

TURKU AMK

TURKU UNIVERSITY OF
APPLIED SCIENCES



Moni-moneen aikasarjaregressiomalli LSTM-neuroverkolla

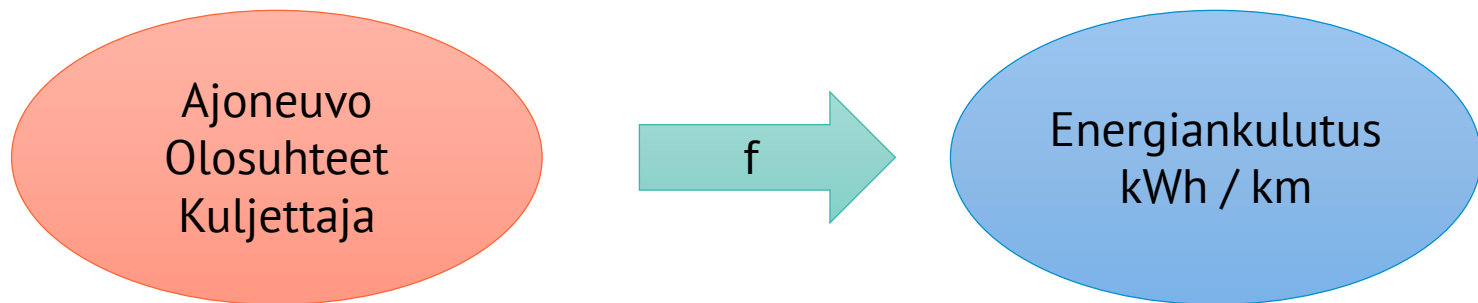
Case : Energiankulutusmalli Turun kaupungin sähköbusseille

Motivaatio

- Suomen ensimmäiset sähköbussioperaatiot ovat käynnissä pk-seudulla, Tampereella ja Turussa
- Liikenteessä olevat bussit on laajasti instrumentoitu ja niistä kerätään jatkuvasti sensoridataa
- Voidaanko syväoppimisella saada datasta irti tietoa, joka hyödyttää:
 - Tulevien reittien suunnittelua?
 - Nykyisten optimointia aikataulutuksen ja latausten suhteen?
 - Koulutuksellisia tavoitteita / kuljettajien ajotavan optimointia?



- Sähköbussin energiankulutus teoriassa:



- Sähköbussin energiankulutus käytännössä:

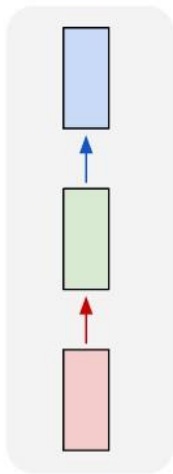
Maaston korkeuserot, liikenteen ruuhkaisuus,
reitin tyyppi
(kaupunkikeskus/taajama/moottoritie),
ajoalustan kunto, pysäkkien määrä ja sijainti
reitillä, pysähdysten lukumäärä ja kesto,
ajoneuvon omamassa, ajoneuvon keulan pinta-
ala, matkustajien määrä, akun ja voimalinjan
kunto sekä kokonaishyötysuhde, lisälaitteiden
kuten paineilmakompressorin käyttö,
rengaspaineet, ulkoilman lämpötila,
matkustamon sisälämpötila, ilmalämpöpumpun
/ lisälämmittimen käyttö, aikataulutus,
keskimääräinen ajonopeus, kiihdytysten ja
jarrutusten lukumäärä sekä suuruus, kuljettajan
kokemus, asenne ja halu ajaa taloudellisesti,
regeneratiivisen jarrutuksen käyttö...

LSTM?

Energiankulutus
kWh / km

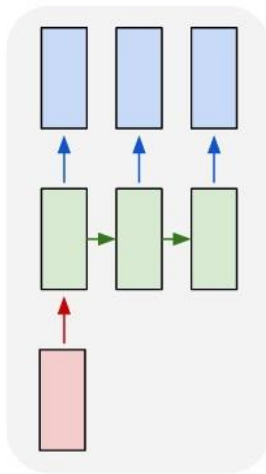
Toistavien neuroverkkojen ominaisuuksia

one to one



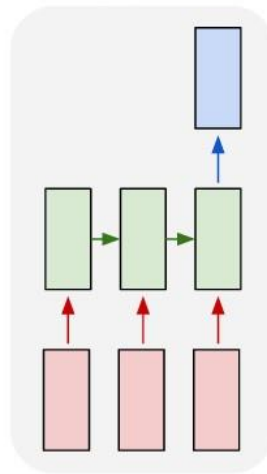
ei aikaulottuvuutta -
ei kovin mielenkiintoinen

one to many



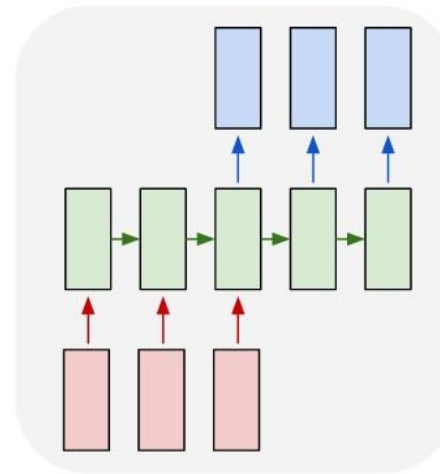
Vaihtelevan pituinen sekvenssi
outputtina staattisen inputin
perusteella – esim. kuvatekstien
luominen

many to one



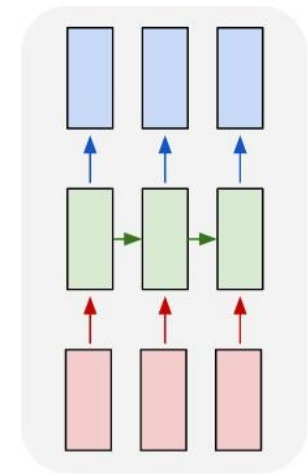
Staattinen output, vaihteleva input
– esim. sentiment analyysi (lause
sisään, luokittelu positiivinen /
negatiivinen mielipide tai
pisteytys regressiolla)

many to many



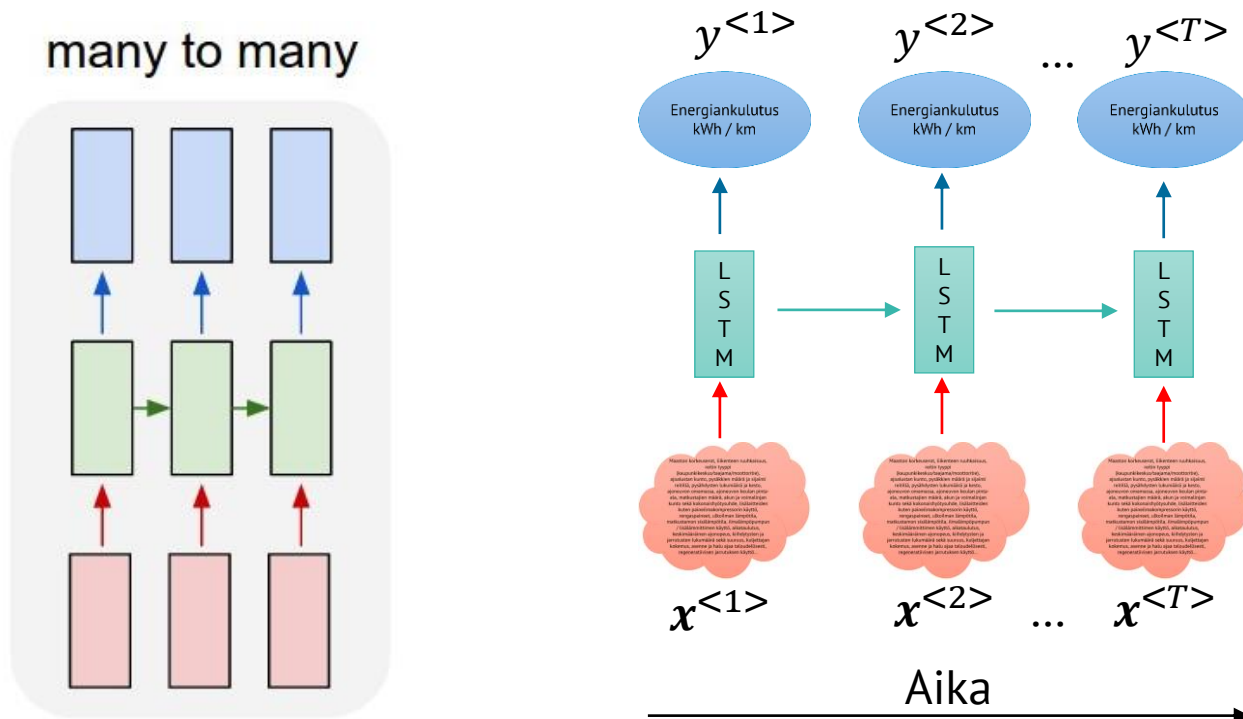
Esim. konekääntäminen –
outputin aika-askelten määrän ei
tarvitse täsmätä inputtiin

many to many

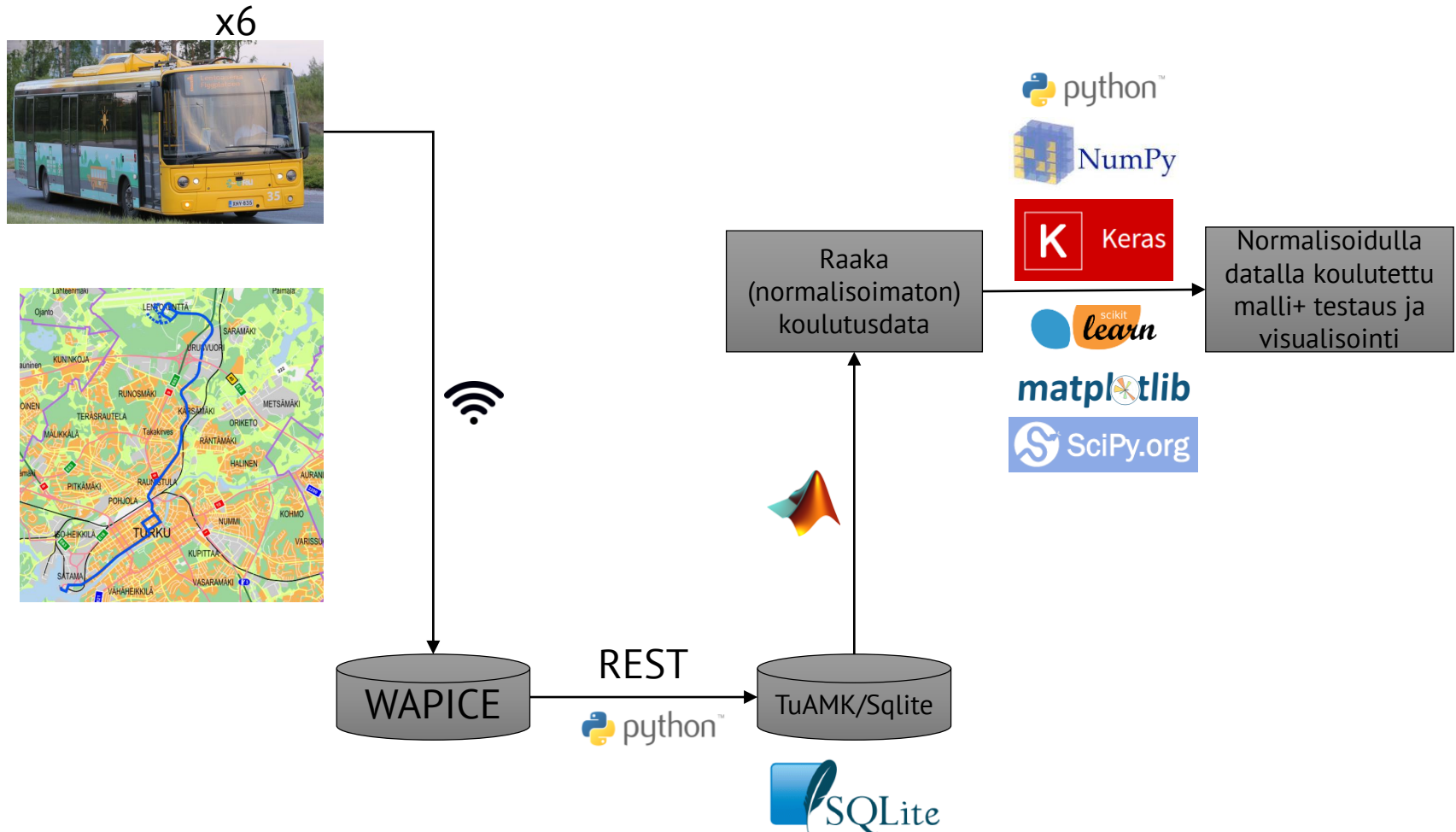


Edellisen erikoistapaus, synkattu
moni-moneen – jokaista inputin
aika-askelta vastaa täsmälleen
yksi outputin aika-askel. Esim.
Videokuvan framejen luokittelu

Mallin yleisperiaate – valittu arkkitehtuuri



Tiedonkeruun yleisperiaate ja käytetyt pääteknologiat



- Bussien sensoridata käsiteltiin MATLAB-ympäristössä vektoreiksi, jotka esittävät järjestelmän (siis bussin) tilan ajanhetkellä t:

$$\mathbf{x}^{<t>} = \begin{bmatrix} \text{GPS – pituus} \\ \text{GPS – leveys} \\ \text{Est. korkeus merenpinnasta}^1 \\ \text{Ajonopeus} \\ \text{Ulkoilman lämpötila} \\ \text{Kellonaika} \end{bmatrix}$$

- Jokaista vektoria $\mathbf{x}^{<t>}$ vastaa koulutusdatassa skalaari $y^{<t>}$

$$y^{<t>} = \text{Akun varaustila, \%}$$

1) Estimoitu GPS-tiedon perusteella; ei saatu luotettavaa tietoa suoraan raakadatatista

- Yksittäinen $(x^{<t>}, y^{<t>})$ pari ei voi vielä muodostaa LSTM-verkon koulutusnäytettä, koska aikaulottuvuus puuttuu
- Parit jatkokäsiteltiin edelleen **sekvensseiksi**

$$< (x^{<0>}, y^{<0>}), (x^{<1>}, y^{<1>}), \dots, (x^{<T>}, y^{<T>}) >$$

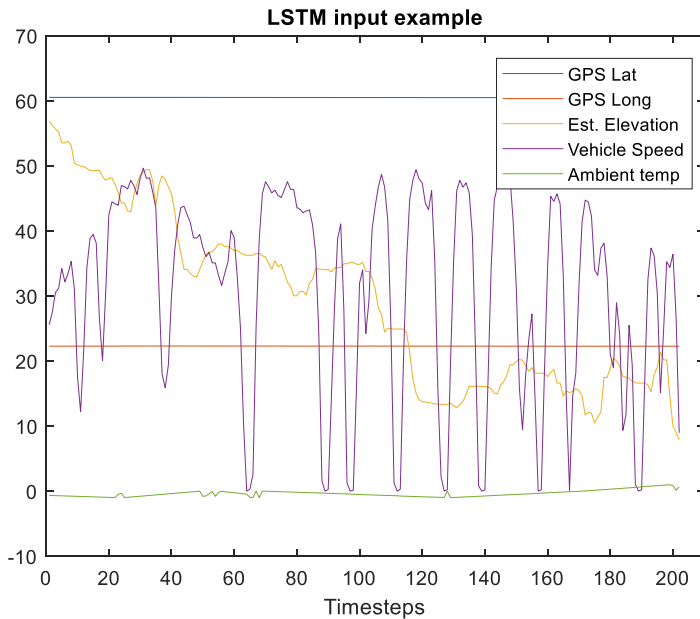
- Sekvensointitapa mielivaltaisesti valittavissa – T voitaisiin esim. lukita etukäteen, jolloin saataisiin kiinteän mittaisia näytteitä. Voidaan käyttää myös jotain muuta ”pätkimissääntöä”, jolloin T vaihtelee näytteestä toiseen. Kumpikin tapa on LSTM-verkon näkökulmasta validi.
- LSTM-verkko ei myöskään vaadi, että kahden aika-askeleen t ja $t + 1$ välinen aika olisi vakio! Näin ei käytännössä olekaan, koulutusdatassa olevista satunnaisista epäjatkuvuuksista johtuen.

Datan esiprosessointi

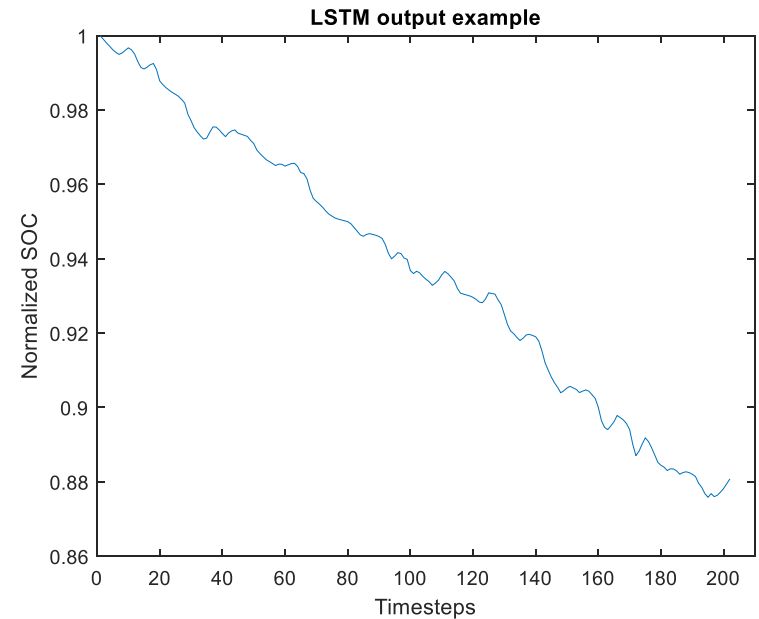
- Tässä sekvensointi tehtiin bussireitin osavälien perusteella
- Raakadata paloiteltiin sekvensseiksi ajoneuvotunnisteen sekä gps-tiedon perusteella
 - Sekvenssi alkaa, kun GPS-tiedon mukaan ajoneuvo lähtee päätepysäkiltä, sekä päättyy kun ajoneuvo saapuu seuraavalle päätepysäkille (haversine-etäisyyskaava)
- Kun käsiteltiin neljän sähköbussin datat kalenterivuoden 2017 ajalta, tuloksena saatiin yhteensä noin 28 600 sekvenssiä

Reittiosuus / ajoneuvo	80035	80037	80038	80040	TOTAL
Lentokenttä - Kauppatori	2306	2110	1524	1358	7298
Kauppatori - Satama	2192	2009	1501	1286	6988
Satama - Kauppatori	2203	2011	1478	1305	6997
Kauppatori - Lentokenttä	2325	2085	1515	1361	7286
TOTAL	9026	8215	6018	5310	28569

Input – Output esimerkki



$$\mathbb{R}^{n \times T} \rightarrow \mathbb{R}^T$$



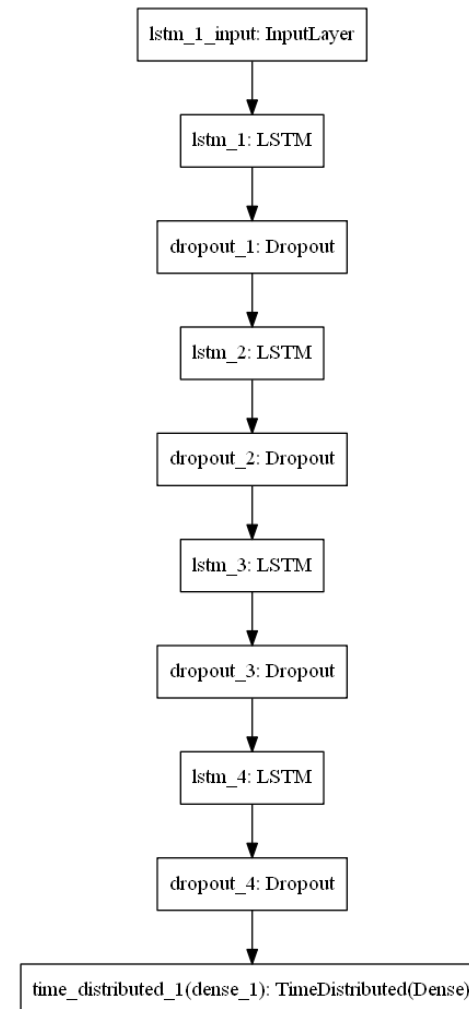
n : selittävien muuttujien määrä → 5
T : näytteen aika-askelten määrä ~100 - 250

Yleisessä tapauksessa T voi vaihdella
näytteestä toiseen

- Malli toteutettiin Pythonin Keras-kirjastolla käyttäen Tensorflow-moottoria
- Apukirjastoina datan käsittelyyn ja visualisointiin mm. numpy, sklearn, pandas, matplotlib
- Malli mahdollisimman yleiskäyttöinen
- Inputit/output dimensiot voidaan valita vapaasti (muuttujien sekä aika-askelten lkm) –ts. malli yleistyy funktioksi:

$$lstm: \mathbb{R}^{n \times T} \rightarrow \mathbb{R}^{n' \times T'}$$

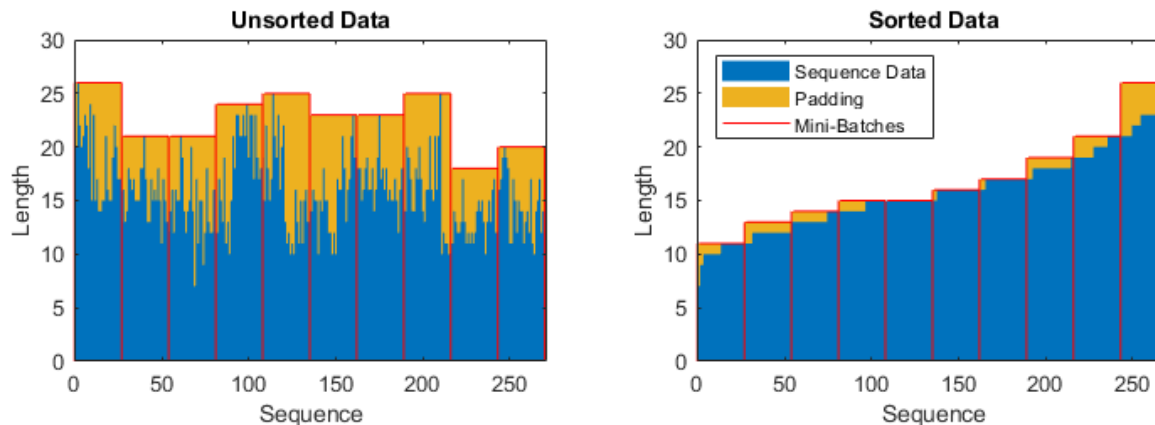
- Tämän hetkinen malli soveltuu regressio-ongelmiin, pienillä muutoksilla (kustannusfunktio ja ulostulokerroksen aktivaatio) myös luokitteluun
- Kustannusfunktiona MSE ja optimointi-algoritmina ADAM



Mallin kontrollit / hyperparametrit

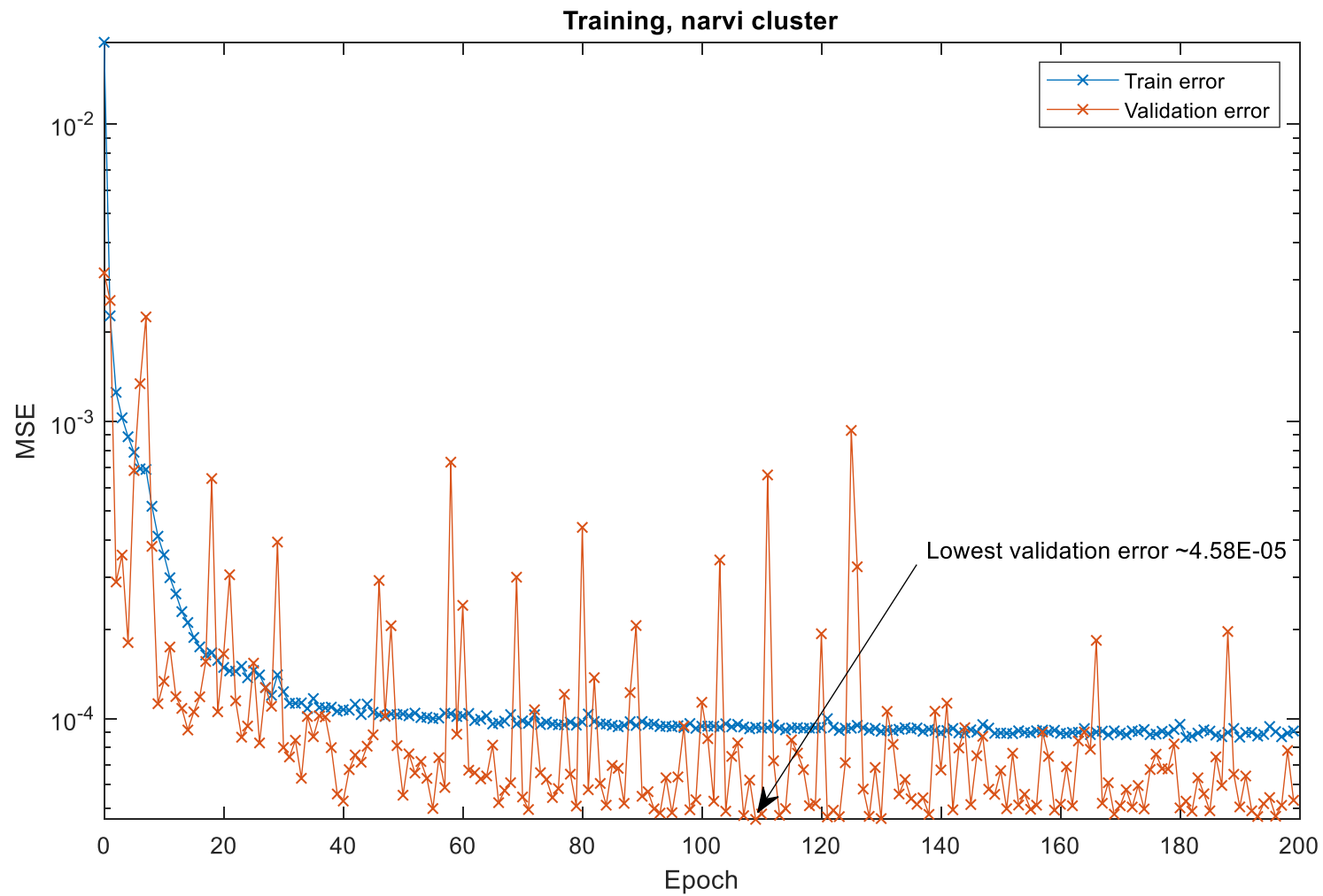
```
35 ### SETTINGS / HYPERPARAMETERS
36
37 #The number of entries in each timestep of input, e.g. data from different sensors
38 input_dim = 6
39
40 #The dimension of the predicted variable
41 output_dim = 1
42
43 #number of neurons per lstm layer
44 LSTM_units = 200
45
46 #Additional model / training hyperparameters
47 dropout_rate = 0.2
48 test_ratio = 0.05
49 batch_size = 32
50 max_epochs = 50
51
52 #static normalizing factors for input variables
53 min_x = np.array(
54     [0,
55      60.415733,
56      22.111310,
57      0.2000000,
58      0,
59      -39.88021])
60 max_x = np.array(
61     [1792,
62      60.520895,
63      22.341193,
64      57.200000,
65      255.99600,
66      51.320079])
67
68 #Set random seed for reproducible train/test split
69 random_seed = 42
```

- Min/max normalisointi input-datalle → kaikki muuttujat (0, 1) välille
- Koulutuksen aikana sekvenssien pituus jokaisen minibatchin sisällä pitää olla vakio (ainakin Kerasin implementaatioissa)
- Zero-padding, täytetään sekvenssejä nolilla tarpeen mukaan pisimmän batchissa olevan sekvenssin mukaan
- Sekvenssit lajiteltuna pituuden mukaan → mahd. vähäinen paddingin tarve
- Minibatch jaottelu lukittuu jo ennen koulutusta – käytännön merkitys?

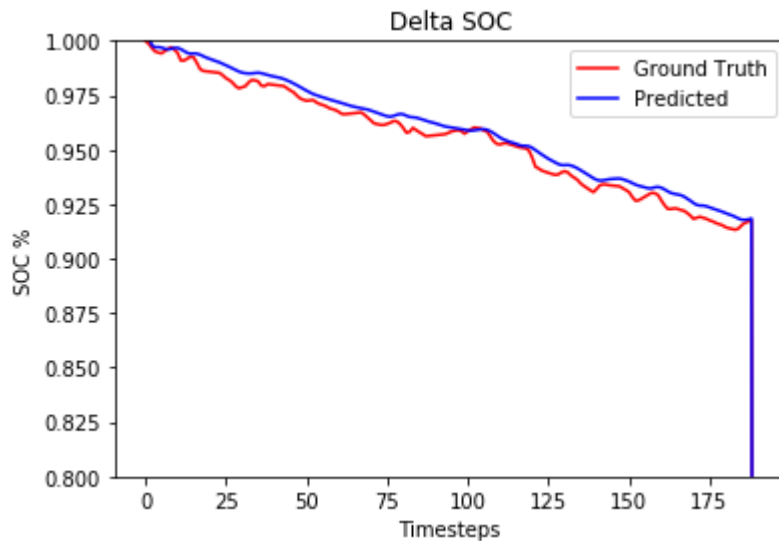


Predictive EBUS soc model with LSTM - Run Diary

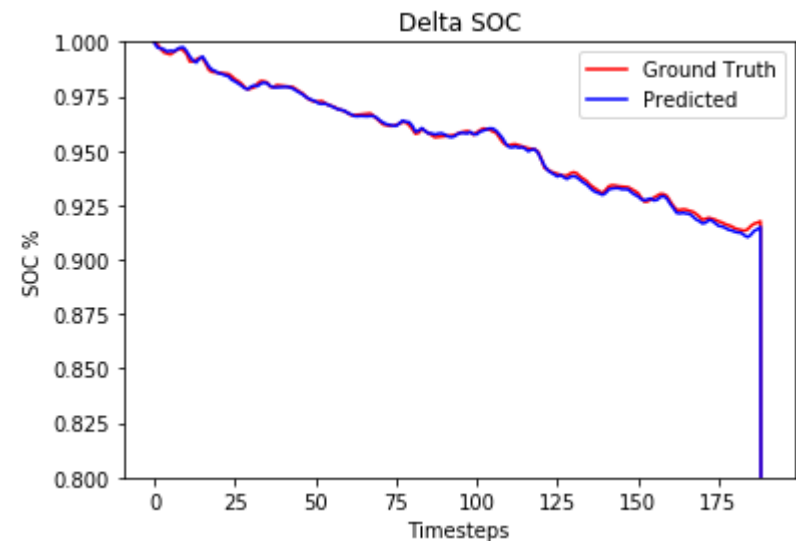
Run #	Machine	Start Time	Duration (minutes)	Actual epochs	max epochs	LSTM_units	MSE - Train	MSE - Validate
1	SEPLT1528	3.5.2018	27	10	10	50	2,48E-04	1,44E-04
2	Panu-PC	3.5.2018	122	10	10	200	4,18E-04	1,23E-04
3	Panu-PC	3.5.2018	666	50	50	200	1,02E-04	6,18E-05
4	NARVI	11.5.2018	~1 day	200	200	200	9,09E-04	5,31E-05



- Yleiskuva – Mallin hyperparametrien optimointi (lähinnä kapasiteetin kasvattaminen ja koulutusajan pidentäminen) auttoi mallia havaitsemaan suurten linjojen lisäksi pienempiä nyansseja ja dynamiikkaa



Pseudosatunnainen testinäyte
baseline-mallilla (ei optimointia)



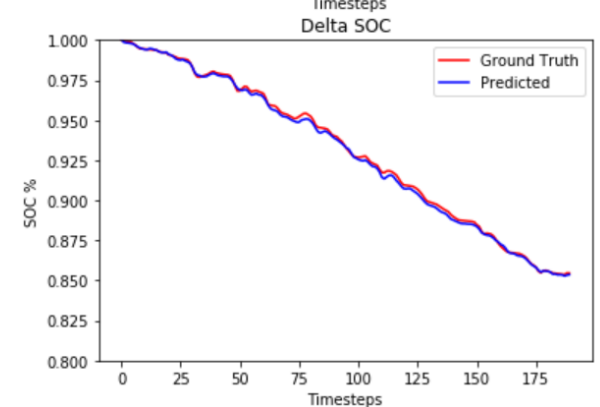
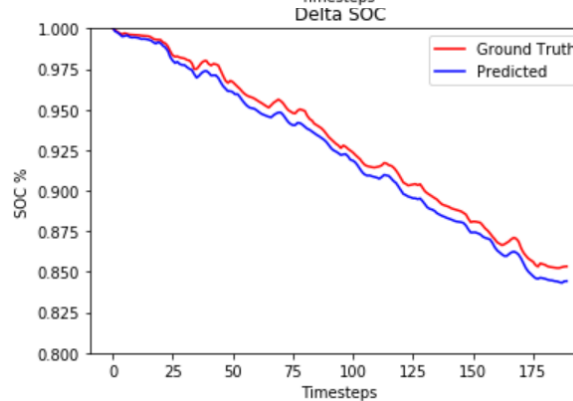
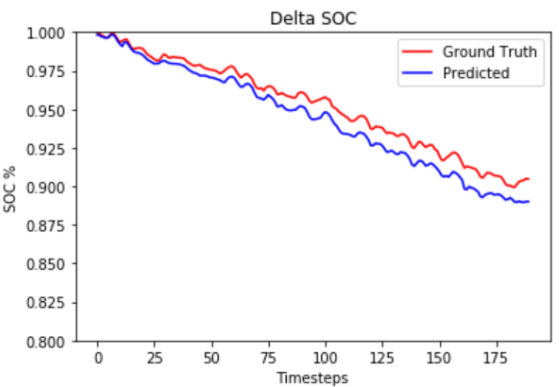
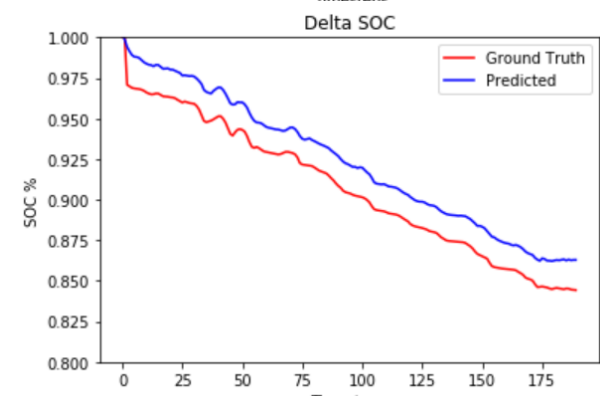
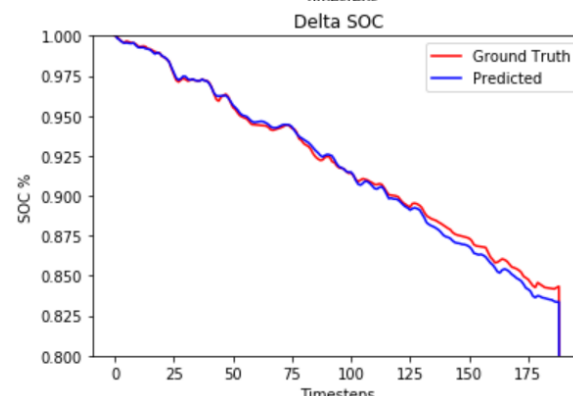
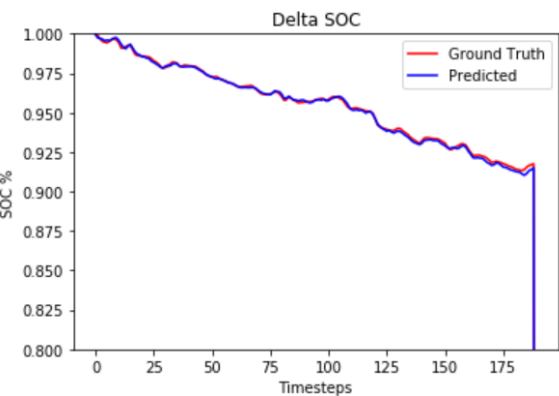
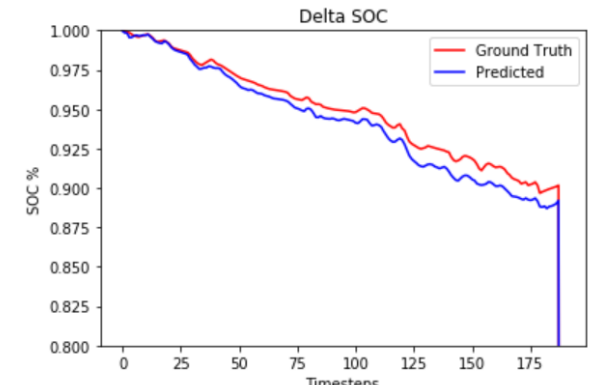
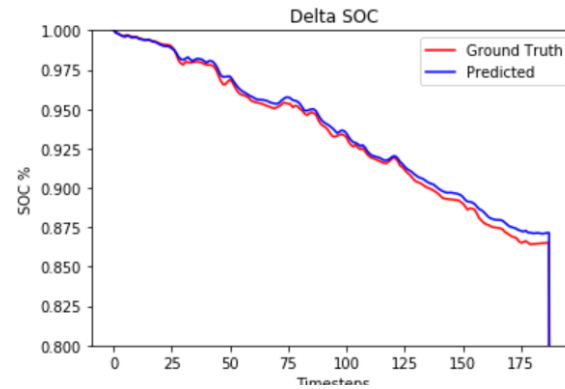
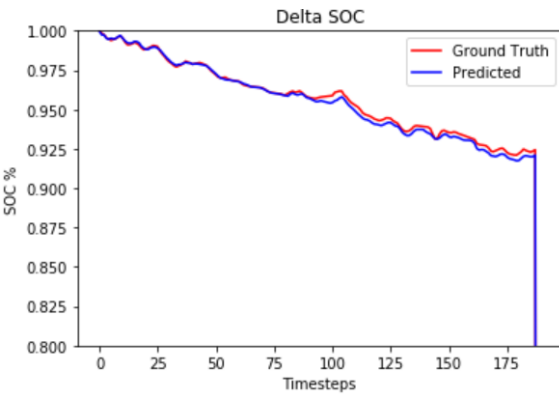
Sama näyte "optimoidulla" mallilla

Lisää tuloksia...



TURKU AMK

TURKU UNIVERSITY OF
APPLIED SCIENCES



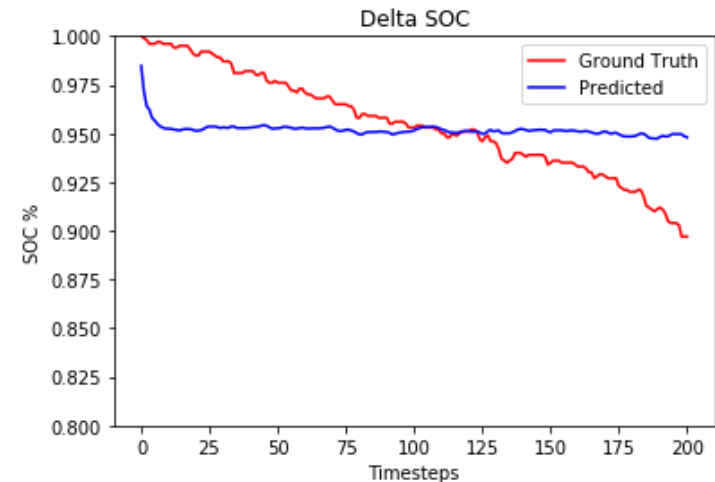
Miljoonan dollarin kysymys

Yleistyykö malli muille reiteille, kuin
millä se on koulutettu?



Vastaus: ei vielä

- Koulutettuun malliin syötettiin Helsingistä saatua sähköbussien ajodataa – huonoin tuloksin
- Ilmeisenä ongelmana koulutusaineiston sidonnaisuus tiettyyn absoluuttiseen geolokaatiojakaumaan (Turun linja 1)
- Kiinnostavaa on siirtymä ajan funktiona, ei niinkään absoluuttinen paikka – tämä tulisi verkon oppia, jotta yleistävyydestä muille reiteille voidaan puhua
- (Lisä)normalisointi?



- Aikasarja-regression tekeminen LSTM-verkolla on mahdollista
- Pienehkö malli (n. 1 000 000 parametria koulutettavissa) jopa kuluttajatasen PC-tietokoneella (kolmas koulutus, jolla päästiin jo suht hyviin tuloksiin kesti 11 tuntia)
- Tulokset rohkaisevia, lisäämällä muuttujia, hankkimalla lisää dataa / parempaa dataa ja säätämällä mallin hyperparametreja voitaisiin pyrkiä vielä parempaan
- NARVI laskentaklusterin käyttö, vaikei se tässä tapauksessa olisi ollut välttämätöntä, vapauttaa henkilökohtaisen työpöytäkoneen resursseja muuhun käyttöön ja lisäksi harjoitustyön yhteydessä opetuksellista funktiota ei voida väheksyä
- Malli ei tällaisenaan ole yleistettävissä reiteille, jotka eivät ole riittävästi edustettuna koulutusaineistossa – tämä saattaa olla toteutettavissa, mutta vaatii lisätutkimuksia
- Kriittistä tarkastelua vaatinee myös kysymys, onko syväoppimisen käyttö ylipäättään perusteltua - ts. ratkeaisiko ongelma myös perinteisin autoregressio yms. ”keveämpien” aikasarjatekniikoiden avulla?

Mahdollisia kaupallisia sovelluksia

- Ekstrapolointi tulevaisuuteen
 - Koulutetaan malli ennustamaan, miten varaustila kehittyy ajanhetkestä T eteenpäin, kun käytettävissä on historiallista tietoa
 - Esim. ennustetaan mikä on akun varaustila, kun bussi saapuu latauspisteelle → Hyödyllinen tieto latausprosessin optimoinnissa
- Joukkoliikennesuunnittelu
 - Esim. henkilöautolla ajetaan suunniteltu reitti läpi bussiliikennettä simuloiden, logataan nopeuskäyrät ja syötetään esikoulutettuun malliin → Mallista saadaan ulos arvioitu sähkönkulutus ko. reitillä → Voidaan käyttää aikataulutuksen ja latausjärjestelmien mitoitukseen
 - Vaatii koulutusaineiston muuttamista paikan suhteen absoluuttisesta suhteelliseksi (yksi ”lisä” normalisointikerros)
- Kuljettajakoulutus
 - Käytetään mallia generatiivisessa moodissa, jolloin kuljettajille voidaan näyttää reaaliaikaisena simulaatioina, miten ajokäyttäytyminen sekä olosuhdevaihtelu vaikuttaa kulutukseen
- Muut kuin sähköbussijärjestelmät, prediktiivinen kunnossapito
 - Esim. Teollisuudessa tai merenkulussa voidaan ennustaa kriittisen komponentin jäljellä oleva elinaika ja ryhtyä ajoissa toimenpiteisiin, jolloin välttyään kalliilta seisokkiajalta

TURKU AMK

TURKU UNIVERSITY OF
APPLIED SCIENCES



Kiitos!

https://github.com/donkkis/TripPredictor_2/