the following unified format:

$$(C_i : P_{i,t}, V_{i,t}, T_{i,t}, Conf(C_i, t), t)$$

where $C_i$ refers to the $i$th concept in the dictionary, which also serves as a key in the dictionary; $P_{i,t}$ is a boolean variable representing the presence or absence of $C_i$ in the text at tick $t$; $V_{i,t}$ is a number representing the value of $C_i$ at tick $t$; $T_{i,t}$ is the normalized original trigger text of $C_i$ at tick $t$, and $Conf(C_i, t)$ indicates the confidence of the concept $C_i$ at tick $t$. Assuming the text from which the concept $C_i$ was detected has a speech-to-text confidence score $Conf_G(C_i)$ provided by Google Speech API and its CUI detected by MetaMap has a similarity score $mmScore(C_i)$, we calculate the confidence score $Conf(C_i)$ for every $C_i$ in $C$ as follows:

$$Conf(C_i) = Conf_G(C_i) \cdot mmScore(C_i)$$  \hspace{1cm} (1)$$

An example piece of text along with the corresponding dictionary elements extracted by the *Information Extraction* phase:
are shown in Figure 3a. These outputs are formatted in a
unified structure and then fed to the next stage for protocol
selection and execution and intervention suggestion.

D. Vectorizer

Once we get the concept set $E$ representing the input text
by EMS related concepts, similar to text vectorization, we can
transfer the concept set $E$ as a vector $V_T$, whose size equals
the length of the concept set $C$ and values are the confidence
scores $Conf(C_i, t)$ for each extracted concept $C_i$ in $E$. Each
item in the input text vector indicates if the concept has
appeared in the input text and how much confidence we have
for its mapping (mapping textual contents to concepts). Thus,
if any concept $C_i$ is detected at tick $t$, the corresponding item
in the text vector will be encoded with a value of $Conf(C_i, t)$.

We also use a set of vectors $V_P$ to represent the concepts
related to signs and symptoms that are required for the execu-
tion of a specific EMS protocol. Each protocol in protocol
set $P$ is represented as a vector $V_P$, whose size also equals
the length of the concept set $C$ but values are assigned with
different weights based on the importance of these concepts in
selecting the protocol. These weights are assigned and ranked
by real first responders participating in our project. Formally,
these two vectors can be represented as follows:

$$V_T = \{Conf(C_i) | \forall C_i \in C\}$$

$$V_P = \{Weight(C_j) | \forall C_j \in C \land Weight(C_j) = \text{Softmax}(Pri_{ij}) \land Pri_{ij} \in \{0, 1, 2, 3\}\}$$

where $Pri_{ij}$ is a priority score assigned based on the
relevance between the protocol $P_i$ and concept $C_j$ (with 3
representing most relevance and 0 representing no relevance).
We apply softmax function to normalize these scores into
weights and make them add up to 1.

E. Protocol Selection

Given the input text vector $V_T$, representing the information
gathered from the scene at tick $t$ and the protocol vector
set $V_P$, representing the required concepts (conditions, signs
and symptoms) for executing a specific EMS protocol, we take
a weakly supervised approach to determine the relevance
between the current situation at the scene and each EMS
protocol in $P$ by calculating the similarity between their
vectors. Cosine similarity, as a commonly used metric in
information retrieval and question answering systems is used.
We calculate the similarity and relevance between the text ($V_T$)
and protocol ($V_P$) vectors, as follows:

$$S_i = \frac{V_T \cdot V_P}{\|V_T\| \cdot \|V_P\|} \quad (4)$$

After calculating the cosine similarity between a given text
vector and all the protocol vectors in our library of EMS
protocols, we rank the protocols based on their similarity
to the input text and select the ones with highest scores
as the appropriate protocol to be executed by the cognitive
assistant system. If multiple protocols have a high relevance
score, an ordered list of candidate protocols will be selected
and used for the feedback generation. We assign the cosine
similarity index calculated for each protocol as a confidence
score for its selection and normalize the confidence scores
such that the sum of all scores in the final list is equal
to 1. For a subset of protocols from $P$, called Candidate,
containing top $N$ protocols based on their cosine similarity
scores, the normalized confidence score of each candidate
protocol, $Conf(P_i)$, is calculated as follows:

$$Conf(P_i) = \begin{cases} \frac{s_i}{\sum_{j=1}^{N} s_j}, & \forall P_j \in \text{Candidate} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

This normalization of the confidence scores provides a
frame of reference to the responders for comparing the scores
and potentially considering the protocols with the higher
scores. It also enables confidence propagation and assignment
to the interventions suggested by the BT framework.

F. Protocol Execution - Intervention Suggestion

Typically, each EMS protocol describes some specific rules
to perform interventions in an emergency scene, and the
conditions in these rules are signs, symptoms or medical
history of the patient, which we extracted and represented
as concepts in the previous components (see example in
Figure 1b). Therefore, we model the execution logic for
each EMS protocol as a separate sub-tree in the BT whose
children implement the conditions to be checked and actions
or interventions to be taken as part of the protocol. All of
the protocol nodes are connected to a parallel parent node, which
makes it possible for all the selected candidate protocols to
be executed concurrently at the same time and suggest most
relevant interventions with highest confidence score to the
responders. Due to the modularity of the BTs structure, we can
easily replace or extend the set of EMS protocols by merely
replacing or adding to the sub-trees under the parent node.

There is an obvious risk to execute protocols concurrently
in this system. In most cases, extra protocols will be executed,
and consequently, inappropriate or even safety-critical feed-
back might be suggested to the responder. To avoid such risks,
we have extended the BT framework with a new capability
for assigning confidence values to the nodes and propagating
them through the execution path on the BT. This enables us
to provide a confidence for each final feedback generated
by the selected protocols and let the responder consider
different interventions with different confidence levels.
When calculating the propagation of the confidence scores on
the paths of the BT, we assume that the appearance of the concepts
in the protocols are independent events from each other and
they are also independent from the event that a protocol is
selected. Thus, we assign a confidence score to every final
feedback node (leaf action or intervention node in the protocol
subtree) by multiplication of confidence scores assigned to
previous nodes in the path to that node, including the concepts
and conditions observed in the input text and the protocols
selected. Figure 2 shows an example of the propagation of
confidence scores from a selected protocol and an observed concept in text into an action node on the BT. Finally, The interventions with a confidence score of less than 0.1 are filtered out from the final list of suggestions presented to the responder.

As a result of applying the above-mentioned mechanism, the safety-critical and inappropriate interventions tend to have a lower confidence scores. Because:

- The initial confidence assigned to each protocol is based on the similarity between the text vector and their protocol vectors, which means the interventions within the less relevant protocols will be assigned with lower confidence scores.
- Even if an irrelevant protocol is selected by the model, some of the interventions suggested by these protocols are less likely to be suggested because the relevant observations are not extracted from the scene and required conditions for those interventions are almost impossible to be satisfied. For example, if chest pain protocol is triggered in a abdominal pain case, "STEMI" is less likely to appear in this case and corresponding interventions will not be suggested.
- In EMS protocols, the safety-critical medications/interventions typically have more conditions/prerequisites to be satisfied and some of them can only be performed when other less safety-critical interventions were not effective (e.g., Fentanyl will only be administered when pain persists after giving Nitroglycerin in Chest Pain protocol). Thus, these interventions tend to have lower confidence scores and more likely to be filtered by the confidence threshold.

As a result of applying the above-mentioned mechanism, the safety-critical and inappropriate interventions tend to have a lower confidence scores. Because:

- The initial confidence assigned to each protocol is based on the similarity between the text vector and their protocol vectors, which means the interventions within the less relevant protocols will be assigned with lower confidence scores.
- Even if an irrelevant protocol is selected by the model, some of the interventions suggested by these protocols are less likely to be suggested because the relevant observations are not extracted from the scene and required conditions for those interventions are almost impossible to be satisfied. For example, if chest pain protocol is triggered in a abdominal pain case, "STEMI" is less likely to appear in this case and corresponding interventions will not be suggested.
- In EMS protocols, the safety-critical medications/interventions typically have more conditions/prerequisites to be satisfied and some of them can only be performed when other less safety-critical interventions were not effective (e.g., Fentanyl will only be administered when pain persists after giving Nitroglycerin in Chest Pain protocol). Thus, these interventions tend to have lower confidence scores and more likely to be filtered by the confidence threshold.

A. Experimental Setup

For these two experiments, we considered 8 commonly used EMS protocols from a regional set of protocol guidelines and a dataset of 8302 pre-hospital call sheets from a regional ambulance authority (RAA). The information inside these reports are originally organized into several categories including the type of the call, priority of the dispatch, chief complaint from the patient, first and second impressions from the first responders, vital signs recorded in the emergency scenes, interventions taken by the first responders, outcome after the interventions and the narratives describing the emergency situations. Narratives and vital signs were fed to our model as inputs because the narratives from the first responders and the vital signs are the only information that we can directly obtain from verbal conversations in the emergency scenes. Note that in these experiments, we directly used the narratives and vitals transcribed by the responders and did not perform the speech to text conversion, so the $\text{Conf}_C(C_i)$ score in Equation 1 was always set as 1. The results of evaluating the speech to text conversion step for both noise-free and noisy realistic audio data from incident scenes were presented in [1].

For a subset of 3657 records, the actual protocol used by the responder was labeled by one of the advanced life support trained responders in our project and was used as grand truth for assessing the accuracy of automated protocol selection component. This was done by developing a set of rules unique to each of the pre-selected protocols in order to filter out cases that were either ambiguous or fell into another treatment protocol. For example, in order for a case to be labeled as an opioid overdose, the medication Naloxone must have been administered and the documented field impression must indicate that the original responder believed the patient’s presentation was due to an overdose. Thus, we marked the cases that the medication Naloxone was given and the impressions included opioid overdoses as overdose opioid protocol. First responders’ interventions recorded in these reports served as the ground truth to evaluate the suggestions generated by our model and the quality of the feedback to responders.

We developed multiple machine learning (ML) models with several variations of hyper-parameters that were trained on the RAA data to perform intervention prediction. These models were used to evaluate the performance of the intervention suggestion by our proposed knowledge-driven BT method. Specifically, we applied the following three supervised data driven ML models to perform the intervention prediction: Support Vector Machines (SVM), Random Forest (RF), and Decision Trees (DT). We applied the intervention column in the RAA reports as the ground truth and the narrative column as inputs to train the supervised ML models. Each narrative was represented in two vector spaces using technical n-grams and uni-grams of the narratives. Our test dataset for intervention prediction included 1000 RAA EMS reports for both BT and ML models. The remaining 7302 EMS reports were used to train the ML models. We applied 5-fold cross-validation by splitting the training data consisting
of 7302 reports into 5841 training cases and 1461 validation cases and trained multi-class classifiers (for 94 intervention classes) using the three supervised ML models. To achieve a fair comparison between the confidence-aware, knowledge driven unsupervised method based on BTs and the data driven, supervised ML models, we also added the following two settings to the ML models: (i) Training the ML models using a class weighting approach in which intervention classes with higher risk scores were assigned lower weights to direct the supervised ML models towards selecting less safety-critical interventions with lower risk factors; (ii) Applying the similar confidence score filter implemented in the BT model to filter the interventions with low confidence.

B. Experimental Results

1) Protocol Selection: To evaluate the automated protocol selection procedure, we compared the ranked list of selected protocols by the cognitive assistant with the grand-truth protocols labeled by the participating first responders in our project. Since the protocol selection component generates a ranked list of top 3 protocols with their confidence scores, we applied a top-3 accuracy metric to evaluate if the target label by the responder is one of the top 3 predictions by the cognitive assistant. Our experiments with the 3657 test cases from RAA dataset showed an average top-3 accuracy of 89.0%. By reviewing the cases for which the cognitive assistant predictions did not match the labels provided by the responders, we identified the following reasons for sub-optimal performance of our protocol selection method:

- Errors occurred in mapping between input text and standard concepts in our Information Extraction component, which led to generation of inaccurate text vectors and consequent generation of wrong ranking for the selected protocols. These errors were due to: 1) MetaMap not recognizing the required concepts as CUIs; 2) Some identified CUIs by MetaMap not appearing in the mapping between CUIs and standard concepts in our model; 3) CUIs and concepts not precisely matching (e.g., We get the CUI “Respiratory Sound” from UMLS mapping to the required concept “Wheezing.” However, they are not the same since wheezing is one kind of respiratory sound. Thus, some other respiratory sounds will be mapped to wheezing, which leads to a mapping errors.); 4) MetaMap not producing the correct negation detection results, leading to missing the presence of some concepts from input text.

- Protocol vectors were manually developed based on the descriptions of signs and symptoms in the set of protocols and the value of each concept in the protocol vector was assigned with different weights based on their importance, as reviewed and ranked by one of the participating responders in our project. Some of the manually assigned weights in the protocol vectors caused errors.

- Missing critical information (e.g., incomplete vital signs) in some of the EMS cases also affected the correctness of the text vector.

2) Intervention Suggestion: To evaluate the performance of the intervention suggestion, we used the list of interventions performed by the responders in the dataset as ground truth and compared it with the list of interventions suggested by the cognitive assistant system. We define the predictions which appear in the ground truth as true positives (TP), while the ones which are not included in the ground truth as false positives.
(FP). We also define the interventions in the ground truth that failed to be predicted by our system as false negatives (FN). By calculating the TPs, FNs and FPs for each RAA case, we use both weighted and micro average precision, recall and F1-score to evaluate the performance of the intervention predictions. The weighted metrics calculate precision, recall, and F1-score for each class, and find their weighted average based on the number of instances for each class to take the imbalance among classes into account.

In addition to traditional methods for evaluation of multi-output prediction results, we also consulted with first responders about the FN and FP intervention predictions because some of the suggested interventions although reasonable, might not be performed by the first responders and some of the suggestions are too risky to be performed at the scene. EMS protocols are written in terms of escalating clinical care, therefore even if an intervention is indicated under a certain protocol the responder may not perform it due to time or resource limitations. Further, EMS protocols prioritize life and limb saving interventions over comfort measures, and simple interventions are preferred over the complex ones whenever possible. Under this consideration, we used another metric to evaluate our intervention suggestion method in terms of the risk incurred by the interventions. All the suggested interventions were classified into four distinct classes of red, orange, yellow and green. This is according to the severity of the condition that the intervention addresses as well as possible side effects they might have for patients when incorrectly suggested (FPs) or not suggested (FNs). These severity levels were then encoded as different risk scores $Risk(I_i)$ from 1 to 4 assigned to each intervention class. Larger scores indicate a higher risk if an incorrect intervention is suggested to the responder. Then for each case with a set of $I$ interventions, we calculated the average risk factor of the suggested interventions by summing the products of the risk scores $Risk(I_i)$ and confidence scores $Conf(I_i)$ of the incorrect interventions (FP or FN) provided by the model and normalized it by dividing by the number of grand truth interventions for each case. The average normalized risk to evaluate the performance of model over $n$ test cases was calculated as follows:

$$Avg.\ Normalized\ Risk = \frac{1}{n} \sum_{I \in I} Conf(I_i) \cdot Risk(I_i)$$ (6)

The evaluation results using the metrics mentioned above are shown in Table I. Note that we also tried other supervised ML methods such as support vector machines (SVM), but do not report their results here due to poor performance. Our results show that the knowledge driven unsupervised BT method performs worse than supervised ML methods when evaluated using the traditional evaluation metrics (precision, recall, and F1) used for a multi-class classification task. However, the average normalized risk factor of the interventions suggested by the BT model is at least 22% lower than the best performing supervised ML model, i.e., DT-unigram model with and without filtering. This means that we can effectively avoid safety-critical suggestions using the confidence propagation and filtering mechanisms of the BT model. The supervised ML methods trained with class weighting perform worse in terms of precision, recall, and F1 than the models with no knowledge of risk scores, and they also yield higher average risk factors as well. Based on the formula of the risk factor in our evaluation, the reason for these results might be the extra FN and FP predictions brought by weight assignments for the ML models. On the other hand, the ML models with filters, which get rid of predictions with confidence scores lower than a threshold, slightly reduced the risk factor compared to the original models.

Furthermore, the proposed BT method has the following advantages compared to supervised ML models:

- The BT model has high modularity, which means when we need to edit/add/remove any EMS protocols in the model, what we need to do is only substituting/inserting/deleting the corresponding protocol subtrees. However, when it comes to supervised data-driven ML methods, re-collection and labeling of data and re-training the whole model is required.
- The proposed BT framework is weakly supervised and knowledge-driven, which means that it does not rely on the availability of training data and correctness of labels. Whereas the performance of supervised ML methods greatly relies on the quantity and quality of the training data and labels. Further, in contrary to ML methods which are black box end-to-end solutions from input text to interventions, the BT framework is transparent and can provide explanation for the decisions made and suggestions provided to the responders.

V. DISCUSSION AND FUTURE WORK

From the evaluations conducted in the previous section, the following major challenges were identified:

- The concept list used in the information gathering phase is currently manually created and is limited to the knowledge of protocols and, thus, it might be a possible reason for missing important concepts from the input text. Also, as the number of the target EMS protocols grows, the cost and amount of effort needed for modeling the protocols and manually extending the concept list significantly increases. Thus, we plan to find an automatic way of obtaining a more complete and accurate concept list in the future. We are now investigating the vector space models and unsupervised machine learning techniques to automatically expand the limited set of manually identified concepts to a larger database of EMS relevant terms to be extracted from the text.
- The inaccuracies in detection of presence or absence of the concepts in text largely affect the results of the protocol selection and execution phases. Currently, we rely on the negation detection features of MetaMap to extract the absence of concepts. Future work will focus on developing techniques for more precise detection of concept presence and absence.


<table>
<thead>
<tr>
<th>Model</th>
<th>Weighted Precision</th>
<th>Micro Precision</th>
<th>Weighted Recall</th>
<th>Micro Recall</th>
<th>Weighted F1 Score</th>
<th>Micro F1 Score</th>
<th>Cross-Validation Micro F1 Score</th>
<th>Avg. Risk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Behavior Tree</td>
<td>0.65</td>
<td>0.76</td>
<td>0.66</td>
<td>0.66</td>
<td>0.64</td>
<td>0.71</td>
<td>NA</td>
<td>0.22</td>
</tr>
<tr>
<td>Random Forest (RF)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ngram</td>
<td>0.77</td>
<td>0.88</td>
<td>0.67</td>
<td>0.66</td>
<td>0.68</td>
<td>0.76</td>
<td>0.75</td>
<td>0.46</td>
</tr>
<tr>
<td>ngram, weighted</td>
<td>0.71</td>
<td>0.81</td>
<td>0.61</td>
<td>0.62</td>
<td>0.63</td>
<td>0.72</td>
<td>0.72</td>
<td>0.46</td>
</tr>
<tr>
<td>ngram, filtering</td>
<td>0.80</td>
<td>0.89</td>
<td>0.66</td>
<td>0.67</td>
<td>0.68</td>
<td>0.76</td>
<td>0.75</td>
<td>0.43</td>
</tr>
<tr>
<td>unigram</td>
<td>0.74</td>
<td>0.91</td>
<td>0.72</td>
<td>0.71</td>
<td>0.71</td>
<td>0.78</td>
<td>0.75</td>
<td>0.42</td>
</tr>
<tr>
<td>unigram, weighted</td>
<td>0.76</td>
<td>0.88</td>
<td>0.63</td>
<td>0.65</td>
<td>0.65</td>
<td>0.74</td>
<td>0.74</td>
<td>0.49</td>
</tr>
<tr>
<td>unigram, filtering</td>
<td>0.84</td>
<td>0.91</td>
<td>0.71</td>
<td>0.71</td>
<td>0.74</td>
<td>0.80</td>
<td>0.78</td>
<td>0.39</td>
</tr>
<tr>
<td>Decision Trees (DT)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ngram</td>
<td>0.73</td>
<td>0.77</td>
<td>0.73</td>
<td>0.75</td>
<td>0.72</td>
<td>0.76</td>
<td>0.74</td>
<td>0.45</td>
</tr>
<tr>
<td>ngram, weighted</td>
<td>0.71</td>
<td>0.75</td>
<td>0.70</td>
<td>0.73</td>
<td>0.70</td>
<td>0.74</td>
<td>0.73</td>
<td>0.54</td>
</tr>
<tr>
<td>ngram, filtering</td>
<td>0.73</td>
<td>0.78</td>
<td>0.72</td>
<td>0.75</td>
<td>0.72</td>
<td>0.76</td>
<td>0.76</td>
<td>0.46</td>
</tr>
<tr>
<td>unigram</td>
<td>0.82</td>
<td>0.84</td>
<td>0.80</td>
<td>0.82</td>
<td>0.81</td>
<td>0.82</td>
<td>0.83</td>
<td>0.28</td>
</tr>
<tr>
<td>unigram, weighted</td>
<td>0.77</td>
<td>0.80</td>
<td>0.79</td>
<td>0.79</td>
<td>0.77</td>
<td>0.80</td>
<td>0.79</td>
<td>0.39</td>
</tr>
<tr>
<td>unigram, filtering</td>
<td>0.80</td>
<td>0.84</td>
<td>0.80</td>
<td>0.81</td>
<td>0.80</td>
<td>0.83</td>
<td>0.81</td>
<td>0.28</td>
</tr>
</tbody>
</table>

TABLE I: Intervention suggestion provided by the unsupervised BT model vs. the supervised random-forest (RF) and decision tree (DT) models. Weighted supervised models represent the models trained with inverse intervention risk scores while filtered models represent models with confidence score filtering that remove interventions with low confidence from the list of suggestions. Overall both of the supervised models result in higher precision, recall, and F1-score than the unsupervised BT model. Specifically, when compared to the unsupervised BT model, the DT models result in 26% and 17% higher weighted F1 score and micro F1 score, respectively. However, the BT model results in at least 22% lower average risk score than the best performing supervised model (i.e., DT-unigram model). This shows the effectiveness of our developed unsupervised model for protocol specific intervention suggestion for the safety-critical, low resource EMS domain.

- The unified dictionary format for representing and collecting the extracted information, the protocol conditions, and the modularity of behavior tree models enable scalability of the BT framework. We plan to study the possibility of automatically extending the behavior trees based on EMS data or protocols, by adding learning capability to the nodes.
- Interventions performed by a first responder are necessarily limited by the underlying context such as transport time, severity of patient illness and resources available. Systematically accounting for these contexts would improve and better account for both the safety and rate of the intervention suggestion false positives.

VI. CONCLUSION

This paper presented a Behavior Tree cognitive assistant system for emergency response which can be implemented as a virtual assistant or social robot interacting with the responders at the incident scenes to provide them with suggestions on the most appropriate protocols and interventions to execute. Our experimental results show that supervised ML methods trained on historical EMS data might outperform the knowledge-driven BT method when compared using traditional accuracy metrics. However, the proposed BT modeling framework provides better guarantees on the safety of interventions suggested to the responder as well as transparency and evidence. The proposed cognitive assistant system has also the potential to be used during simulation training experiments for preparing responders with the knowledge of protocol guidelines and scoring their performance in executing the protocols.

REFERENCES
