

Napovedovanje vremena s konvolucijskimi nevronskimi mrežami (Weather prediction using convolutional neural networks)

Avtor: Uroš Perkan
Mentor: viš. pred. dr. Žiga Zaplotnik

29. april 2023

Povzetek

V tem seminarju predstavljam koncept napovedovanja vremena s pomočjo modelov strojnega učenja. Na kratko opisujem zgodovino numeričnega napovedovanja vremena in omenjam nekaj pomanjkljivosti numeričnih modelov, ki bi se jih lahko znebili s pomočjo modelov strojnega učenja. Izvedem tudi pregled trenutnih raziskav, pojasnim delovanje konvolucijskih nevronskih mrež, opišem izbiro spremenljivk in proces srednjeročnega napovedovanja vremena in predstavim rezultate svojega in izbranega modela strojnega učenja GraphCastNet. Na koncu omenim, kakšen bi lahko bil nadaljnji razvoj modelov na tem področju.

Abstract

In this seminar I introduce the concept of weather prediction using machine learning (ML) models. I briefly describe the history of numerical weather prediction and its weaknesses, which could potentially be alleviated using ML models. I also briefly explain the current state of research in ML weather forecasting and describe convolutional neural networks in more detail. Next, I discuss the choice of variables in medium-range weather forecasting and present the results of my model as well as one of the current state-of-the-art ML models GraphCastNet. Finally, I write about the possible future work that could be done in this field.

1 Uvod

Napovedovanje vremena spada med poglobitve naloge meteorologije. Zasnove modernega numeričnega napovedovanja vremena segajo v začetek 20. stoletja, ko je L. F. Richardson pokazal, da lahko sistem parcialnih diferencialnih enačb, ki opisujejo atmosfero (gibalne enačbe, kontinuitetna enačba, termodinamska enačba in plinska enačba) diskretiziramo z metodo končnih razlik in rešujemo na mreži točk [1]. Z napredovanjem računalniške tehnologije in izpopolnjevanjem sistema enačb so numerični modeli v zadnjih desetletjih močno napredovali in postali daleč najboljše orodje za napovedovanje vremena. Vremenske napovedi so tako naredile velik

preskok iz subjektivih sinoptičnih napovedi meteorologov v objektivne rešitve fizikalnih enačb.

Kljub znatni izboljšavi vremenskih napovedi s pomočjo numeričnih modelov, obstaja še prostor za izboljšave. Zgoraj omenjeni sistem parcialnih diferencialnih enačb je po fizikalni naravi kaotičen, kar pomeni, da v faznem prostoru prognostičnih spremenljivk majhne razlike med začetnima pogojevoma eksponentno rastejo s časom. Napake v vremenskih napovedih so tako posledica razlik v začetnih pogojih, poleg tega pa k napaki prispeva tudi izbira numerične metode za reševanje danega sistema enačb, parametrizacija pod mrežnih procesov in netočnost robnih pogojev. Z nadomestitvijo numeričnih modelov s statističnimi metodami se pri napovedovanju vremena znebimo potrebe po ločeni obravnavi parametrizacij in robnih pogojev. Statistični modeli lahko samostojno prepoznajo povezave med atmosferskimi spremenljivkami in napovejo nadaljnji razvoj vremena na podlagi njihovega statističnega razumevanja vremena v preteklosti.

Z razmahom uporabe metod strojnega učenja v procesiranju slik se je v zadnjih letih globoko učenje razširilo tudi na področje meteorologije, kjer trenutno poteka aktivno raziskovanje uporabe različnih tipov nevronske mreže kot nadomestilo numeričnih globalnih modelov. Primeri takšnih raziskav so navedeni v [2] in [3], kjer je za napovedovanje vremena uporabljena U-Net konvolucijska nevronska mreža, [4] in [5], kjer uporabijo transformer nevronske mreže in [6] ter [7], kjer uporabijo grafične nevronske mreže. Navedena literatura se osredotoča na srednjeročno vremensko napoved in v primerih [5] in [7] sta modela strojnega učenja že dosegla nivo determinističnega visokoresolucijskega numeričnega modela Evropskega centra za srednjeročno napovedovanje vremena (ECMWF). Metode strojnega učenja v splošnem niso omejene le na omenjeni časovni interval srednjeročne napovedi (1-14 dni). Raziskave potekajo tudi na področju kratkoročnega napovedovanja - "nowcastinga", kjer izkoriščajo hitrost, s katerimi modeli strojnega učenja napoved izvedejo. Napovedovanje razvoja konvekcije ([8]) lahko na ta način posodobimo ob vsaki novi satelitski sliki in radarski meritvi, kar nam lahko omogoči natančnejše izdajanje opozoril pred neurji. Izboljšamo lahko tudi napoved kratkoročnega razvoja vremenskih front in ostalih vremenskih procesov na omejenih domenah [9]. Še večji potencial ima strojno učenje na področju sezonskih in subsezonskih napovedi, kjer modeli iščejo korelacije med počasneje spreminjajočimi se komponentami atmosfere ("Quasi-Biennial Oscillation", "Madden-Julian oscillation", "El Niño-Southern Oscillation", ...) in z njihovo pomočjo napovejo razvoj vremena nekaj tednov ali mesecev v prihodnost. Raziskave kažejo, da obstajajo obdobja povečane subsezonske prediktabilnosti, ki bi jih lahko modeli strojnega učenja izkoristili pri svojih napovedih ([10] in [11]).

2 Treniranje nevronske mreže

Izdelavo modela strojnega učenja za napovedovanje vremena lahko v grobem razdelimo na obdelavo podatkov, konstrukcijo modela in trening nevronske mreže.

Večina omenjenih člankov je podatke pridobila iz ERA5 reanalize, ki predstavlja naš najboljši približek dejanskega stanja atmosfere. Dobimo jo iz procesa asimilacije meritev, kjer podatke iz vertikalnih sondaž, letal, morskih boj, satelitskih in ostalih meritev s pomočjo numeričnega modela združimo v analizo stanja atmosfere. To storimo tako, da kratkoročno modelsko napoved, izračunano iz zadnje analize, objektivno združimo z meritvami, tako da definiramo cenilko, ki meri odstopanje od obeh virov informacije na podlagi znanih statistik napak teh virov. Ko je cenilka minimizirana, dobimo najbolj verjetno stanje atmosfere v danem trenutku. ERA5 reanaliza je izračunana za obdobje med 1940 in 2023. Pri tem je uporabljen en sam

numeričen model, zato ne prihaja do sprememb modelske pristranskosti kot posledica modelske napake v različnih obdobjih reanalize. Njena kakovost je predvsem odvisna od kakovosti meritev, ki jih je uporabila v procesu asimilacije. Reanaliza vsebuje podatke o temperaturi, vlagi, vetru, geopotencialu in mnogih drugih spremenljivkah. Podani so na Zemljinem površju in 37 različnih tlačnih nivojih na horizontalni resoluciji 0.25 °.

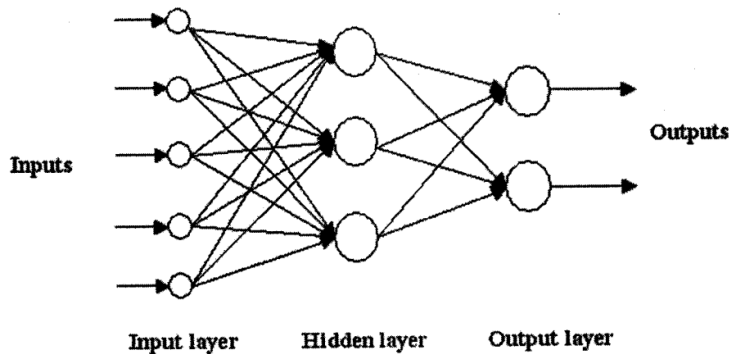
Izbira nevronske mreže igra pomembno vlogo, saj je od nje odvisno, česa se je model sposoben naučiti. Obstaja visoka korelacija med uspešnostjo nevronskih mrež pri procesiranju slik in napovedovanju vremena. To je smiselno, saj je oblika podatkov v obeh primerih zelo podobna. Pri slikah analiziramo polja intenzitet rdeče, modre in zelene barve, v meteorologiji pa polja atmosferskih spremenljivk. Tip nevronske mreže lahko torej izberemo na podlagi uspešnosti na področju obdelave slik. Kot najbolj uspešne so se izkazale konvolucijske nevronske mreže, transformerji in grafične nevronske mreže. Konvolucijske nevronske mreže se napovedovanja učijo prek konstrukcije matričnih konvolucijskih filtrov, ki prepoznajo koristne vzorce v prognostičnih poljih. Transformerji se napovedovanja učijo tako, da polje najprej razdelijo na manjše kose - "tokene", nato pa z mehanizmom pozornosti iščejo za napoved pomembne povezave med različnimi tokeni. Grafične nevronske mreže pa vhodno polje kodirajo na grafična vozlišča, jih procesirajo in nato rezultate v vozliščih dekodirajo nazaj na originalno mrežo. V seminarju se osredotočam na uporabo konvolucijskih nevronskih mrež.

Za razumevanje delovanja konvolucijskih nevronskih mrež je treba najprej razumeti usmerjene - "feed-forward" - nevronske mreže (slika 1). Vsak nevron ima pripadajočo vrednost v in izhod $y = \phi(v)$, kjer s ϕ označimo aktivacijsko funkcijo. Povezava med dvema nevronoma je sinapsa z utežjo w . Posamezne plasti nevronov označimo z indeksom $l \in \{0, 1, 2, \dots, L, L + 1\}$, kjer 0 predstavlja vhodno, $L + 1$ izhodno, vse ostale pa t.i. skrite plasti. Zaporeden indeks nevrona v posamezni plasti pišemo z indeksom i . Tedaj $w_{ji}^{(l)}$ predstavlja utež sinapse med j -tim nevronom plasti $l - 1$ in i -tim nevronom plasti l , $v_i^{(l)}$ pa je stanje i -tega nevrona plasti l , ki ga iz prejšnje plasti izračunamo kot linearno kombinacijo uteži in izhodov $v_i^{(l)} = \sum_j w_{ji}^{(l)} y_j^{(l-1)}$. Pri tem $y_i^{(l-1)} = \phi(v_i^{(l-1)})$ predstavlja izhod i -tega nevrona plasti $l - 1$. Vse nevronske plasti vsebujejo en dodaten nevron z vrednostjo 1, ki doda parameter pristranskosti. Posamezno plast nevronov zato pišemo kot vektor $\mathbf{v} = (1, v_1, v_2, \dots, v_m)$, pripadajoči vektor uteži pa $\mathbf{w} = (w_0, w_1, w_2, \dots, w_m)$, kjer je w_0 utež pristranskosti ("bias"). Pri izbiri linearne aktivacijske funkcije $\phi(x) = x$ dobimo le matrično množenje uteži z nevroni, zato bi v tem primeru večslojno nevronske mrežo lahko nadomestili z enim samim matričnim množenjem in preslikava bi bila linearna. Aktivacijska funkcija torej predstavlja ključen vir nelinearnosti nevronskih mrež in signifikantno spremeni dinamiko modela.

Napako napovedi modela tipično merimo kot vsoto kvadratov razlik med napovedjo \hat{y} in dejansko vrednostjo y : $E = \frac{1}{2} \sum_i (\hat{y}_i - y_i)^2$. Pri treniranju modela spreminjamo uteži w_{ji} z vzvratno propagacijo napake. V ta namen moramo izračunati $\frac{\partial E}{\partial w_{ji}^{(l)}}$, ki ga dobimo z verižnim odvajanjem napake E :

$$\frac{\partial E}{\partial w_{ji}^{(l)}} = \frac{\partial E}{\partial y_i^{(l)}} \frac{\partial y_i^{(l)}}{\partial v_i^{(l)}} \frac{\partial v_i^{(l)}}{\partial w_{ji}^{(l)}} = \frac{\partial E}{\partial y_i^{(l)}} \phi'(v_i^{(l)}) y_j^{(l-1)}. \quad (1)$$

Izračunati moramo še $\frac{\partial E}{\partial y_i^{(l)}}$, kar dobimo iz odvoda napake E po izhodih nevronov plasti $l + 1$,



Slika 1: Skica usmerjene nevronske mreže. Vir: https://doi.org/10.1501/commua1-2_0000000026.

zato temu rečemo vzvratna propagacija gradienta:

$$\frac{\partial E}{\partial y_i^{(l)}} = \sum_k \frac{\partial E}{\partial v_k^{(l+1)}} \frac{\partial v_k^{(l+1)}}{\partial y_i^{(l)}} = \sum_k \frac{\partial E}{\partial y_k^{(l+1)}} \frac{\partial y_k^{(l+1)}}{\partial v_k^{(l+1)}} \frac{\partial v_k^{(l+1)}}{\partial y_i^{(l)}} = \sum_k \frac{\partial E}{\partial y_k^{(l+1)}} \phi'(v_k^{(l+1)}) w_{ik}^{(l+1)}. \quad (2)$$

Vrednost $\frac{\partial E}{\partial y_k^{(l+1)}}$ izračunamo v predhodnem koraku, zato jo na tem mestu direktno vstavimo. Če je $\frac{\partial E}{\partial w_{ji}^{(l)}} > 0$, potem z večanjem $w_{ji}^{(l)}$ večamo tudi E. Po drugi strani $\frac{\partial E}{\partial w_{ji}^{(l)}} < 0$ pomeni, da z večanjem $w_{ji}^{(l)}$ napako E manjšamo. Če uteži popravljamo prek enačbe

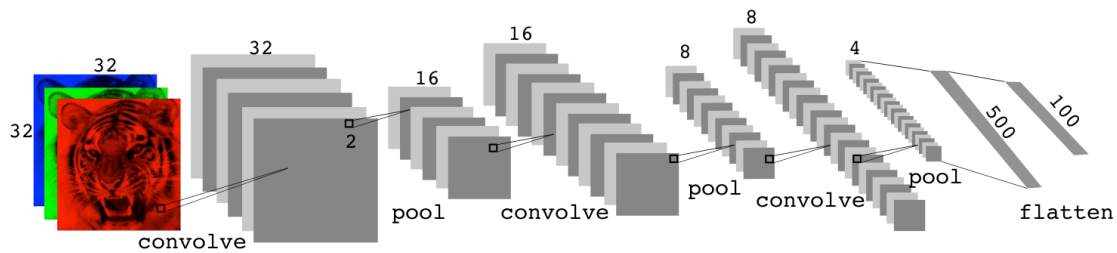
$$\Delta w_{ji}^{(l)} = -\eta \frac{\partial E}{\partial w_{ji}^{(l)}} = -\eta \phi'(v_i^{(l)}) y_j^{(l-1)} \sum_k \frac{\partial E}{\partial y_k^{(l+1)}} \phi'(v_k^{(l+1)}) w_{ik}^{(l+1)}, \quad (3)$$

si torej zagotovimo, da se bo napaka na učni množici manjšala. Uteži lahko popravljamo tudi z drugimi optimizacijskimi algoritmi, najbolj običajno z optimizatorjem Adam, kjer pri izračunih upoštevamo tudi glajeno povprečje in kvadrat gradienta $\frac{\partial E}{\partial w}$ ([12]). Napaka modelske napovedi se na učni množici manjša v nedogled, zato vemo, da bo v nekem trenutku prišlo do prepreleganja. Med treningom zato model testiramo tudi na validacijski množici, ki je med učenjem nima na voljo. Ko se začne napaka na validacijski množici povečevati, učenje ustavimo. V enačbo 3 smo vpeljali hiperparameter hitrosti učenja η , tipično imenovan “learning rate”, s katerim kontroliramo hitrost učenja. Ker se uteži iterativno približujejo optimalnim, se spleča hitrost učenja med treningom spreminjati. Temu rečemo načrtovanje hitrosti učenja oz. “learning rate scheduling”. Obstaja več različnih načinov načrtovanja hitrosti učenja. Med najbolj uspešne sodita eksponentno manjšanje hitrosti učenja (“exponential learning rate”) ali koračno zmanjševanje, kjer η zmanjšamo, ko pride do stagnacije učenja (“reduce learning rate on plateau”).

Konvolucijske nevronske mreže delujejo na podoben način, vendar so plasti nevronov organizirane v matrike, ki so v različnih plasteh lahko različnih dimenzij (slika 2). Namesto linearno obtežene vsote uporabljajo operator konvolucije. Uteži $w_{k,m}$ so zato organizirane v konvolucijske matrike velikosti $c \times d$. Nevron na poziciji (i, j) v (l) -ti plasti izračunamo po enačbi

$$v_{i,j}^{(l)} = w_0 + \sum_{k=1}^c \sum_{m=1}^d w_{k,m} y_{i+k-1, j+m-1}^{(l-1)}, \quad (4)$$

kjer w_0 predstavlja pristranskost (“bias”).



Slika 2: Konvolucijska nevronska mreža (za klasifikacijo). Vir: [13]

Vhodna plast dimenzij (H, W) se po konvoluciji zmanjša na $(H - c + 1, W - d + 1)$. Številu konvolucijskih matrik uteži rečemo število kanalov in v vsaki plasti jih je lahko poljubno mnogo. Vhodna matrika nevronov se konvolira z vsako izmed konvolucijskih matrik uteži in na izhodu se plasti združijo v en tenzor s številom kanalov C . Po eni ali več konvolucijah lahko matriko nevronov “poolamo” ali “upsamplamo”. V prvem primeru matriko nevronov agregiramo tako, da matrike $n \times n$ sosednjih nevronov združimo v enega, tipično s povprečenjem (average pooling) ali izračunom maksimuma (max pooling). V drugem primeru se matriko nevronov razširi z dodajanjem vrednosti. Za koliko mest po matriki nevronov premikamo konvolucijsko ali pooling matriko, opisujemo s hiperparametrom stride, razmik med posameznimi elementi konvolucijske ali pooling matrike pa opisuje parameter dilation. Na sliki 2 se v zaključni fazi matriko nevronov preslika v 1D vektor in poveže z usmerjeno nevronska mrežo. Ta korak ni nujen in je v primeru napovedovanja vremena, kjer želimo imeti na izhodu matriko, tudi neobičajen.

Množenje odvodov v vzvratni propagaciji napake privede do znatnega zmanjšanja ali eksplozije gradientov, saj se zaradi množenja v verižnih odvodih vrednosti močno približajo 0, ali pa močno narastejo. Posledično se uteži prvotnih skritih plasti spreminjajo zelo počasi, ali pa zavzamejo popolnoma arbitrarne vrednosti. Model se posledično med treningom ne izboljšuje več, ali pa se celo poslabša. Temu se izognemo s preskočnimi povezavami (ang. “skip connections”), kjer izhodi ene plasti preskočijo nekaj vmesnih plasti in postanejo vhodi kasnejših plasti. Gradienti lahko v tem primeru preskočijo nekaj vmesnih plasti in se zato med vzvratno propagacijo manjkrat množijo, kar izboljša učenje. Pomembno vlogo pri izboljšanju vzvratne propagacije igra tudi normalizacija vrednosti po vsaki konvoluciji, zaradi katere so tudi velikosti odvodov omejene, kar zmanjša izgubo gradienta. Najbolj pogost način normalizacije je centralizacija in normalizacija izhodov celotne gruče podatkov, ki gredo skozi sloj, kar imenujemo “batch normalization”.

Podatke tipično razdelimo na učno, validacijsko in testno množico, pri čemer model učimo na učni in testiramo na validacijski množici. Po optimizaciji hiperparametrov model testiramo še na testni množici, da se prepričamo, da se niso izbrani hiperparametri preprilegali na validacijsko množico.

3 Spremenljivke

Izbira spremenljivk je pomembna, saj je od njih odvisno, koliko informacij o stanju atmosfere lahko model iztrži v nekem trenutku. Modeli v vhodnih podatkih iščejo povezave med atmosferskimi spremenljivkami, iz katerih znajo razbrati nadaljnji razvoj vremena. Z vsako dodano spremenljivko pa podatkom povečamo tudi dimenzionalnost, s čimer modelu otežimo napove-

dovanje. Pomembno je, da najdemo ravnotežje med številom spremenljivk in kompleksnostjo modela. Hkrati moramo poskrbeti, da model treniramo na podatkih, ki so zares relevantni za nadaljnji razvoj vremena, sicer delujejo kot šum, ki oteži učenje. Za globalne modele so ključni procesi na sinoptični skali, ki jih dobro opiše kvazi-geostrofska teorija. Podatki ERA5 reanalize so na voljo na tlačnih nivojih. Posledično sistem enačb, ki opisuje časovno spreminjanje osnovnih atmosferskih spremenljivk, zapišemo na izobaričnih ploskvah, t.j. ploskvah z enakim tlakom. Na njih horizontalno gibalno enačbo, hidrostatsko enačbo, kontinuitetno enačbo in termodinamsko enačbo zaporedno zapišemo kot:

$$\begin{aligned}
\frac{d\mathbf{v}}{dt} + f\mathbf{k} \times \mathbf{v} &= -\nabla\phi \\
\frac{\partial\phi}{\partial p} &= -\frac{RT}{p} \\
\nabla \cdot \mathbf{v} + \frac{\partial\omega}{\partial p} &= 0 \\
\left(\frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla\right)T - S_p\omega &= \frac{J}{c_p},
\end{aligned} \tag{5}$$

kjer je $\mathbf{v} = (u, v)$ horizontalen vektor hitrosti vetra, sestavljen iz zonalne in meridionalne komponente, f Coriolisov parameter, ϕ geopotencial, p tlak, R specifična plinska konstanta zraka, T temperatura, $\omega = \frac{dp}{dt}$ vertikalna hitrost tlačne ploskve, $S_p = -T\frac{\partial \ln \theta}{\partial p}$ statična stabilnost, $J = \frac{dq}{dt}$ hitrost diabatnega segrevanja, pri čemer je $q = \frac{Q}{m}$ toplota na enoto mase, c_p specifična toplota zraka pri konstantnem tlaku in $\frac{d}{dt} = \frac{\partial}{\partial t}\Big|_p + (\mathbf{v} \cdot \nabla)\Big|_p + \omega\frac{\partial}{\partial p}$ totalni odvod. V velikem delu atmosfere lahko predpostavimo geostrofsko ravnovesje med silo gradienta geopotenciala $-\nabla\phi$ in Coriolisovim pospeškom $f\mathbf{k} \times \mathbf{v}$. Geostrofski veter tedaj definiramo kot

$$\mathbf{v}_g = \frac{1}{f}\mathbf{k} \times \nabla\phi. \tag{6}$$

V splošnem je veter superpozicija geostrofskega in ageostrofskega: $\mathbf{v} = \mathbf{v}_g + \mathbf{v}_a$, pri čemer v kvazi-geostrofskem približku predpostavimo $\mathbf{v}_a \ll \mathbf{v}_g$.

V približku beta ravnine lahko za majhne meridionalne odmike y predpostavimo linearno spreminjanje Coriolisovega parametra $f = f_0 + \beta y$, kjer je $\beta = \frac{\partial f}{\partial y}\Big|_{\phi_0} = \frac{2\Omega \cos \phi_0}{R_z}$ in R_z predstavlja polmer Zemlje, Ω pa krožno frekvenco rotacije Zemlje okrog lastne osi. Če v geostrofskem vetru obdržimo $f = f_0$, povsod drugod v enačbah 5 pa predpostavimo približek beta ravnine, lahko enačbe poenostavimo v obliko kvazi-geostrofskega ravnovesja. Pri tem predpostavimo, da advekcija poteka z geostrofskim vetrom \mathbf{v}_g . Kvazi-geostrofske enačbe tedaj zapišemo kot

$$\begin{aligned}
\frac{d_g \mathbf{v}_g}{dt} &= -f_0\mathbf{k} \times \mathbf{v}_a - \beta y\mathbf{k} \times \mathbf{v}_g \\
\nabla \cdot \mathbf{v}_a + \frac{\partial\omega}{\partial p} &= 0 \\
\left(\frac{\partial}{\partial t} + \mathbf{v}_g \cdot \nabla\right)\left(-\frac{\partial\phi}{\partial p}\right) - \sigma\omega &= \frac{RJ}{c_p p},
\end{aligned} \tag{7}$$

kjer je $\frac{d_g}{dt} = \frac{\partial}{\partial t} + \mathbf{v}_g \cdot \nabla$ geostrofski totalni odvod in $\sigma = -\frac{RT_0}{p} \frac{d \ln \theta_0}{dp}$ statična stabilnost. Polje temperature in potencialne temperature smo razdelili na osnovno stanje in perturbacijo, torej velja $T_{\text{tot}} = T_0 + T$ in $\theta_{\text{tot}} = \theta_0 + \theta$. Iz kvazi-geostrofskih enačb 7 se da izpeljati enačbo tendence

geopotenciala

$$\left[\nabla^2 + \frac{\partial}{\partial p} \left(\frac{f_0^2}{\sigma} \frac{\partial}{\partial p} \right) \right] \frac{\partial \phi}{\partial t} = -f_0 \mathbf{v}_g \cdot \nabla \left(\frac{1}{f_0} \nabla^2 \phi + f \right) - \frac{\partial}{\partial p} \left[-\frac{f_0^2}{\sigma} \mathbf{v}_g \cdot \nabla \left(-\frac{\partial \phi}{\partial p} \right) \right] - f_0^2 \frac{\partial}{\partial p} \left(\frac{RJ}{\sigma p} \right). \quad (8)$$

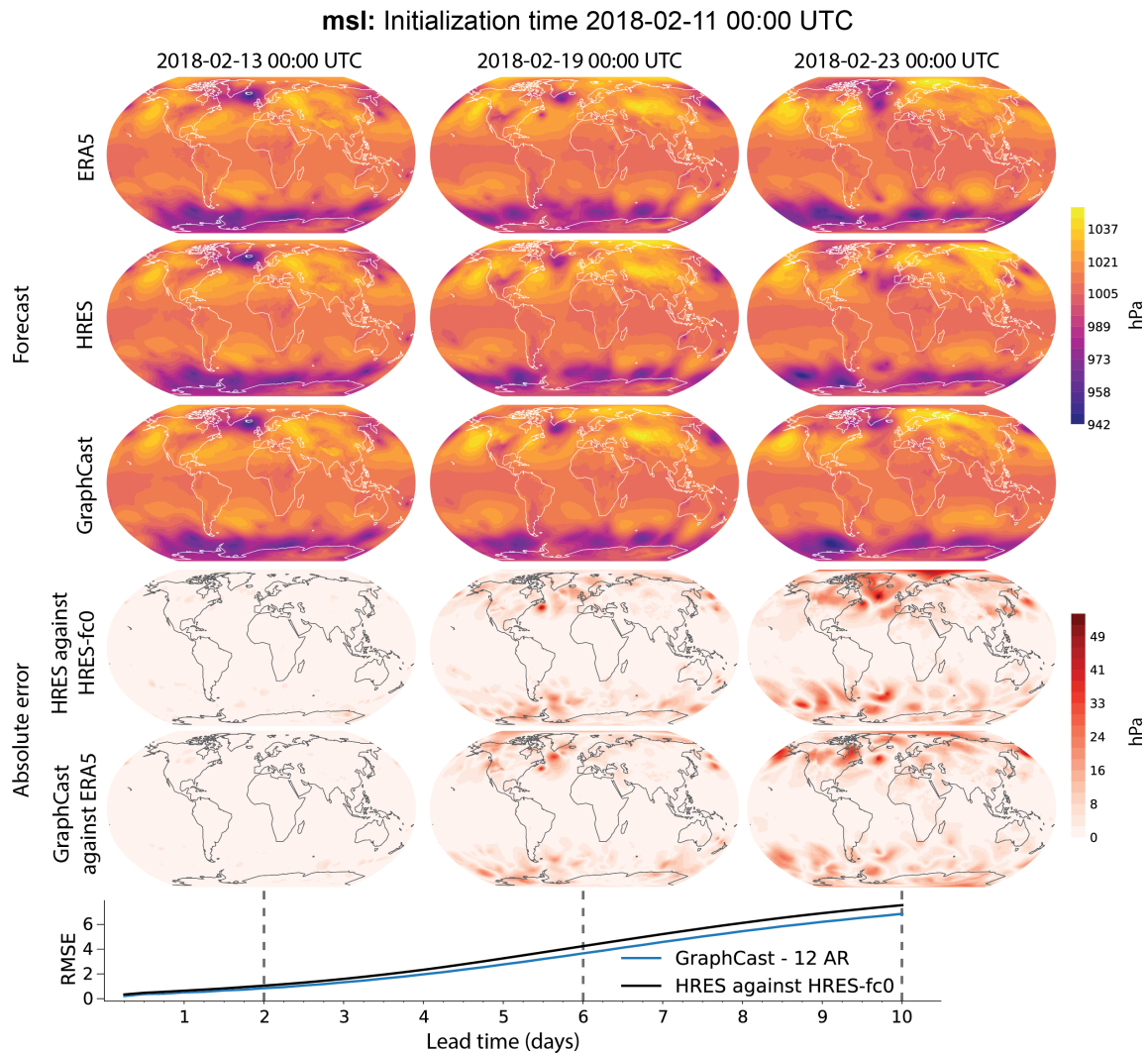
Ta enačba nam poda dober vpogled v razvoj geopotenciala na sinoptičnih skalah. Pomembna lastnost te enačbe je, da je sprememba geopotenciala v prihodnosti odvisna le od trenutnega polja geopotenciala (geostrofski veter lahko izrazimo iz geopotenciala) in od diabatnih procesov. Geopotencial ϕ je torej odlična spremenljivka za modele strojnega učenja, saj se teoretično model z njeno pomočjo in z vedenjem o diabatnih procesih lahko nauči napovedovati nadaljni razvoj polja. Prvi člen enačbe 8 je člen lokalne tendence geopotenciala. Drugi člen predstavlja geostrofsko advekcijo absolutne vrtinčnosti, tretji člen opisuje vertikalno spreminjanje geostrofske advekcije relativne topografije, četrti člen pa vertikalno spreminjanje diabatnega segrevanja. Na podlagi tega lahko sklepamo, da je treba pri srednjeročni napovedi izbrati polje geopotenciala na čim več vertikalnih nivojih. Treba je tudi čim bolj vključiti spremenljivke, ki prispevajo k diabatnim procesom. K temu spadajo procesi povezani s sončnim obsevanjem, kondenzacijo, izhlapevanjem in na sploh s prenosom toplote. Dobra lastnost modelov strojnega učenja je, da jim diabatnih procesov ni treba podati eksplicitno, ampak se lahko o njih naučijo posredno, prek drugih spremenljivk. Primer takšnega posrednega učenja bi lahko bilo izhlapevanje vlage iz morja. V numeričnih modelih bi se v tem primeru parametriziralo turbulenten kinematičen flux vlage, kar lahko storimo z enačbo $\overline{w'r'_v}|_0 = -c_e(\overline{u}_a^2 + \overline{v}_a^2)^{1/2}(\overline{r}_{va} - r_{v0})$ ([14], enačba 4.37), kjer sta \overline{u}_a in \overline{v}_a povprečni hitrosti zonalnega in meridionalnega vetra na višini anemometra (10 m), \overline{r}_{va} je povprečno razmerje mešanosti na višini anemometra, r_{v0} pa razmerje mešanosti pri površini morja (enako nasičenemu razmerju mešanosti pri temperaturi površine morja). Koeficient c_e se določi empirično in v resnici ni čisto konstanten. Ker je takšna parametrizacija dokaj nenatančna, po daljšem času integracije akumulira precej napake in prispeva k slabšanju vremenske napovedi. Modelu strojnega učenja lahko namesto takšne parametrizacije podamo horizontalno polje vetra na 10 m, temperaturo površine morja in zraka na 2 m. Iz tega in ostalih polj, ki bi jih imel na voljo, bi se lahko sam empirično naučil, koliko vodne pare v danih okoliščinah izhlapi v atmosfero. Modeli se na ta način posredno naučijo o vplivu robnih pogojev na stanje atmosfere. Na tem mestu je treba še dodati, da kvazi-geostrofska teorija ne drži v bližini front, na območjih močne ciklonogeneze in za mezoskalne procese. Tedaj lahko sila gradienta tlaka prevlada in ageostrofski veter postane primerljive velikosti geostrofskemu, s čimer postane pomembna tudi ageostrofska advekcija. V model je torej treba vključiti tudi dejanske hitrosti vetra.

Kvazi-geostrofska teorija drži v srednjih geografskih širinah, kjer ima Coriolisova sila močan vpliv na atmosferske tokove. V bližini ekvatorja je Coriolisova sila zanemarljiva, pomembno vlogo pa prevzame konvekcija. Intenzivna kondenzacija v konvektivnih celicah vzbuja nastanek ekvatorialnih valov, ki se na srednjeročnih časovnih skalah v obliki Rossbyevih valovnih paketov širijo tudi v srednje geografske širine in prispevajo k povečanju napak v napovedih. Tu je zato smiselno upoštevati tudi vertikalne hitrosti zraka in količino padavin.

4 Trenutna zmogljivost modelov strojnega učenja

V tem poglavju opisujem trenutno zmogljivost modelov strojnega učenja. Modela [5] in [7] sta decembra 2022 prvič premagala ECMWF-jev HRES model. Na sliki 3 je prikazana primerjava GraphCastNet in HRES napovedi povprečnega zračnega tlaka na nivoju morja (mslp) z ERA5

reanalizo. Napoved tlaka lahko poda koristen vpogled v kakovost modela. Zračni tlak v hidrostatski atmosferi, kar na sinoptičnih skalah večinoma drži, opisuje enačba $p(z) = g \int_z^\infty \rho dz$. Spremembe tlaka so torej posledica spreminjanja gostote v stolpcu zraka $\frac{\partial p}{\partial t} = g \int_z^\infty \frac{\partial \rho}{\partial t} dz$. Kontinuitetna enačba poveže spreminjanje gostote z masnim tokom $\frac{\partial \rho}{\partial t} = -\frac{\partial(\rho u)}{\partial x} - \frac{\partial(\rho v)}{\partial y} - \frac{\partial(\rho w)}{\partial z}$. Če to vstavimo v integral in integriramo od $z = 0$ navzgor ter upoštevamo, da je pri tleh $w = 0$, dobimo enačbo, ki opisuje spremembe mslp v hidrostatski atmosferi (podrobneje opisano v [14], v poglavju 2.5.1)



Slika 3: Primerjava ERA5 reanalize z GraphCastNet in HRES napovedjo zračnega tlaka, reduciranega na nivo gladine morja. Vir: [7]

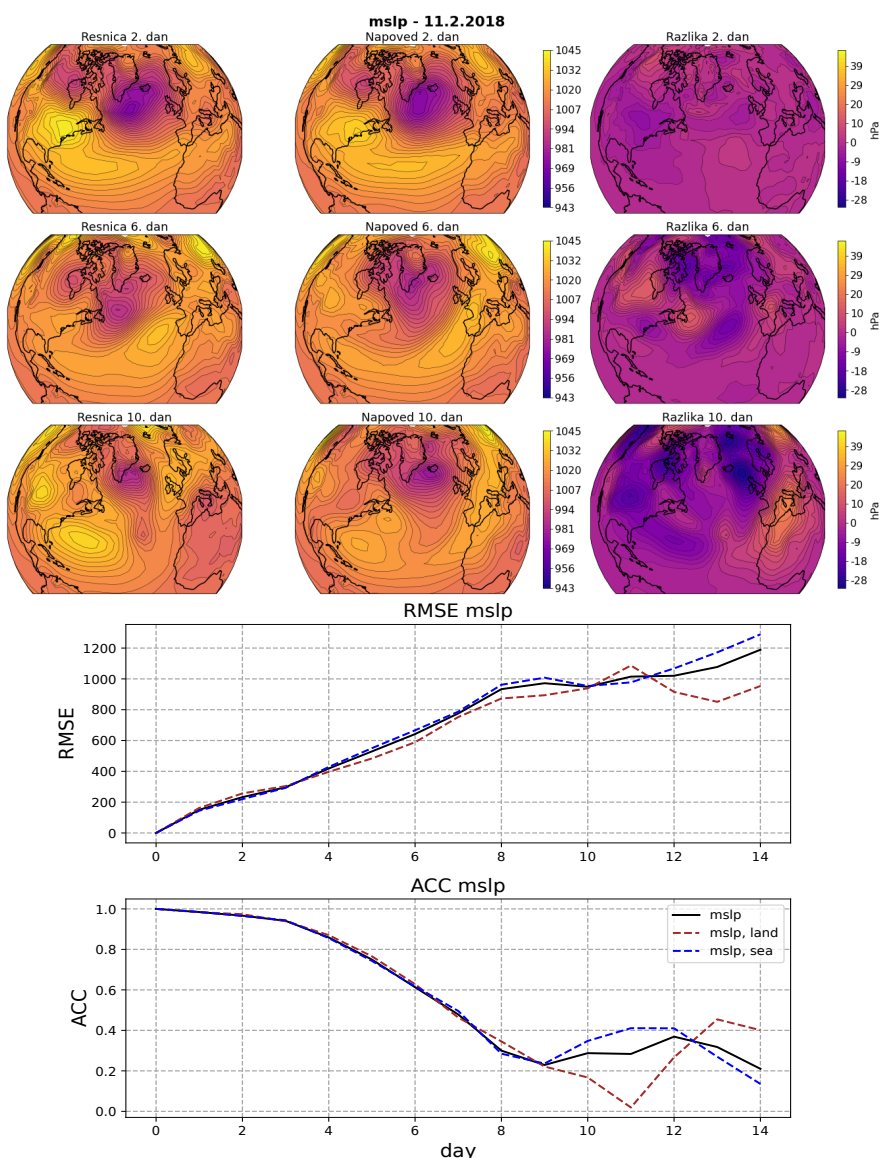
$$\frac{\partial p}{\partial t} = -g \int_0^\infty \left[\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} \right] dz. \quad (9)$$

Razberemo lahko, da je sprememba tlaka posledica divergence masnega toka v stolpcu nad tlemi. Pravilen opis razvoja mslp torej implicitno namiguje na kakovostno napoved prerazporejanja zraka v celotnem atmosferskem stolpcu. Največje težave imajo modeli (tako numerični kot modeli strojnega učenja) na močno baroklinih območjih, kjer lahko poteka intenzivna ciklonogeneza.

Na sliki 3 vidimo, da je HRES 10-dnevno napoved razvoja barokline nestabilnosti na območju Severnega Atlantika napovedal nekoliko slabše kot GraphCastNet. Napako tipično merimo kot

$$\text{RMSE} = \sqrt{(\hat{\mathbf{y}} - \mathbf{y})^2}, \quad (10)$$

ki predstavlja korenjeno povprečje kvadratov razlik med napovedjo $\hat{\mathbf{y}}$ in ERA5 reanalizo \mathbf{y} . Na opazovani sliki jo prikazuje krivulja na dnu, iz katere razberemo, da oba modela po 6 dneh v povprečju zgrešita tlak za približno 4 hPa, vendar GraphCastNet zgreši kanček manj kot HRES.



Slika 4: Primerjava ERA5 reanalize z napovedjo mojega modela konvolucijskih nevronske mreže. Prikazana je za zračni tlak, reduciran na nivo gladine morja. Izračunana je za iste datume, kot na sliki 3. Enote RMSE metrike so [Pa]. Avtor: Uroš Perkan

Napoved GraphCastNet in HRES sem primerjal še z napovedjo svojega modela za prognozo vremena z uporabo nevronske mreže. Arhitektura modela je podobna nevronske mreži U-Net, sestavljeni iz petih pooling in upsampling slojev (globina 5). Na vsaki stopnji se izvajajo 4 zaporedne konvolucije s konvolucijskimi matrikami velikosti 5×5 in 3×3 . Model vsebuje

preskočne povezave - tako med zaporednimi konvolucijami, kot med sloji enakih dimenzij (na enakih globinah). Model uporablja batch normalizacijo in ReLu aktivacijsko funkcijo. Optimiziral sem arhitekturo mreže, hiperparametre in proces treniranja. Večina modelov je treniranih na podatkih o trenutnem stanju vremena z 1, 3 ali 6 urnimi časovnimi koraki. Sam sem model treniral na dnevno povprečenih poljih in korakal z $\Delta t = 1$ dan. Na ta način sem zgladil nekatere hitre procese, zato je v tem primeru dinamika atmosfere nekoliko poenostavljena in v splošnem lažje napovedljiva.

Napoved razvoja barokline nestabilnosti sem za iste dni, kot na sliki 3, izvedel tudi s svojim modelom, ki upošteva 24 spremenljivk (geopotencial na 250, 500, 700 in 850 hPa, zonalen in meridionalen veter na 200, 500, 700 in 900 hPa ter 10 m, zračni tlak, reduciran na nivo morja, temperaturo na površju tal in višinah 2 m, 500 in 850 hPa, dnevno sončno obsevanje, maso vlage v stolpcu atmosfere, nadmorsko višino tal, lokacijo morja in kopnega ter geografsko širino), kar je približno 10 krat manj kot GraphCastNet, ki jih vsebuje 227. Poleg tega ima moj model horizontalno ločljivost 3° , kar je 12 krat manj kot GraphCastNet-ovih $0,25^\circ$. Prikazana je na sliki 4. Tako kot HRES in GraphCastNet tudi moj model ni uspel predvideti natančnega razvoja ciklonov, kljub temu pa je napoved dokaj blizu reanalizi in napaka ni pretirano velika. Na spodnjem grafu lahko odčitamo, da je globalen RMSE mojega modela po 6 dneh 6 hPa, medtem ko imata HRES in GraphCastNet na tem mestu približno 4 hPa. Prikazan je tudi Anomaly Corelation Coefficient (ACC), ki meri korelacijo med anomalijami (od klimatološkega povprečja) napovedi in reanalize:

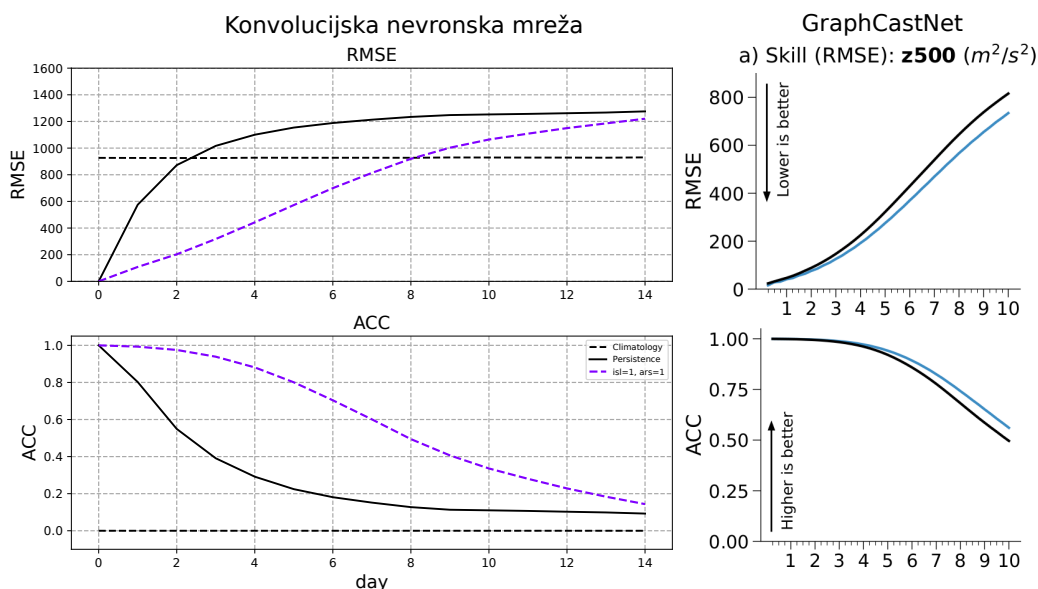
$$ACC = \frac{\overline{(\hat{\mathbf{y}} - \mathbf{y}_c) \cdot (\mathbf{y} - \mathbf{y}_c)}}{\sqrt{\overline{(\hat{\mathbf{y}} - \mathbf{y}_c)^2} \overline{(\mathbf{y} - \mathbf{y}_c)^2}}}. \quad (11)$$

Tipično se kot mejno vrednost, nad katero je napoved koristna, vzame $ACC = 0.6$. To je moj model v tem primeru dosegel po 6 dneh. Prikazana sta tudi RMSE in ACC nad kopnim in nad morjem. Nad kopnim je v primerjavi z morjem po eni strani napoved nekoliko bližje pravim vrednostim, po drugi strani pa nekoliko slabše korelirana z resnico.

Na sliki 5 sta za geopotencial na 500 hPa prikazana RMSE in ACC za GraphCastNet, HRES in moj konvolucijski model. Še enkrat velja opozoriti, da je moj model učen na dnevno povprečenih poljih, medtem ko sta GraphCastNet in HRES učena na trenutnih stanjih vremena. Hkrati pa ima moj model približno 10 krat manj spremenljivk od ostalih dveh. Po 6 dneh ima konvolucijski model RMSE 700, HRES približno 400, GraphCastNet pa $350 \text{ m}^2/\text{s}^2$. Konvolucijskemu modelu pade ACC pod mejo 0.6 po 7, HRES modelu po 9, GraphCastNet-u pa po približno 9.5 dneh. GraphCastNet in HRES imata torej nekaj prednosti pred konvolucijskim modelom, kar je tudi pričakovano, saj sta razvita z uporabo veliko večjih računskih zmogljivosti.

5 Zaključek

Modeli strojnega učenja so v zadnjih letih močno napredovali tudi na področju srednjeročnega napovedovanja vremena in postali konkurenčni celo najboljšim numeričnim modelom na svetu. Preizkušenih je bilo že veliko različnih arhitektur, od katerih so se za dobre izkazale konvolucijske in grafične nevronske mreže ter transformerji. Konvolucijske nevronske mreže imajo sposobnost prepoznavanja vremenskih vzorcev in z njihovo pomočjo ugotovijo stanje vremena v naslednjem časovnem koraku. Poleg omenjenih obstaja tudi veliko nepreizkušenih arhitektur, kot so npr. kombinirane konvolucijske nevronske mreže in transformerji, ki bi lahko združile



Slika 5: RMSE in ACC geopotenciala na 500 hPa mojega modela (levo), GraphCastNet modela (desno, modro) in HRES modela (desno, črno). RMSE je v enotah m^2/s^2 . Na levem grafu je prikazana tudi persistenca (črna), ki predstavlja vrednosti RMSE in ACC, če bi bila vremenska napoved vedno identična vhodnim poljem. Avtor levih grafov: Uroš Perkan, Vir desnih grafov: [7]

najboljše iz obeh svetov. Glavna prednost modelov strojnega učenja tiči v njihovi sposobnosti iskanja povezav med spremenljivkami, ki jih numerični modeli težko simulirajo. Na ta način se znebimo potrebi po empiričnih parametrizacijah, ki predstavljajo dodaten vir napake v napovedih. Poleg tega je napoved modelov strojnega učenja za razliko od numeričnih hitra in energijsko učinkovita.

Kljub temu imajo nevronske mreže tudi nekatere slabosti. Ena izmed njihovih glavnih težav je izguba detajlov v atmosferskih poljih zaradi iterativnega izvajanja napovedi. Posledično prihaja do glajenja spektralnega prostora, zaradi česar je po daljšem času napoved nerealistična in napoved ekstremnih vremenskih dogodkov slabša. Poleg tega so nevronske mreže učene na ERA5 podatkih, ki imajo ponekod znatna odstopanja od dejanskega stanja atmosfere, torej se nevronske mreže učijo na nekoliko napačni atmosferski dinamiki. Za popolno uporabo nevronskih mrež bi bilo treba z njihovo pomočjo izvesti tudi asimilacijo meritev. Nevronske mreže so v zadnjih letih pokazale, da so sposobne zelo učinkovitega napovedovanja vremena. Vse kaže, da lahko v bližnji prihodnosti nadomestijo numerične vremenske modele in postane glavno orodje za napovedovanje vremena.

Literatura

- [1] Lewis Fry Richardson, *Weather Prediction by Numerical Process*, doi: <https://doi.org/10.1017/CBO9780511618291>
- [2] Weyn, J. A., Durran, D. R., & Caruana, R. (2019). *Can machines learn to predict weather? Using deep learning to predict gridded 500-hPa geopotential height from historical weather data*. *Journal of Advances in Modeling Earth Systems*, 11, 2680–2693. doi: <https://doi.org/10.1029/2019MS001705>

- [3] Weyn, J. A., Durran, D. R., Caruana, R., & Cresswell-Clay, N. (2021). *Sub-seasonal forecasting with a large ensemble of deep-learning weather prediction models*. Journal of Advances in Modeling Earth Systems, 13, e2021MS002502. doi: <https://doi.org/10.1029/2021MS002502>
- [4] Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadene-sheli, Pedram Hassanzadeh, Karthik Kashinath, Animashree Anandkumar. *FourCastNet: A Global Data-driven High-resolution Weather Model using Adaptive Fourier Neural Operators*. doi: <https://doi.org/10.48550/arXiv.2202.11214>
- [5] Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, Qi Tian. *Pangu-Weather: A 3D High-Resolution Model for Fast and Accurate Global Weather Forecast*. doi: <https://doi.org/10.48550/arXiv.2211.02556>
- [6] Ryan Keisler. *Forecasting Global Weather with Graph Neural Networks*. doi: <https://doi.org/10.48550/arXiv.2202.07575>
- [7] Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Alexander Pritzel, Suman Ravuri, Timo Ewalds, Ferran Alet, Zach Eaton-Rosen, Weihua Hu, Alexander Merose, Stephan Hoyer, George Holland, Jacklynn Stott, Oriol Vinyals, Shakir Mohamed, Peter Battaglia. *GraphCast: Learning skillful medium-range global weather forecasting*. doi: <https://doi.org/10.48550/arXiv.2212.12794>
- [8] Shreya Agrawal, Luke Barrington, Carla Bromberg, John Burge, Cenk Gazen, Jason Hickey. *Machine Learning for Precipitation Nowcasting from Radar Images*. doi: <https://doi.org/10.48550/arXiv.1912.12132>
- [9] Lasse Espeholt, Shreya Agrawal, Casper Sønderby, Manoj Kumar, Jonathan Heek, Carla Bromberg, Cenk Gazen, Rob Carver, Marcin Andrychowicz, Jason Hickey, Aaron Bell, Nal Kalchbrenner. *Deep learning for twelve hour precipitation forecasts*. doi: <https://doi.org/10.1038/s41467-022-32483-x>
- [10] Annarita Mariotti, Cory Baggett, Elizabeth A. Barnes, Emily Becker, Amy Butler, Dan C. Collins, Paul A. Dirmeyer, Laura Ferranti, Nathaniel C. Johnson, Jeanine Jones, Ben P. Kirtman, Andrea L. Lang, Andrea Molod, Matthew Newman, Andrew W. Robertson, Siegfried Schubert, Duane E. Waliser and John Albers. *Windows of Opportunity for Skillful Forecasts Subseasonal to Seasonal and Beyond*. doi: <https://doi.org/10.1175/BAMS-D-18-0326.1>
- [11] Judah Cohen, Dim Coumou, Jessica Hwang, Lester Mackey, Paulo Orenstein, Sonja Tetz, Eli Tziperman. *S2S reboot: An argument for greater inclusion of machine learning in subseasonal to seasonal forecasts*. doi: <https://doi.org/10.1002/wcc.567>
- [12] Diederik P. Kingma, Jimmy Lei Ba. *Adam: A method for stochastic optimization*. doi: <https://doi.org/10.48550/arXiv.1412.6980>
- [13] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. *An Introduction to Statistical Learning*.
- [14] Paul Markowski, Yvette Richardson (2010). *Mesoscale Meteorology in Midlatitudes*.