Forklift Truck Activity Recognition from CAN Data

Kunru Chen[1,2(✉)], Sepideh Pashami[1,2], Sławomir Nowaczyk[1,2], Emilia Johansson[1,2], Gustav Sternelöv[1,2], and Thorsteinn Rögnvaldsson[1,2]

[1] Center for Applied Intelligent Systems Research, Halmstad University, Halmstad, Sweden
kunru.chen@hh.se
[2] Toyota Material Handling Europe, Mjölby, Sweden

Abstract. Machine activity recognition is important for accurately estimating machine productivity and machine maintenance needs. In this paper, we present ongoing work on how to recognize activities of forklift trucks from on-board data streaming on the controller area network. We show that such recognition works across different sites. We first demonstrate the baseline classification performance of a Random Forest that uses 14 signals over 20 time steps, for a 280-dimensional input. Next, we show how a deep neural network can learn low-dimensional representations that, with fine-tuning, achieve comparable accuracy. The proposed representation achieves machine activity recognition. Also, it visualizes the forklift operation over time and illustrates the relationships across different activities.

Keywords: Machine Activity Recognition · Learning representation · Autoencoder · Forklift truck · CAN signals · Unsupervised learning

1 Introduction

In recent years, a field of study has emerged as Machine Activity Recognition (MAR), i.e. the study of how to label machine activities from data streams (video, audio, or other types of data). MAR enables monitoring machine productivity, understanding customer needs better, and designing improved maintenance schemes. So far, work on MAR has almost exclusively been applied to construction equipment. A recent overview of the field is provided by Sherafat et al. [7], who categorize the approaches into kinematic methods, computer vision based methods, and audio based methods. The first two are the most common approaches. The approaches tend to be based on external sensors placed on (or nearby) the machine for the sole purpose of activity recognition. It is rare to use the streaming on-board data on the controller area network (CAN), which is multidimensional and consist of control and sensor signals to and from different parts of the equipment. We are only aware of some early works by Vachkov et al. [8,9], who used CAN data and different variants of self-organizing maps for

J. Gama et al. (Eds.): ITEM 2020/IoT Streams 2020, CCIS 1325, pp. 119–126, 2020.
https://doi.org/10.1007/978-3-030-66770-2_9

this. There have been important developments in the last decade, both regarding the volume of on-board data and the capacity of machine learning algorithms. Thus, it is worthwhile to explore how well MAR can be done using more recent machine learning techniques and CAN data.

In this paper we present ongoing work on MAR for forklift trucks, a type of equipment that is widely used in the industry, but rare in MAR research. Forklift trucks are considered exceptionally challenging, even "unrecognizable", due to few "articulated moving parts" [7]. We show that the situation is not poor when one uses multidimensional CAN data.

A contribution of this work concerns learning nonlinear representations from unlabeled data. Representation learning [1] is a way to make use of large unlabeled data sets to visualize relations, and improve classification performance. This can lead to deep learning classifiers that significantly outperform shallow classifiers. We have used autoencoders to learn the CAN data embedding, and then fine-tune it with supervised learning, in order to visualize the forklift operations. We are not aware of any work, prior to our study, that has approached this visualization question for MAR.

2 Definition of the Task

The task is to recognize the activities of forklift trucks. Forklift experts defined expected actions and grouped them into six levels of detail. The top level includes engine off and on. The next level contains 1) engine off, 2) engine on but forklift idle, and 3) engine on and forklift working. This activity breakdown was continued to level six. The complexity in the activities depend on the number and order of signals, driver behavior, and time. The same activity can vary in duration. The top activity levels can be recognized with rules and a single signal; the actions in levels 4–6 are much more complex. The results in this paper are reported for the fifth level, with five different activities, excluding engine off.

3 The Data

The forklift trucks used in this study were reach forklifts with 1.6-ton load capacities and 10-meter maximum lift height. Data collection was made with a Vector GL1000 compact Logger and signals from two CAN buses were saved. The original frequency of the CAN bus data is 50 Hz, but the data was extracted with a frequency of 10 Hz. Data were collected from two different warehouse sites, one in Sweden and one in Norway. During the data collection at the Swedish site, the operators' hand actions were filmed for later activity labeling.

Two data sets were collected with the same driver at the Swedish site: one with 58 min and one with 27 min. The long represented normal operation, including picking up orders and waiting, whereas the short focused on load handling operations. Another data set collected at the Norway site spanned two weeks of normal operation, with different drivers. However, no labels could be created for forklift activities, due to the lack of videos.

Each second of the videos from the Swedish site was labeled manually by an expert from watching the operators' actions. The fifth level activity recognition has five categories. The recognition of each activity was evaluated with one-against-all, presenting data imbalance. In the 58-min data set, the class proportions are 35.6%, 24.3%, 15.8%, 15.1%, and 9.1%, corresponding to the five labels *other, drive without load, take load, drive with load*, and *leave load*. In the 27-min data set, the corresponding proportions are 2.5%, 31.1%, 21.8%, 23.1%, and 21.6%.

The collected data contained more than 260 signals, of which 14 were selected by the experts as especially relevant for recognizing the activities. These 14 signals were also available in the data from the Norwegian site. These 14 signals contain information about the fork adjustment functions, the fork height, the weight on the fork, the speed and the heading of the vehicle.

4 Methods

The goal is to recognize the activities from these 14 signals. Our idea is to use time window snapshots by the size of two seconds to define the activities. Two seconds correspond to 20 time samples, so the total size of each pattern was 280 (14 signals $\times$ 20 time steps). The activity label for each pattern was determined by the label for the last second in the time window snapshot. The overlap between sliding windows is 50% (10 time steps). Random forests (RF) [3] was applied as the baseline classifier model.

The choice of a two-second time window was made after plotting the data with t-Distributed Stochastic Neighbor Embedding [5] with different time window lengths. Data corresponding to different activities became more separated when increasing the window size, which is reasonable since data points should become more and more unique as the time window is increased. However, there is a trade-off where a large window can give more information about the corresponding activity while can also increase the difficulty in finding the patterns of

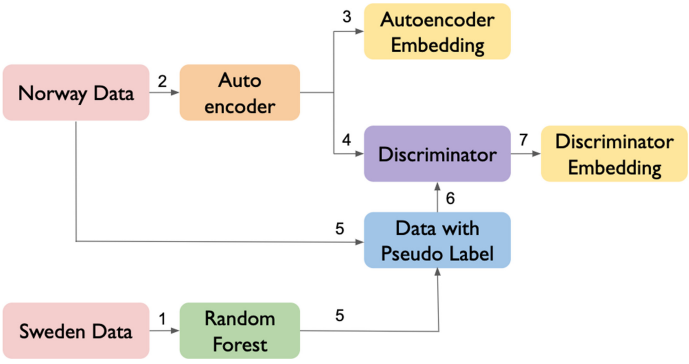


Fig. 1. Flow chart for forklift truck activity recognition.

the data in the window, i.e., more than one activities can be captured in a single window.

Figure 1 shows a flow chart of the method, divided into seven steps: (1) training a baseline RF with the Sweden (labeled) data sets, (2) training an autoencoder [4] with the Norway (unlabeled) data, (3) extracting the output of the bottleneck layer of the trained autoencoder, this embedding is named *autoencoder embedding*, (4) transforming the autoencoder into a discriminator by removing the decoder and adding one layer as the output layer connected to the bottleneck layer, (5) generating pseudo-labels on the Norway data by using the baseline RF trained at step 1, (6) training the discriminator using the data with pseudo-labels from step 5, (7) extracting the output of the last hidden layer of the discriminator, which is named *discriminator embedding*.

In this case, overly optimistic values can be given by reporting the result using, e.g., the area under the receiver-operating curve (AUC), or the accuracy. We therefore report the recognition results using two measures that are more suitable with imbalanced data: the balanced accuracy (BA) and the area under the precision-recall curve (APRC) [6]. The BA is the average of the true positive rate (TPR) and the true negative rate (TNR). The precision-recall curve describes the trade-off between precision and recall, which are measurements focusing on the true positive (TP), hense it is more appropriate for imbalanced data [2]. The APRC has similar characteristics as the AUC: it is a score across different probability thresholds, and it needs to be compared with the lower bound from a random classifier. For a classifier that makes random decisions, its AUC score will be 0.5 while its APRC value will be the ratio of positive samples over all samples. In the result section, for each activity, we report the APRC value achieved with a random classifier as "Random APRC".

5 Results

Throughout the experiments in this paper, the 27-min labeled data set was used as a hold-out test set. The 58-min labeled data set was used for supervised training. All hyper-parameter selection was done with 10-fold cross validation on the training set. Additionally, for the results from deep neural networks, the loss into training and validation was checked to confirm that overfitting did not happen.

5.1 Baseline Classifier

The best RF model was selected by doing a randomized grid search on three significant hyper-parameters. The final RF model was generated with 60 trees, a maximum depth of 11, and maximum 2 features to consider while looking for the best split. The baseline classifier's results on the test set are shown in Table 1.

Table 1. One-against-all recognition performance of baseline RF classifier (BA is Balanced Accuracy, and APRC is Area under the Precision-Recall Curve). For comparison, the performance of a random classifier is shown in the bottom row.

Measure	Other	Drive w/out load	Drive w load	Take load	Leave load
BA	0.858 ± 0.022	0.811 ± 0.026	0.715 ± 0.026	0.684 ± 0.042	0.601 ± 0.078
APRC	0.661 ± 0.380	0.602 ± 0.156	0.535 ± 0.032	0.546 ± 0.036	0.636 ± 0.042
"Random" APRC	0.025	0.311	0.218	0.231	0.216

5.2 Classifiers Trained from Unlabeled Data

The experiment processed to train classifiers from unlabeled data according to Sect. 4. Autoencoders of different sizes (breadth and depth) were trained to obtain a stable signal reconstruction performance. The results reported in this paper use an encoder with the 280 dimensional input and three layers connecting the input layer and the bottleneck layer, with 128, 64, and 32 units, respectively. The decoder had a symmetric structure with three layers. The total autoencoder architecture can be described as $280-128-64-32-N-32-64-128-280$, where N denotes the number of bottleneck units. The activation function in the bottleneck layer is linear while the other layers have ReLu units. Each autoencoder was trained with backpropagation and early stopping.

The first findings were that an autoencoder trained on the unlabeled data set was efficient also for encoding the labeled data sets, and the reconstruction of the signals improved as the number of bottleneck units increased, see the left panel in Fig. 2. This showed that data from one site could be used to learn a representation applicable to other sites. It also showed that the data is so complex that it requires a high-dimensional manifold to be represented accurately.

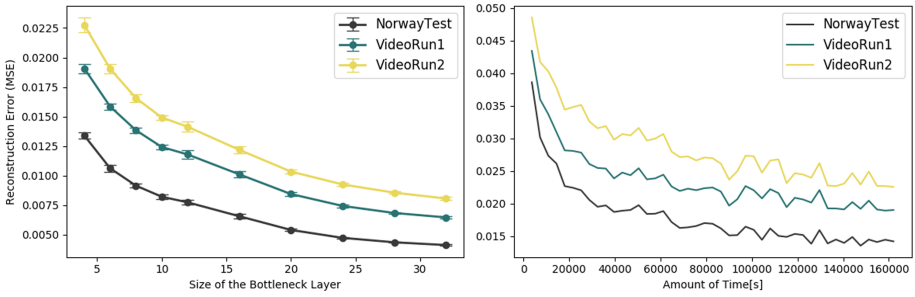


Fig. 2. The left panel shows the reconstruction error (MSE) when changing the size of the bottleneck layer, for all three data sets. The right panel shows how the reconstruction error decreased when the amount of data used for training increased. In both cases the autoencoders were trained only on the large unlabeled data set.

Another finding was that having more data is helpful in learning better representations for reconstructing the signal. The right panel in Fig. 2 shows how the reconstruction performance improved as adding the training set size.

Table 2. Balanced accuracy (BA) with different representations

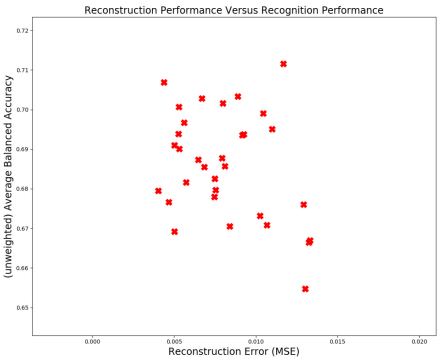
	Other	Drive w/out load	Drive w load	Take load	Leave load
Baseline (280D)	0.858 ± 0.022	0.811 ± 0.026	0.715 ± 0.026	0.684 ± 0.042	0.601 ± 0.078
Autoencoder (3D)	0.722 ± 0.286	0.684 ± 0.042	0.597 ± 0.044	0.624 ± 0.030	0.609 ± 0.022
Discriminator (3D)	0.848 ± 0.024	0.818 ± 0.040	0.718 ± 0.022	0.689 ± 0.016	0.615 ± 0.018

Table 3. Area under the precision-recall curve (APRC) with different representations

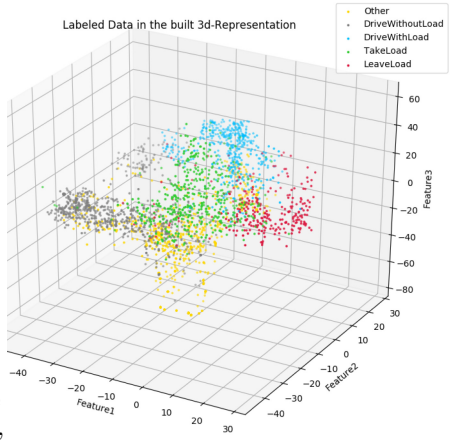
	Other	Drive w/out load	Drive w load	Take load	Leave load
Baseline (280D)	0.661 ± 0.380	0.602 ± 0.156	0.535 ± 0.032	0.546 ± 0.036	0.636 ± 0.042
Autoencoder (3D)	0.125 ± 0.136	0.557 ± 0.070	0.390 ± 0.036	0.382 ± 0.050	0.551 ± 0.042
Discriminator (3D)	0.256 ± 0.286	0.722 ± 0.054	0.507 ± 0.048	0.529 ± 0.024	0.528 ± 0.044

Tables 2 and 3 summarize the recognition results on the test set, for the baseline RF classifier with 280-dimensional input, and classifiers built using autoencoder and discriminator representations. We used three-dimensional representations in both cases (i.e. $N = 3$). However, when an RF classifier was constructed using the learned autoencoder representation, its recognition performance was worse than the one from the baseline RF model. Figure 3a shows the recognition performance (BA) plotted versus the reconstruction error and there is very little correlation (a negative correlation is expected). It shows that the most variance preserving representation did not coincide with the most discriminative representation. The results are concluded with that training a "better" autoencoder did not produce a more discriminative representation for the activity recognition task, indicating a misalignment of the two criteria.

In order to instead build a more discriminative representation, the autoencoder network was fine-tuned in a supervised manner with the pseudo-labeled data. The decoder part of the autoencoder was removed and an additional output layer was placed after the bottleneck layer (see Sect. 4). Since there are five target activities, the structure of the classification network was $280-128-64-32-3-5$, where the activation function in the output layer is softmax. After the training, the 3-dimensional bottleneck before the output layer is extracted to be the discriminator representation. Figure 3b is a visualization of the 3D-representation applied to the 58-min labeled data set. In this space, different activities are better separated from each other and it is possible to see the relationships between neighbors activities.



Comparison between recognition performance (BA) and autoencoder reconstruction error, when using autoencoder representations with different network structures.



The three-dimensional discriminator representation, after fine-tuning the autoencoder.

Fig. 3. Performance comparison between reconstruction and recognition (left) and visualization of the discriminator representation (right).

6 Conclusions

It was shown that streaming CAN data can be used to achieve machine activity recognition for a forklift truck. It was also presented that forklift activities at different sites are similar, i.e. autoencoders trained on one site data were also good for encoding data from another site, but that several hours of activity data are needed to build a good autoencoder for the signals. However, autoencoder representations yielded poor activity recognition, and there is a weak connection between reconstruction performance and recognition performance. It was shown how discriminative encoders could be trained with pseudo-labeled data, making it possible to construct a three-dimensional representation that achieved equal recognition results to the 280-dimensional baseline. This representation allowed a better visualization of the learned activities and their relationships.

A future work is to explore if a general supplementary criterion for representation learning for MAR can be inferred from comparing the autoencoder representation and the discriminative representation, avoiding the need for a pseudo labeled data set.

References

1. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**, 1798–1828 (2013)
2. Branco, P., Torgo, L., Ribeiro, R.: A survey of predictive modelling under imbalanced distributions (2015)
3. Breiman, L.: Random forests. Mach. Learn. **45**, 5–32 (2001)

4. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016). http://www.deeplearningbook.org
5. van der Maaten, L., Hinton, G.: Visualizing high-dimensional data using t-SNE. J. Mach. Learn. Res. **9**, 2579–2605 (2008)
6. Saito, T., Rehmsmeier, M.: The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLoS ONE **10**, e0118432 (2015)
7. Sherafat, B., et al.: Automated methods for activity recognition of construction workers and equipment: state-of-the-art review. J. Constr. Eng. Manag. **146**, 03120002 (2020)
8. Vachkov, G.: Classification of machine operations based on growing neural models and fuzzy decision. In: 21st European Conference on Modelling and Simulation (ECMS 2007) (2007)
9. Vachkov, G., Kiyota, Y., Komatsu, K., Fujii, S.: Real-time classification algorithm for recognition of machine operating modes by use of self-organizing maps. Turkish J. Electr. Eng. **12**, 27–42 (2004)