
	Politechnika Bydgoska im. J. J. Śniadeckich Wydział Telekomunikacji, Informatyki i Elektrotechniki <b>Zakład Systemów Teleinformatycznych</b>		
<b>Przedmiot</b>	ARiSC		
<b>Prowadzący</b>	prof. dr hab. inż. prof. PBŚ Piotr Cofta		
<b>Temat</b>	<i>Project Raport 2</i>		
<b>Student</b>	Cezary Tytko		
<b>Ocena</b>		<b>Data oddania spr.</b>	

## Etap 2. – przewidywanie wartości

Należałoby zacząć od wyznaczenia celu, tzn. co chcemy przewidywać i o ile w przód, jest to podstawowe założenie dla problemu biznesowego, które nie zostało jasno sprecyzowane w zadaniu. Mamy do dyspozycji 3 parametry:  $V_1$ ,  $F$ ,  $THDV_1$ , na początek odrzuciłbym  $THDV_1$ , gdyż nie widzę sensu przewidywania tej wartości (procent harmonicznych w napięciu na pierwszej), jest to raczej parametr mogący pomóc w analizie i przewidywaniu parametru  $V_1$ , sądzę że dwa pozostałe parametry można by przewidywać z praktycznego punktu widzenia i znalazłoby to swoje zastosowanie, postanowiłem jednak skupić się na  $V_1$  i przewidywać napięcie.

Powinniśmy się zastanowić, czy istnieją przesłanki do tego, że przewidywanie napięcia jest możliwe, bazując na danych historycznych. Na aktualną wartość napięcia, mają wpływ dwa dwie podstawowe grupy urządzeń, generatory napięcia i jego odbiorniki, mają do dyspozycji dodatkowe dane takie jak stan aktywności urządzeń podłączonych do sieci z pewnością można by przewidywać zmiany napięcia z tym związane, jednak my mamy do dyspozycji tylko dane historyczne z 3 parametrów stanu napięcia i częstotliwości w punkcie pomiarowym.

Spodziewałbym się że możemy wykryć pewne zależności krótko terminowe, związane z włączaniem i wyłączeniem urządzeń podłączanych do sieci, które mogą generować charakterystyczne zmiany napięcia, jak np. silniki elektryczne, kondensatory, zatem będę starał się przewidzieć napięcie co najwyżej kilka sek. w przód. Chciałbym zaznaczyć że przeglądając dane zauważyłem również znaczne zmiany napięcia ( $\pm 2V$ ), które występują przez długi czas, kilka godzin, domyślam się że będą to zmiany związane z cyklem dobowym i powiązanym z nim stanem aktywności urządzeń jak i całej sieci energetycznej, w takim przypadku można by prowadzić prognozę długo terminowa, będzie ona jednak oparta o czas, a nie o zmierzone parametry, na których mam się opierać zgodnie z poleceniem zadania.

Predykcji będę chciał dokonywać za pomocą modelu sieci neuronowych, które dostaną fragment szeregu czasowego, a wartością do przewidzenia, będzie napięcie jakie wystąpi za wskazaną liczbą cykli pomiarowych (sekund). Takie dane możemy przetwarzać za pomocą sieci rekurencyjnych, będących pierwszym co przychodzi mi do głowy dla takiego problemu, jak również za pomocą sieci splotowych 1D. Jako funkcję starty wykorzystam MSE, czyli błąd średnio kwadratowy.

Dokładna implementacja sieci znajduje się w kodzie załączonym do sprawozdania.

Trzeba ustalić także sposób oceny modelu, jako że jest to regresja wybrałem MAE i  $R^2$ , jednak jak się okazało wyniki są przekłamane, ponieważ średnia z wartość wszystkich pomiarów używana jako punkt odniesienia w przypadku  $R^2$  ( $R^2 = 0$  model taki sam jakby zwracał zawsze średnią z pomiarów (wszystkich przewidywanych pomiarów  $V_1$ )) dlatego wartości te były zbliżone do 1, modelu idealnego, jednak nie jest to dla nas miarodajne w przypadku parametru który waha się w zakresie 226-234 V, ale zmiany do skrajnych wartości następuje bardzo powoli,  $R^2 > 0$  mówi nam że nasz model jest lepszy od modelu który zwracałby zawsze 230V, a nie jest to dla nas satysfakcjonujące w przypadku przewidywania krótko terminowego, MAE po standaryzacji również nie mówi nam za dużo, możemy powiedzieć jedynie że im mniej tym lepiej. Postanowiłem więc znaleźć punkt odniesienia dla MAE i  $R^2$ . Takim punktem okazał się model który, który jako wartość przewidywaną zwraca ostatnią wartość z analizowanego fragmentu szeregu czasowego, czyli model który mówi

że od ostatniego czasu nic się nie zmieniło (należy zauważyć że dla takiego modelu nie ma wpływu długość analizowanego fragmentu), jedynie czas od którego chcemy przewidzieć wartość), dla tak prostego modelu, przewidującego napięcie 5 sek. w przód, uzyskałem spektakularny wynik:

```
Train MAE: 0.07421860098838806, R2: 0.9883258348765861
Test MAE: 0.06672847270965576, R2: 0.9904608031853894
```

Ten wynik powinien być dla nas punktem odniesienia, i jeżeli uda nam się uzyskać wynik lepszy od tego, możemy mówić już o małym sukcesie.

Dla sieci LESM wykorzystałem fragmenty sekwencji z 30 sekund

```
1. class LSTMRegressor(nn.Module):
2.     def __init__(self):
3.         super(LSTMRegressor, self).__init__()
4.         self.lstm = nn.LSTM(3, 64, batch_first=True)
5.         self.fc = nn.Linear(64, 1)
6.
7.     def forward(self, x):
8.         h0 = torch.zeros(1, x.size(0), 64).to(x.device)
9.         c0 = torch.zeros(1, x.size(0), 64).to(x.device)
10.        out, _ = self.lstm(x, (h0, c0))
11.        out = self.fc(out[:, -1, :])
12.        return out
13.
```

Uzyskałem wynik:

Po 10 Epokach:

Train Loss: 0.0097, Train MEA: 0.1078, Train R<sup>2</sup>: 0.9806

Test Loss: 0.0171, Test MEA: 0.1031, Test R<sup>2</sup>: 0.9829

Po 20 Epokach:

Train Loss: 0.0090, Train MEA: 0.0889, Train R<sup>2</sup>: 0.9864

Test Loss: 0.0116, Test MEA: 0.0833, Test R<sup>2</sup>: 0.9884

Po 30 Epokach:

Train Loss: 0.0089, Train MEA: 0.0878, Train R<sup>2</sup>: 0.9867

Test Loss: 0.0115, Test MEA: 0.0829, Test R<sup>2</sup>: 0.9885

Jest to wynik bardzo zbliżony do modelu odniesieni, natomiast nie jest on od niego lepszy.

Dla Sieci splotowej wykorzystałem sekwencję długości 30 i 100 pomiarów, znacznie więcej niż dla LSTM, nie spodziewałbym się poprawy jakości ze względu na aż taką długość sekwencji, natomiast dla Sieci splotowych w odróżnieniu od LSTM nie spowoduje to tak znacznego zwiększenia czasu potrzebnego na obliczenia.

```
1. class CNN_TS(nn.Module):
2.     def __init__(self, input_channels):
3.         super(CNN_TS, self).__init__()
4.         self.conv1 = nn.Conv1d(input_channels=input_channels, out_channels=64, kernel_size=3)
5.         self.conv2 = nn.Conv1d(input_channels=64, out_channels=128, kernel_size=3)
6.         self.conv3 = nn.Conv1d(input_channels=128, out_channels=256, kernel_size=3)
7.         self.fc1 = nn.Linear(256 * 10, 128)
8.         self.fc2 = nn.Linear(128, 1)
9.         # self.dropout = nn.Dropout(0.2)
10.
11.     def forward(self, x):
12.         x = F.relu(self.conv1(x.permute(0, 2, 1)))
13.         x = F.max_pool1d(x, 2)
14.         x = F.relu(self.conv2(x))
15.         x = F.max_pool1d(x, 2)
16.         x = F.relu(self.conv3(x))
17.         x = F.max_pool1d(x, 2)
18.         x = torch.flatten(x, 1)
19.         x = F.relu(self.fc1(x))
20.         # x = self.dropout(x)
21.         x = self.fc2(x)
22.         return x
23.
```

Dla sekwencji 100 uzyskałem wynik:

Po 10 Epokach:

Train Loss: 0.0199, Train MAE: 0.1021, Train R2: 0.9801

Val Loss: 0.0985, Val MAE: 0.2417, Val R2: 0.9015

Po 20 Epokach:

Train Loss: 0.0182, Train MAE: 0.0983, Train R2: 0.9818

Val Loss: 0.0777, Val MAE: 0.2156, Val R2: 0.9223

Po 30 Epokach:

Train Loss: 0.0173, Train MAE: 0.0956, Train R2: 0.9827

Val Loss: 0.0554, Val MAE: 0.1869, Val R2: 0.9446

Dla sekwencji 30 uzyskałem wynik:

Po 10 Epokach:

Train Loss: 0.0125, Train MAE: 0.0823, Train R2: 0.9875

Val Loss: 0.0805, Val MAE: 0.2213, Val R2: 0.9195

Po 20 Epokach:

Train Loss: 0.0118, Train MAE: 0.0793, Train R2: 0.9882

Val Loss: 0.0496, Val MAE: 0.1686, Val R2: 0.9504

Po 30 Epokach:

Train Loss: 0.0114, Train MAE: 0.0776, Train R2: 0.9886

Val Loss: 0.0441, Val MAE: 0.1617, Val R2: 0.9559

Wnioski:

Dla Sieci LSTM po treningu błąd względem wartości przewidywanej jest zbliżony do wartości uzyskanej dla modelu bazowego, natomiast nie jest on od niego lepszy, co nie świadczy dobrze, plusem jest natomiast brak występowania overfitingu, patrząc po samych metrykach dokładności i starty. Dla sieci splotowych wyniki są bardzo podobne, z tą różnicą że zwiększenie długości sekwencji nie poprawiło wyniku (a nawet jest on minimalnie gorszy), widoczny jest jednak efekt overfitingu, co sprawia że wynik dla testu jest zdecydowanie gorszy od prostego modelu zakładającego brak zmiany.

Spodziewałem się takiego rezultatu, analizując korelację i auto korelację nie zauważyłem przesłanek, które mogły wskazywać że na pewno musi się to udać, ale nigdy nie wiadomo co sieć neuronowa wymyśli, dlatego warto było spróbować, być może dla innych paramentów sieci i innych założeń przewidywania udało by się uzyskać nam coś lepszego niż zaproponowany, model bazowy. Uzyskując wyniki dla sieci zbliżone do modelu bazowego, wydaje mi się że proces uczenia sieci wykonałem najlepiej jak potrafię, ale może nie dać się uzyskać tymi metodami czegoś lepszego niż model bazowy, ale przynajmniej wyniki nie są gorsze, co mogło by świadczyć np. o błędzie w kodzie lub podejściu do problemu.

Podsumowując, dla założonego przez siebie problemu, czyli krótkoterminowego przewidywania napięcia w sieci, stwierdzam że z obecnymi parametrami ciężko jest wykonać to zadanie uzyskując wynik lepszy, niż stwierdzenia braku zmiany, który dawał najlepsze wyniki dla MAE i  $R^2$ .