A Simple Parameter Inference with Transformers

Lidiia Pylyp

31. prosince 2024

1 Úvod

Cílem tohoto projektu je předpověď parametrů amplitudy A_i a frekvence w_i periodické funkce $y_i(A_i, w_i; t) = A_i \sin(w_i t)$, přičemž vstupními daty jsou vektory $V_i(A_i, w_i)$ reprezentované 100 hodnotami.

Projekt se zaměřuje na:

- Optimalizaci hyperparametrů modelů,
- Analýzu a srovnání efektivity různých architektur transformátorů (Encoder-Only, Decoder-Only, Encoder-Decoder) pro predikci amplitudy a frekvence,
- Zkoumání vlivu Fourierovy transformace na kvalitu predikce,
- Identifikaci frekvenčních rozsahů, které model dokáže efektivně předpovědět.

Použití transformátorových modelů je motivováno jejich schopností efektivně zpracovávat sekvenční data a odhalovat složité nelineární závislosti. Fourierova transformace byla zahrnuta do experimentů s cílem zlepšit reprezentaci dat a usnadnit modelu extrakci informací o frekvenci.

2 Vstupní data

Vstupní data byla generována synteticky. Pro každou kombinaci amplitudy A a frekvence ω byl vytvořen vektor V_i o délce 100 vzorků, přičemž časová osa t byla omezena na interval $[0, 2\pi]$. Rozsah hodnot pro A a ω byl zvolen jako $[0.1, 10]$, což umožňuje pokrýt široké spektrum možných scénářů, od nízkých po vysoké frekvence.

Data byla před trénováním normalizována pomocí `MinMaxScaler`, což zajišťuje, že hodnoty vstupních vektorů a cílových parametrů budou omezeny na interval $[0, 1]$. Tento krok je zásadní pro zlepšení stability a rychlosti konvergence modelu.

Pro některé experimenty byla na vstupní data aplikována Fourierova transformace, která umožňuje extrahovat frekvenční složky signálu a potenciálně zlepšit predikci parametrů.

Předzpracovaná data byla rozdělena na trénovací a validační sadu v poměru 80:20. Normalizace byla aplikována samostatně na vstupní vektory V_i a cílové parametry A a ω.

3 Metody

Pro řešení úlohy byly použity následující metody:

- **Optimalizace hyperparametrů:** Počáteční model Encoder-Only byl vytvořen s parametry $A_{\text{range}} = (0.1, 10)$, $\omega_{\text{range}} = (0.1, 10)$, $t_{\text{points}} = 100$. Optimalizace hyperparametrů byla provedena ve dvou iteracích ($N = 1000$, $N = 5000$) pomocí frameworku **Optuna**. Po optimalizaci byly nalezeny následující hodnoty pro Encoder Transformer:
  - $d_{\text{model}} = 128$
  - Počet hlav: 4
  - Počet vrstev: 4
  - Dimenze feedforward vrstvy: 256
  - Dropout: 0.2
  - Batch size: 32
  - Learning rate: 0.0001

  Byla také testována efektivita různých aktivačních funkcí (viz Tabulka 1). Funkce `silu` vykázala nejlepší výsledky s minimální chybou (MSE a MAE) a lepší predikcí jak amplitudy, tak frekvence.

Tabulka 1: Výsledky modelu s různými aktivačními funkcemi

Activation Function	**MSE A**	**MSE w**	**Total MSE**	**MAE A**	**MAE w**
relu	0.158452	0.255883	0.414335	0.327034	0.345411
leaky_relu	0.237921	0.487192	0.725113	0.415252	0.419455
gelu	0.120631	0.437550	0.558181	0.257524	0.576794
silu	0.043539	0.147587	0.191126	0.163827	0.254661
tanh	0.308856	0.427381	0.736237	0.485203	0.456884
elu	0.157638	0.392405	0.550043	0.327979	0.522544

- **Porovnání architektur:** Byly testovány různé architektury transformátorů (encoder-only, decoder-only, encoder-decoder), jejichž výkon byl vyhodnocen pomocí metrik MSE, MAE a R2 Score (viz Tabulka 2). Výpočet R2 Score byl proveden podle vzorce:

$$R^2 = 1 - \frac{\sum(\text{true\_values} - \text{predictions})^2}{\sum(\text{true\_values} - \text{mean\_true\_values})^2}.$$

Tabulka 2: Porovnání architektur transformátorů

Model	**MSE A**	**MSE w**	**MAE A**	**MAE w**	**R2**
Encoder	0.076939	0.265539	0.207789	0.337926	0.979101
Decoder	0.028098	0.079430	0.109377	0.140570	0.993438
Encoder-Decoder	0.045229	0.110106	0.138678	0.190743	0.990521

Decoder-Only Transformer dosáhl nejlepších výsledků ve všech metrikách, což naznačuje jeho vyšší efektivitu při predikci parametrů. Encoder-Only Transformer měl nejhorší výkon, zejména u predikce frekvence, pravděpodobně kvůli omezené schopnosti této architektury zachytit složité frekvenční vzory.

- **Fourierova transformace:** Fourierova transformace obecně zlepšila predikci frekvence, zatímco predikce amplitudy byla přesnější bez použití této metody (viz Tabulka 3).

Tabulka 3: Výsledky s Fourierovou transformací a bez ní.

Method	MSE A	MSE w	MAE A	MAE w	R2
Without Fourier	0.074051	0.083018	0.197286	0.186467	0.990769
With Fourier	0.101256	0.040785	0.245737	0.125910	0.991652
Without Fourier	0.148505	0.099650	0.286074	0.201202	0.985415
With Fourier	0.091786	0.112735	0.240786	0.197463	0.987980

Pro zlepšení predikce byla testována Fourierova transformace ve 3 variantách (viz Tabulka 4).

Tabulka 4: Porovnání různých metod Fourierovy transformace.

Method	MSE A	MSE F	MAE A	MAE F	R2
Simple Fourier	0.081999	0.155247	0.197955	0.273719	0.986057
Log Fourier	0.063269	0.058093	0.173678	0.178337	0.992867
Hann Window Fourier	0.087342	0.081492	0.169505	0.195399	0.990077

Nejlepší výsledky byly dosaženy metodou Log Fourier, která minimalizovala chyby (MSE, MAE) a dosáhla nejvyšší hodnoty R2 Score. Log Fourier provádí aplikaci Fourierovy transformace na vstupní data pomocí funkce torch.fft.fft, následně vrací logaritmovanou hodnotu absolutní hodnoty výsledku transformace, čímž se získá stabilnější a lepší predikce pro data s vysokými frekvencemi.

Kód projektu je napsán v Python s pomocí PyTorch v Google Colab, je tady.

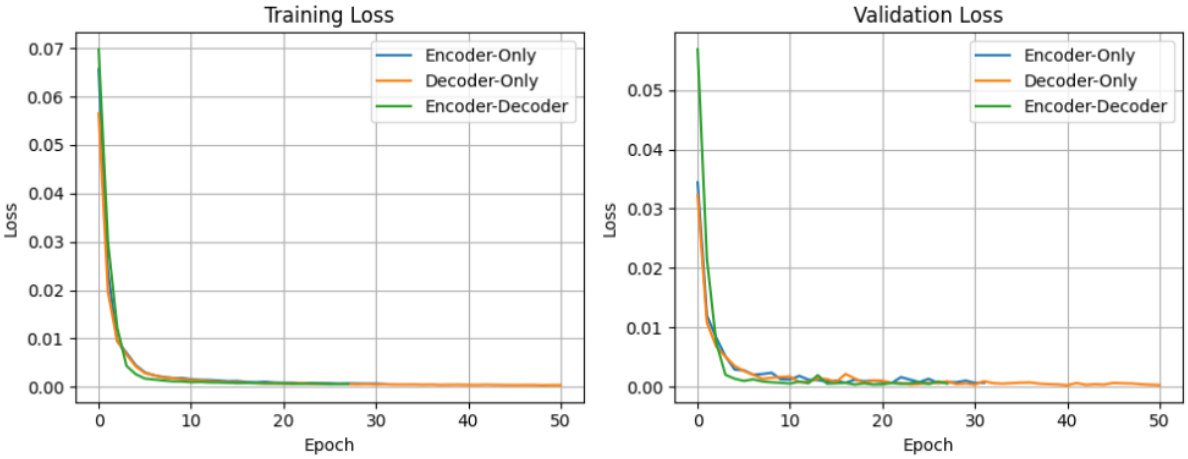
4 Výsledky

4.1 Modely s Fourierovou transformací

Tabulka 5 zobrazuje výsledky pro tři architektury transformátorů (Encoder-Only, Decoder-Only a Encoder-Decoder) s použitím Fourierovy transformace (N=10000). Nejlepších výsledků dosáhl model Decoder-Only, který vykazoval nižší hodnoty MSE a MAE a vyšší R2 skóre pro amplitudu i frekvenci. To naznačuje, že model Decoder-Only lépe zachycuje vzory ve signálech, zejména v sekvenčním charakteru dat. Model Encoder-Decoder dosáhl mírně horších výsledků, ale stále byl lepší než Encoder-Only, což naznačuje, že kombinace enkodéru a dekodéru může lépe zachytit složitější struktury než čistě enkodér.

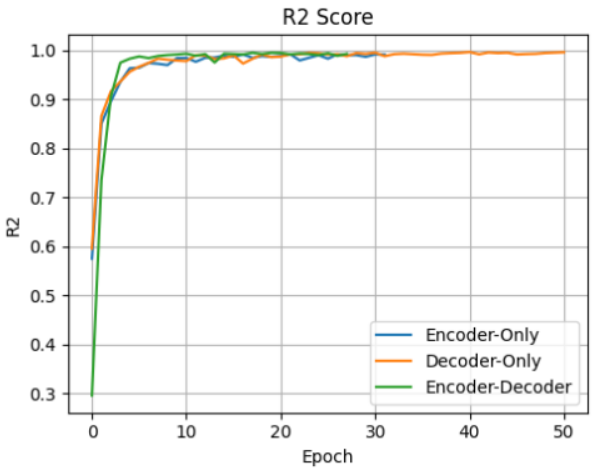
Tabulka 5: Výsledky pro konečné modely s Fourierovou transformací (N=10000).

Model	MSE A	MSE w	MAE A	MAE w	R2
Encoder-Only	0.0610	0.0580	0.1558	0.2020	0.9928
Decoder-Only	0.0417	0.0099	0.1260	0.0777	0.9969
Encoder-Decoder	0.0586	0.0164	0.1477	0.0880	0.9954



Obrázek 1: Průběh tréninkové a validační ztráty pro různé modely

Model Decoder-Only může být efektivnější v zachycování sekvenčních závislostí v datech, což je klíčové pro analýzu signálů. Fourierova transformace poskytla frekvenční charakteristiky, které model Decoder-Only dokázal efektivně využít pro lepší predikci.



Obrázek 2: R2 pro různé modely

4.2 Predikce v různých frekvenčních rozsazích

Tabulka 6 ukazuje výsledky pro tři architektury modelů v různých frekvenčních pásmech (N=5000). Zjištění ukazují na významné problémy při předpovědi vyšších frekvenčních složek, což naznačuje, že modely mají potíže se zvládnutím složitosti signálů v těchto pásmech.

- Nízké frekvence (0.01–0.1): Všechny modely vykazují nízké hodnoty MSE a MAE, ale hodnoty R^2 jsou záporné, což naznačuje přetrénování. Model Decoder-Only dosahuje nejlepších výsledků, což ukazuje jeho silné stránky při zpracování sekvenčních dat.
- Střední frekvence (0.1–10.0): Chyby se zvyšují, ale modely zůstávají efektivní. Model Decoder-Only nadále překonává ostatní, zejména pokud jde o zpracování složitějších

frekvenčních složek.

- Vysoké frekvence (10.0–1000.0): Chyby vzrůstají prudce, zejména mezi 100.0 a 1000.0 Hz, což ukazuje na neschopnost modelů účinně předpovědět vysokofrekvenční složky.

Tabulka 6: Výsledky pro různé modely v různých frekvenčních rozsazích A=1.

Model	**MSE w**	**MAE w**	**R2**
Frekvenční rozsah (0.01, 0.1)			
Encoder	0.000631	0.021592	-2.110562
Decoder	0.000946	0.026515	-0.624858
Encoder-Decoder	0.000709	0.023228	-0.476813
Frekvenční rozsah (0.1, 1.0)			
Encoder	0.069023	0.195465	-19.558853
Decoder	0.250356	0.414728	-10.614365
Encoder-Decoder	0.140780	0.291002	-10.262549
Frekvenční rozsah (0.1, 10)			
Encoder	1.553503	1.007744	0.811423
Decoder	1.815034	1.087288	0.823230
Encoder-Decoder	1.665024	1.038595	0.812812
Frekvenční rozsah (1.0, 10.0)			
Encoder	0.863925	0.772706	0.862912
Decoder	1.445779	0.936289	0.846817
Encoder-Decoder	1.308577	0.914377	0.856884
Frekvenční rozsah (10.0, 50.0)			
Encoder	212.463669	11.947613	0.614542
Decoder	252.318756	13.109768	0.549494
Encoder-Decoder	252.153687	13.697452	0.553136
Frekvenční rozsah (50.0, 100.0)			
Encoder	272.211060	12.958292	0.907378
Decoder	291.688995	14.811506	0.899772
Encoder-Decoder	243.030319	12.868630	0.918334
Frekvenční rozsah (100.0, 1000.0)			
Encoder	87440.945312	246.654404	0.595273
Decoder	116972.679688	279.757935	0.465887
Encoder-Decoder	90195.156250	253.202927	0.584181

Přes všechny pokusy, Decoder-Only model, který vykazuje nejlepší výsledky, stále nedokáže adekvátně zvládnout vysokofrekvenční složky signálu. Zdá se, že výkon modelu velmi závisí na parametru amplitudy a také na rozsahu parametrů pro generování dat, na kterých jsou modely trénovány. Proto byl tento experiment zopakován při A=(0.1, 10) v tom samém rozsahu, jaký byl použit pro tréninková data (Tabulka 7).

Klíčové pozorování

- Vliv rozsahu amplitudy (A): Rozšíření rozsahu amplitudy zlepšuje výkon v nízkofrekvenčních pásmech, ale má minimální vliv na vysokofrekvenční složky.

Tabulka 7: Výsledky pro různé modely v různých frekvenčních rozsazích A=(0.1, 10).

Model	MSE w	MAE w	R2
Frekvenční rozsah (0.01, 0.1)			
Encoder	0.000693	0.022745	-0.483278
Decoder	0.001307	0.029829	0.522197
Encoder-Decoder	0.000986	0.026405	0.547293
Frekvenční rozsah (0.1, 1.0)			
Encoder	0.067137	0.211147	-2.791920
Decoder	0.162788	0.316422	-0.151325
Encoder-Decoder	0.114648	0.267011	-1.194358
Frekvenční rozsah (0.1, 10)			
Encoder	1.200093	0.889999	-0.022545
Decoder	3.428635	1.474503	0.249401
Encoder-Decoder	2.092221	1.163417	0.200367
Frekvenční rozsah (1.0, 10.0)			
Encoder	0.974267	0.819214	-0.320020
Decoder	6.116010	2.033845	-0.021837
Encoder-Decoder	1.083877	0.834870	0.508552
Frekvenční rozsah (10.0, 50.0)			
Encoder	196.224335	11.594696	0.512040
Decoder	235.031540	12.999815	0.468524
Encoder-Decoder	196.505157	11.961046	0.522631
Frekvenční rozsah (50.0, 100.0)			
Encoder	238.967804	11.870941	0.908007
Decoder	325.517426	15.554658	0.878742
Encoder-Decoder	246.858856	13.116399	0.903227
Frekvenční rozsah (100.0, 1000.0)			
Encoder	91173.265625	253.530365	0.578492
Decoder	120296.179688	284.150818	0.456438
Encoder-Decoder	89494.609375	250.810242	0.591695

- Modely typu Decoder-Only se neustále osvědčují v nízkých a středních frekvencích, ale mají problémy s vysokými frekvencemi.
- Modely typu Encoder-Only vykazují větší stabilitu ve vysokofrekvenčních pásmech, ale celkově zůstávají méně efektivní.

4.3 Závěry:

Nízké frekvence (0.01–0.1): Výsledky jsou lepší než v předchozích experimentech, zejména pro Decoder-Only a Encoder-Decoder modely. Tento zlepšení může být způsobeno širším rozsahem amplitud, což umožňuje modelům lépe generalizovat data. Nicméně, záporné hodnoty R^2 naznačují, že modely mohou být přetrénovány na nízkých frekvencích, což vede k problému s generalizací na nová data.

Střední frekvence (0.1–10.0): Modely vykazují rostoucí chyby, ale stále se ukazují jako efektivní. Decoder-Only model stále vykazuje lepší výsledky, což ukazuje na jeho

silnou schopnost zpracovávat složitější signály. Omezená schopnost Encoder-Only modelu zpracovávat složité frekvenční složky je patrná z jeho horších výsledků.

Vysoké frekvence (10.0–1000.0): Chyby stále rostou, zejména v rozsahu 100.0–1000.0. To ukazuje, že modely mají problémy s efektivním zpracováním vysokofrekvenčních složek signálu. Tento problém může být způsoben nízkou variabilitou dat na vysokých frekvencích, kde může být signál více šumový nebo složitější pro přesné modelování. Další faktor může být nedostatečná kapacita modelů k zachycení rychlých změn signálů, což je běžné u rychlých, vysokofrekvenčních složek.

5 Diskuse

Pro dosažení vyšší přesnosti při predikci parametrů amplitudy a frekvence je nezbytné provést několik dalších kroků, zaměřených na optimalizaci metodiky. Klíčovým aspektem je podrobná optimalizace hyperparametrů pro každou zkoumanou architekturu modelu (Encoder-Only, Decoder-Only, Encoder-Decoder). Precizní ladění parametrů, jako jsou velikost modelu, počet vrstev, počet hlav v multi-head attention, velikost dávky nebo rychlost učení, může zásadně ovlivnit výkon modelů v různých scénářích.

Další směr výzkumu zahrnuje optimalizaci aplikace Fourierovy transformace na vstupní data. Různé metody této transformace mohou vést k odlišným výsledkům v přesnosti predikce. Proto je vhodné zkoumat specifické techniky a jejich vliv na různé architektury modelů. Například může být užitečné analyzovat, zda je efektivnější aplikovat transformaci na celý dataset, nebo pouze na určité jeho části, či zda zpracování pomocí frekvenčních filtrů přispívá ke zvýšení kvality predikcí.

Neméně důležitým krokem je omezení rozsahu predikovaných frekvencí a amplitud. Trénování modelů na specifických intervalech parametrů umožňuje vytvoření specializovaných modelů s vyšší přesností a stabilitou. Tyto modely by pak mohly být kombinovány do systému, který by pokrýval širší spektrum parametrů. Tento přístup nejen zlepšuje výsledky, ale také snižuje pravděpodobnost selhání modelu na krajních hodnotách parametrů.

Další oblast výzkumu spočívá ve zlepšení schopnosti modelu zaměřit pozornost na predikci frekvence. Dosavadní experimenty ukazují, že amplituda má často dominantní vliv na predikce, což může vést ke snížení přesnosti při určování frekvence. Je proto vhodné zkoumat, jak přizpůsobit modelové mechanismy pozornosti tak, aby více reflektovaly frekvenční informace. To by mohlo zahrnovat úpravy váhování v rámci pozornostních mechanismů nebo zavedení specifických ztrátových funkcí, které penalizují chyby v predikci frekvence.

Závěrem lze říci, že zúžení úlohy a adaptace metod pro každou architekturu a rozsah parametrů mohou představovat perspektivní cestu pro zvýšení efektivity modelů. Tyto přístupy přispívají k lepším výsledkům a mohou také poskytnout hlubší vhled do toho, jak různé architektury transformátorů pracují s periodickými daty, zejména v kontextu využití Fourierovy transformace.

Seznam přečtených článků

1. Sachinsoni How Inference is done in Transformer?.
2. Aaron A. King Introduction to inference: parameter estimation.

3. Yash Gondhalekar, Kana Moriwak Convolutional Vision Transformer for Cosmology Parameter Inference.

4. Kip S. Thorne GRAVITATIONAL WAVES.

5. Optuna Framework Documentation