Linköping University | IDA - Department of Computer and Information Science
Independent thesis Basic level, 16 hp/bachelor | Bachelor’s Programme in Programming
Spring 2021 | LIU-IDA/LITH-EX-G--21/041—SE

Conversational Engine for Transportation Systems

Albin Sidås
Simon Sandberg

Rita Kovordanyi
Jalal Maleki

LINKÖPINGS
UNIVERSITET

Copyright

Conversational Engine For Transportation Systems

Albin Sidås
Linköping, Sweden
albsi727@student.liu.se

Simon Sandberg
Linköping, Sweden
simsa999@student.liu.se

ABSTRACT

Today's communication between operators and professional drivers takes place through direct conversations between the parties. This thesis project explores the possibility to support the operators in classifying the topic of incoming communications and which entities are affected through the use of named entity recognition and topic classifications. By developing a synthetic training dataset, a NER model and a topic classification model was developed and evaluated to achieve F1-scores of 71.4 and 61.8 respectively. These results were explained by a low variance in the synthetic dataset in comparison to a transcribed dataset from the real world which included anomalies not represented in the synthetic dataset. The aforementioned models were integrated into the dialogue framework Emora to seamlessly handle the back and forth communication and generating responses.

AUTHOR KEYWORDS

Natural Language Processing, Topic Classification, Named Entity Classification, NLP, NER, NERC

INTRODUCTION

Think of the scenario where a driver contacts an operator to ask for directions, instead the driver talks to a Natural Language Processing (NLP) model which responds to the inputs of the driver. The NLP-model could then convert speech to text, analyse why the driver made contact and then extract information from the conversation to appropriately give responses and instructions to the driver through speech. This thesis project will focus on analysing the dialogue in text format. The primary focus will be on two issues: 1) *topic classification*, for example, a request for clearance for parking and 2) *named entity recognition*, for example, identification of the vehicle and/or the location where the driver wishes to park.

During the past years, NLP systems have had substantial breakthroughs and there are many sophisticated chatbots for various sorts of dialogues [12]. One of those was Emora, which used NLP, topic classification, and ontology resources to produce dialogues when interacted with [1]. Emora used topic handlers and dialogue intent to identify what the intention of a conversation was, and to know how it should generate responses. This thesis project used Emora as a NLP-tool to recognize the topic and entities of the communication between a driver and an operator.

There have also been breakthroughs in the area of Intelligent Personal Assistants (IPAs) such as Google Assistant, Amazon's Alexa and Apple's Siri[13]. These IPAs handle topic classification and entity recognition in general dialogues when a user talks to them, which was identical to the functionality we sought to achieve in this project. The parts which separate the general IPAs and this project was to handle identification patterns which were combinations of words, letters and numbers as identifiers. As this article will research a particular area, entities and locations will be more specific than the general cases IPAs handle. For example, to identify a car from its registration plate, sample plate BAT 111, this can be formatted as B A T ONE ONE ONE or BAT HUNDRED ELEVEN etc. which still refers to the same registration plate and same entity.

Purpose

Today's communications between larger vehicles, such as cargo ships or planes, to operators are frequent and in between persons. There are operators who still use paper to write down important information about the vehicles they are responsible for. Air traffic controllers have been noted to use strips of paper for taking notes during dialogues with the pilot [2]. This information is not digital, and it takes unnecessary time for an operator to manually put in the systems. If the communication was used through a NLP model instead, the information could be used by the digital systems which help with safety functions and planning. The workload for air traffic controllers was significantly reduced when shifting from manual communication to a communication which was influenced by a NLP model [7]. The goal for the model of this thesis project is to be a step forward for the manual communication between an operator and a driver to be more automatic.

Research questions

By using NLP techniques on text dialogues between a driver and an operator, with what precision and F1-score can the following be predicted:

- the topic of the dialogue
- the entities in the dialogue

Delimitations

This thesis focuses on communication between sea captains and operators. The situation examined is ship traffic which travels in and out of a port where the ships have different objectives.

The dataset received is an audio dataset where parts of it will be manually transcribed. The transcriptions will be analyzed and then be used as the basis for generating a training dataset with similar dialogue structure as the transcribed dialogues. This is a small subset of the total audio dataset and may not give an accurate representation of the variance in the dataset.

The generated data is transcribed from the assumption that a future speech-to-text model will be able to transcribe equally as good as the generated dataset, and thereby can follow the results in this article.

THEORY

In this chapter the base is presented which will be needed to follow the rest of the work through the paper and the previous research done within and related to the methods which will be used.

Prerequisites

To handle the topic classification and named entity recognition of a dialog the first task is to handle the textual representation of a dialog. For the topic classification and named entity recognition, machine learning (ML) techniques were used. To enable ML-techniques the textual representation had to be converted to a numerical representation through tokenization. A simple tokenization would be to map all unique words in the sense that words are space separated, of a large dataset of dialogs to a sequence of integer numbers which in turn gives each word a numerical representation [3]. By iterating the words of a dialogue (or a sentence) it could be represented as a numerical vector with the conversion of the tokenization and thereby be fed into a neural network.

The concept of training a neural network is central to building an accurate NLP model. The training relates closely to iterating a training dataset together with a labeled dataset which gives the correct representation of what the model is meant to achieve. By making an own prediction of the sequence and then comparing the prediction to the label for the same sequence a loss function calculates the degree of correctness and then updates the model's weights. This continues over the training set until a predefined accuracy is reached or the training set ends[4]. This way of training a model is called supervised training. Another approach is unsupervised learning which instead analyzes the text sequence by context of a word. By using clustering-techniques can words which are used in similar contexts be classified together and can in turn be represented as the same classified entity class[5].

Emora

Emora is a dialogue handling framework which parses an input sentence and based on the contents of the sentence branches into a specified subtree of the conversation. By handling conversations this way it’s possible to create acyclic graphs and expand a conversation through multiple paths depending on the last inputs of the user[1].

Emora provides standard implementations to handle both NER and topic classification through an external framework.

SpaCy

SpaCy[1] is a NLP framework with various implementations to handle preprocessing and provide pre-trained models (on general text data) which can be used to be further trained in specified areas to increase performance of a model and decreasing the training data needed in comparison to a model which is trained from the standard initialized weights.

SpaCy provides tools for named entity recognition but have not been shown to be the best performing in the area, Stanford NLP have reported better results[16].

SpaCy also includes a model to handle topic/intent classification which has been shown to have good performance when initialized with pretrained weights[17].

The spaCy models for topic- and entity recognition were used in this project.

Named Entity Recognition

To extract information from unstructured data as text, structure must be created in some way. Part of speech (POS) tagging is a way to use previously annotated datasets which give grammatical information about words. As words can have different semantic meanings depending on the sequence of words this also becomes a ML task to assign the correct POS-tag to a word which in turn will be used for the named entity recognition[6].

The ML-algorithm which was used for this task is Named Entity Recognition (NER). The NER-algorithm has a variety of ways to learn entities but the most common is with supervised learning[5].

There are problems in handling the NER classifications, as words may come in many formats. For example, entities which are referred to with multiple words or models which have been trained on entities with capitalization which then occur in text without capitalization will be a lot harder to classify correctly[5, 6].

Topic Classification

The task of classifying the topic of a dialog is a multiclass prediction problem. The problem can be formulated as the model must know of the topics and be trained with supervised learning to be able to handle future data to predict the unseen data’s topic into one of the previously known topics. An important concept to bear in mind is that much of the meaning in text is context-based. The Long Short Term Memory (LSTM) ML-model is exceedingly good at taking in a data sequence and from that data compute a context based prediction, this will take into consideration the full sequence of words rather than the individual words in the dialogue[8,10]. There are contesting techniques such as Convolutional Neural Networks which have been shown to handle text classification tasks just as well or better as LSTM-models[14,15].

Emora Iris

Another research team has developed a topic and intent recognition model which they integrated into the Emora Chatbot and had prize winning performances with their implementation. They used a three step process with three classification models; a classifier which vectorized the current sentence and ran it through a CNN, an entity recognition model and an Amazon processor which annotated the inputs. The output of the three models were merged and classified to be incorporated in the final

[1] https://spacy.io/

prediction of the text. The team extended this architecture to make use of previous sentences to have contextual information when classifying topics. This research team won the Alexa prize 2018 [18].

METHOD

In this chapter the methods that were used in preprocessing, named entity recognition, and topic classification are explained. The dataset is also described as this was generated for the purpose of this article.

Dataset

Transcribing an audio dataset from an audio file can be done in many ways. One is using existing tools such as DeepSpeech [19], another is to manually transcribe the data by hand. The audio dataset was recorded in a test session where multiple operators acted as operators and drivers of cargo ships. The speakers in the test were primarily Swedish while speaking english which presented another problem, accents. There have been attempts in handling general accents but the results have not been very promising as the state-of-the-art models achieve around 60% accuracy in sequence to sequence transcriptions [21, 23].

The approach which was used is towards Natural Language Generation (NLG). The generation of data which was the most similar to the audio dataset the audio dialogues had to be thoroughly examined to find structural and grammatical structures which were repeating over multiple different dialogues. From this information it was possible to draw assumptions of future dialogue structures and create the possibility of generating a good dialogue dataset [20, 26]. The final dataset was generated rule-based and utilized randomizations. It was rule-based in the sense that specified word sequences were always in one order, and that the sequences have a randomized chance of being changed to form full sentences.

To get closer to the erratic nature of real dialogues, a noise generator was implemented. It had two options when iterating over the words of a dialogue;

a) *FULLY RANDOM:* This option happened in 5% of the words and replaced the word to another randomized word taken from the English Web Treebank[2]. This option simulated the possibility of the Speech to Text model to predict the wrong word and includes possibilities of the speaker saying a random word.

b) *ALIKE RANDOM:* This option happened in 10% of the words and generated a word which was grammatically very similar to the analyzed word. We choose to grade a similar word by finding words in the English Web Treebank with one character differentiation to the currently analyzed word.

Moving entities, locations and directions were also randomly generated for each dialogue. This was done by keeping these elements in separate arrays and randomizing the corresponding element to correctly form a sentence structure which matched the dialogues in the audio dataset.

The transcribed test dataset which was used to evaluate the final performance of the models was transcribed by hand from the audio files and the topics and entities were annotated by separately running them through previously used functions of the datagenerator which automatically labels entities and topics. This was done to ensure the same format of the transcribed test dataset and the generated dataset.

Preprocessing

The dataset which was used needed to be preprocessed before it could be used by the NLP-model. This was done by tokenization where all words of the sentences were converted to a numeric representation by creating a dictionary which could convert between the numerical value and the word. To further clean our data lemmatization was used, which is when the words are processed and reduced to their base word. This was solved by using stemming, which is when a word's common prefixes and suffixes are processed and removed to get the original word. To retrieve whether a word is an adjective or a verb, part-of-speech (POS) tagging was used. The POS tagger was called individually for all words in our dataset. To make the model further understand the meaning of the sentence, dependency parsing was used. Words in a sentence were run through an embedding layer to increase the dimension of the word and add weights which determined the vectorized version of the word. As the model was trained it updated the weights of the words to represent the meaning of the word and created a vector which could be compared to other word vectors and thereby find other words which are similar through distances in a vector space.

Named entity recognition

SpaCy's pretrained entity recognition model was used to detect entities in the dialogues. The dataset were preprocessed by the procedure described before. A document containing all of the custom entities which occurred was presented to the model to make it possible to predict custom entities. With the help of spaCy's built in functions it was trained to recognize these entities.

Before training the name entity recognition model, the dataset first needed to be labeled. Every entity in the dialogues which was created in our dataset was labeled by either being *text* (COM), *text* (IPL), or *text* (GPE). The custom categories of entities are communicating entities (COM), in port locations (IPL) and geopolitical entities(GPE). The entities were labeled by the startposition of the word and the endposition of the word within the dialogue.

Topic classification

SpaCy's pretrained text classification model was used to initialize the topic classification part of the model. The first step was for the data to be loaded and tokenized which was described before. When the data were preprocessed the next step was to generate a matrix which helped us to classify

[2] https://github.com/UniversalDependencies/UD_English-EWT

the text. The method to solve this problem is called bag of words, and it builds the matrix based on how many times the word has occurred in a document. This was done by using the tool scikits countvectorizer and feeding it our tokens. Another variable was the Term Frequency-Inverse Document Frequency (TF-IDF). TF-IDF is a way of knowing how important a word is in the document, and is calculated based on the total times it occurs in the given data context. By analyzing dialogues which have inbound or outbound vessels the TF-IDF method will categorize the keywords "inbound" and "outbound" as highly related to the associated category. When the model is built, and the data is preprocessed, the training of the model can start on the training dataset.

To train the topic classifier the dataset was modified to hold information regarding the labels required by adapting the available dataset module. The labels were added by looking at the contents of the third back-and-forth communication in the dialogue as this section was the first request by any part (operator or driver) in the communication. The added labels were:

Traffic information: This topic refers to dialogues which are purely information based rather than requests of how to proceed. This information is for example a position of the communicating vessel and replies regarding other vessels.

Inbound: Inbound refers to vessels which are on their way into the harbour.

Outbound: Outbound refers to vessels which are on their way out of the harbour.

Pilot required: This is the only label which can be combined with others. Pilot required refers to vessels which need more guidance to find a correct path to its final path or destination.

These labels describe the most common topics of dialogues based on the audio dataset. The "Pilot required" label could be combined with either the "outbound" or the "inbound" label as this indicated the need for further guidance of a physical person.

Training considerations

With the datagenerationmodule considerations about overfit and underfit models had to be made. We choose the approach of splitting the dataset into training, validation and a test dataset. By training the models on the training dataset and keeping the scores how well the model performs on the validation dataset we choose the instance of the model performing the best on the validation dataset instead of the training dataset. The chosen model has not been overfit on the training dataset and has been trained enough to not be underfit [29]. While training the models the performance was evaluated per batch with the F1-score of the model following the function:

$$\text{F1-score} = \frac{2 \times (Recall \quad \times Precision\ )}{Precision + Recall}$$

This is a metric which is used to calculate the most common F1-score which collects values from precision which is the number of correct predictions divided by the sum of correct predictions and wrong predictions, and the recall is a score that collects the correct predictions divided by the sum of the correct predictions and the number of missed predictions (as in there were entities which were not predicted).

The F1-score was used as the metric as there were label imbalances within the dataset to give an accurate representation of the performance as it gives indications of how correctly the model predicts entities and topics but also how often it forgets to predict a topic or entity [30].

Emora

The Emora chatbot uses a wide variety of NLP tools to handle sentence generation and to control the dialogue structure. To keep a context of previously mentioned entities or topics, the dialogue manager within Emora has an integrated natural language understanding (NLU) model which is used when generating responses from the Emora chatbot [1].

Emora had opened up the possibility to integrate pretrained NLP-models into the NLU module by defining custom functions which were used to implement the developed entity- and topic classification models.

RESULTS

This chapter goes into the results of the different models and how they unite within the Emora chatbot to fully combine into a conversational engine which knows about entities and topics of conversations.

Dataset

The generated data consisted of a variation of entities, locations, directions and a limited range of filler concepts to build a sentence structure which were similar to the dialogue data from the audio dataset.

An example dialogue looks like this where U is the user input and O denotes the operator response:

> O: VTS Ostergotland Jasmin Øresund
>
> U: Jasmin Øresund VTS Ostergotland
>
> O: I see you passed SAAB You rave Benny inbound heading to south . Moon inbound towards south . Naranja inbound for north soft Betut inbound via northwest.
>
> U: 3 vessels inbound understood .

In this example there was one alike noise added which changed "have" to "rave" and a fully random word where "soft" now stands. The resulting dialogue still holds the structure of the audio dataset yet keeps room for misinterpretation and wrong predictions by an automated speech to text model.

2500 dialogues were generated to obtain an amount of dialogues with enough differences in sentence structures and entities. A smaller dataset would have had risks of too few samples of each label class while a larger dataset could introduce a lot of similar examples which easily lead to overfitting.

The transcribed evaluation dataset was ultimately 37 dialogues which held differing dialogue structures than the generated dataset. This was important to note as the evaluation of this dataset gave an indication of how well the models generalize. The transcribed dataset included dialogues which repeated identifications, missayings which were corrected in the same sentence and other random filler words than previously used in the generated dataset.

The transcribed dataset was divided into two datasets where one was referred to as a *validation dataset* and the other as the *test dataset*. The validation dataset comprised 25% of the data while the test dataset held the remaining 75%.

NER classification

A baseline model was imported and created from Spacy, which was tested with the training and validation dataset. The untrained baseline had an accuracy of approximately 4%.

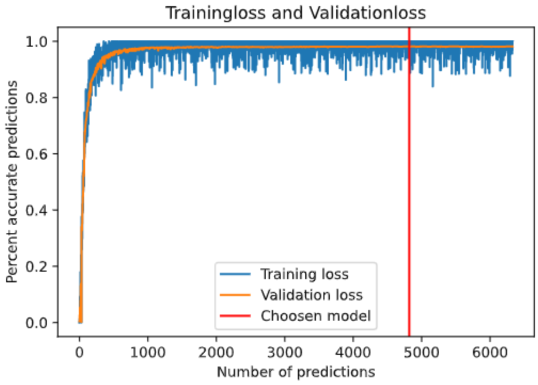


Figure 1. Training- and validation loss by the NER-model when training and validating on the generated dataset. The vertical red line shows the best performing model.

When training the model it got a F1-score of approximately on the generated validation dataset 98.22%. The chosen model which is marked by the vertical red line was used to measure F1-score on the generated test dataset and achieved a score of approximately 98.94%.

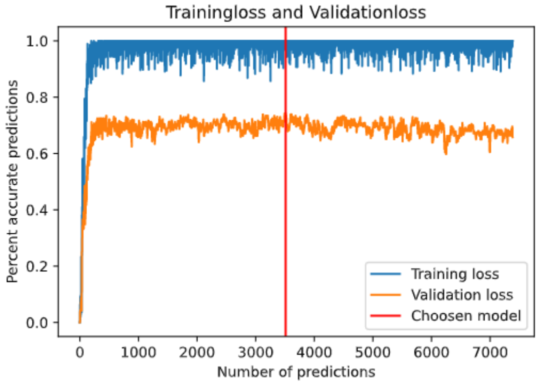


Figure 2. Training- and validation loss by the NER-model when training on the generated dataset and using a subset of the transcribed data as validation dataset. The vertical red line shows the best performing model.

Moving onto the transcribed dataset, this instance was trained together with the transcribed validation dataset. As can be seen in Figure 2 the model found an optimal performance closer to the start than the model training on generated data (Figure 1) and with a peak at approximately 73.88 F1-score on the transcribed validation dataset. This model was used to predict the rest of the transcribed dataset and achieved a F1-score of approximately 71.38 and with a precision of 70.91.

Topic classification

Following a similar approach as the NER classification first a model was trained and evaluated on the generated dataset. The best performing model on the generated validation dataset can be found in Figure 3.

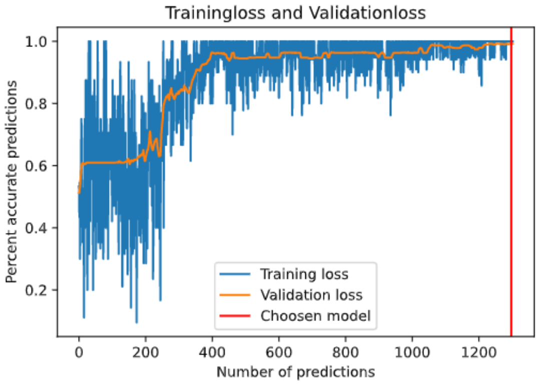


Figure 3. Training- and validation loss by the Topic-model when training and validating on the generated dataset. The vertical blue line shows the best performing model.

The topic classification performed a 99.19 F1-score on the generated validation dataset. This model was also used to evaluate F1-score on the generated test dataset where it achieved a score of 99.33.

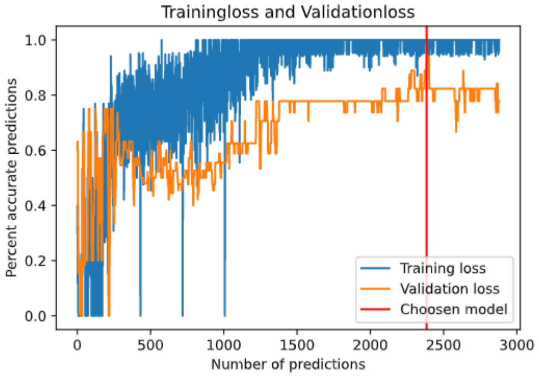


Figure 4. Training- and validation loss by the Topic-model when training on the generated dataset and using a subset of the transcribed data as validation dataset. The vertical red line shows the best performing model.

Using the transcribed dataset, which is visualized in Figure 4, the model finds a peak optimum at 94.11 F1-score together with the transcribed validation dataset. This model was then used to predict the transcribed test dataset and achieved a F1-score of 61.81 and a precision of 60.71.

Emora

The presented models where integrated into the Emora chatbot within the NLU-module and by feeding this dialogue structure it found the correct entities and topic of the incoming communication:

> U: Rosida VTS Ostergotland.
>
> O: vts ostergotland rosida
>
> U: Anchor away outbound philipines to northeast and towards northwest at fina fyren pilot onboard.
>
> O: There is Gokker anchor away outbound through southwest and Jaws Ms anchor away outbound for Lysekil. outbound vessel Momentum was fina fyren.
>
> U: 2 vessels nearby voted .
>
> O: thank you vts ostergotland
>
> Found entities within the user input: [Rosida, VTS Ostergotland, fina fyren]
>
> Found topic from the dialogue: outbound

This presents a generated dialogue which can be fully automated and give the operator information regarding what entity is doing the communication and with which intent and thereby can prioritize the communication accordingly.

DISCUSSION

In this chapter we delve deeper into a discussion regarding the different modules of the article.

Method

Dataset: The dataset was randomized but was still limited to which words and sequences which were an option to build the sentences. By further developing this part there is a greater opportunity to make the model better at handling more general cases as this became a problem when we ran the models on the transcribed test dataset. The dialogues in the transcribed dataset held larger differences in the dialogue structures than anticipated such as repeatings of identifications, missayings and many filler words such as “over” as ending of a communication or questions which were not part of the generated dataset. Another reason the dataset was not as varied as the audio data is related to the randomization of the sentence structures where the sentences were built from previously created sequences of words which then were randomized into different sequences. Within the prepared sequences there was close to no variance (except from the possibility of the noise generator changing words) which in turn gives many sentences very similar sentence structures.

The dataset was generated by controlling about 20 dialogues in the audio data which should have been more. In hindsight the full transcribed dataset should have been created before starting the development of the data generation module to include more variance in the sentence structures. This was a time consuming task and we finished 37 dialogues which were not within the original scheduling. Because of the original planning, transcribing the audio data could not be prioritized.

Further development of the dataset could include incorporating an NLG model which generates the dialogue structure after training on similar data to the audio dataset. This could reduce the similarity which can be found in the small filler words or different structures of sentences which does not change the meaning of the sentence. Including this into the dataset would most likely make the model more robust in predicting dialogues which were previously unseen.

See the appendix section for a complete rundown of the distribution of classes in the NER data and the topic classification data.

Topic classification: The labels which were used for topic classification should have been confirmed and iterated together with experts and operators to find the most relevant labels. Unfortunately the chosen labels were based on the authors knowledge of the audio dataset which was not known to be of highest relevance. Also by extending the possible labels of topics it becomes exponentially more difficult for the model to make correct predictions as multiple labels are allowed. This means that the model would need a larger dataset to train on and the distribution of the labeled training data has to be relatively evenly distributed within the example dialogues to not create a biased model.

NER classification: The spaCy model did not comply with overlapping entities which could appear in dialogues, an example would be when an entity “Jasmin Øresund” could respond “Jasmin” as short. This was problematic to handle within the spaCy implementation. This is a problem with the use of beginning-inside-outside tagging which marks beginnings and ends of entities and would require enabling multiple tags for a word to make it possible to include overlapping entities.

To prevent overfitting or underfitting the models to the generated dataset we choose to complete the training sequence and keep track of the best evaluation score on the validation dataset through continuous evaluations after each trained batch. This can be seen in the graphs in the result chapter. It was not the most efficient way when experimenting with the transcribed dataset as it overfitted early in the training and could have been stopped but it did find the best performing model of the training iteration.

Emora: The integration of models into the emora chatbot framework through the NLU-module was very smooth as the developers of the framework have made it very simple to use custom functions.

Results

Dataset: The resulting dataset which was generated held to low variance in comparison to the transcribed dataset, this led to a lower performance which was not comparable to the performances on the generated dataset. This was problematic as the purpose of the generated dataset was to generalize better towards the transcribed dataset and be used as a training extension rather than another alike dataset which it became.

The 37 dialogues which were transcribed from the audio dataset were transcribed to the best of ability of the transcribers. Since there is no speech-to-text (STT) model which can transcribe 100% correctly the result of the transcription was equivalent to a STT model which transcribes english speech without accent.

Topic classification: The F1-score on the generated validation data was alarmingly high which could be explained by the structure of the generated data which decides the topic was limited to one section of the dialogue. While the topic label information can be found in the same structure within the audio data it is common to find more variance within the sentence structure of that sequence than within the generated training dataset.

The performance on the transcribed validation dataset was expected and can be explained due to the small amount of samples in the validation dataset. Another factor could have been the limited number of labels which were included as possible predictions. As the F1-score takes into consideration true positives, false positives and false negatives having a topic classifier which is biased towards predicting one label, predicting one label wrong gives a double wrong. As the imbalance of classes within the dataset in both the training data and test data the model was biased towards predicting "traffic information" on a majority of the dialogues. See appendix 12 for further detailed information of the class specific results of the topic classifier.

The final F1-score on the test dataset was far below the state-of-the-art performances which are close performance at around 90 F1-scores.

NER classification: The high F1-score of predictions made on the generated dataset can be explained by the nature of the dialogue structure. As the generated dataset holds a strict dialogue structure where entities often appear in the same places within the dialogue the model will learn this a lot faster and more accurately compared to other NER-models which extracts entities from free text or non structured dialogues. As the F1-score holds about the same results on both the validation and the test dataset it raises a few red flags that the generated data samples are very alike and comparing the generated data to the transcribed data shows larger differences than within the generated dataset.

The drawdown from the generated dataset to the transcribed dataset could be explained by the new dialogue structures which are introduced in the transcribed data and the more erratic nature of human language than anticipated. This performance is worse than state-of-the-art performances of NER models. The developed model was trained on new entities which were not well known and followed a wide variance of structure such as one up to four word entities which made the dataset different to the datasets used for classic NER predictions. There was a low number of entities which occured in the training dataset and then appeared within the transcribed dataset. As the model then did its predictions based on the structures of the named entities rather than knowing the entities the performance was over expectations. Keeping a larger overlap of entities within the test and validation dataset to the training dataset would most likely have a big impact on the performance of the NER model.

While investigating the individual classes of the NER-model it could be concluded that the model had problems in finding "in port locations" which could be caused by the similarity to geopolitical entities as both will be referred to and spoken of in similar ways. As previously mentioned, an improvement to be made is also to have a greater overlap of "communication entities" in the training dataset to the test dataset. This should be possible to implement in the future as operators know before the day begins which ships are planned to come into the area. For more detailed information see appendix 11.

In the topic classification model, when trained together with the transcribed validation data (Figure 2), it's clear to see that the model is overfitting to the training dataset as the optimal model configuration is found early in the training session and the validation performance gets lower with further training.

Emora: The results presented with the Emora framework with the integrated models gives a preview of what the project can result in with further development of the prediction models. This was the key part of the project as the next step in development is to implement entity linking which can assign the location to the entity and thereby be able to track the locations of the moving entities by locations or in the future even by coordinates.

Ethical implications

In the future where this system could be continually developed and integrated into larger systems and possibly more autonomous systems there are ethical concerns. One angle is in prioritizing which communication is the most important and how the information is presented to the operator, another is more critical information where some entities may put on queue for longer due to biases within the AI model [24].

Another consideration in automating specific tasks is the time saved can be used by an operator elsewhere, by this assumption the time requirement for each operator goes down and it is less time needed, if any, and therefore less operators may be needed [25].

Another positive consideration is the environmental impact which an automated model can optimize towards when communicating with a cargo ship. The fuel usage formula is often described as a cubic function of the speed of the ship and by having more continually dialogues with ships and giving them optimal paths and predicted arrivals into the port the speed can be optimized and fuel usage minimized [27,28].

Risk Factors To Validity of Results

The way the dataset is generated is heavily biased towards the transcriptions of the audio files and also biased towards how well the transcriptions are made. If there are a lot of filler sounds or words which would be predicted to other words rather than the transcribed ones it poses a problem where the dataset is too different from the real data which in the future can be fed into the model [22]. These problems were partly handled by using tools to find filler words which could be randomized into the sentences (so that these were not chosen) and also by fully randomizing words from the English Web Treebank.

There may also be unintentional help while transcribing as is a bias towards presenting a successful study. Unfortunately this issue could not be handled as it is secrecy on the audio files which prevents 3rd party persons from doing transcriptions which could have helped with this issue.

A major threat which this article relies on is the future development of a speech to text model which can transcribe english while spoken with a heavy swedish accent and correctly predict swedish entities such as locations and vessel names.

CONCLUSIONS

Given a model which can handle the Swedish accent to convert STT this approach is promising. Given this STT-model it would be easy to retrieve a large dataset for training on real dialogues rather than generated data. Training the spaCy models and following the format of this study we believe a future study could find promising results.

Another future approach could be to keep on developing the training dataset as improvements there will make the training more relevant to generalize better towards the real use case. This should be done incorporating NGL models to closely mimic human language and find new sentence structures which might show up in the audio data.

The models did not achieve the high performances we had hoped for, yet it provides a benchmark of what performance can be achieved from a purely synthetic dataset.

The developed models achieved precision scores of 70 for the NER model and 60 for the topic classification model and F1-scores at 61.8 and 71.4 which is a long way below state-of-the-art models, yet the performance can be explained by the differences between the generated data and the transcribed.

The Emora chatbot implementation has been investigated to handle the dialogue states and implemented entity- and topic classification models. The dialogue handling was handled as expected and integration with the pretrained models could be done in a short time.

REFERENCES

[12] Adamopoulou E, Moussiades L. *An Overview of Chatbot Technology*. Artificial Intelligence Applications and Innovations. 2020;584:373-383. Published 2020 May 6. doi:10.1007/978-3-030-49186-4_31

[6] Alan Ritter, Sam Clark, Mausam and Oren Etzioni. July 2011. *Named Entity Recognition in Tweets: An Experimental Study.* University of Washington, WA.

[18] Ali Ahmadvand, Ingyu (Jason) Choi, Harshita Sahijwani, Justus Schmidt, Mingyang Sun, Sergey Volokhin, Zihao Wang, Eugene Agichtein, *Emory IrisBot: An Open-Domain Conversational Bot for Personalized Information Access*, 2018, 1st Proceedings of Alexa Prize (Alexa Prize 2018).

[17] Anda Stoica, Tibor Kadar, Camelia Lemnaru, Rodica Potolea, Mihaela Dînşoreanu, *The Impact of Data Challenges on Intent Detection and Slot Filling for the Home Assistant Scenario*, 2019 IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP), Cluj-Napoca, Romania, 2019, pp. 41-47, doi: 10.1109/ICCP48234.2019.8959642.

[3] Chunyu Kit, Jonathan J. Webster. 1992. *Tokenization as the initial phase in NLP.* Proceedings of the 14th conference on Computational linguistics -, Vol. 4.

[25] Dahlin, E. (2019). *Are Robots Stealing Our Jobs?* Socius. https://doi.org/10.1177/2378023119846249

[26] W. H. Deason, D. B. Brown, K. -. Chang and J. H. Cross, "*A rule-based software test data generator,*" in IEEE Transactions on Knowledge and Data Engineering, vol. 3, no. 1, pp. 108-117, March 1991, doi: 10.1109/69.75894.

[28] Fagerholt, K., Laporte, G. & Norstad, I. *Reducing fuel emissions by optimizing speed on shipping routes*. J Oper Res Soc 61, 523–529 (2010). https://doi.org/10.1057/jors.2009.77

[1] Finch, S. E.. 2020 *Emora: An Inquisitive Social Chatbot Who Cares For You*. Published in 3rd Proceedings of Alexa Prize

[22] Geva, Mor, Y. Goldberg and Jonathan Berant. "*Are We Modeling the Task or the Annotator? An Investigation of Annotator Bias in Natural Language Understanding Datasets.*" ArXiv abs/1908.07898 (2019): n. pag.

[8] Giovanni Di Gennaro, Amedeo Buonanno, Antonio Di Girolamo, Armando Ospedale, Francesco A.N. Palmieri. Jan 2020. *Intent Classification in Question-Answering Using LSTM Architectures*. Universit´a degli Studi della Campania "Luigi Vanvitelli". Roma 29, Aversa (CE), Italy.

[19] Hannun, Awni Y., Carl Case, J. Casper, Bryan Catanzaro, G. Diamos, Erich Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates and A. Ng. *"Deep Speech: Scaling up end-to-end speech recognition.*" ArXiv abs/1412.5567 (2014): n. pag.

[2] H. Helmke, O. Ohneiser, T. Mühlhausen and M. Wies, *Reducing controller workload with automatic speech recognition.* 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), Sacramento, CA, 2016, pp. 1-10, doi: 10.1109/DASC.2016.7778024.

[10] Jenset GB, McGillivray B. *Enhancing Domain-Specific Supervised Natural Language Intent Classification with a Top-Down Selective Ensemble Model. Machine Learning and Knowledge Extraction*. 2019; 1(2):630-640.

[27] James J. Corbett, Haifeng Wang, James J. Winebrake,

The effectiveness and costs of speed reductions on emissions from international shipping, Transportation Research Part D: Transport and Environment, Volume 14, Issue 8, 2009, Pages 593-598, ISSN 1361-9209, https://doi.org/10.1016/j.trd.2009.08.005.

[23] Kardava, I., Jemal Antidze and Nana Gulua. "*Solving the Problem of the Accents for Speech Recognition Systems.*" (2016).

[4] Karol Grzegorczyk. 2019. *Vector representations of text data in deep learning,* Ph.D Dissertation. AGH University of Science and Technology, Kraków.

[21] Kitashov, Fedor, Elizaveta Svitanko and Debo Dutta. "*Foreign English Accent Adjustment by Learning Phonetic Patterns.*" ArXiv abs/1807.03625 (2018): n. pag.

[7] Kleinert M. et al. (2019) *Adaptation of Assistant Based Speech Recognition to New Domains and Its Acceptance by Air Traffic Controllers.* In: Karwowski W., Ahram T. (eds) Intelligent Human Systems Integration 2019. IHSI 2019. Advances in Intelligent Systems and Computing, vol 903. Springer, Cham. https://doi.org/10.1007/978-3-030-11051-2_125

[29] Koehrsen, W. Overfitting vs. Underfitting: *A Conceptual Explanation*. Jan. 2018. URL:

https://towardsdatascience.com/overfitting-vsunderfitting-a-conceptual-explanation-d94ee20ca7f9.

[5] David Nadeau, Satoshi Sekine. 2007.*A survey of named entity recognition and classification*. Lingvisticæ Investigationes 30

[13] Oktay Bahcec, *Analysis and Comparison of Intelligent Personal Assistants,* 2016, Kungliga Tekniska Högskolan, Stockholm

[20] REITER, E., & DALE, R. (1997). *Building applied natural language generation systems*. Natural Language Engineering, 3(1), 57-87. doi:10.1017/S1351324997001502

[14] Shaojie Bai, J. Zico Kolter, Vladlen Koltun., *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*, arXiv e-prints, 2018.

[24] K. Shahriari and M. Shahriari, "*IEEE standard review — Ethically aligned design: A vision for prioritizing human wellbeing with artificial intelligence and autonomous systems,*" 2017 IEEE Canada International Humanitarian Technology Conference (IHTC), 2017, pp. 197-201, doi: 10.1109/IHTC.2017.8058187.

[16] Xavier Schmitt, Sylvain Kubler, Jérémy Robert, Mike Papadakis, Yves LeTraon, *A Replicable Comparison Study of NER Software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate,* 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS),

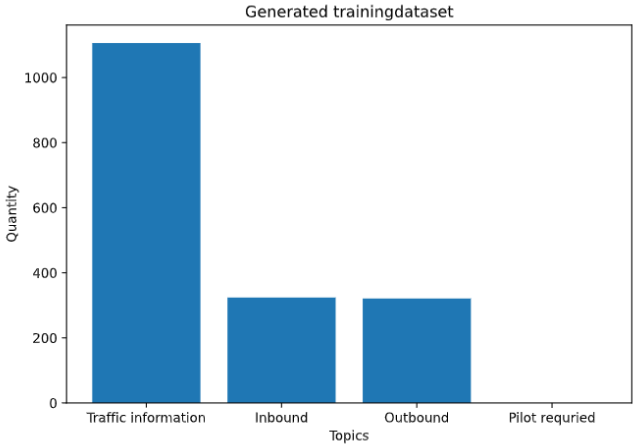
Granada, Spain, 2019, pp. 338-343, doi: 10.1109/SNAMS.2019.8931850.

[15] Xiang Zhang, Junbo Zhao, Yann LeCun, *Character-level Convolutional Networks for Text Classification*, arXiv e-prints, 2015.

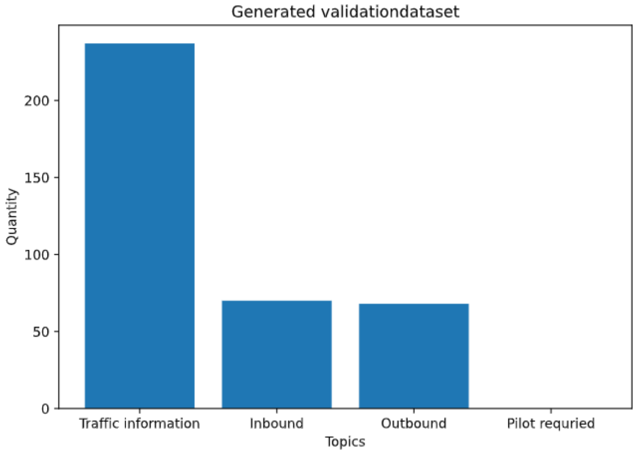
[30] Yutaka Sasaki, 2007, *The truth of the F-measure*, School of Computer Science, University of Manchester

APPENDIX

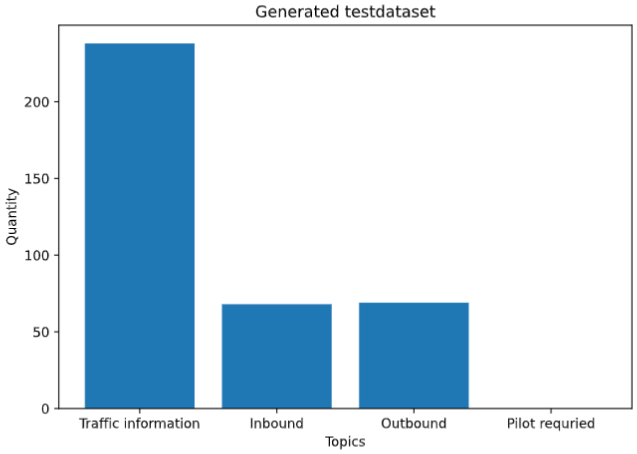
Topic Classification data distribution



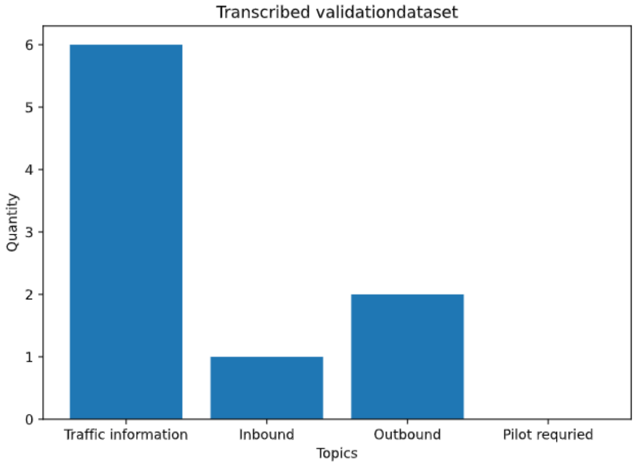
Appendix 1. The data distribution of the generated training dataset.



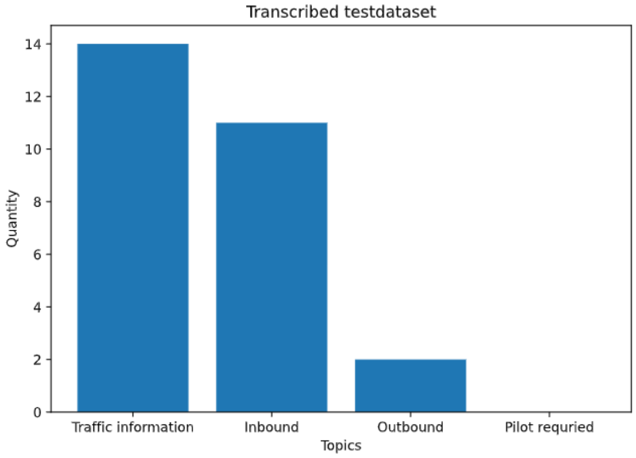
Appendix 2. The data distribution of the generated validation dataset.



Appendix 3. The data distribution of the generated test dataset.

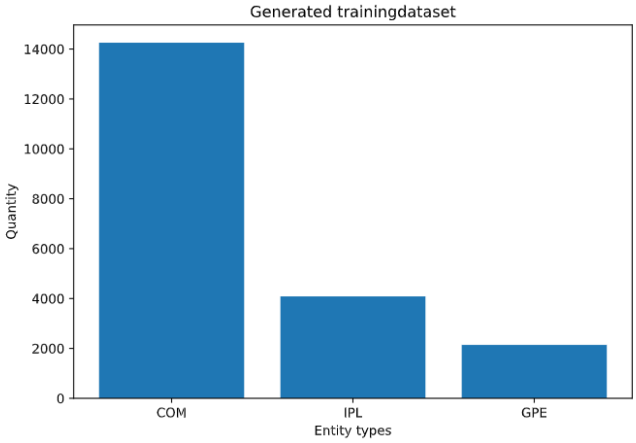


Appendix 4. The data distribution of the transcribed validation dataset.

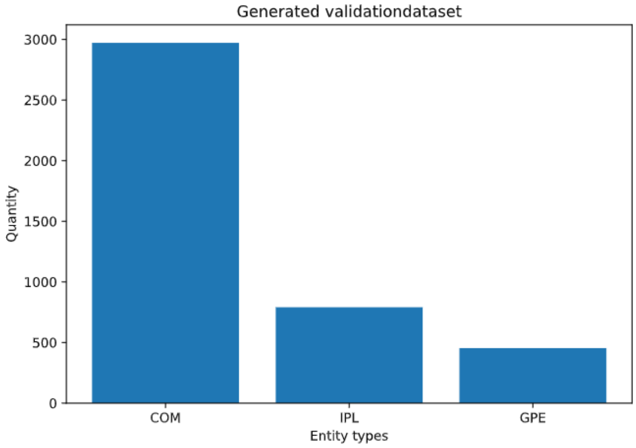


Appendix 5. The data distribution of the transcribed test dataset.

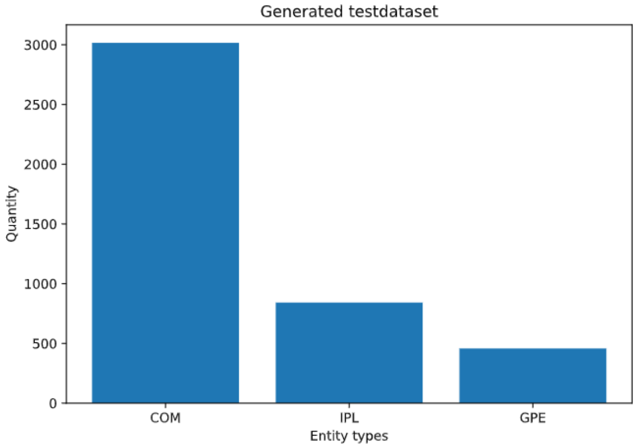
NER Classification datadistribution



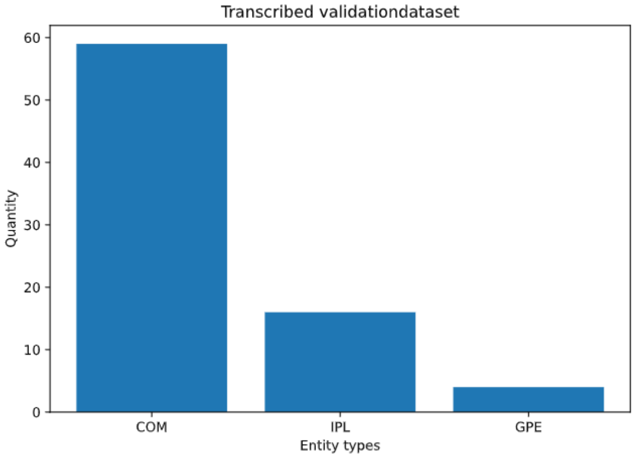
Appendix 6. The data distribution of the generated training dataset.



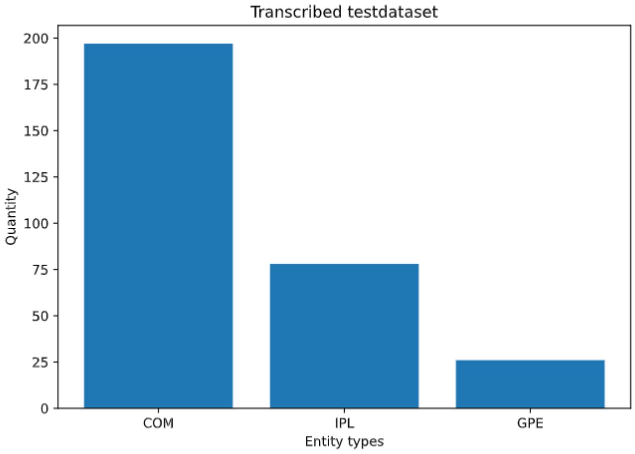
Appendix 7. The data distribution of the generated validation dataset.



Appendix 8. The data distribution of the generated test dataset.

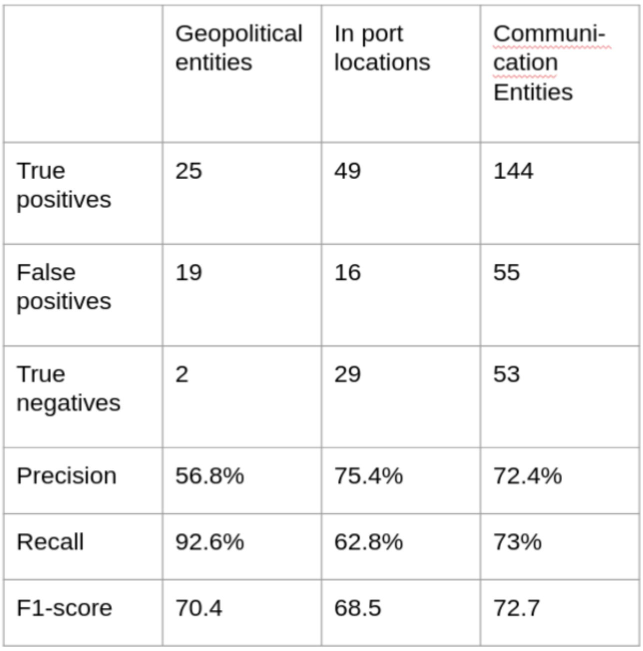


Appendix 9. The data distribution of the transcribed validation dataset.



Appendix 10. The data distribution of the transcribed test dataset.

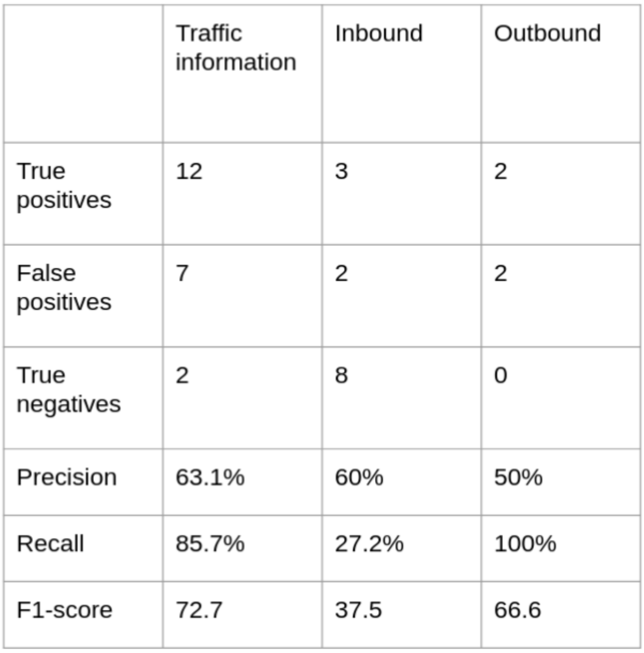
More precise data of NER result



	Geopolitical entities	In port locations	Communi-cation Entities
True positives	25	49	144
False positives	19	16	55
True negatives	2	29	53
Precision	56.8%	75.4%	72.4%
Recall	92.6%	62.8%	73%
F1-score	70.4	68.5	72.7

Appendix 11.Table with the results of the NER-prediction model.

More precise data of Topic classification result



	Traffic information	Inbound	Outbound
True positives	12	3	2
False positives	7	2	2
True negatives	2	8	0
Precision	63.1%	60%	50%
Recall	85.7%	27.2%	100%
F1-score	72.7	37.5	66.6

Appendix 12.Table with the results of the topic classification model.