Proceedings of Machine Learning Research vol 120:1–10, 2020

A Spatially and Temporally Attentive Joint Trajectory Prediction Framework for Modeling Vessel Intent

Jasmine Sekhon JS3CN@VIRGINIA.EDU
University of Virginia, Charlottesville, VA

Cody Fleming CF5EG@VIRGINIA.EDU
University of Virginia, Charlottesville, VA

Abstract

Ships, or vessels, often sail in and out of cluttered environments over the course of their trajectories. Safe navigation in such cluttered scenarios requires an accurate estimation of the intent of neighboring vessels and their effect on the self and vice-versa well into the future. In manned vessels, this is achieved by constant communication between people on board, nautical experience, and audio and visual signals. In this paper we propose a deep neural network based architecture to predict intent of neighboring vessels into the future for an unmanned vessel solely based on positional data.

Keywords: intent modeling, trajectory prediction, long short term memory networks, spatial attention, temporal attention

1. Introduction

Autonomous navigation is increasingly being adopted in land and airborne vehicles. The success of autonomy in other modes of travel has led to its advent in the maritime industry with the development of Autonomous Surface Vessels or ASVs. However, like all other autonomous vehicles, ASVs also come with their safety and reliability concerns. These autonomous vessels, or other autonomous agents in general, are expected to negotiate safely through crowded environments, like harbors or urban streets, that involve complex social interactions.

Any autonomous agent that is required to safely navigate through such crowded environments must possess the ability to actively and accurately forecast the future intent of neighboring entities in order to adjust own trajectory accordingly to avoid collisions.

The problem of predicting the future intent of a vessel based on observations of its positional data over several timesteps can be viewed as a sequence-to-sequence modeling tasks. Long Short Term Memory Networks (LSTMs), introduced by Hochreiter and Schmidhuber (1997), are a special variant of deep neural networks known for their ability to model long sequences. The primary component of an LSTM is a gate-regulated cell state that allows LSTMs to remember information from a longer history. Consequently, LSTMs are achieving almost human-level performance in sequence generation tasks such as text generation, speech recognition, language translation, time series prediction, and others. However, despite their success in learning and reproducing long sequences, LSTMs are not capable of modeling interactions between multiple correlated sequences such as spatially co-located autonomous agents.

Inspired by the success of LSTMs in sequence modeling tasks and motivated by their inability to capture dependencies between correlated sequences, in this work we propose a novel temporally and spatially attentive deep learning architecture that aims to predict future intent for vessels by variably attending to observations of past spatial situations. Conceptually, in our architecture, LSTM hidden states are no longer constrained to the LSTM they are associated with, and instead are also allowed to 'affect' the cell states of other spatially close LSTMs. Our model is described in greater detail in section 2.

For an agent attempting to navigate safely in a crowded environment, the agent's *domain* can be defined as the safe space surrounding the agent, the intrusion of which by any neighboring agent would cause both to have a direct impact on each other's future intent. The concept of ship domain has been crucial for safe navigation and collision-avoidance in marine transportation. Several works have used deterministic methods such as systems of equations to determine geometric dimensions of the domain(Coldwell (1983); Goodwin (1975); Pietrzykowski and Uriasz (2009); Pietrzykowski (2001)). In our work, we propose to use data-driven methods to determine a *ship domain* in order to take into account the non-procedural knowledge that comes from nautical experience of a navigator on board. We use this inferred domain to model the impact of a vessel on another based on their distances and relative orientations. Such insights or information about a system's so-called domain, along with its decisions, can be used for knowledge transfer to other deep learning models, other safety-critical domains using autonomy, or non-ML models applied to the same domain.

When trying to make a certain decision, the human brain has the natural capability to suppress idle details and focus more on certain other details. Attention networks are variants of deep learning models that mimic this capability of variably attending to different details in the input. They do this by learning a *weighting* over inputs or internal features that governs the flow of information through the network and consequently, the decision. Two variants of attention networks are relevant to our work:

Temporal Attention. Given a sequential input data, a typical auto-encoder encodes the input into a fixed embedding and decodes the embedding into a future sequence prediction under the assumption that every future timestep is uniformly dependent on observed timesteps. This causes information loss because in reality, different timesteps in an observed sequence variably affect future behavior. Using temporal attention the model is able to overcome this limitation and learn what to 'attend' to based on the input sequence and its prediction so far. Bahdanau et al. (2014) and Luong et al. (2015) proposed temporal attention mechanisms that have been successfully applied to sequence modeling tasks such as sentence translation, image caption generation, dynamic visual control problems (Vaswani et al. (2017); Xu et al. (2015); Mnih et al. (2014)).

Spatial Attention. As mentioned earlier, a conventional LSTM lacks the ability to model interactions across sequences. In our work, we attempt to overcome this limitation by modifying the conventional LSTM architecture, allowing the hidden state associated with an LSTM to not only recursively propagate to its own cell at the next time step, but also communicate some information about its own cell to other spatially close cells. The amount of information communicated is dependent on *spatial weights*, explained in greater detail in Section 2.3.

The goal of this work is to develop a deep learning based approach to predicting the future intent of socially-interacting agents. This paper:

- improves on the sequence modeling capabilities of a conventional LSTM by adding the ability to model relationships between interacting sequences, such as spatially co-located agents.
- introduces a novel interleaved temporal and spatial attention mechanism that enables variably attending to observations of such correlations to generate predictions.
- adopts a data-driven approach for inferring useful knowledge such as ship domain based on observation data, that can be used for knowledge transfer to other safety-critical domains.

2. Model Architecture

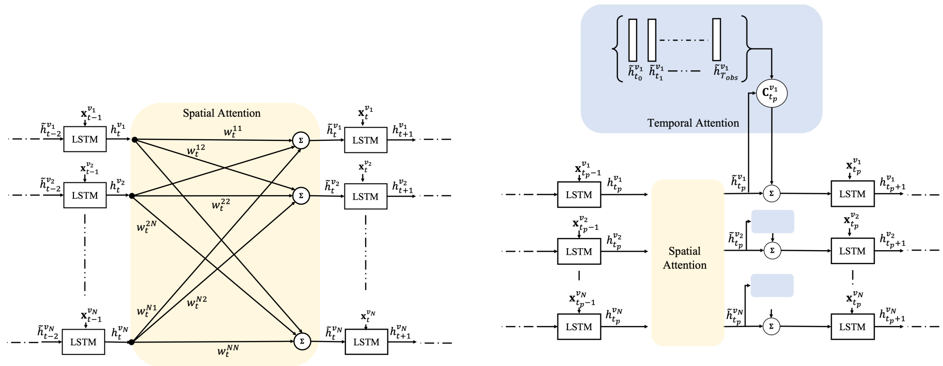
Given N vessels present in a given area and actively transmitting AIS data at the beginning of an observation time window $t_s = t_0$ to $t_s = T_{obs}$, our model uses an LSTM-based autoencoder to identically model the observed sequences of the N vessels. The observed sequence for a vessel v is denoted by $\mathbf{x}^v_{t_0:T_{obs}}$ and is composed of its positional information (latitude, longitude, speed, heading) extracted from the AIS data.

2.1. Encoding Stage

At each timestep t_s in the observed sequence spanning over time interval $[t_0, T_{obs}]$, the hidden state of every vessel v, denoted by $h^v_{t_s}$ is updated by feeding the hidden state from the previous timestep $h^v_{t_s-1}$ and the observed features at t_s, $\mathbf{x}^v_{t_s}$ to the encoder. However, the hidden state at t_s is also variably influenced by the hidden states of spatially close neighbors. As mentioned earlier, a conventional LSTM cannot take this influence into consideration. To take this spatial effect into account, we incorporate a *spatial attention mechanism*, explained in greater detail in Section 2.3. In summary, the spatial attention mechanism aggregates variable amount of information from hidden states of spatially close neighbors. The amount of information extracted from each neighbor is computed based on a weighting mechanism, and is influenced by different factors such as distance from v, relative bearing and relative heading with respect to v. The spatially-weighted hidden state of v, $\tilde{h}^v_{t_s-1}$ is then fed into the encoder at the next time step to update the hidden state of the LSTM.

2.2. Decoding Stage

Every spatially weighted hidden state, $\tilde{h}^v_{t_s}$ corresponding to every vessel v is a vector representation of the *spatial situation* at t_s. It summarises the orientation of neighbors around v, their distances from v, their headings with respect to v and their resulting influence on v. The decoding LSTM receives a sequence of these spatially weighted hidden states for each vessel v for every t_s in the observation time window $[t_0, T_{obs}]$. Similar to the encoding stage, for every time step t_p in the prediction time window from $T_{obs} + 1$ to T_{pred}, the decoder computes the spatial influence of the *future intent* of neighbors on the *future intent* of the self and vice versa using the same spatial attention mechanism. This is analogous to a pedestrian altering their path if they *anticipate* collision with another pedestrian at a future time step. Further, in order to predict the intent of v given a sequence of observed trajectory, it is useful to compare the *anticipated* situation at every timestep t_p in the prediction time window, $[T_{obs+1}, T_{pred}]$ with the history of observed situations, $\tilde{h}^v_{t_s}$. This is similar to a pedestrian using knowledge from past experiences to determine a safe future trajectory. In the maritime domain, this is similar to a cargo ship recollecting from past experiences, the safest way to maneuver around a fishing boat when the fishing boat is present at a certain distance and relative bearing from it. Therefore, to make the model better gauge the spatial influence of the future intent of neighbors on the future intent of the self and vice versa, we interleave the spatial attention mechanism with the *temporal attention* mechanism, as shown in Figure 1(*b*). The temporal attention mechanism compares the spatially weighted hidden state at a time step t_p in the prediction time window to all spatially weighted hidden states in $[t_0, T_{obs}]$. This is analogous to a vessel reacting similarly to situations it has observed previously and is used to make the model aware of similarity in spatial situations, hence enabling it to learn from the encoded input and react similarly. The temporally spatially weighted hidden state at a time step is then used to compute the hidden state corresponding to v at the next time step, and the predicted intent at the next time step. The temporal attention mechanism is explained in further detail in Section 2.4.



(a) Spatial Attention Mechanism (b) Temporal Attention Mechanism

Figure 1: The spatial attention mechanism is used in the encoding and decoding stage to model the spatial influence of neighbors on the intent of self and vice-versa. The temporal attention mechanism is used in the decoding stage to enable learning from observed ‘situations’ by comparing the current hidden state of each vessel with its history of spatially-weighted hidden states.

2.3. Spatial Attention

A socially interacting agent’s intent is not only influenced variably by neighbors depending on their distance from it, it is also affected by other factors, such as relative bearing from the agent and their heading angle. For instance, in the pedestrian domain, a human is most likely to be influenced by neighboring pedestrians in its line-of-sight than those behind it. In the same way, in the maritime domain, the effect of a neighbor on a vessel’s intent would vary with its orientation around the vessel. To incorporate this multimodal spatial effect, we introduce a spatial attention mechanism to model the influence of spatially close vessels on each other. While data-driven approaches to vessel intent modeling are limited, several pioneering works that model human-human interaction in the pedestrian domain have introduced some forms of spatial attention (Gupta et al. (2018); Alahi et al. (2016); Sadeghian et al. (2019); Fernando et al. (2017)). However, these methods are replete with limiting assumptions on the (equal) number of neighbors that identically affect the intent of a pedestrian in each direction, or alternatively grid size. In contrast to these approaches, we let the model deduce the *vessel domain* from the observed data. Any neighboring agent that violates this area around a vessel would be deemed as a threat to its navigational safety and would cause the vessel to initiate timely maneuvers to avoid risk of collision. We denote this domain by a learn-able parameter S. This parameter S is treated like any other trainable parameter in the model and is learned from training on observed data. At time t, the spatial influence of a neighboring vessel v_2 on a vessel, v_1 is dependent on three prominent factors: the distance of v_2 from v_1 at t, d_t^{21}; the heading angle of v_2 with respect to v_1 at t, denoted by ϕ_t^{21}; and, the relative bearing of v_2 with respect to the heading of v_1 at time t, denoted by θ_t^{21}. At a time step t, the spatial influence of v_2 on v_1 is then determined by computing its spatial weight, w_t^{21},

$$w_t^{21} = \mathbf{ReLU}(S(\theta_t^{21}, \phi_t^{21}) - d_t^{21}) \tag{1}$$

$\mathbf{ReLU}$ is a non-linear activation function commonly used in deep neural networks. For any input i, $\mathbf{ReLU}(i) = max(0, i)$. Here, this activation function ensures that if the distance of v_2 from v_1, d_t^{21} is greater than the corresponding domain value $S(\theta_t^{21}, \phi_t^{21})$, v_2 would have no effect on the intent of v_1. The *spatially weighted* hidden state of v_1 is then computed as:

$$\tilde{h}_t^{v_1} = w_t^{11} h_t^{v_1} + w_t^{21} h_t^{v_2} + \ldots + w_t^{N1} h_t^{v_N} \tag{2}$$

This spatially weighted hidden state is then fed to the encoder or the decoder at the next time step to update the hidden state corresponding to v_1, $h_{t+1}^{v_1}$. Our spatial attention mechanism is shown in Figure 1(*a*).

2.4. Temporal Attention

At every timestep t_p in the prediction time window $[T_{obs+1}, T_{pred}]$, the decoder first uses the spatial attention mechanism to summarise the 'situation' or the orientation of neighbors around v_1 and their influence on v_1 thereof. It then compares this spatially weighted hidden state $\tilde{h}_{t_p}^{v_1}$ with all $\tilde{h}_{t_s}^{v_1}$, $t_s \in [t_0, T_{obs}]$, to understand from similar past experiences the best way to navigate through this situation. This is done using a temporal attention mechanism, shown in Figure 1(*b*). In our model, we specifically use the attention mechanism introduced by Luong et al. (2015). At each time step t_p in the prediction sequence, the LSTM associated with v computes a *context vector*, $\mathbf{C}_{t_p}^v$ as the weighted sum of (spatially-weighted) hidden states from the observed time window:

$$\mathbf{C}_{t_p}^v = \sum_{t_s=t_0}^{T_{obs}} = \alpha_{t_p} \tilde{h}_{t_s}^v \tag{3}$$

The alignment vector α_{t_p}, with length equal to the number of time steps in the observed sequence, is derived by comparing the current spatially-weighted hidden state $\tilde{h}_{t_p}^p$ with each spatially-weighted hidden state $\tilde{h}_{t_s}^p$ from the observed sequence:

$$\alpha_{t_p} = \mathbf{align}(\tilde{h}_{t_s}^v, \tilde{h}_{t_p}^v) = \frac{\mathbf{exp}(\mathbf{score}(\tilde{h}_{t_s}^v, \tilde{h}_{t_p}^v)}{\sum_{s'} \mathbf{exp}(\mathbf{score}(\tilde{h}_{t_{s'}}^v, \tilde{h}_{t_p}^v)} \tag{4}$$

where $\mathbf{score}$ is called *content-based function* and is used to quantify the similarity of a source hidden state and a target hidden state. An observed experience or situation being identical to the current situation would cause the two spatially weighted hidden states being compared to be equal. To allow such similar observed experiences to be assigned a higher $\mathbf{score}$ in Equation 4, we use dot product to compute the $\mathbf{score}$. This is because dot product is maximum when the two hidden states being compared are 'equal', which would mean that the spatial situations being summarized by the two spatially weighted hidden states being compared are identical. Therefore,

$$\mathbf{score}(\tilde{h}_{t_s}^v, \tilde{h}_{t_p}^v) = \tilde{h}_{t_s}^v \cdot \tilde{h}_{t_p}^v \tag{5}$$

The soft attention context vector $\mathbf{C}_{t_p}^v$ is computed at every $t_p \in [T_{obs}+1, T_{pred}]$. At every time step, it is concatenated with the computed spatially weighted hidden state, $\tilde{h}_{t_p}^v$ and is further used to update the hidden state of the decoder at the next timestep, t_p+1, $h_{t_p+1}^v$. A fully connected linear layer is used to convert the updated hidden state into a predicted intent for v_1 at t_p+1.

$$\tilde{h}_{t_p}^v = \mathbf{concat}(\mathbf{C}_{t_p}^v, \tilde{h}_{t_p}^v) \tag{6}$$

$$\mathbf{x}_{t_p+1}^v = \mathbf{linear}(h_{t_p+1}^v) \tag{7}$$

where $\mathbf{x}_{t_p+1}^v$ is the predicted position or intent at t_{p+1} for v.

For more details on the model architecture, please refer to the full technical report Sekhon and Fleming (2019).

3. Implementation

Dataset and Pre-processing. To evaluate our model, we use AIS records within U.S. coastal waters from January 2017[1]. Because we are interested in being able to predict intent in crowded environments, we train and validate our model on available AIS data around San Diego Harbor (UTM Zone 11) from January 2017. Vessels update their AIS information at different rates, and because our model processes concurrent AIS information from all vessels within a certain area, we resample and interpolate the raw AIS data to one minute intervals. We evaluate the intent prediction of our model for 5 time steps (5 minutes) in the future given a history of positional data for all vessels in a scene over the past 5 time steps. We extracted 8676 such samples from the processed AIS data, using 80% for training, 10% for validation and the remaining 10% for testing the trained models. We observed that in many cases, the recorded AIS speed and Heading values are not consistent with the recorded positional data (latitude, longitude values). Therefore, we use only two input features, i.e., latitude and longitude values.

Architecture Details. To substantiate our choice of architecture, we trained and evaluated our model in an ablative setting:

- *LSTM+Spatial+Temporal Attention.* This refers to our proposed model, with a spatial attention mechanism to incorporate the spatial interactions with other agents in close proximity and a temporal attention mechanism to enable the model to learn variably from its history of observed experiences.
- *LSTM+Spatial Attention.* This refers to our model with only the spatial attention mechanism to incorporate spatial interactions with other neighbors in close proximity. This model does not take into account temporal attention mechanism to understand the variable effect of observed situations on the predicted intent. The encoding and decoding stage for this model are essentially identical.
- *LSTM+Temporal Attention.* This model consists of a vanilla-LSTM with a temporal attention mechanism. This model is agnostic to spatial interactions with neighbors in close proximity while predicting intent for a certain vessel v. It, however, does incorporate the variable temporal effects of different timesteps in the observed time window for each vessel v while predicting intent.
- *Vanilla-LSTM.* This baseline model consists of a single-layer vanilla-LSTM that tries to model intent while being agnostic to any spatial or temporal influences.

4. Evaluation

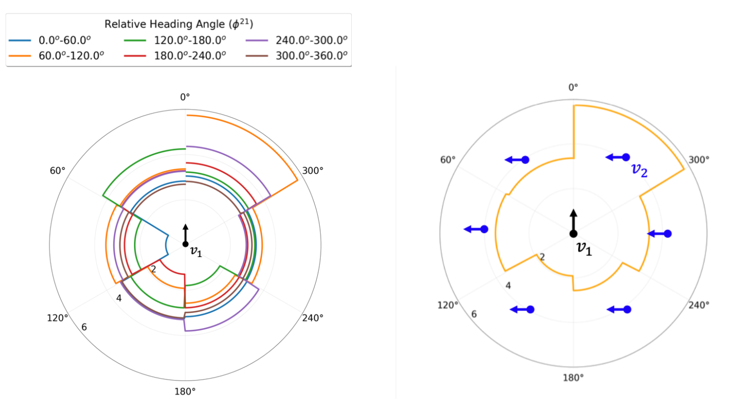
We evaluate the performance of our model in different ablative settings on data from UTM Zone 11[2]. We report performance on two metrics commonly used in the pedestrian domain for evaluating trajectory prediction methods(Alahi et al. (2016); Gupta et al. (2018); Sadeghian et al. (2019)). *Average Displacement Error* (ADE) is defined as the average displacement between the predicted trajectory and ground truth trajectory over the prediction time span $[T_{obs+1}, T_{pred}]$ across all the vessels in the frame. *Final Displacement Error* (FDE) is the displacement error between the final predicted positions and ground truth positions at the end of the prediction time span, i.e. at T_{pred} averaged over all the vessels in the frame.

1. retrieved from `https://marinecadastre.gov`
2. Code available at: `https://github.com/coordinated-systems-lab/VesselIntentModeling`

Metric	*Vanilla-LSTM*	*LSTM + Temporal Attention*	*LSTM + Spatial Attention*	*LSTM + Spatial + + Temporal Attention*
ADE	0.04567	0.04152	0.03912	**0.03314**
FDE	0.05377	0.05601	0.04292	**0.03840**

Table 1: Quantitative Results for all models on evaluation dataset from UTM Zone 11. The ADE and FDE values are reported in nautical miles and are computed for predicted intent over 5 minutes using observed AIS information from 5 minutes.

Table 1 shows the ADE and FDE values for different variants of our model. Since the *vanilla-LSTM* does not incorporate spatial interactions and solely uses the vessel's own observed history to predict its intent, the *vanilla-LSTM* and its variant with temporal attention perform the worst. The vanilla-LSTM + spatial attention model is able to perform better than the models without any spatial attention mechanism because of its ability to understand the causal relationship between a vessel's neighborhood and its intent. Adding temporal attention to this model further improves performance because the model is then able to learn from past "situations" as observed by the self and variably attend to these while predicting intent, alongwith understanding and incorporating spatial influences. The hidden layer dimensions of LSTM across all models is 6. Despite the LSTM encoder and decoder being single-layer LSTMs with very small hidden dimensions, our model performs well because of its interleaved spatial and temporal attention mechanisms that are able to intelligently capture the complex cause-effect relationships among neighbors, their observed experiences and each vessel's individual intent. Please see the full technical report for other training and implementation details (Sekhon and Fleming (2019)).



(a) Learned Domain for various ϕ_{21} (b) Learned Domain for ϕ_{21}=90^o

Figure 2: Vessel domain parameter as learned by our spatially and temporally attentive model via training on vessel AIS data from UTM Zone 11, January 2017.

5. Discussion

As mentioned earlier, prior literature on data-driven modeling intent of interacting agents model spatial interactions under strong assumptions such as uniform influence of all neighbors in a certain grid space. By virtue of introducing a learnable vessel domain parameter, our model is able to differentiate and variably attend to different agents at the same distance from an agent, based on

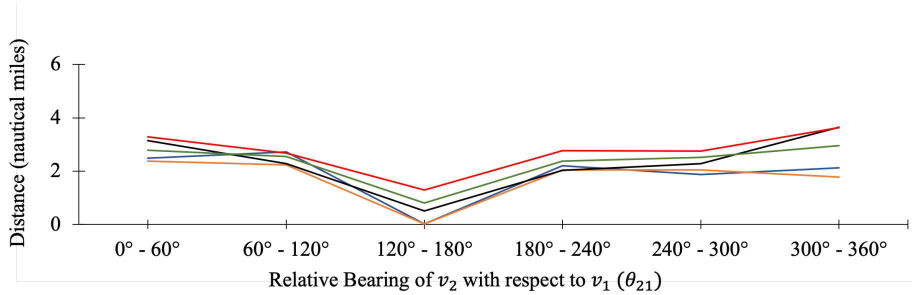


Figure 3: Robustness of learned domain parameter to random initializations for $120^o < \phi_{21} \leq 180^o$

their relative headings and relative bearings from the self. The vessel domain parameter as learned by our spatially and temporally attentive model is shown in Figure 2(a). In general, the model learns a farther distance from the self for relative bearings that fall in the line-of-sight of the vessel, and closer distances from the self for relative bearings that fall behind the vessel. Further, the model learns a farther distance for all neighbors v_2 that are approaching v_1 head-on, with $120^o \leq \phi_{21} < 180^o$. This implies that between two neighbors, both at equal distances from v_1 and heading in the same direction, v_1 would be more influenced by the one that is approaching it head-on than another with the same relative heading but at a different relative bearing from v_1. Figure 2(b) shows the vessel domain as learned by the model for a vessel v_2 with $\phi_{21} = 90^o$ for various θ_{21} values. As can be seen from the figure, the model attends more to v_2 when it tries to cross it from its starboard side, as compared to other relative bearings. This is understandable because neighbors with the same orientation at other relative bearings have no influence on its intent or high-level trajectory, and pose no immediate risk of collision to v_1.

In practice, deep neural networks are initialized to random weights before beginning the training process. Since this randomness causes the optimal parameter search to initiate at a different point and progress differently each time the model is trained on the same dataset, it may cause the model to converge at a different parameter configuration each time. To evaluate the robustness of our model to randomness in learning, we train our model using 5 different random initialization seeds. Figure 3 shows the learned domain values for a scenario with a neighboring vessel v_2 at a relative heading ($120^o < \phi_{21} \leq 180^o$) with respect to v_1 for 5 different random initializations. As can be seen from the figure, the model is nearly able to reproduce the learned domain parameter across all the initializations.

6. Conclusion

In this work, we propose a learning-based method for modeling intent of vessels, hence enabling safe navigation in cluttered environments such as harbors. Despite being trained on only positional data, our novel architecture is able to accurately model vessel intent and is also able to infer knowledge such as vessel domain from observed data. Our model can be used alongside other sophisticated data sources, such as sensors like LiDARs, radars, etc. for improved accuracy and user trust in safety-critical scenarios. While we validate our approach on the maritime domain, this method can be easily adopted to model intent and spatial interactions for other socially interacting autonomous agents, such as pedestrians, automobiles and unmanned aerial vehicles.

Acknowledgments

This material is based upon work supported in part by the National Science Foundation under Grant No. CNS: 1650512, conducted in the NSF IUCRC Center of Visual and Decision Informatics, through the guidance of Leidos Corporation.

References

Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014.

T. G. Coldwell. Marine traffic behaviour in restricted waters. *Journal of Navigation*, 36(3):430444, 1983. doi: 10.1017/S0373463300039783.

Tharindu Fernando, Simon Denman, Sridha Sridharan, and Clinton Fookes. Soft + hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection. *Neural networks : the official journal of the International Neural Network Society*, 108:466–478, 2017.

Elisabeth M. Goodwin. A statistical study of ship domains. *Journal of Navigation*, 28(3):328344, 1975. doi: 10.1017/S0373463300041230.

Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, number CONF, 2018.

Sepp Hochreiter and Jrgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997. doi: 10.1162/neco.1997.9.8.1735.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation, 2015.

Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 2204–2212, Cambridge, MA, USA, 2014. MIT Press. URL http://dl.acm.org/citation.cfm?id=2969033.2969073.

Zbigniew Pietrzykowski. The analysis of a ship fuzzy domain in a restricted area. *IFAC Proceedings Volumes*, 34(7):45 – 50, 2001. ISSN 1474-6670. doi: https://doi.org/10.1016/S1474-6670(17)35057-7. URL http://www.sciencedirect.com/science/article/pii/S1474667017350577. IFAC Conference on Control Applications in Marine Systems 2001, Glasgow, Scotland, 18-20 July 2001.

Zbigniew Pietrzykowski and Janusz Uriasz. The ship domain a criterion of navigational safety assessment in an open sea area. *Journal of Navigation*, 62(1):93108, 2009. doi: 10.1017/S0373463308005018.

Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Jasmine Sekhon and Cody Fleming. A spatially and temporally attentive joint trajectory prediction framework for modeling vessel intent, 2019. URL https://arxiv.org/abs/1912.09429.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention, 2015.