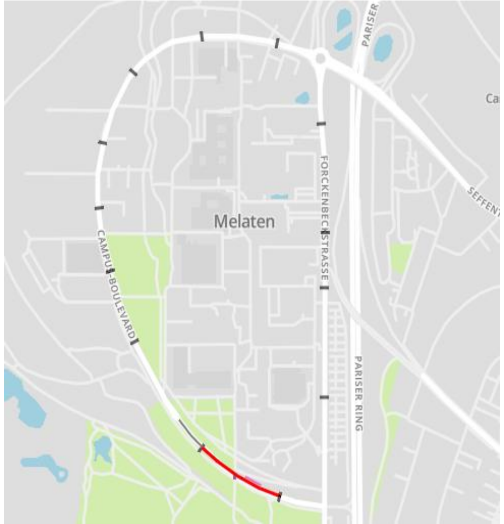
Masterarbeit

Maneuver identification in urban traffic using machine learning

Xiaoman Liu

Supervisor: Thilo Braun

Examiner: Prof. Dr-Ing. Eric Sax



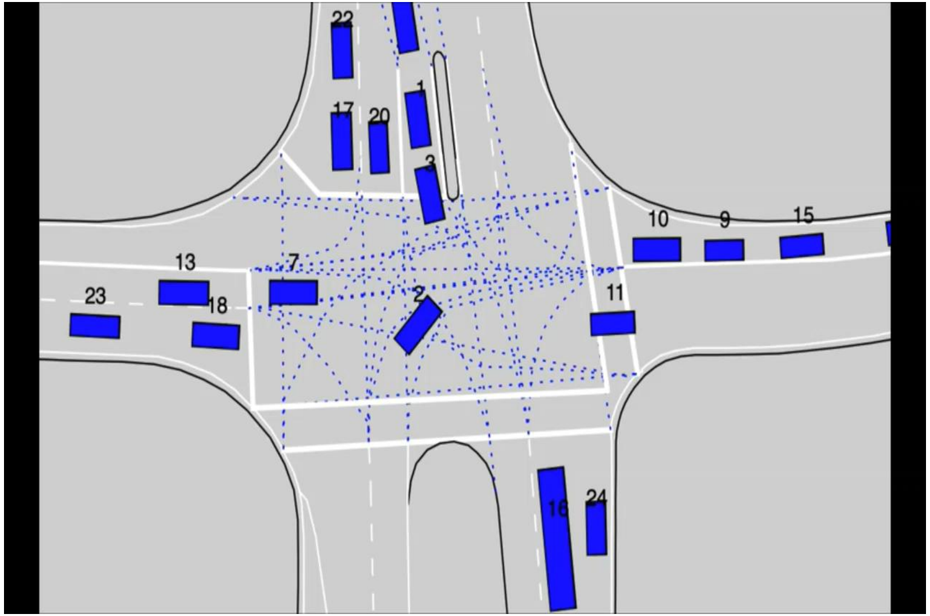
FZI FORSCHUNGSZENTRUM INFORMATIK

Contents

Motivation

- Validate autonomous driving algorithms in simulation platform
- Different cases can be created in simulation platform by quantitative modification of driving data
- For realistic simulation, real-world knowledge is needed
- Require abstract forms of description for different driving maneuvers in urban area

- Thesis: investigate methods to identify the driving maneuvers in recorded real world



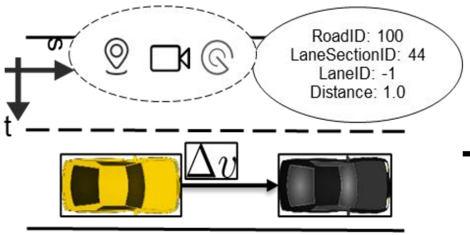
Motivation

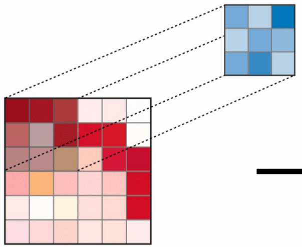
- Maneuver Identification to understand the recorded data
- Investigate the performance of neural networks for this task

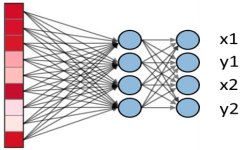
Numerical features
- Ego position & relevant position
- Speed & relevant speed
- Time
- ...
- Category features
- RoadID

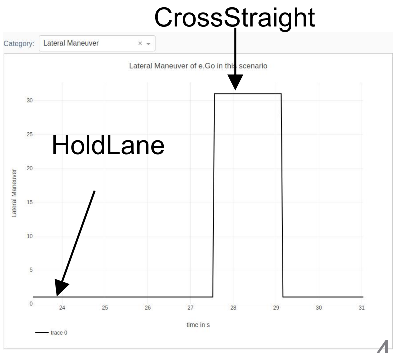
Neural networks
- Convolutional Neural Network
- Recurrent neural network
- ...

Lateral maneuver
- HoldLane
- Crossstraight
- ...
- Longitudinal maneuver
- CruiseFree
- ...









State of the art

- Motivation
- State-of-the-art
  - Convolutional Neural Network
    - Residual Neural Network (ResNet)
    - Fully Convolutional Networks (FCN)
  - Recurrent Neural Network
    - Long Short Term Memory (LSTM)
    - Bidirectional Long Short-Term Memory (Bi-LSTM)
- Datasets
- Experiment
- Conclusion
- Outlook

State of the art

- Different Methods of Maneuver Identification
  - Rule-based (already implemented)
  - Machine Learning (great performance in many problems) → Master thesis
- Different Machine Learning Methods
  - Unsupervised learning
    - Understand patterns behind input data
  - Reinforcement learning
    - An agent interacts with the environment & learn from errors or reward
  - Supervised learning
    - Learn the mapping function from the input to the output → Master thesis

Criterions	Supervised Learning	Unsupervised learning	Reinforcement learning
Data	Labeled data	Unlabeled data	No predefined data
Problem type	Classification & Regression	Clustering & Association	Rewards based
Real time learning	*Offline*	Real time	Real time

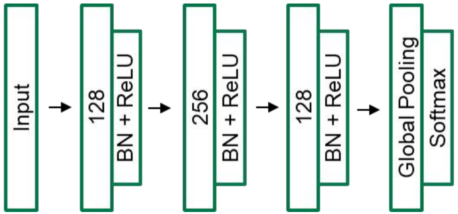
Criterions	LSTM	Bi-LSTM	Resnet	FCN
Accuracy	+	+	+	+
Space/ Time	o	o	+	+

+ best o medium - worst

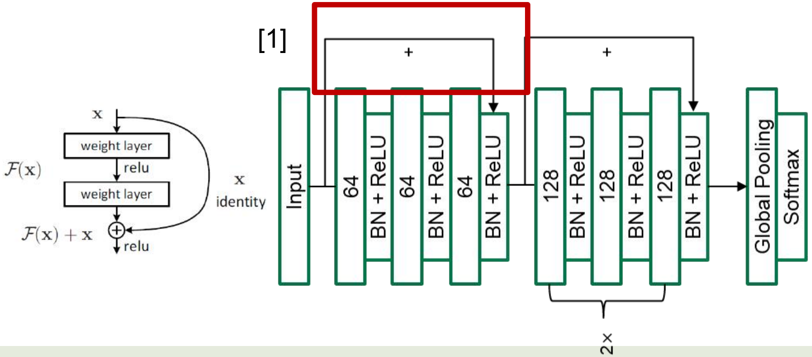
State of the art

FCN	ResNet

[1]



[1]



- FCN has achieved great performance in pixel-level semantic segmentation[2]
- Deep layers easily lead to gradient vanishing or explosion
- FCN baseline structure: 3 convolution blocks + GAP layer + softmax classifier[1]

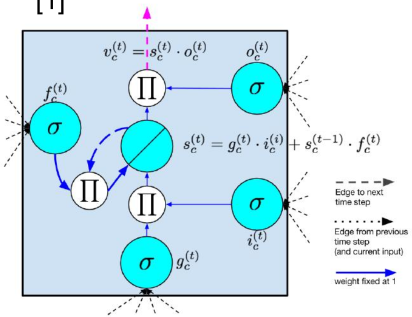
- ResNet: Winner in 2015 ImageNet
- Residual connection improves neural network gradient vanishing or explosion
- ResNet baseline structure: 3 residual blocks + GAP layer + softmax classifier[1]

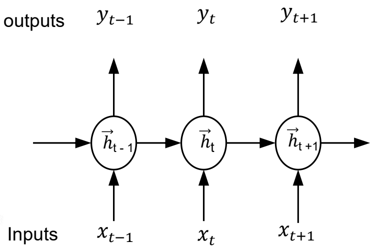
[1] Zhiguang Wang etc. Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline
[2] E. Shelhamer, J. Long, and T. Darrell. “Fully Convolutional Networks for Semantic Segmentation

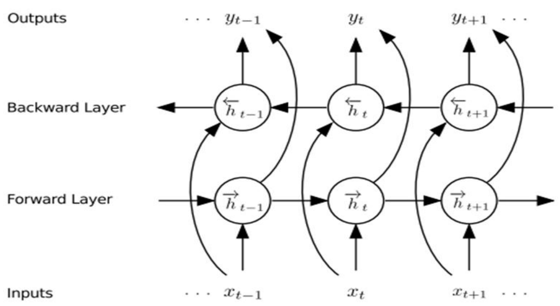
State of the art

LSTM	Bi-LSTM

[1]

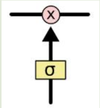






[2]

- Widely used in image captioning, speech recognition
- Overcome the problem of vanishing gradients
- Learning long-term dependencies
  - Input gate
  - Forget gate
  - Output gate



- Widely used in machine translation, speech recognition
- Overcome the problem of vanishing gradient
- Train with using all available input information in the past and future of a specific time frame[3]

[1] Zachary C. Lipton etc. A Critical Review of Recurrent Neural Networks for Sequence Learning
[2] Alex Graves etc. Hybrid Speech Recognition With Deep Bidirectional Lstm
[3] Mike Schuster and Kuldip K. Paliwal, Bidirectional Recurrent Neural Networks

Datasets

- Motivation
- State-of-the-art
- Datasets
  - Format
  - Challenges
    - Class imbalance
    - Variable length sequence
  - Data preprocessing
- Experiment
- Conclusion
- Outlook

Overview

Feature engineering

Datasets

- Challenges
  - Data imbalance
  - Variable length sequence
- Data preprocessing

Models

- FCN
- ResNet
- LSTM
- Bi-LSTM

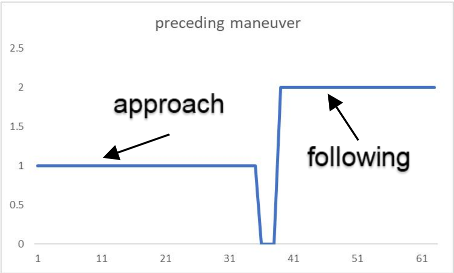
Evaluate methods

- Accuracy
- Precision
- Recall
- F1-score
- Confusion Matrix

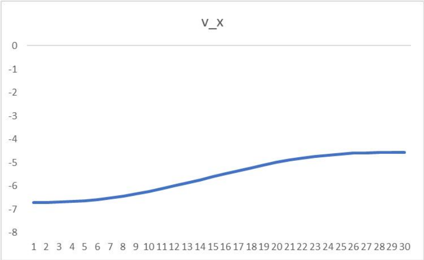
Datasets 1 - INTERACTION dataset - Format

- Interaction dataset visualization – in urban scene
- Total 724 samples
- 18 features (numerical variable + categorical variable)
- 4 labels (lane + preceding + turn + vehicle state)

Preceding maneuver

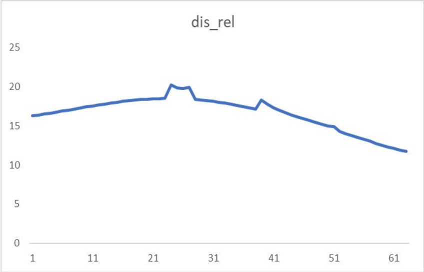


Vehicle speed



Vehicle state maneuver

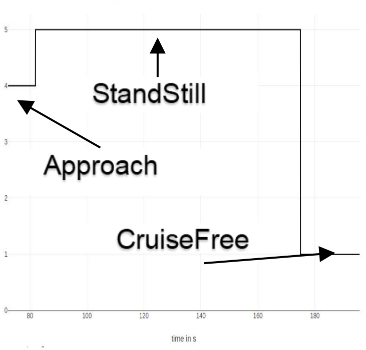
Relative distance



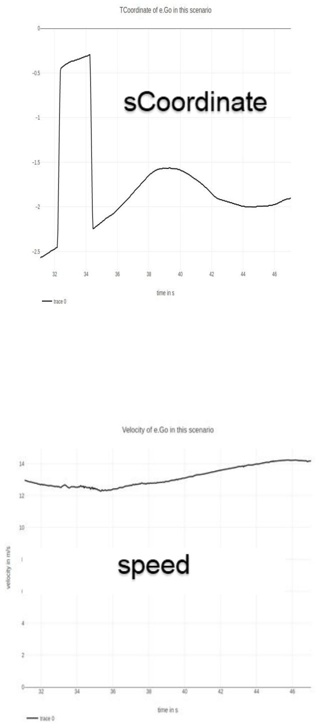
Datasets 2 - FZI dataset - Format

- FZI dataset visualization – in urban scene
- Total 699 samples
- 14 features
- 2 labels (lateral maneuver + longitudinal maneuver) labeled from rule based algorithm

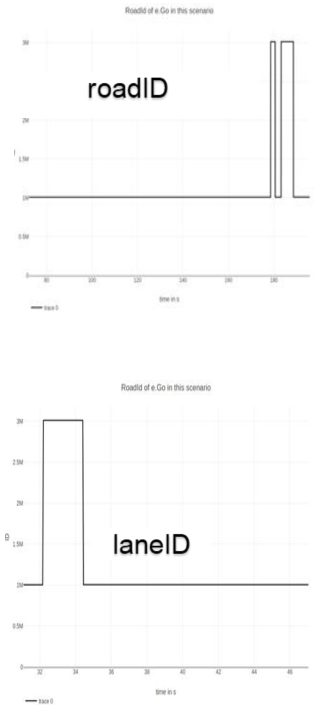
Longitudinal maneuver



Numerical variable



Categorical variable



List of maneuvers

Lateral maneuver	NoneLat	HoldLane	ChangLane	CrossStraight	TurnLeft	TurnRIght	CrossRoad
Longitudinal maneuver	NoneLong	CruiseFree	Follow	Approach	Stop	Standstill	

Datasets - Challenges

Class imbalance

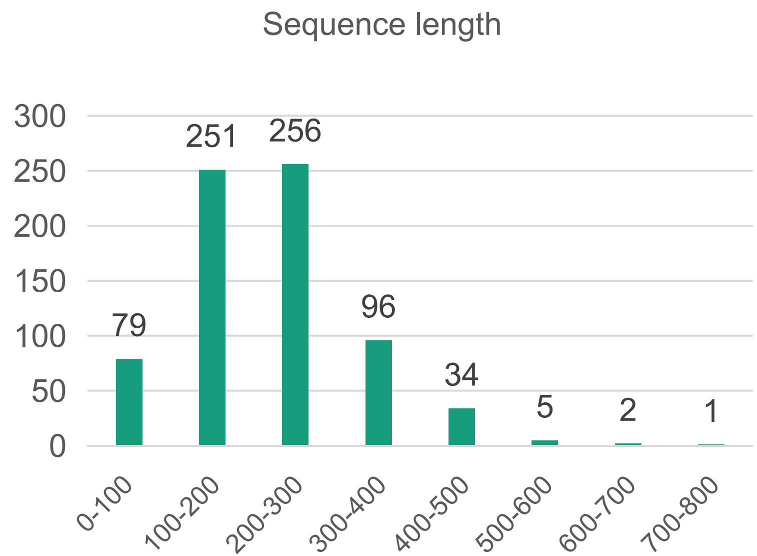
- Result in poor performance of model: tend to classify all samples to majority class

Lateral maneuver		Number	Longitudinal maneuver	Number
NoneLat		3770	NoneLong	**99085**
ChangeLane		17697	CruiseFree	31817
CrossJunction	CrossStraight	14991	Follow	1548
	TurnLeft	1257		
	TurnRight	3108		
CrossRoad		**1430**	Approach	1415
HoldLane		**97474**	Stop	**1167**
--------			StandStill	4695

- Lateral maneuver：
  70% HoldLane, 1 % CrossRoad
- Longitudinal maneuver：
  71% NoneLong, 1 % Stop

Variable sequence length

- Sequence length in a batch should be consistent



- Maximum: 777
- Minimum: 16
- 90 % of sequence length are in range 0-300

Datasets - Challenges - Solution

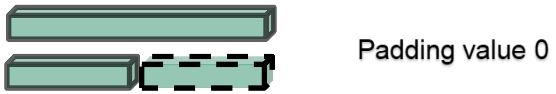
Class imbalance - Solution

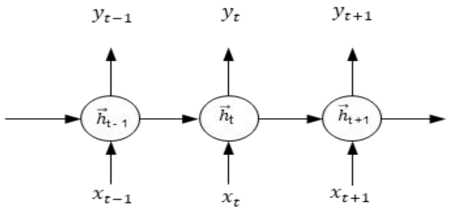
- Data
- Evaluation methods
- Model

Aspect	Solution	Status
Data	Get more small sample data	×
	Upsampling Downsampling	×
	create new features	√
	Weighted loss function	√
Evaluation methods	Precision	√
	Recall	√
	F1-score	√
Model	Change model	√

Variable sequence length - Solution

- LSTM model
  - Padding 0 + masking layer
  - Padding value do not update the weights
- FCN model
  - Padding 0 to the same length





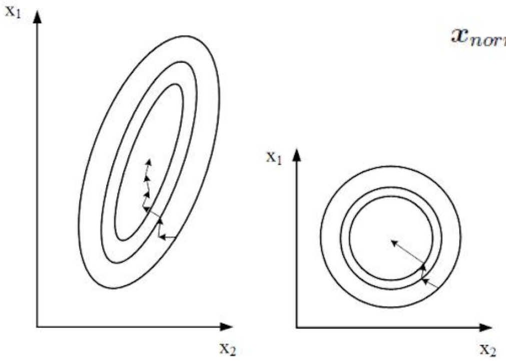
Datasets - Data preprocessing

Numerical variables → Normalization

- Optimization of the loss function is based on the gradient descent method
- Normalization speeds up convergence

Categorical variables → Onehot encoding

- Transform categorical variables into a vector from the Euclidean space
- For computing distances between features or similarities between features
- Easily compute distances in Euclidean space



$$x_{norm} = \frac{x - x_{mean}}{x_{max} - x_{min}}$$

Gradient descent process: unnormalized data vs Normalized data

Type
Car - 1
Ped - 2
Bike - 3

→

1	2	3
1	0	0
0	1	0
0	0	1

Onehot encoding process

Datasets - Feature engineering

- Process of using domain knowledge to extract powerful features from raw data via data mining techniques
- Feature engineering
  - Onehot encoding on categorical variable
  - Data normalization numerical variable
  - Add new features:
    - Polar angle
    - Sample type
  - Feature relabeling
    - RoadID
    - LaneID
  - Forward selection algorithm for feature selection

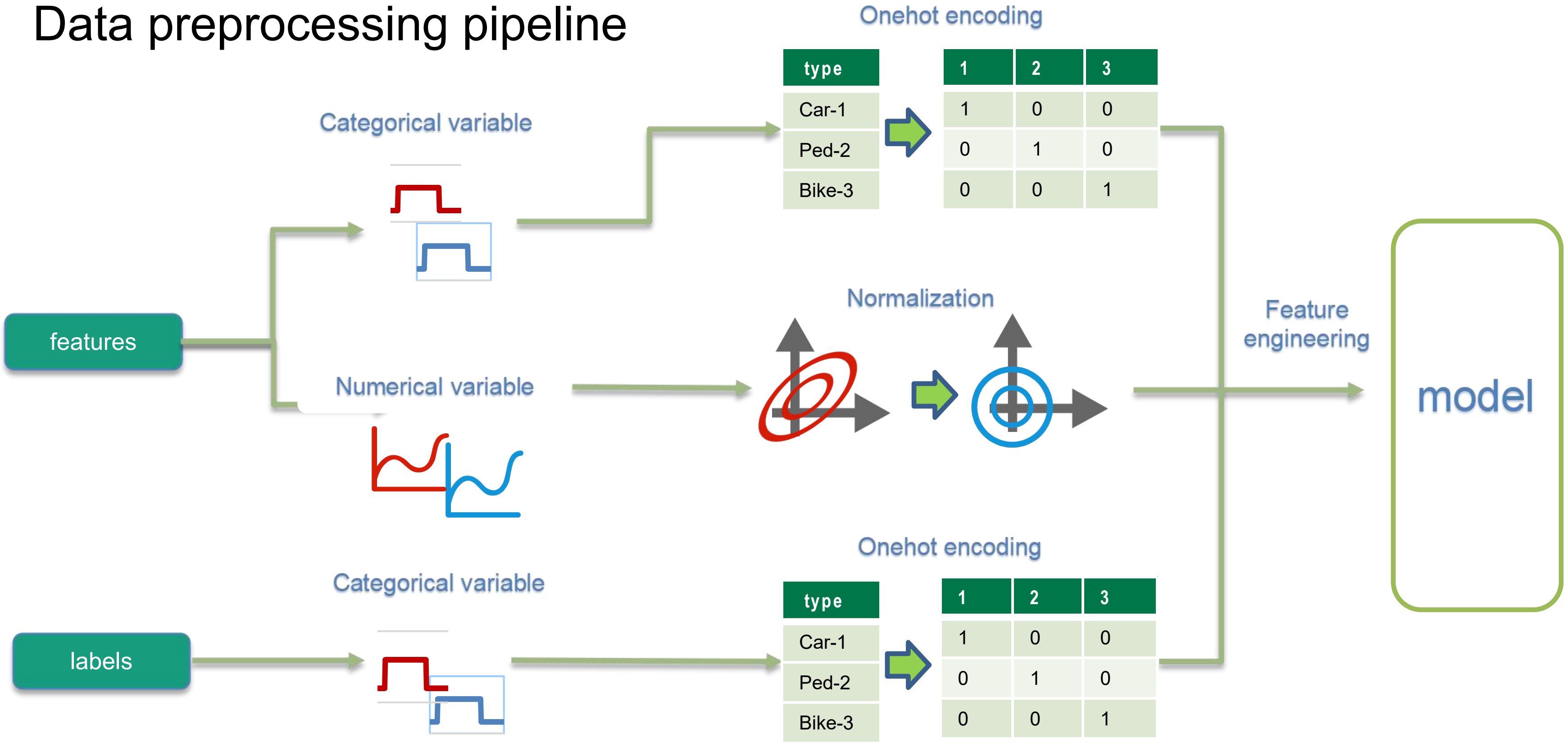
Row data performance

Features	Precision	Recall	F1-score	Accuracy
Time, PosX, PosY, sCoordinate, tCoordinate, speed, Yaw, roadID, LaneID, tLane, roadYaw, heading	0.2352	0.2696	0.2512	0.7102
Time, PosX, PosY, sCoordinate, tCoordinate, speed, Yaw, roadID, LaneID, tLane, roadYaw, heading **(with onehot encoding)**	0.5182	0.5548	0.5341	0.7936
Time, sCoordinate, tCoordinate, roadID,laneID, Sample type (with onehot encoding)	0.5668	0.5181	0.5327	0.7730
Time,sCoordinate, tCoordinate, roadID, laneID, **roadYaw**, **heading**, Sample type (with onehot encoding)	0.6652	0.7103	0.6633	0.8188
Time,sCoordinate, tCoordinate, roadID, laneID, roadYaw, heading, Sample type (with onehot encoding, **with weighted loss and data normalization**)	0.6919	0.7148	0.6929	0.8394
Time,sCoordinate, tCoordinate, roadID, laneID, roadYaw, heading, Sample type (with onehot encoding,with weighted loss and data normalization, **relabelled laneid**)	0.6190	0.6944	0.6403	0.8531
Time,sCoordinate, tCoordinate, roadID, laneID, roadYaw, heading, Sample type (with onehot encoding, with weighted loss and data normalization, relabelled laneid, **relabelled roadid**)	0.6200	0.7322	0.6458	0.8693
Time,sCoordinate, tCoordinate, roadID, laneID, **tlane**, roadYaw, heading, Sample type (with onehot encoding,with weighted loss and data normalization, relabelled laneid, relabelled roadid)	0.6585	0.7136	0.6657	0.8844
Time,sCoordinate, tCoordinate, roadID, laneID, **polar angle**, roadYaw, heading, Sample type (with onehot encoding,,with weighted loss and data normalization, relabelled laneid, relabelled roadid)	0.6992	0.7290	0.6882	0.8730

Performance after feature engineering

Datasets

- Data preprocessing pipeline



Experiments

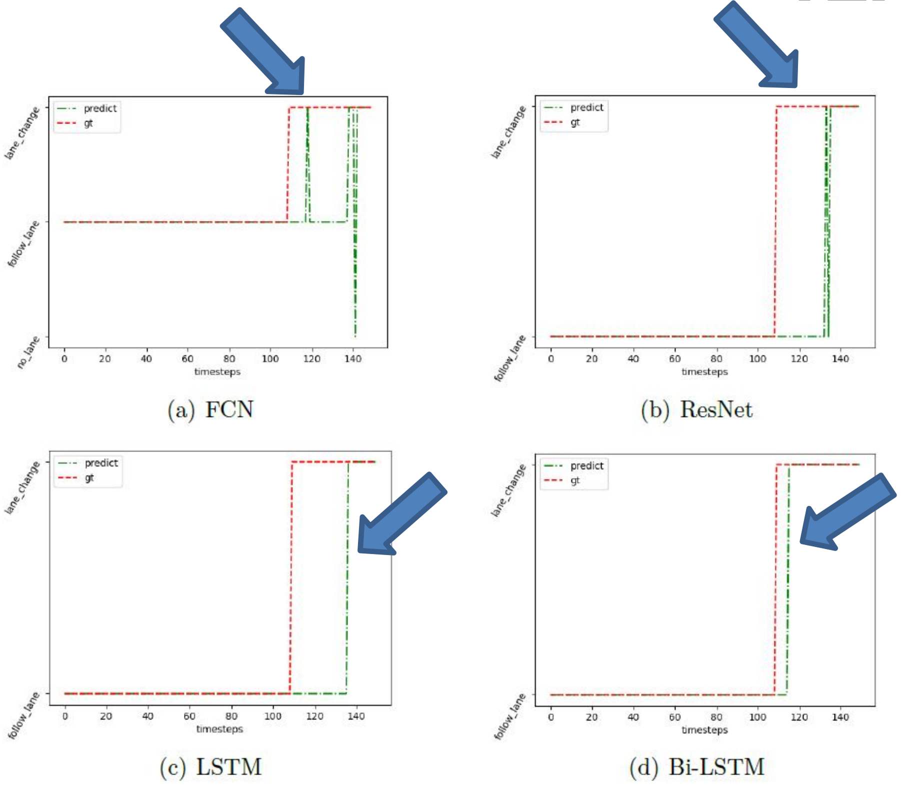
- Motivation
- State-of-the-art
- Datasets
- Experiments
  - Interaction dataset
  - FZI dataset
- Conclusion
- Outlook

Experiments - Interaction dataset

- Training dataset: 456 samples, Test dataset 268 samples
- **Lane Maneuver:**
  - no_lane
  - follow_lane
  - lane_change
- Selected sample covers as many maneuvers as possible

methods	precision	recall	f1-score	accuracy
FCN	0.7435	0.5251	0.5904	0.9343
ResNet	0.7164	0.5542	0.6103	0.9363
LSTM	0.6826	0.5167	0.5664	0.9305
Bi-LSTM	**0.7528**	**0.7149**	**0.7314**	**0.9467**

Model performance on lane maneuver



Model performance on lane maneuver

Experiments - Interaction dataset

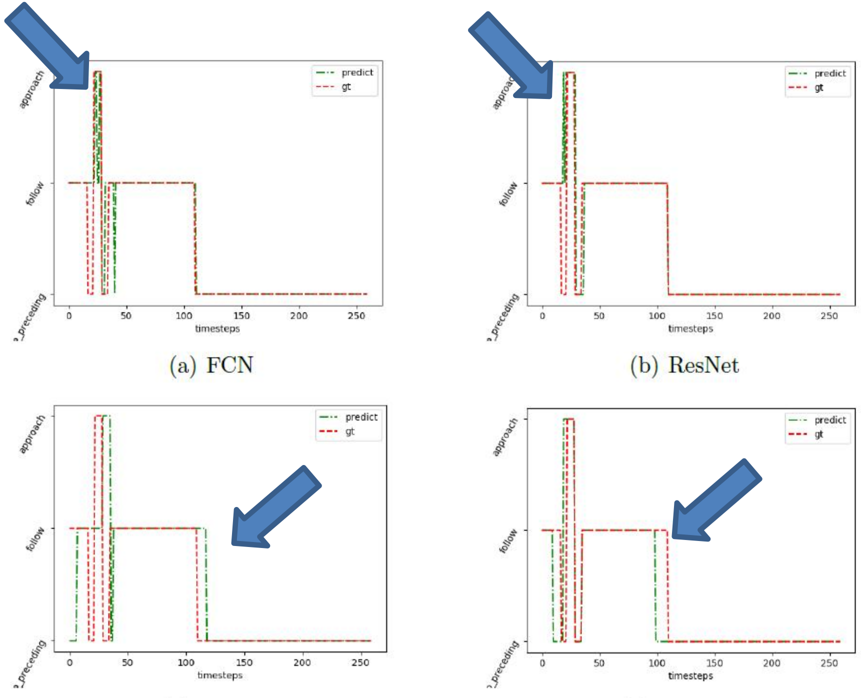
- Training dataset: 456 samples, Test dataset 268 samples

- **Preceding Maneuver:**
  - no_preceding
  - follow
  - approach

- Selected sample covers as many maneuvers as possible

methods	precision	recall	f1-score	accuracy
FCN	0.9119	0.9252	0.9185	0.9600
ResNet	**0.9229**	**0.9545**	**0.9382**	**0.9738**
LSTM	0.8953	0.8942	0.8947	0.9524
Bi-LSTM	0.9150	0.9343	0.9239	0.9620

Model performance on preceding maneuver



(a) FCN (b) ResNet (c) LSTM (d) Bi-LSTM

Model performance on preceding maneuver

Experiments - Interaction dataset

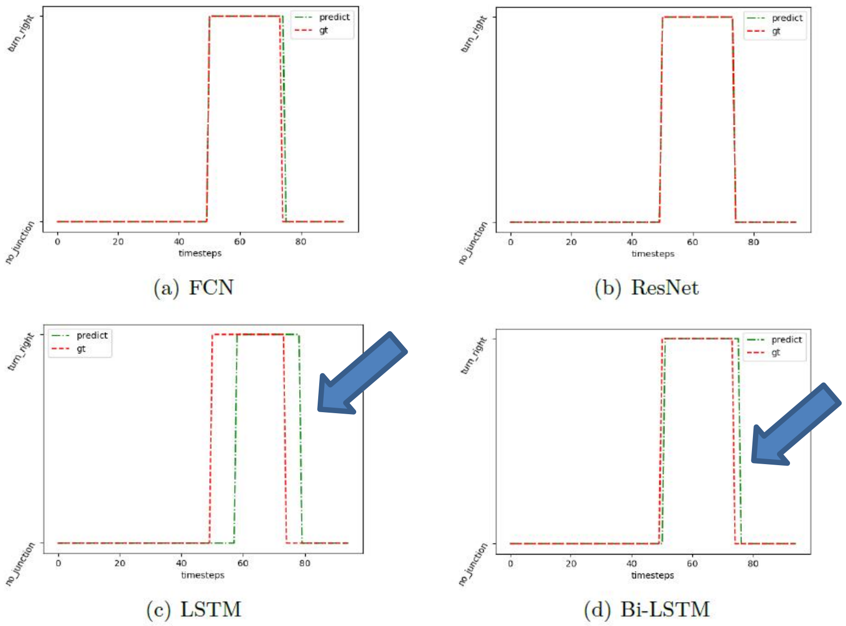
- Training dataset: 456 samples, Test dataset 268 samples

- **Turn Maneuver:**
  - no_junction
  - turn_left
  - turn_right

- Selected sample covers as many maneuvers as possible

methods	precision	recall	f1-score	accuracy
FCN	0.9073	0.9090	0.9081	0.9556
ResNet	**0.9338**	0.9175	**0.9250**	**0.9695**
LSTM	0.9092	0.8691	0.8878	0.9472
Bi-LSTM	0.9305	**0.9178**	0.9236	0.9614

Model performance on turn maneuver



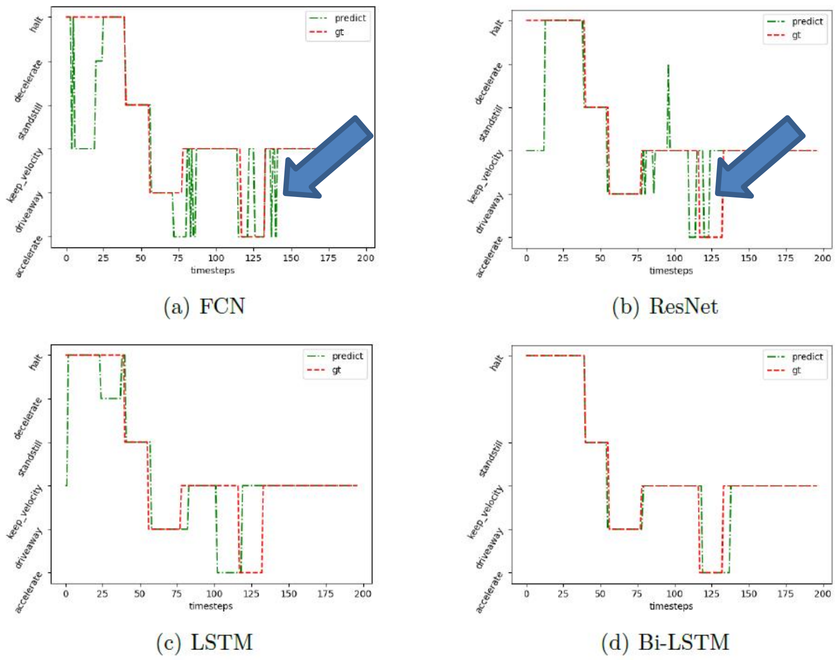
Model performance on turn maneuver

Experiments - Interaction dataset

- Training dataset: 456 samples, Test dataset 268 samples
- **Vehicle_state Maneuver:**
  - halt
  - standstill
  - driveaway
  - keep_velocity
  - accelerate
  - decelerate
- Selected sample covers as many maneuvers as possible

methods	precision	recall	f1-score	accuracy
FCN	0.7075	0.7205	0.7129	0.7420
ResNet	0.7473	0.7385	0.7414	0.7806
LSTM	0.6498	0.6464	0.6442	0.6708
Bi-LSTM	**0.8804**	**0.8800**	**0.8795**	**0.8828**

Model performance on vehicle state maneuver

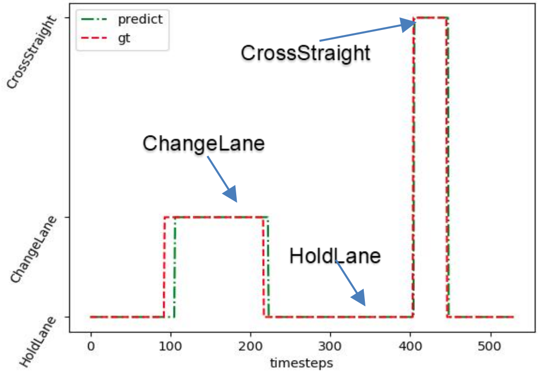


Model performance on vehicle state maneuver

Experiments - FZI dataset

- Lateral maneuver classification
- Training dataset: 559 samples, Test dataset 140 samples

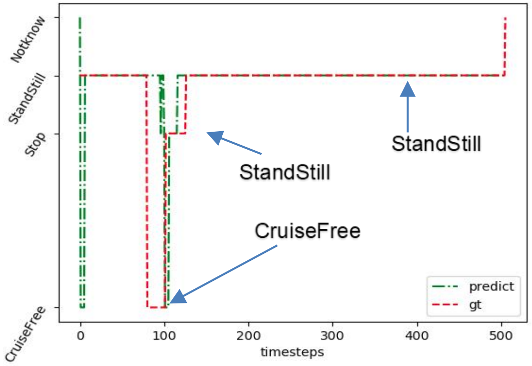
methods	precision	recall	f1-score	accuracy
FCN	**0.7216**	**0.7821**	**0.7282**	0.8635
ResNet	0.7037	0.7328	0.7179	**0.8787**
LSTM	0.6993	0.7290	0.6882	0.8730
Bi-LSTM	0.6612	0.6567	0.6589	0.8536



FCN performance

- Longitudinal maneuver classification
- Training dataset: 559 samples, Test dataset 140 samples

methods	precision	recall	f1-score	accuracy
FCN	**0.6180**	**0.6176**	**0.6135**	0.8914
ResNet	0.6069	0.5839	0.5817	0.8785
LSTM	0.5605	0.5727	0.5644	0.8785
Bi-LSTM	0.5561	0.5739	0.5580	**0.8942**



FCN performance

Conclusion

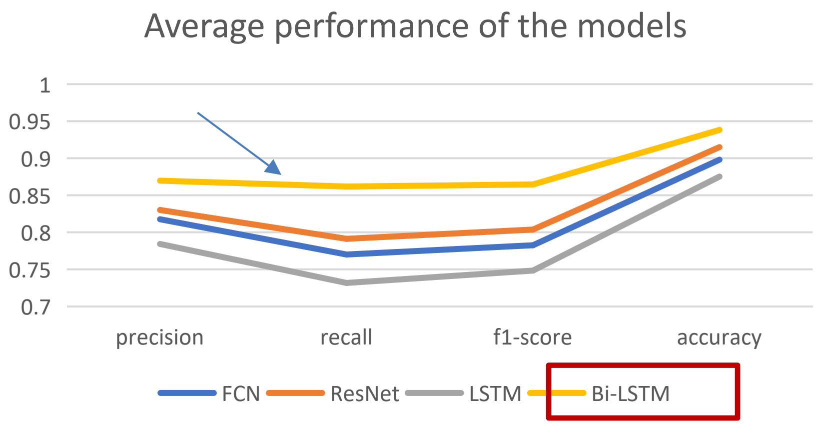
- Motivation
- State-of-the-art
- Datasets
- Experiments
- Conclusion
- Outlook

Conclusion

- Time series classification models have successfully created based on state of art CNN (FCN, ResNet) and RNN (LSTM, Bi-LSTM) model
- ResNet can learn better feature expression compared to FCN due to the structure of short connections
- Bi-LSTM can learn the dependencies in sequences better than LSTM because it can use contextual information
- Overall, Bi-LSTM has the most stable performance in different classes

Average performance of the models

methods	precision	recall	f1-score	accuracy
FCN	0.8176	0.7700	0.7825	0.8980
ResNet	0.8301	0.7912	0.8037	0.9150
LSTM	0.7842	0.7316	0.7483	0.8752
Bi-LSTM	**0.8697**	**0.8618**	**0.8646**	**0.9382**



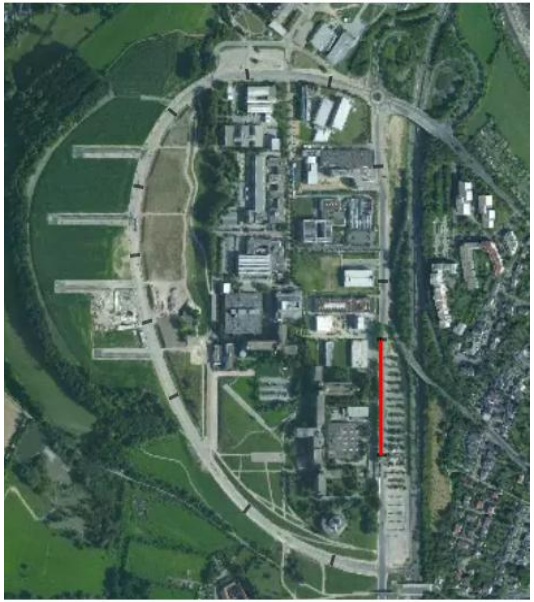
Future work

- Motivation
- State-of-the-art
- Datasets
- Experiments
- Conclusion
- Outlook

Outlook

- Use more data and other data to verify the generalization ability of the models
- Attention Mechanisms is a promising method, which extracts more critical and important information by assigning different weights to each part of the input
- Bi-LSTM can be merged with ResNet or FCN together to obtain a more stable and strong model

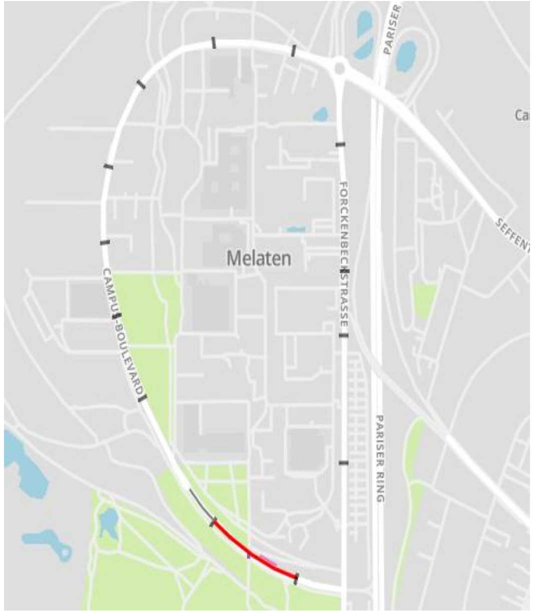
Thank you for your interest !



Xiaoman Liu

Supervisor: Thilo Braun

Examiner: Prof. Dr-Ing. Eric Sax



FZI FORSCHUNGSZENTRUM INFORMATIK