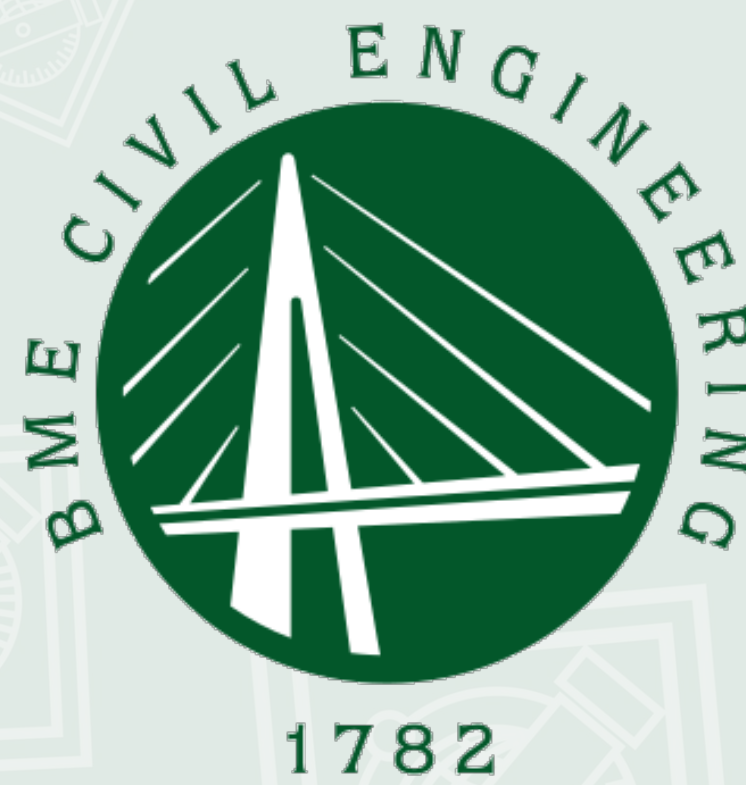




Idősorok feldolgozása mély tanulással

Siki Zoltán



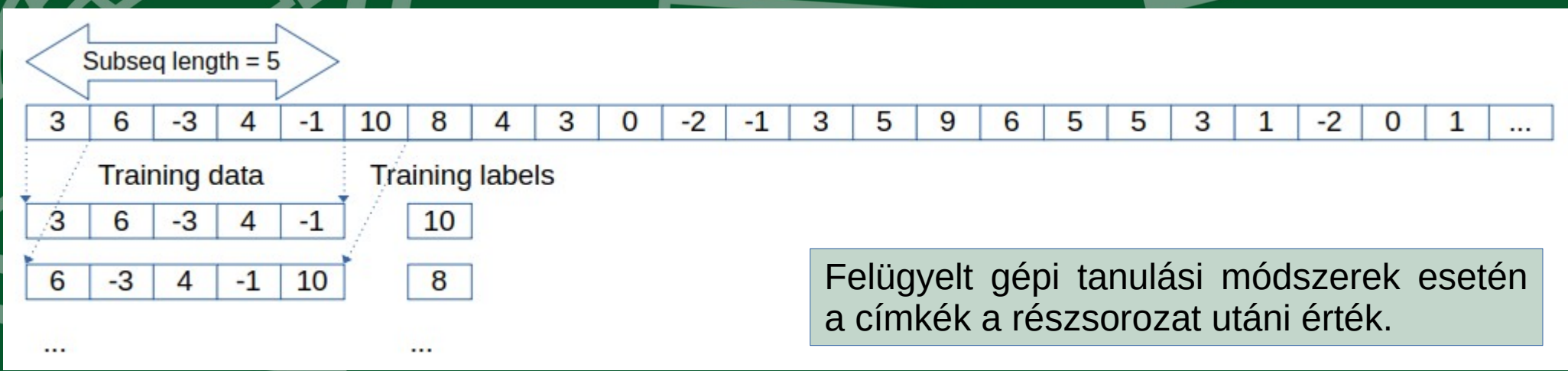
MI Specialista

Az autonóm környezetekben a szenzortechnológia gyors fejlődése hatalmas mennyiségű idősoros adat generálásához vezetett, ami szükségessé tette az automatizált módszerek fejlesztését a hatékony feldolgozás érdekében. A mesterséges intelligencia (MI) jelentős fejlődésével, különösen a beszéd, a természetes nyelvi feldolgozás és a generatív MI terén, új mély tanulási technikák jelentek meg az idősoros adatok kezelésére és előrejelzésére.

Azonban az ezekben a rendszerekben gyakran használt alacsony költségű szenzorok gyakran zajos adatokat állítanak elő, ami további kihívásokat jelent az elemzés során. Ez a poszter a speciális mesterséges neurális hálózatok, köztük az egydimenziós konvolúciós neurális hálózatok (CNN), a hosszú rövid távú memóriájú (LSTM) hálózatok, a rekurrens neurális hálózatok (RNN) és a kapuzott rekurrens egységek (GRU) integrációját vizsgálja mind az egyváltozós, mind a többváltozós idősoros elemzéshez. Az adatfeldolgozásban a Python domináns programozási nyelvű vált, amelyet a nyílt forráskódú könyvtárak, például a Keras, a TensorFlow, a PyTorch és a Scikit-Learn robusztus ökoszisztémája támogat. Ezek az eszközök komplex alkalmazásokat tesznek lehetővé, beleértve a görbeillesztést, a regressziót, az osztályozást, az előrejelzést, a szegmentálást, a klaszterezést és az anomáliadetektálást. Ez a poszter kiemeli a Python és a Keras képességeit az idősoros adatok feldolgozásában, és bemutatja azok hatékonyságát különféle gyakorlati példákon keresztül.

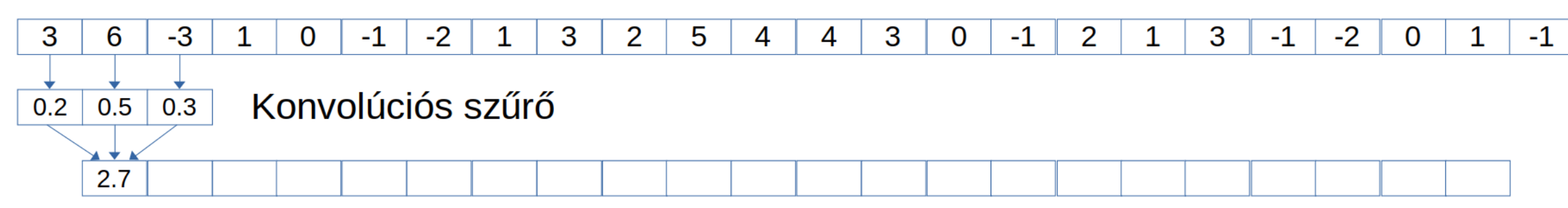
Adatelőkészítés

- Időben egyenletesen eloszló adatok
- Hiányzó adatok feltöltése
- Normalizáció (több változós idősorokhoz)
- Adatsorok felbontása rövidebb, egyforma hosszúságú részekre



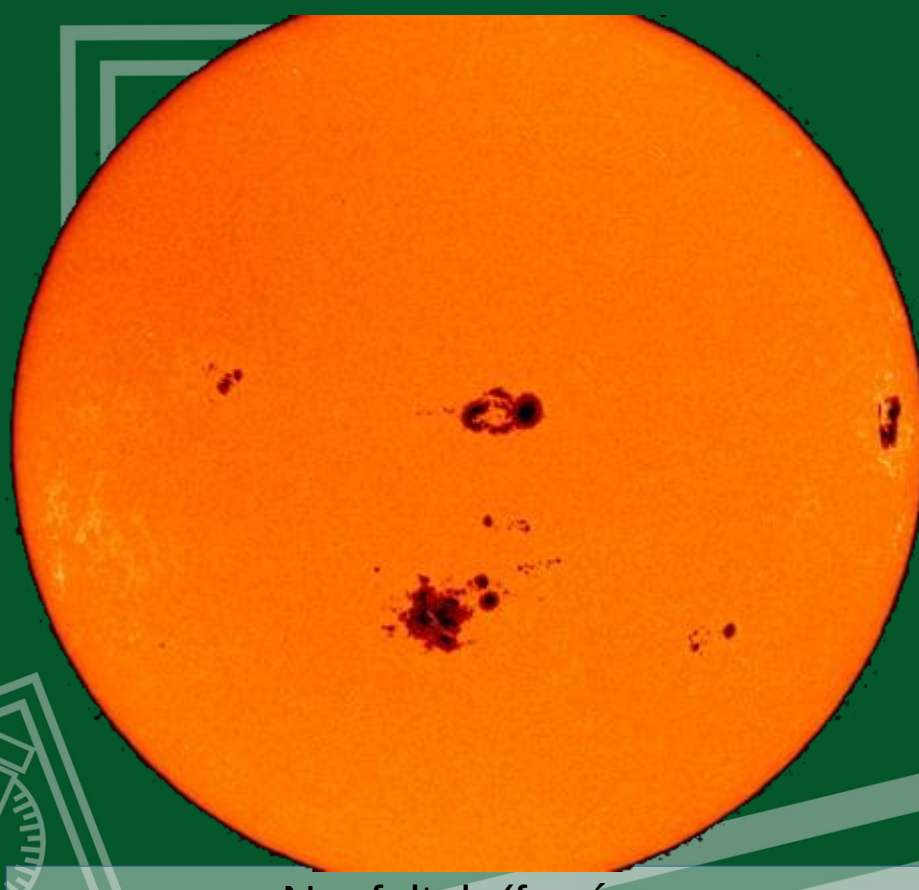
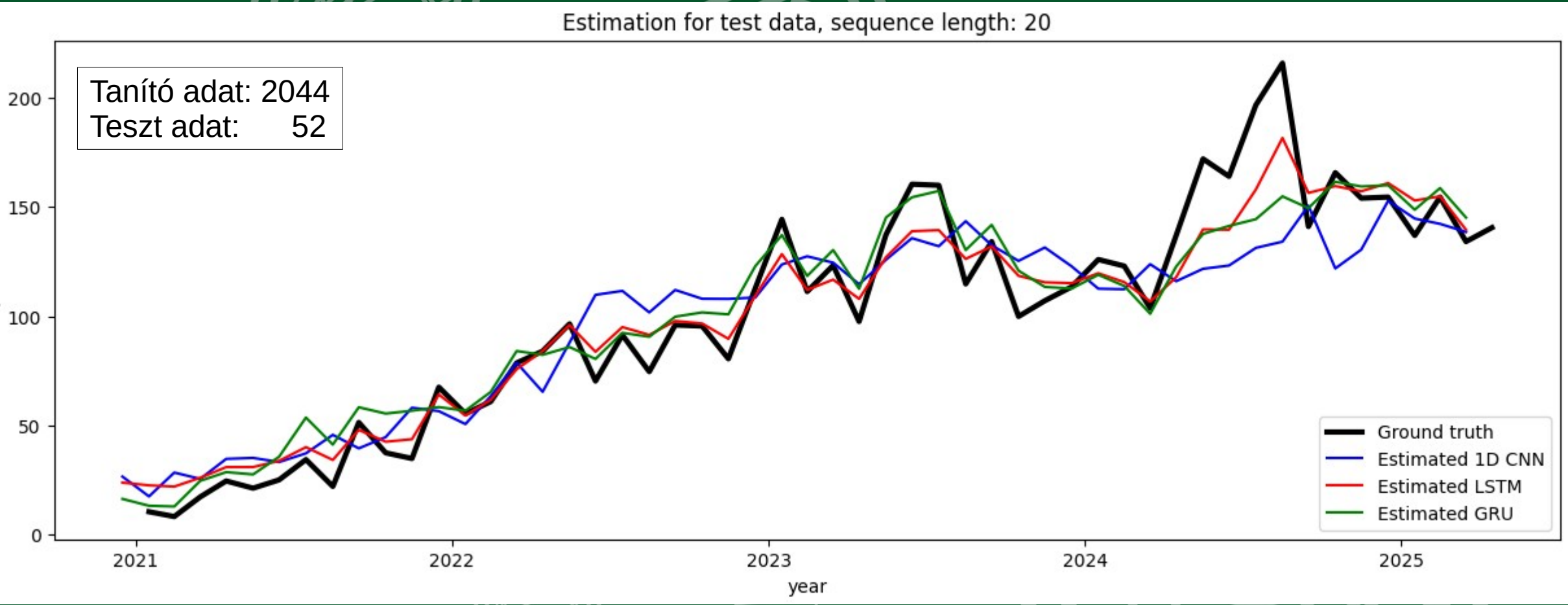
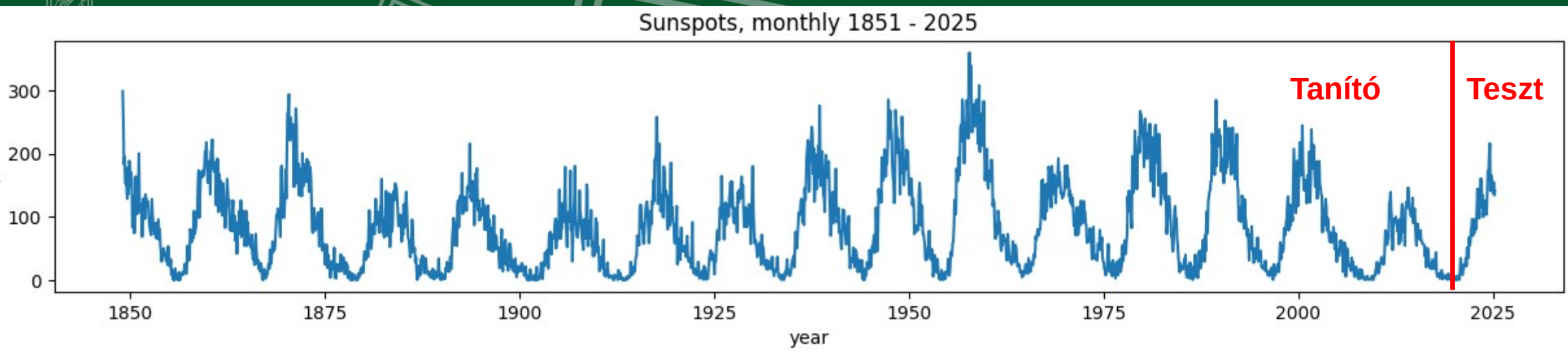
1D CNN

Az egydimenziós konvolúciós neurális hálózati rétegek képesek lokális mintázatokat detektálni az idősoros adatokban, és felhasználhatók előrejelzésre. Kerneleket (szűrővektorokat) mozgatnak a sorozatok mentén.

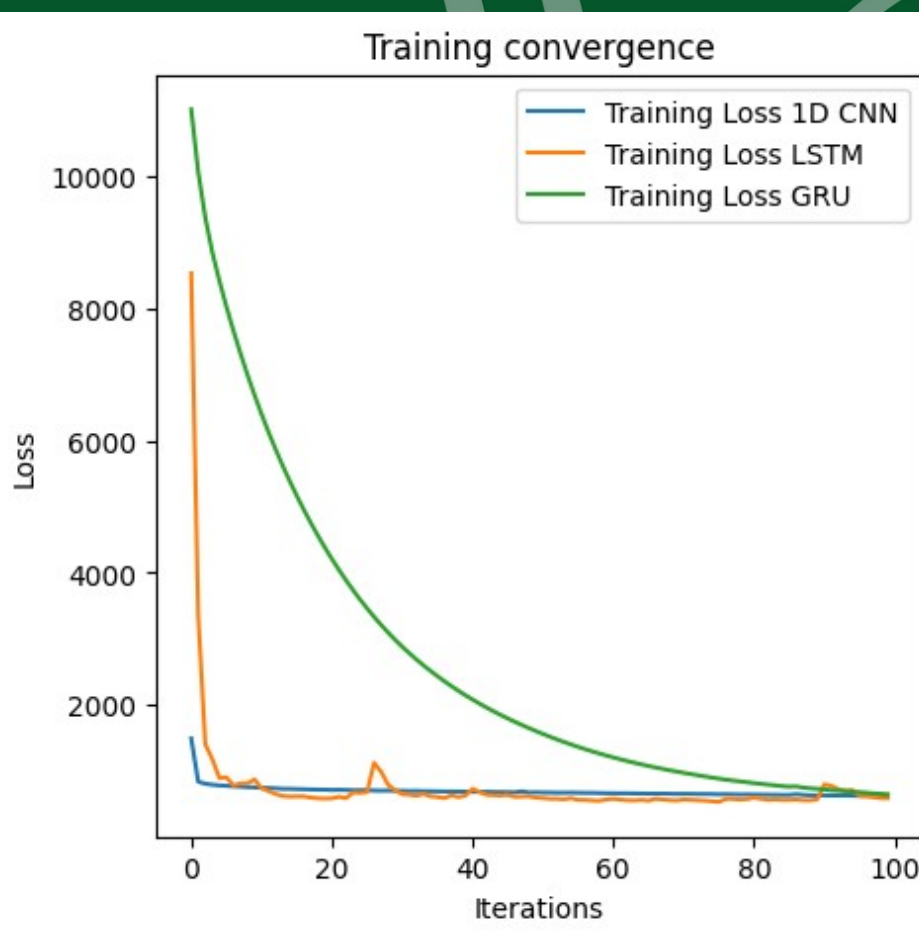


Napfolt adatok

A napfoltok számát 1700 óta regisztrálják, és letölthetők a Belga Királyi Obszervatóriumból (<https://www.sidc.be/SILSO/datafiles>). A különböző modellekben az idősoros előrejelzéshez havi adatokat használtam. 1850-től nincsenek hiányzó értékek, ezért az 1850-2025 közötti intervallummal dolgoztam.



Napfoltok (forrás: <https://www.weather.gov/fsd/sunspots>)



Modellek

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 19, 64)	192
max_pooling1d (MaxPooling1D)	(None, 9, 64)	0
flatten (Flatten)	(None, 576)	0
dense_3 (Dense)	(None, 50)	28,850
dense_4 (Dense)	(None, 1)	51
Total params: 29,093 (113.64 KB)		MSE: 622.5
Trainable params: 29,093 (113.64 KB)		MAE: 17.7
Non-trainable params: 0 (0.00 B)		

1D CNN

Layer (type)	Output Shape	Param #
Input_Layer (InputLayer)	(None, 20, 1)	0
lstm (LSTM)	(None, 50)	10,480
dense (Dense)	(None, 1)	51
Total params: 10,451 (40.82 KB)		MSE: 514.9
Trainable params: 10,451 (40.82 KB)		MAE: 17.5
Non-trainable params: 0 (0.00 B)		

LSTM

Layer (type)	Output Shape	Param #
input_layer_2 (InputLayer)	(None, 20, 1)	0
gru_1 (GRU)	(None, 50)	7,950
dense_2 (Dense)	(None, 1)	51
Total params: 8,001 (31.25 KB)		MSE: 460.4
Trainable params: 8,001 (31.25 KB)		MAE: 16.9
Non-trainable params: 0 (0.00 B)		

GRU

Anomália detektálás autoenkóderrel

Az autoenkóderek egyfajta felügyelet nélküli neurális hálózatok, melyeket a gépi tanulásban használnak dimenziócsökkentésre, jellemzőtanulásra és adattómörítésre. Úgy működnek, hogy megtanulják a bemeneti adatok mintázatát, majd minimális veszteséggel rekonstruálják (dekódolják). Anomáliaészlelés során egy normál (hibátlan) adatsort használnak a modell betanításához, majd a modellt alkalmazzák a tesztelt adatsorokra.

Layer (type)	Output Shape	Param #
conv1d_2 (Conv1D)	(None, 180, 32)	256
dropout_2 (Dropout)	(None, 180, 32)	0
conv1d_3 (Conv1D)	(None, 90, 16)	3,600
conv1d transpose 3 (Conv1DTranspose)	(None, 180, 16)	1,808
dropout_3 (Dropout)	(None, 180, 16)	0
conv1d transpose 4 (Conv1DTranspose)	(None, 360, 32)	3,616
conv1d transpose 5 (Conv1DTranspose)	(None, 360, 1)	225
Total params: 9,505 (37.13 KB)		
Trainable params: 9,505 (37.13 KB)		
Non-trainable params: 0 (0.00 B)		

This research was supported by the Ministry of Culture and Innovation and the National Research, Development and Innovation Office within the Cooperative Technologies National Laboratory of Hungary (grant No. 2022-2.1.1-NL-2022-00012).

