A MODULAR, DATA-FREE PIPELINE FOR MULTI-LABEL INTENTION RECOGNITION IN TRANSPORTATION AGENTIC AI APPLICATIONS

Xiaocai Zhang, Ph.D.
Department of Infrastructure Engineering,
Faculty of Engineering and Information Technology,
The University of Melbourne

Hur Lim
School of Computing,
National University of Singapore

Ke Wang, Ph.D., Corresponding Author
Institute of High Performance Computing (IHPC),
Agency for Science, Technology and Research (A*STAR)

Zhe Xiao, Ph.D.
Institute of High Performance Computing (IHPC),
Agency for Science, Technology and Research (A*STAR)

Jing Wang, Ph.D.
Civil and Environmental Engineering,
The Hong Kong University of Science and Technology

Kelvin Lee, Ph.D.
Institute of High Performance Computing (IHPC),
Agency for Science, Technology and Research (A*STAR)

Xiuju Fu, Ph.D.
Institute of High Performance Computing (IHPC),
Agency for Science, Technology and Research (A*STAR)

Zheng Qin, Ph.D.
Institute of High Performance Computing (IHPC),
Agency for Science, Technology and Research (A*STAR)

Word Count: 2963 words + 7 table(s) $\times$ 250 = 4713 words

Submission Date: July 27, 2025

ABSTRACT

In this study, a modular, data-free pipeline for multi-label intention recognization is proposed for agentic AI applications in transportation. Unlike traditional intent recognition systems that depend on large, annotated corpora and often struggle with fine-grained, multi-label discrimination, our approach eliminates the need for costly data collection while enhances the accuracy of multi-label intention understanding. Specifically, the overall pipeline, named DMTC, consists of three steps: 1. Using prompt engineering to guide large language models (LLMs) to generate diverse synthetic queries in different transport scenarios; 2. Encoding each textual query with a Sentence-T5 model to obtain compact semantic embeddings; 3. Training a lightweight classifier using a novel online focal-contrastive (OFC) loss that emphasizes hard samples and maximizes inter-class separability. The applicability of the proposed pipeline is demonstrated in an agentic AI application in the maritime transportation context. Extensive experiments show that DMTC achieves a Hamming loss of 5.35% and an AUC of 95.92%, outperforming state-of-the-art multi-label classifiers and recent end-to-end SOTA LLM-based baselines. Further analysis reveals that Sentence-T5 embeddings improve subset accuracy by at least 3.29% over alternative encoders, and integrating the OFC loss yields an additional 0.98% gain compared to standard contrastive objectives. In conclusion, our system seamlessly routes user queries to task-specific modules (e.g., ETA information, traffic risk evaluation and other typical scenarios in transportation domain), laying the groundwork for fully autonomous, intention-aware agents without costly manual labelling.

Keywords: multi-intention recognition, online focal-contrastive loss, data-free, agentic AI system

INTRODUCTION

Over the past decades, Artificial Intelligence (AI) has transformed transportation, empowering applications such as estimated time of arrival (ETA) prediction (*1*), fuel consumption estimation (*2*), traffic hotspot detection (*3*), vessel trajectory forecasting (*4*), and vessel type recognition (*5*) etc. Building on advances in large language models (LLMs) and reinforcement learning, agentic AI—autonomous systems that perceive, plan, and act with minimal oversight—is poised to deliver even greater autonomy in routing, scheduling, and decision-making in transportation sector. However, accurate multi-label intention understanding remains a bottleneck: mapping free-form user queries to precise operational goals requires fine-grained discrimination across high-dimensional outputs, label correlations, and imbalanced classes. Besides, collecting large, annotated datasets in the transportation domain is labor-intensive, and existing classifiers often struggle with hard samples. To address the challenges of costly data collection and low accuracy in multi-label intention understanding within agentic transportation systems, we introduce **DMTC** (Data-less Multi-label Text Classification), a modular, data-free framework designed to minimize data requirements while achieving high accuracy in multi-label intention understanding. The key contributions of this work are:

- **Modular pipeline for multi-label intention recognition**: A scalable framework that facilitates seamless integration with agentic AI systems in diverse transportation domains.
- **Zero-shot synthetic data generation via prompt engineering**: An approach that eliminates the need for manual data annotation by generating synthetic datasets, providing valuable benchmarks for the transportation AI community.
- **Novel online focal-contrastive loss function**: A novel loss function that enhances model performance in multi-label classification tasks by focusing on hard-to-classify samples and improving feature discrimination.

RELATED WORK

Intention-based text classification has been a core task in natural language processing, underpinning applications in domains as diverse as healthcare, finance, and transportation. Advances in agentic AI—autonomous systems that perceive, plan, and act with minimal oversight—have further extended this functionality, enabling automated model orchestration through precise intent detection from user textual queries.

Text classification methods in transportation typically fall into five categories: rule-based approaches, traditional machine learning (excluding neural networks and LLMs in this study), neural network models, large language models (LLMs), and hybrid systems.

Rule-based methods apply predefined linguistic rules or keyword dictionaries to interpret text. In the study by (*6*), they constructed a tweet-based activity classifier using a manually curated keyword list to label posts into seven travel-related categories (*6*).

Traditional machine learning methods in text classification predominantly employ SVM and Naive Bayes classifiers. In the work by (*7*), they trained an SVM to distinguish transport-related from non-transportation texts. Styawati et al. (*8*) generated Word2Vec embeddings for Gojek and Grab user reviews before applying SVM for sentiment analysis. Klaithin et al. (*9*) used Naive Bayes to categorize Twitter comments into six traffic-event classes (e.g., accidents, announcements). Suat et al. (*10*) combined Doc2Vec embeddings with SVM to classify tweets as accident-related or not. (*11*) employed a Bayesian network to identify maritime incident descriptions in news articles.

Advanced neural network methods—particularly RNNs (LSTM/GRU) and CNNs—have become the dominant approach for transportation-related text classification. (*12*) combined CBOW embeddings with an LSTM–CNN hybrid to distinguish traffic-relevant from traffic-irrelevant content. (*7*) employed Bi-LSTM for sentiment analysis on social media posts and ITS office reports. (*13*) used CBOW embeddings with CNN to detect traffic-related microblogs, outperforming both SVM and MLP baselines. (*14*) evaluated GloVe embeddings with RNN, GRU, and LSTM for traffic-tweet detection and severity classification, with LSTM achieving the highest accuracy (94.2 %). (*15*) applied byte-pair encoding for data augmentation followed by a CNN–Bi-LSTM pipeline to classify tweets into traffic, accident, disaster, and social-event categories. (*16*) introduced an Ernie-Gram–Bi-GRU attention architecture for air-traffic-control instruction recognition.

Among LLM-based methods in transportation, BERT and its variants have become the de facto standard. (*17*) applied the bidirectional encoder representations from BERT to classify tweets as traffic-related or unrelated. (*18*) leveraged BERT for sentiment analysis and multi-class classification of Madrid Metro tweets—covering incidents, road closures, construction, delays, public transport, and other information—achieving 99.37 % accuy against NB, decision tree, and SVM baselines. (*19*) explored BERT for named-entity recognition in ATC commands and for detecting speaker roles (e.g., controller vs. pilot) in dialog. (*20*) demonstrated that RoBERTa—an optimized BERT pretraining—outperformed Word2Vec, GloVe, FastText, BERT, and XLNet on sentiment classification, reaching 97 % accurac (*21*) introduced a focal-loss-trained ConvBERT model for maritime versus non-maritime text classification, achieving an F1 score of 99.97 %. Finally, (*22*) evaluated multiple transformer architectures (GPT, BERT, RoBERTa, BERTweet, CrisisTransformer) for public sentiment analysis of transportation issues on Twitter. Hybrid methods combine multiple techniques to leverage their complementary strengths. (*23*) used BERT for contextual embeddings, then applied ANN, CNN, and LSTM architectures to classify maritime-incident texts. (*24*) similarly extracted BERT features from tweets and employed a CNN to detect traffic events, reporting that BERT outperformed ELMo and Word2Vec, with LSTM achieving 94.4% accuracy. (*25*) semantically tagged NAVTEX safety messages using a Bi-LSTM+CRF pipeline aligned with the Common Maritime Data Structure. (*26*) combined TF-IDF, POS tagging, and n-grams with χ^2 feature selection, then used a hybrid LSTM-CNN model to classify customer service claims—routing valid requests via gradient tree boosting. (*27*) fine-tuned BERT on ATC communications with additional masked-language model training, enabling few-shot intent recognition in air-traffic control dialogues.

While rule-based, traditional ML, neural networks, LLMs, and hybrid approaches each offer valuable capabilities for transportation text classification, none simultaneously eliminate the need for annotated data, support fine-grained, multi-label intent detection, and integrate seamlessly into agentic AI workflows. To fill this gap, a modular, data-free pipeline is designed specifically for zero-annotation, multi-label intention recognition in autonomous transportation systems.

METHOD

Figure. 1 outlines the workflow of DMTC for data-free multi-label intention understanding, as implemented in a maritime context. The pipeline comprises three core components: (A) Synthetic data generation via prompt-engineered LLMs; (B) Semantic representation with sentence-T5; (C) Online focal-contrastive learning for multi-label classification.

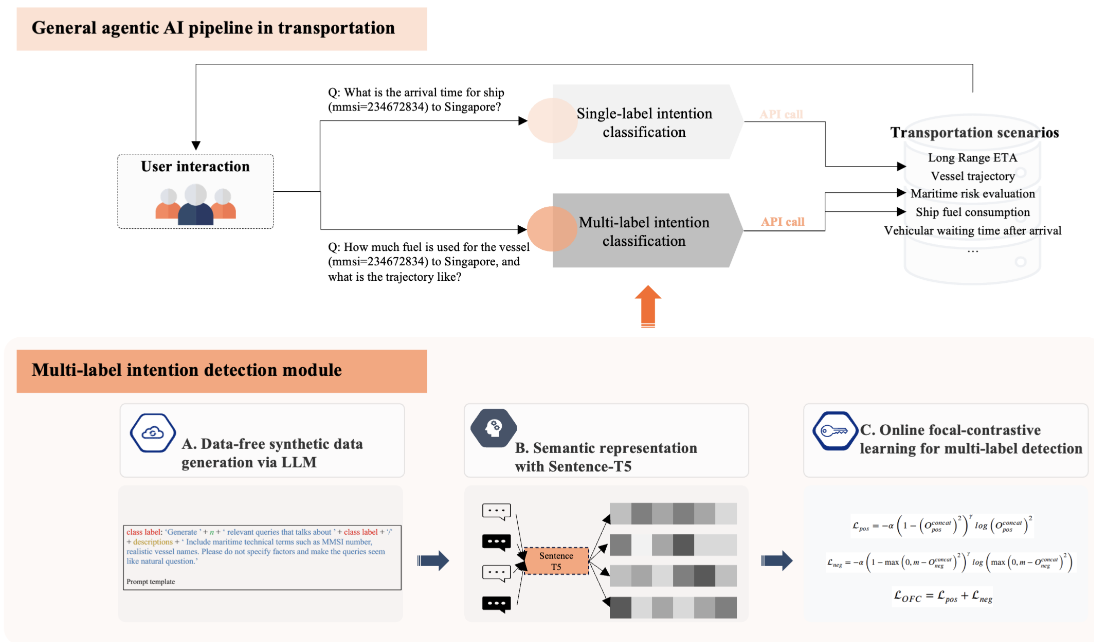

FIGURE 1 The proposed DMTC pipeline for multi-label intention recognization in the maritime transportation

Data-free synthetic data generation with LLM

Prompt engineering is employed to generate synthetic datasets without relying on pre-existing data. As illustrated in Figure 2, a unified prompt template is used for this data generation process. The prompt includes the class label, its description, and the desired sample size, where the ’description’ provides a clear and concise sentence elaborating on the characteristics of the specified class.

> class label: ‘Generate ’ + *n* + ‘ relevant queries that talks about ’ + class label + ‘/’ + descriptions + ‘ Include maritime technical terms such as MMSI number, realistic vessel names. Please do not specify factors and make the queries seem like natural question.’
>
> Prompt template

FIGURE 2 Template of prompt for generating data for each class.

Figure. 3 illustrates the prompt for the “long-range ETA in maritime” class. Here, the class label is set to "long-range ETA in maritime" and n is set to 100, targeting the generation of 100 samples. The prompt also includes the description “estimated time of arrival for maritime vessels and ships” to clarify the scenario. Once each class template is defined, its prompt is submitted to LLaMA 2 to automate synthetic data creation. For multi-label sample generation, we concatenate the individual text segments corresponding to each label. This plug-and-play process can be readily applied to other agentic AI systems.

> 'long-range ETA prediction': 'Generate ' + '100' + ' relevant queries that talks about ' + 'long-range ETA prediction' + '/' + 'estimated time of arrival of maritime vessels and ships prediction' + ' Include maritime technical terms such as MMSI number, realistic vessel names. Please do not specify factors and make the queries seem like natural question.'
>
> Prompt example

FIGURE 3 An example of constructing the prompt for generating data.

Semantic representation with Sentence-T5

Sentence-T5 (*28*), a derivative of the T5 model, is implemented to learn the contextual representations of texts. It consists of an encoder E, and a decoder D, both built upon the Transformer architecture. In the pre-training stage, the outputs of the Sentence-T5 are formulated as

$$O = D\left(\mathrm{Pool}\left(E\left(S;\theta_E\right)\right);\theta_D\right), \tag{1}$$

Where S is the input text, θ_E and θ_D are the parameters of the encoder and decoder, respectively, and Pool represents the average pooling operation. The encoder E transforms the input text into a sequence of contextual embedding blocks, as represented by

$$E\left(S;\theta_E\right) = E_N\left(\cdots E_1\left(S;\theta_{E_1}\right);\theta_{E_N}\right), \tag{2}$$

where θ_{E_i} denotes the parameters of the ith layer of the Transformer. The average pooling operation is applied across the sequence of encoder outputs to create a pooled representation H, which is a distinct attribute of the Sentence-T5 to T5 model. The process is formulated as

$$H = \mathrm{Pool}\left(E\left(S;\theta_E\right)\right). \tag{3}$$

The decoder D uses the pooled representation H to generate the output, as described by

$$O = D\left(H;\theta_D\right). \tag{4}$$

To retain the sequential structure of the input text, positional encodings (PE) are incorporated into the embeddings, ensuring the model accounts for word order, as shown by

$$E_1\left(S\right) = E_{word}\left(S\right) + PE, \tag{5}$$

where PE represents the positional encoding vector.

Online focal-contrastive (OFC) learning for multi-label classification

A novel contrastive loss function, termed OFC loss, is introduced for multi-label intention classification. Designed to capture distinctive features by emphasizing hard samples, OFC incorporates hard-negative sampling strategies to strengthen inter-class separability and handle complex label combinations. The initial step in OFC involves the creation of pairs comprising positive samples (S_{pos}), representing instances with shared labels within the pair, and negative samples (S_{neg}), consisting of samples with disparate labels,

$$S_{pos} = \left\{\left(a_i^{l_a}, b_i^{l_b}\right) \mid i = 1, 2, \cdots, n\right\}, \tag{6}$$

where $l_a = l_b$ in S_{pos} and $l_c \neq l_d$ in S_{neg}. The similarity between two instances, i.e., a^{l_a} and b^{l_b}, is defined as

$$Sim\left(a^{l_a}, b^{l_b}\right) = \frac{a^{l_a} \cdot b^{l_b}}{\left\|a^{l_a}\right\| \cdot \left\|b^{l_b}\right\|}. \tag{7}$$

Subsequently, we compute the similarity for each instance pair in S_{pos} and S_{neg}, for example,

$$D_{pos} = \left\{Sim\left(a_i^{l_a}, b_i^{l_b}\right) \mid i = 1, 2, \cdots, n\right\}, \tag{8}$$

Next, hard sample pairs—instances that are challenging to classify correctly—are introduced. Specifically, for positive sample pairs, hard sample pairs is defined using the following formulations, as elucidated by

$$H_{pos} = \left\{\left(a_i^{l_a}, b_i^{l_b}\right), \text{ if } Sim\left(a_i^{l_a}, b_i^{l_b}\right) > T_{neg} \mid i = 1, 2, \cdots, n\right\} \tag{9}$$

and

$$T_{neg} = \min\left(D_{neg}\right), \tag{10}$$

where T_{neg} is a threshold determined by the minimum similarity within the similarity set of the negative samples. Similarly, the identification of hard sample pairs from negative samples H_{neg} follows an opposite process. Furthermore, the refined positive samples O_{pos} are derived by calculating the relative complement of H_{pos} within S_{pos},

$$O_{pos} = S_{pos} - H_{pos}. \tag{11}$$

Similarly, the refined negative samples O_{neg} are obtained by

$$O_{neg} = S_{neg} - H_{neg}. \tag{12}$$

Following that, we re-arrange the refined positive samples O_{pos} in descending order, as demonstrated by

$$O_{pos}^{sorted} = \left\{Sim\left(a_{(1)}^{l_a}, b_{(1)}^{l_b}\right), Sim\left(a_{(2)}^{l_a}, b_{(2)}^{l_b}\right), \cdots, Sim\left(a_{(k)}^{l_a}, b_{(k)}^{l_b}\right)\right\}. \tag{13}$$

On the contrary, we organize the refined positive samples O_{neg} in ascending order. We choose the initial p percent from the sorted positive and negative samples to create O_{pos}^{top} and O_{neg}^{top}, as defined by

$$O_{pos}^{top} = \text{select\_top\_element}\left(O_{pos}^{sorted}, p\right), \tag{14}$$

where p is a percentage ratio and select_top_element is utilized to extract the initial $p\%$ of the whole elements. Subsequently, we combine these initially selected $p\%$ elements with the extracted hard sample pairs to form both the positive samples O_{pos}^{concat} and the negative samples O_{neg}^{concat},

$$O_{pos}^{concat} = \text{concat}\left(O_{pos}^{sorted}, H_{pos}\right), \tag{15}$$

The loss for the positive samples is calculated by

$$\mathscr{L}_{pos} = -\alpha\left(1 - \left(O_{pos}^{concat}\right)^2\right)^{\gamma} log\left(O_{pos}^{concat}\right)^2. \tag{16}$$

Meanwhile, the loss for the negative samples is determined by

$$\mathscr{L}_{neg} = -\alpha\left(1 - \max\left(0, m - O_{neg}^{concat}\right)^2\right)^{\gamma} log\left(\max\left(0, m - O_{neg}^{concat}\right)^2\right) \tag{17}$$

where m is a positive margin parameter and $m > 0$. The parameters α and γ refer to the weight factor and the focusing parameter, respectively. Following that, we define the OFC loss as the sum of the positive loss and the negative loss,

$$\mathscr{L}_{OFC} = \mathscr{L}_{pos} + \mathscr{L}_{neg}. \tag{18}$$

The overall learning process can be described as below: Given a sequence of tokens denoted as a series of tuples, as shown by

$$X = \{x_t \mid t = 1, 2, \cdots, n\}, \tag{19}$$

this sequence X is fed into the pre-trained Sentence-T5 model to generate its corresponding representation. This resulting representation is expressed as

$$y_{rep} = f_{ST5}\left(X; \theta_E\right), \tag{20}$$

where f_{ST5} signifies the Sentence-T5 model that has been pre-trained. The newly-proposed OFC loss is utilized for calculate the loss. The objective of network pre-training is to obtain a discriminative text representation from Sentence-T5 over the training data by minimizing the loss function, as shown by

$$\theta_{ST5} = \underset{\theta_{ST5}}{\arg\min}\, \mathscr{L}_{OFC}\left(X, \hat{y}; \theta_{ST5}\right). \tag{21}$$

Algorithm 1 outlines the procedure of the DMTC algorithm.

Algorithm 1 DMTC algorithm for multi-label text classification

---

Input: Class labels $L = \{l_1, l_2, \cdots, l_m\}$ and test instance x
Output: Probability distribution y of x

$X \leftarrow \{\}$
for l_i in L **do**
  Construct prompt templates T_{l_i} based on l_i;
  $x_{l_i} \leftarrow LLaMA\left(T_{l_i}\right)$;
  Put x_{l_i} into X;
end for
$\theta_E \leftarrow$ Load the pre-trained Sentence-T5 model;
$\theta_{ST5} \leftarrow$ Pre-train Sentence-T5 to minimize $\mathscr{L}_{OFC}$ (Eq. (18));
$(\theta_{ST5}, \theta_{FC}) \leftarrow$ Fine-tune Sentence-T5 to minimize overall loss function (cross-entropy loss function $\mathscr{L}_{CE}$);
$y \leftarrow f_{Sigmoid}\left(f_{FC}\left(f_{ST5}\left(x; \theta_{ST5}\right); \theta_{FC}\right)\right)$;
return y

EXPERIMENTS AND EVALUATION

Dataset

Since DMTC is "data-less", no external training data is required. Instead, a test set of 918 manually expert-curated samples was established, covering eight distinct AI model categories in maritime transportation. Table 1 presents an example query for each single scenario, while Table 2 provides examples of multi-labelled scenarios.

Results and analyses

Table 3 summarizes the performance of the proposed pipeline (DMTC) in multi-label intention classification task: 70.15% in accuracy, 95.92% in AUC, and 5.35% of Hamming loss. A comprehensive benchmark against popular baselines reveals clear trends. Two dense embeddings (GloVe, Word2Vec) and two contextual embeddings (BERT, MPNet) were paired with conventional machine learning classifiers (NB, SVM) and neural network classifiers (MLP, LSTM, CNN, Bi-LSTM). Models using BERT or MPNet consistently outperform those using GloVe and Word2Vec: all BERT/MPNet variants exceed 60% accuracy, 93% AUC, and 81% F1-score, whereas GloVe/Word2Vec models cap at 38% accuracy, 89% AUC, and 64% F1-score. Among dense-embedding classifiers,

TABLE 1 Example for each single class (involve only one maritime AI model) in our constructed data set

Scenarios	Query example
Long-range ETA in maritime	Forecast the ETA to next port of vessel with MMSI 564765123
Arrival time to pilotage boarding ground	what is the arrival time of vessel MOUNT ST to the Eastern Boarding Ground “B" (PEBGB) of Singapore?
Direct berthing to port	Vessel name: Achéron; MMSI: 409011089, chance to get a direct berth when she arrives at the port
Vehicular waiting time after arrival	Tanker vessel ACTIVE (IMO: 6578109, MMSI: 2877619019, call sign: TYRA). How long time will she wait for a berth after arrival?
Ship fuel consumption	How much LSFO is consumed by the tanker vessel for 10 hours?
Berth staying	estimate the time to unberth (ETU) of the container ship NYK CONSTELLATION (IMO: 9337626)
Maritime risk evaluation	Does there any cases about piracy/terrorism detected for the voyage from the Port of Ningbo to Singapore?
Vessel trajectory	Predict the route for container ship OOCL POLAND (IMO 9622588)

TABLE 2 Example for some multi-label classes (involve multiple maritime AI models) in our constructed data set

<table>
<tr><th>Scenarios</th><th>Query example</th></tr>
<tr><td>Direct berthing to port</td><td rowspan="2">How big is the change to get a direct berth for the vessel? If not, how long time will vessel Aeternum Dread wait at the anchorage?</td></tr>
<tr><td>Vehicular waiting time after arrival</td></tr>
<tr><td>Long-range ETA in maritime</td><td rowspan="3">How many hours left for the vessel Ore Italia to arrive at the destination, and the direct and indirect berthing rate? How much LSFO can be utilized during the past 3 hours?</td></tr>
<tr><td>Direct berthing to port</td></tr>
<tr><td>Ship fuel consumption</td></tr>
</table>

SVM and Bi-LSTM lead; with contextual embeddings, MLP, LSTM, and CNN perform similarly, yielding 60.02–62.96% accuracy, 93.80–95.79% AUC, and 81.21–83.58% F1-score. This disparity underscores the inadequacy of traditional embeddings for fine-grained multiclass intent classification.

TABLE 3 Performance comparison between different machine learning methods

Model	Accuracy (%)	Hamming Loss (%)	Jaccard Similarity (%)	F1-Score (%)	Precision (%)	Recall (%)	MCC (%)	AUC (%)
GloVe + NB	15.47	21.83	25.82	41.04	43.02	39.24	28.00	75.64
GloVe + SVM	31.70	14.39	42.64	59.76	65.15	55.20	52.37	84.72
GloVe + CNN	23.53	16.26	32.20	48.71	62.58	39.87	41.74	76.48
GloVe + Bi-LSTM	37.91	12.91	45.33	62.38	71.58	55.27	56.18	83.42
Word2Vec + NB	13.40	21.96	25.60	40.76	42.66	39.03	27.62	74.98
Word2Vec + SVM	37.04	12.39	46.94	63.89	73.32	56.61	56.91	88.75
Word2Vec + CNN	35.29	13.73	44.22	61.32	67.48	56.19	56.00	86.04
Word2Vec + Bi-LSTM	36.17	12.06	46.24	63.24	77.13	53.59	57.89	84.50
BERT + CNN	60.02	7.41	68.70	81.45	79.07	83.97	77.81	94.70
BERT + LSTM	61.44	6.69	71.80	83.58	79.76	**87.90**	80.48	95.79
MPNet + CNN	62.96	6.97	69.86	82.26	81.08	83.47	78.56	94.19
MPNet + MLP	62.31	7.12	69.45	81.97	80.39	83.61	78.31	94.04
MPNet + LSTM	61.98	7.43	68.37	81.21	79.51	82.98	77.63	93.80
DMTC	**70.15**	**5.35**	**75.53**	**86.06**	**86.83**	85.30	**83.32**	**95.92**

Ablation studies

Table 4 compares the introduced Sentence-T5 with other popular LLM-based contextual embedding methods, including MiniLM, RoBERTa, BERT, and MPNet. Overall, Sentence-T5 outperforms other LLMs in text representation. Among the baselines, Sentence-T5 leads across all metrics, with MPNet a close second (66.88% accuracy, 83.66% F1, 93.92% AUC). BERT and RoBERTa deliver moderate performance, while MiniLM trails, underscoring the importance of high-quality contextual embeddings for fine-grained intent tasks.

TABLE 4 Performance comparison between different embedding models

Model	Accuracy (%)	Hamming Loss (%)	Jaccard Similarity (%)	F1-Score (%)	Precision (%)	Recall (%)	MCC (%)	AUC (%)
wth MiniLM	54.47	7.90	63.52	77.69	85.74	71.03	74.21	93.97
wth RoBERTa	60.78	7.67	66.65	79.99	80.88	79.11	76.17	89.91
wth BERT	60.89	7.16	69.82	82.23	79.13	**85.58**	78.44	94.04
wth MPNet	66.88	6.29	71.91	83.66	84.14	83.19	80.54	93.92
wth Sentence-T5	**70.15**	**5.35**	**75.53**	**86.06**	**86.83**	85.30	**83.32**	**95.92**

Table 5 evaluates the impact of various contrastive losses on classification accuracy. The proposed OFC loss achieves 75.15% accuracy—at least 0.98% higher than angular-margin (AMC), N-pair (NP), cosine-similarity (CS), and online-contrastive (OC) losses. The results show that AMC loss and NP loss are not well-suited for this task. This is probably because AMC loss is particularly beneficial tasks where embeddings require clear class separation along with a specific angular relationship within the same class. NP loss is designed for tasks involving multi-label classification, which may not be suitable for this multi-label classification problem. CS and OC offer stronger baselines, with OC benefiting from hard-sample mining, yet both fall short of the OFC's emphasis on challenging examples and label combination complexity.

TABLE 5 Performance comparison between different contrastive loss functions

Model	Accuracy (%)	Hamming Loss (%)	Jaccard Similarity (%)	F1-Score (%)	Precision (%)	Recall (%)	MCC (%)	AUC (%)
wth AMC loss	45.64	9.83	56.03	71.82	80.70	64.70	66.92	91.04
wth NP loss	52.61	8.27	61.80	76.39	85.47	69.06	72.51	92.59
wth CS loss	67.54	5.69	73.64	84.82	87.69	82.14	81.88	93.28
wth OC loss	69.17	5.53	74.89	85.64	86.13	85.16	82.63	95.69
wth OFC loss	**70.15**	**5.35**	**75.53**	**86.06**	**86.83**	**85.30**	**83.32**	**95.92**

Table 6 examines the effect of incorporating contrastive learning into the overall DMTC framework. Adding the OFC loss yields consistent gains: accuracy jumps from 68.08% to 70.15% (+2.07%), F1-score climbs from 84.58% to 86.06% (+1.48%), and Hamming loss drops from 5.80% to 5.35% (–0.45%). All eight evaluation metrics improve, confirming that contrastive objectives bolster multi-label classification robustness.

TABLE 6 Effects of using the contrastive learning paradigm

Model	Accuracy (%)	Hamming Loss (%)	Jaccard Similarity (%)	F1-Score (%)	Precision (%)	Recall (%)	MCC (%)	AUC (%)
w/o contrastive learning	68.08	5.80	73.27	84.58	87.16	82.14	81.26	95.65
wth contrastive learning	**70.15**	**5.35**	**75.53**	**86.06**	**86.83**	**85.30**	**83.32**	**95.92**

Table 7 benchmarks DMTC against end-to-end LLM pipelines (GPT-4, GPT-4o) under identical prompts and test data. DMTC achieves a subset accuracy of 70.15% and Jaccard similarity of 75.53%, far exceeding GPT-4's 32.14%, and GPT-4o's 29.30%. These results highlight the limitations of prompt-only LLM approaches in specialized, multi-label intent classification and validate the effectiveness of the DMTC design.

TABLE 7 Performance comparison between different LLMs

Model	Accuracy (%)	Hamming Loss (%)	Jaccard Similarity (%)	F1-Score (%)	Precision (%)	Recall (%)	MCC (%)
GPT-4	32.14	16.00	40.51	57.66	59.13	56.26	53.05
GPT-4o	29.30	16.37	40.99	58.15	57.59	58.72	53.22
DMTC	**70.15**	**5.35**	**75.53**	**86.06**	**86.83**	**85.30**	**83.32**

Note: Since LLMs do not generate class probability distributions, measuring the AUC is not feasible.

CONCLUSIONS

This work has presented **DMTC**, a modular, data-free pipeline for multi-label intention classification in transportation applications. The key components includes: 1) **Zero-shot synthetic data**

generation via prompt-engineered large language models, removing the reliance on costly, manually labeled corpora; 2) **Sentence-T5 semantic embeddings**, which capture nuanced contextual information in user queries and enable robust multi-class discrimination; 3) **Online focal-contrastive (OFC) loss**, a novel contrastive objective that focuses on hard sample pairs to sharpen inter-class separability and improve generalization. Extensive experiments demonstrate that DMTC achieves state-of-the-art performance—exceeding 86 % F1-score, 95% AUC, and reducing Hamming loss below 5.5%—while outperforming both classical and LLM-based baselines. These results validate the effectiveness of data-free learning and contrastive feature learning for fine-grained intent understanding in complex, multi-label scenarios. Looking ahead, DMTC's plug-and-play design can be extended beyond maritime use cases to broader transportation domains, including logistics, traffic management, and autonomous mobility. Future work will explore end-to-end integration with downstream agentic AI modules, dynamic prompt adaptation for evolving intent taxonomies, and unsupervised refinement of embedding spaces to further reduce human intervention and enhance scalability.

ACKNOWLEDGEMENT

This study is supported under Maritime AI Research Programme (Grant number SMI-2022-MTP-06 funded by Singapore Maritime Institute).

REFERENCES

1. Zhang, X., X. Fu, H. Xu, X. Wei, J. Koh, D. Ogawa, and Z. Qin, Prediction of vessel arrival time to pilotage area using multi-data fusion and deep learning. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2023.
2. Chen, Z. S., J. S. L. Lam, and Z. Xiao, Prediction of harbour vessel fuel consumption based on machine learning approach. *Ocean Eng.*, Vol. 278, 2023, p. 114483.
3. Xiao, Z., X. Fu, L. Zhao, L. Zhang, T. K. Teo, N. Li, W. Zhang, and Z. Qin, Next-generation vessel traffic services systems—From “passive” to “proactive”. *IEEE Intell. Transp. Syst. Mag.*, Vol. 15, No. 1, 2022, pp. 363–377.
4. Zhang, X., X. Fu, Z. Xiao, H. Xu, W. Zhang, J. Koh, and Z. Qin, A Dynamic Context-Aware Approach for Vessel Trajectory Prediction Based on Multi-Stage Deep Learning. *IEEE Trans. Intell. Veh.*, 2024.
5. Zhang, X., Z. Xiao, X. Fu, X. Wei, T. Liu, R. Yan, Z. Qin, and J. Zhang, A Viewpoint Adaptation Ensemble Contrastive Learning framework for vessel type recognition with limited data. *Expert Syst. Appl.*, Vol. 238, 2024, p. 122191.
6. Maghrebi, M., A. Abbasi, T. H. Rashidi, and S. T. Waller, Complementing travel diary surveys with twitter data: application of text mining techniques on activity location, type and time. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, IEEE, 2015, pp. 208–213.
7. Ali, F., S. El-Sappagh, and D. Kwak, Fuzzy ontology and LSTM-based text mining: a transportation network monitoring system for assisting travel. *Sensors*, Vol. 19, No. 2, 2019, p. 234.
8. Styawati, S., A. Nurkholis, A. A. Aldino, S. Samsugi, E. Suryati, and R. P. Cahyono, Sentiment analysis on online transportation reviews using Word2Vec text embedding model feature extraction and support vector machine (SVM) algorithm. In *2021 International Seminar on Machine Learning, Optimization, and Data Science (ISMODE)*, IEEE, 2022, pp. 163–167.
9. Klaithin, S. and C. Haruechaiyasak, Traffic information extraction and classification from Thai Twitter. In *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, IEEE, 2016, pp. 1–6.
10. Suat-Rojas, N., C. Gutierrez-Osorio, and C. Pedraza, Extraction and analysis of social networks data to detect traffic accidents. *Inf.*, Vol. 13, No. 1, 2022, p. 26.
11. Teske, A., R. Falcon, R. Abielmona, and E. Petriu, Automatic identification of maritime incidents from unstructured articles. In *2018 IEEE Conference on Cognitive and Computational Aspects of Situation Management (CogSIMA)*, IEEE, 2018, pp. 42–48.
12. Chen, Y., Y. Lv, X. Wang, L. Li, and F.-Y. Wang, Detecting traffic information from social media texts with deep learning approaches. *IEEE Trans. Intell. Transp. Syst.*, Vol. 20, No. 8, 2018, pp. 3049–3058.
13. Chen, Y., Y. Lv, X. Wang, and F.-Y. Wang, A convolutional neural network for traffic information sensing from social media text. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2017, pp. 1–6.
14. Azhar, A., S. Rubab, M. M. Khan, Y. A. Bangash, M. D. Alshehri, F. Illahi, and A. K. Bashir, Detection and prediction of traffic accidents using deep learning techniques. *Cluster Comput.*, Vol. 26, No. 1, 2023, pp. 477–493.

15. Raksachat, T. and R. Chawuthai, Improving a text classifier using text augmentation: road traffic content from Twitter. In *2023 20th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, IEEE, 2023, pp. 1–4.
16. Pan, W., P. Jiang, Z. Wang, Y. Li, and Z. Liao, Ernie-Gram BiGRU Attention: An Improved Multi-Intention Recognition Model for Air Traffic Control. *Aerosp.*, Vol. 10, No. 4, 2023, p. 349.
17. Wan, X., M. C. Lucic, H. Ghazzai, and Y. Massoud, Empowering real-time traffic reporting systems with nlp-processed social media data. *IEEE Open J. Intell. Transp. Syst.*, Vol. 1, 2020, pp. 159–175.
18. Osorio-Arjona, J., J. Horak, R. Svoboda, and Y. García-Ruíz, Social media semantic perceptions on Madrid Metro system: Using Twitter data to link complaints to space. *Sustainable Cities Soc.*, Vol. 64, 2021, p. 102530.
19. Zuluaga-Gomez, J., K. Veselỳ, I. Szöke, P. Motlicek, M. Kocour, M. Rigault, K. Choukri, A. Prasad, S. S. Sarfjoo, I. Nigmatulina, et al., ATCO2 corpus: A Large-Scale Dataset for Research on Automatic Speech Recognition and Natural Language Understanding of Air Traffic Control Communications. *arXiv preprint arXiv:2211.04054*, 2022.
20. Babbar, S. and J. Bedi, Real-time traffic, accident, and potholes detection by deep learning techniques: a modern approach for traffic management. *Neural Comput. Appl.*, Vol. 35, No. 26, 2023, pp. 19465–19479.
21. Zhang, X., H. Lim, X. Fu, K. Wang, Z. Xiao, and Z. Qin, Maritime-Context Text Identification for Connecting Artificial Intelligence (AI) Models. In *2024 IEEE Conference on Artificial Intelligence (CAI)*, IEEE, 2024, pp. 899–904.
22. Pullanikkat, R., S. Poddar, A. Das, T. Jaiswal, V. K. Singh, M. Basu, and S. Ghosh, Utilizing the Twitter social media to identify transportation-related grievances in Indian cities. *Social Network Anal. Min.*, Vol. 14, No. 1, 2024, p. 118.
23. Jidkov, V., R. Abielmona, A. Teske, and E. Petriu, Enabling maritime risk assessment using natural language processing-based Deep Learning Techniques. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, 2020, pp. 2469–2476.
24. Neruda, G. A. and E. Winarko, Traffic event detection from Twitter using a combination of CNN and BERT. In *2021 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, IEEE, 2021, pp. 1–7.
25. Lee, C., H. Cho, and S. Lee, Analysis of Bi-LSTM CRF Series Models for Semantic Classification of NAVTEX Navigational Safety Messages. *J. Mar. Sci. Eng.*, Vol. 12, No. 9, 2024, p. 1518.
26. Khodadadi, A., S. Ghandiparsi, and C.-N. Chuah, A natural language processing and deep learning based model for automated vehicle diagnostics using free-text customer service reports. *Mach. Learn. Appl.*, Vol. 10, 2022, p. 100424.
27. Hui, Y., K. Cai, M. Zhang, S. Qian, and Y. Yang, ATCBERT: Few-shot Intent Recognition for Air Traffic Control Instruction Understanding. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2023, pp. 4510–4517.
28. Ni, J., G. H. Ábrego, N. Constant, J. Ma, K. B. Hall, D. Cer, and Y. Yang, Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *arXiv preprint arXiv:2108.08877*, 2021.