

# Developing a Deep Learning forecasting system for short term and high resolution prediction of Sea Ice Concentration

Masters' thesis in Computational Science: Geoscience, Spring 2023

Are Frode Kvanum<sup>1,2</sup>

<sup>1</sup>Department of Geosciences, University of Oslo

<sup>2</sup>Development Centre for Weather Forecasting, Norwegian Meteorological Institute

January 5, 2023

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Datasets</b>	<b>6</b>
2.1	AROME Arctic . . . . .	6
2.2	Sea Ice Charts . . . . .	6
2.3	Osi-Saf . . . . .	7
2.4	AMSR2 . . . . .	8
2.5	NeXtSIM . . . . .	8
2.6	Barents-2.5 . . . . .	8
<b>3</b>	<b>Theory</b>	<b>9</b>
3.1	Convolution . . . . .	9
3.2	Batch normalization . . . . .	9
<b>4</b>	<b>Methodology</b>	<b>9</b>
4.1	Pixel classification . . . . .	9
4.2	Image to image classification . . . . .	9

<b>5 Forecast verification metrics</b>	<b>9</b>
5.1 Defining the Ice Edge . . . . .	10
5.2 Integrated Ice Edge Error . . . . .	11
<b>6 Impact of increased resolution on the IIEE</b>	<b>11</b>
6.1 Computing distances with regards to an Ice Edge . . . . .	12
<b>7 Data pipeline</b>	<b>14</b>
7.1 Data sources . . . . .	14
7.1.1 Sea Ice Charts . . . . .	15
7.2 AROME-Arctic . . . . .	15
7.3 OSI-SAF . . . . .	16
<b>8 Developing a U-Net</b>	<b>16</b>
<b>9 Model Architecture</b>	<b>16</b>
9.1 CategoricalCrossEntropy-Loss . . . . .	18
9.2 FocalLoss . . . . .	20
9.3 Cumulative probability distribution model . . . . .	20
9.3.1 Separate convolutional layers as output . . . . .	21
9.4 Model Selection . . . . .	21
<b>10 physical connections</b>	<b>23</b>
10.1 Variograms . . . . .	23
10.2 Case study . . . . .	23
10.3 Synthetic AA forcing . . . . .	23
<b>11 Comparing against physical models</b>	<b>23</b>
11.1 Preparing data . . . . .	24

# 1 Introduction

Arctic Sea Ice extent has continuously decreased since the first satellite measurements of the Arctic was obtained in 1978 [38], with an average decrease of 4% per decade [4]. Moreover, the summer months are experiencing the largest recession of the extent [5], with models from the Coupled Model Intercomparison Project Phase 6 (CMIP6) projecting the first sea ice-free Arctic summer before 2050 [33]. As a consequence of the sea ice retreat during the summer months, previously inaccessible areas has opened up causing an increase in maritime operations in the Arctic waters [10].

Current information of Arctic sea ice can be discerned into several types of products with different spatial and temporal resolutions. The Ice Service of the Norwegian Meteorological Institute (NIS) is tasked with monitoring the sea ice within the Atlantic sector of the Arctic. As part of their monitoring routine, the NIS produces daily high resolution (1km) Ice Charts for the Atlantic Arctic waters on weekdays, which are manual interpretations of available satellite products. Another source of high resolution information for the position and short term movement of sea ice is delivered by operational sea ice forecasting systems such as neXtSIM [43] and Barents-2.5 [36] which pose a spatial resolution of 3 and 2.5km. Finally, satellite products such as [SSMIS global sea ice concentration](#) and [AMRS-2 global sea ice concentration](#) utilize different sensors to produce daily sea ice concentration with a global coverage at a 10km resolution for the mentioned products or greater.

The previously mentioned sea ice products serve different use cases, and it is possible to infer a correlation between the spatial and temporal resolution of a product and its use case. As such, lower resolution products at larger temporal resolutions may aid in long term planning whereas regional products delivered at a high frequency assist in strategic decision making and short term route planning [42]. However, it is currently reported by end users that available operational satellite products are of a too low resolution, partly due to their insufficient ability to detect leads and other high-resolution information necessary for maritime safety. Moreover, it is reported that sea ice forecasting systems lack verification, are inadequate for operational use as well as a missing a suitable integration to the vessel where computational resources and data-bandwidth are limited [41]. Though sea ice charts provides personnel in the Arctic with information regarding where sea ice has been observed in the time after the previous ice chart has been published, the ice charts lack information on how the sea ice will move, such that a continued safe navigation is delegated to the end-user which relies on their experience [41].

As such, it is of our belief that a different approach to short-range sea ice forecasting is necessary to deliver short-term sea ice information on a spatial scale that is usable by end-users. Thus, this thesis proposes an alternative forecasting scheme that applies Convolutional deep learning in the form of a modified U-NET architecture [35] to deliver a short lead time (1 - 3 days), 1km resolution forecasting product over a subsection

Finn  
en  
måte  
å vite  
”osisaf”  
og  
”amsr2”  
på

of the Atlantic Arctic by utilizing the aforementioned Ice Charts as the ground truth. Moreover, the product is verified with regards to the position of the ice edge, which aims to demonstrate the operational relevance of the product [41, 30].

The U-Net architecture is part of the supervised learning paradigm of machine learning, which require labelled samples in order to train the network [35]. Furthermore, U-Nets perform pixel-level prediction where each pixel is classified according to a category. This work will utilize the image-to-image predictive capabilities of the U-Net to create a semantic segmentation based on its input variables simulating a forward in time propagation of the sea ice concentration akin to a physical model.

There has been made previous attempts to develop deep learning sea ice forecasting systems. The authors of [1] propose IceNet, a pan-arctic covering U-NET which predicts monthly averaged sea ice concentration (SIC) with 6 month lead time at a 25 km spatial resolution [1]. The model classifies sea ice concentration into one of the three classes open-water, marginal ice or full ice. IceNet showed an overall improvement over the numerical SEAS5 seasonal forecasting system [21] for 2 months lead time and more, with the greatest improvement seen in the late summer months. The model is trained on SIC data provided by the European Organization for the Exploitation of Meteorological Satellites (EUMETSAT) Ocean and Sea Ice Satellite Application Facilities (OSI-SAF) dataset [24], as well as other climate variables obtained from the ERA5 reanalysis [15].

Similarly, the authors of [26] propose a Convolutional long short-term memory network (ConvLSTM) which forecasts SIC with a lead time up to 6 weeks. The model uses climate variables and SIC from two reanalysis products ERA-Interim [7] and ORAS4 [2], covering the Barents Sea with a domain size of 24 (latitude) x 56 (longitude). Their results showed skill in beating numerical models as well as persistence.

Models such as those noted above consider input variables obtained from climatologies, and represent SIC on spatial scales far larger than what is needed for an operational sea ice forecast. The possibility of using higher resolution input data was explored by the authors of [11], which combined OSISAF SIC, sea surface temperature from the Multi-scale Ultra-high Resolution product, 2 meter air temperature from the ERA5 reanalysis as well as SIC from sea ice charts produced by the NIS. Fritzner et.al. developed a Fully Convolutional Network (FCN), which achieved similar performance to the Metroms coupled ocean and sea ice model version 0.3 [22]. However, due to computational constraints of training the FCN, the subdomain was reduced to a resolution of 224 x 224 pixels. Thus, the product has a limited accuracy for short term operational usage, similar to [1] and [26].

As mentioned in [1, 11], the computational cost of producing a forecast using a pre-trained model is low, such that a laptop running consumer hardware is able to generate a forecast in seconds or minutes depending on the availability of a Graphics Processing Units (GPU). This is in stark contrast to numerical sea ice models, which could run for

several hours on high-performance systems [1]. Training a model is a one time expense, and can be efficiently performed on a GPU. With the increased complexity, efficiency and availability of high end computing power, smart usage of the available memory allows for model training using high resolution fields. Current GPUs have seen a significant increase in the available video memory, which allows for higher resolution data to be utilized during training. This work will exploit the recent advances in GPU development, as well as incorporating techniques to reduce the floating point precision of the input meteorological variables, circumventing a reduction of the spatial resolution as seen in previous work.

In the present work, the development of a deep learning forecasting system will be explored. The choice and tuning of hyperparameters will be reasoned in light of the physical processes surrounding sea ice and the surrounding variables. Furthermore, the quality of the machine learning forecasting system will be assessed against relevant benchmarks such as persistence, physical models and satellite products. Due to the operational nature of the developed forecasting product, ice edge aware validation metrics such as the Integrated Ice Edge Error [13] will be central to the performance analysis. Furthermore, this thesis aims at providing the framework for which a future operational sea ice prediction system can be built upon. As such, the choice and structure of data will be made with a potential operational transition in mind. A consequence of the operational aspect is the possibility to force decoupled NWP-systems with updated Sea Ice Concentration.

The following research questions will be focused on:

- Can a deep learning system resolve regional sea ice concentration for high resolution, short lead time forecasts?
- How does a high resolution, short lead time unet forecasting system resolve the translation and accumulation of sea ice compared to a physical based model
- In what sense can a deep learning model be explainable / made transparent to explain the statistical reasoning behind the physical decision-making

The thesis is structured as follows. The First section will describe the datasets used, followed by the second section which will do a rundown of the methodological framework necessary to develop the U-Net as well as validation metrics used to assess forecast skill. The third section will detail the development process behind the U-Net, with the fourth section exploring the physical connections of the model. The fifth section will detail the performance assessment of the forecasts. In the sixth section, a discussion of the findings will be conducted, with the seventh section presenting conclusions and future outlook.

## 2 Datasets

The aim of this section is to perform a rundown of the different datasets used for the purpose of this thesis. This includes both internal datasets, in the sense that they are used for training the deep learning system, but also external datasets as in datasets only used for validating model performance. The term *dataset* will for the scope of this thesis refer to a sequence of spatio-temporal structured arrays containing gridded values, and serves as a catch-all for model outputs and satellite products.

### 2.1 AROME Arctic

A deep neural network can increase the skill of its predictions by using correlated variables which provide additional information of the current state of Sea Ice. For example, near surface winds influence the sea ice drift speed [39], with the sea ice drift speed being inverse proportional to the sea ice concentration [44]. Moreover, two meter temperature can also impact the growth of sea ice. AROME Arctic is a non-hydrostatic, convection resolving high-resolution weather forecasting system which covers the European Arctic [31].

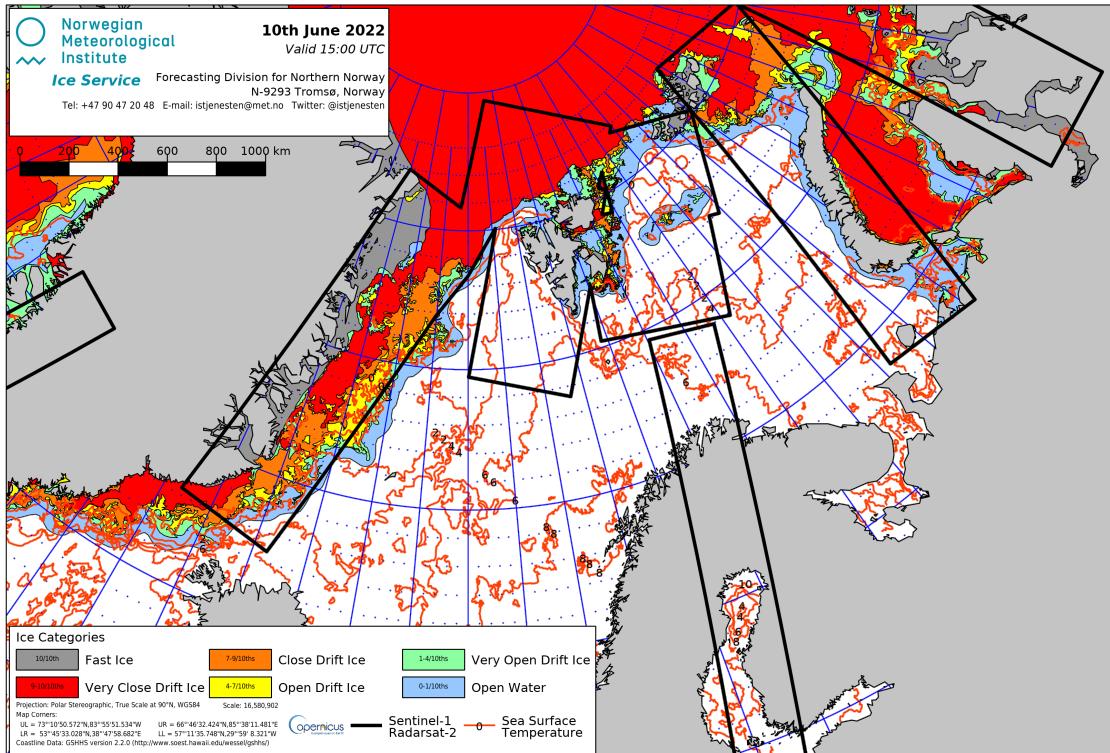
### 2.2 Sea Ice Charts

The Sea Ice charts is an operational Sea Ice Concentration product provided by MET Norway. The product is manually drawn by a Sea Ice Specialist, and is distributed every workday at 15:00 UTC. The Sea Ice specialist assesses available SAR data from Sentinel 1 and Radarsat 2. However, due to the spatial variability in daily SAR coverage, visual, infrared and low resolution passive microwave observations are supplied to achieve a consistent spatial coverage [8]. The Sea Ice charts are drawn in an ArcGIS production environment, and is as such intrinsically not projected onto a defined grid. Yet, the operational product available for download on [Copernicus](#) is provided as mean values on a 1km grid.

From the description of the Sea Ice charts given above, it is worth addressing the spatial inconsistency following the projection onto a uniformly sized grid. As the Sea Ice specialist draws polygons based on data from different satellite sources with a wide range of spatial resolution (80m from SAR, 1000m from visible / infrared and even lower resolution for passive microwave), the underlying uncertainty and detailed structures in the Sea Ice chart varies [8]. Furthermore, I was made aware by one of the Sea Ice Analysts that time

Dette må enten refreres til bakover i oppgaven eller siteres

Få inn en figur som viser månedlig SIC fordeling fra Ice Chartsene, gjerne over en ti års periode. Som i [Grigoryev



constraints also limits the hours different sections of the Ice chart is allotted. Moreover, the Sea Ice charts is an operational product aimed at end users in industries such as fishing, tourism, shipping or other maritime operations. This influences the decision-making when creating the final operational product. . As a consequence, the Sea Ice analyst spends approximately half of the total time draw polygons around the Svalbard archipelago.

In conclusion, concerning the limited resources available both with regards to data availability as well as total hours available, the Sea Ice charts represents a dataset with a spatial uncertainty that is non-uniform across a single sample, and that changes in time. In spite of that, the involvement of a Sea Ice specialist which manually assures each Sea Ice charts, the temporal consistency as well as their high resolution has led us to believe that the Sea Ice charts is the overall best Sea Ice Concentration product available for the current study region.

## 2.3 Osi-Saf

ask  
Trond  
on  
email

Mention how when using Osi Saf trend as predictor, the trend

## 2.4 AMSR2

## 2.5 NeXtSIM

The NeXtSIM-f forecasting system uses a standalone sea ice model (NeXtSIM), and is not coupled with an ocean model [43]. Furthermore, NeXtSIM differentiates itself from comparative physical sea ice models as it does not apply a rheology based on the Viscous-Plastic scheme. Note that the rheology of a sea ice model refers to how the model relates ice deformation and ice thickness with the internal stresses in the ice [16]. internal stress. Instead, NeXtSIM applies a brittle sea ice rheology, specifically the Maxwell elasto-brittle rheology which treats the sea ice as a brittle material rather than a viscous fluid [6].

## 2.6 Barents-2.5

Barents-2.5 is an in-development operational coupled ocean and sea ice forecasting model at MET Norway [36]. The model has been in operation since September 2021. Barents-2.5 poses the same resolution and projection as AA, i.e. Lambert Conformal Conic with a 2.5 kilometer resolution [36, 31]. The sea ice model used in Barents-2.5 is the Los Alamos sea ice model (CICE) version 5.1, which uses an Elastic Viscous Plastic sea ice Rheology [19]. Thus, the CICE model represents sea ice as a viscous fluid which creeps slowly given small stresses and deforms plastically under large stress. It is also noted that the elastic behavior was introduced to benefit the numerical aspects of the model, and can be considered unrealistic from a physical point of view [18].

As part of the forcing routine, Barents-2.5 performs non-homogenous atmospheric forcing of its ensemble members, with one member of each ensemble being forced with AA while the rest of the members is forces using atmospheric data from ECMWF. As such, the member forced with AA seem to perform best with regards to ocean currents, but the atmospheric forcing's impact on SIC performance is unknown at the time of writing. However, there is generally little ensemble spread with regards to sea ice (Johannes Röhres, 2022, pers. commun.)

The data assimilation scheme applied for Barents-2.5 is a Deterministic Ensemble Kalman filter, which solves for the analysis though with a background error covariance matrix estimated as the variance of the ensemble of background members [36]. Furthermore, it has been expressed by the developers of the model that the model performance was unsatisfactory up until May / June 2022 due to spin up time of the data assimilation system (Johannes Röhres, 2022, pers. commun.). This coincides with the formulation of the Ensemble Kalman filter as a Monte Carlo formulation of the Kalman filter [37]. Hence, it is expected that the data assimilation scheme used in Barents-2.5 improves the forecasts

over time.

## 3 Theory

### 3.1 Convolution

### 3.2 Batch normalization

## 4 Methodology

### 4.1 Pixel classification

### 4.2 Image to image classification

Architectures such as [23] could be used for semantic segmentation, given a sliding windows approach across the image to be classified. However, this approach would prove to be time consuming, as each pixel would have to be classified independently, each pixel would only have a receptive field limited by the extent of the sliding window and the edges would be difficult to classify. Thus, network architectures such as [27] and [35] provide a translation invariant framework for image to image prediction.

The U-Net architecture was originally proposed by [35] in 2015.

## 5 Forecast verification metrics

A robust verification scheme is essential to gain insight into how the developed forecasting product performs. Both from the point of view of a developer which aim to increase the skill of the prediction but also from the user which may utilize the verification score to assess the quality of a given forecast [3]. In the context of Sea Ice forecasting, a spatial field of continuous or discrete sea ice concentration is predicted, the latter being the case for the current work. Given the uneven distribution intra sea ice concentration classes as well as sea ice compared to ice free open water, simply comparing pixels for correctness

Wrigglesworth et al., 2011  
the persistence of sea ice anomalies are very high at weekly - sub monthly timescales, making it difficult to beat

Write about how deep neural networks fits a model to some output according to the

Syk  
udoku-  
mentert  
pås-  
tand,  
må  
moder-  
eres

would be biased by the large portion of open water and result in difficult to interpret values devoid of physical reasoning. Furthermore, as the rate of maritime activity such as commercial shipping increases in the Arctic due to the sea ice decline [17], having user relevant metrics can aid and alleviate the risks surrounding Arctic navigation. As such, several studies have proposed calculating the position of the ice edge as a user relevant metric which also provides information of the distribution of the Sea Ice Concentration [9, 13, 12]. However, there is no agreement with how to best calculate the position of the Ice Edge, with the currently available metrics posing different advantages/disadvantages [34, 30]. For the purpose of this thesis, The ice edge position and length will be calculated according to [30, Melsom 2019 et.al], whereas the IIEE originally proposed by Goessling. H. [13] will also be utilized.

## 5.1 Defining the Ice Edge

The ice edge for a given Sea Ice Concentration product is derived on a per pixel basis, and defined as the grid cells which meet the condition

$$c[i, j] \geq c_q \wedge \min(c[i - 1, j], c[i + 1, j], c[i, j - 1], c[i, j + 1]) < c_e \quad (1)$$

i.e. a pixel is marked as a ice edge pixel if the current pixel itself is larger than some given concentration threshold  $c_e$  and the minimum of the pixel's 4-neighbors is less than the same threshold. Following this notation, E is defined as the set of grid cells which constitutes the ice edge [30]. Moreover, the marked grid cells each contribute to the total length of the ice edge, with each pixel's length contribution determined based on the number neighbors also marked as an ice edge pixel. Consequently, a neighborless pixel is assumed to yield a contribution the length of the diagonal to the ice-edge ( $l = \sqrt{2}s$ ) where s is the side length of the pixel. A pixel with one neighbor a contributes a mixed horizontal - diagonasl length  $l = \frac{s+\sqrt{2}s}{2}$ . Finally a pixel with two or more neighbors contributes with a pixel side-length  $l = s$ . The final length of the ice edge length then become

$$L = \sum_{e \text{ in } E} l_c \quad (2)$$

where E represent the set of pixels constituting the ice edge as previously mentioned [30].

## 5.2 Integrated Ice Edge Error

The Integrated Ice Edge Length (IIEE) is an error metric which compares the forecast to some ground truth target [13]. The metric is defined as

$$\text{IIEE} = O + U \quad (3)$$

where

$$O = \int_A \max(c_f - c_t, 0) dA \quad (4)$$

and

$$U = \int_A \max(c_t - c_f, 0) dA \quad (5)$$

with  $c_t$  and  $c_f$  being the target and forecast concentration respectively, attaining a value of 1 if the concentration for a given pixel  $i$  above a set threshold, and 0 elsewhere. From the definition of the metric, it can be seen that the IIEE is a sum of the forecast overshoot and undershoot compared to the ground truth target. For the current work, the IIEE is an easily interpreted metric as it quantifies the total forecast error and reports on the error spatially.

As an additional remark, note that  $O$  the current implementation in Equation (4) is given as an area with sidelength  $dA$ , and is computed as one product against some other product. Conversely, for this work subscripts  $f$  and  $t$  denote forecast and target respectively, with the directionality of the computations in Equation (4) and (5) indicating that the forecast is inspected with respect to the target. However, the metric can and has also been used to define the set of pixels which constitutes its area. To clearly distinguish between the area  $O$  and the set of pixels used to compute  $O$ ,  $A^+$  will be used to note the latter. Similarly,  $A^-$  will represent the set of pixels constituting  $U$ . Finally, it can be seen that  $A^+$  and  $A^-$  represent the False Positive and False Negatives of the forecast respectively.

Furthermore, the IIEE can be combined with the length of the Ice Edge which was derived in the previous section 5.1. Thus, the metric is seasonally normalized, assuming that the IIEE and Ice Edge Length is seasonally correlated.

## 6 Impact of increased resolution on the IIEE

From both the definition of Equation (1) and (3), it can be seen that there is a dependance on the number of pixels which constitutes the ice edge. However, what effect would a change of resolution, i.e. change in number of pixels, have on the IIEE / ice edge length ratio? To answer this question, the "two day" ice edge targets where compared

against persistence on all valid two day forecasts samples for the period 2019 - 2021. Furthermore, the original target resolution of 1km will be assessed, as well as a regridded product downsampled onto a 10km resolution.

The Pearson correlation coefficient was computed directly from the DataFrames containing the metrics for both 1km and 10km resolution, using the statistical Python package Pandas [40, 29]. For clarity, the correlation between mean\_length and IIEE for 1km and 10km was calculated. From the resulting computations, the correlation coefficient gets reported as  $r_{mean\_length} = 0.9620$  and  $r_{IIEE} = 0.9995$ . Furthermore, the IIEE divided by the mean length correlation is reported as  $r_{normalized\_IIEE} = 0.9667$ . Finally, Figure (1) displays the mean monthly IIEE computed along the inspected three year period.

By inspecting Figure (1) in conjunction with the reported correlation coefficients, it can be seen that increasing the working resolution of the dataset from 1km to 10km has a negligible impact on the reported metrics. Though the 1km ice edge is about 2 times longer than the 10km, coarser resolution ice edge, it does not impact the overall stability of the normalized IIEE metric. Moreover, the normalized IIEE is reduced for the 1km grid. This could be a consequence of each pixel's spatial extent covering less area. Note that from the definition of the IIEE given in Equation (3), the un-normalized metric is dependent on pixel spatial size. Thus, the number of pixels constituting the sum is inverse proportional with the size of each pixel, hence keeping the stability of the values. However, a discrepancy may arise due to the coarser resolution pixels covering a larger area, thus losing out on the fine details. However, for the current work, the ratio of 1km IIEE and 10km IIEE is 0.9938, indicating that they are close to equal.

## 6.1 Computing distances with regards to an Ice Edge

For the scope of this thesis, the above defined set E will be subscripted with  $f$  and  $t$  which represents *forecast* and *target* respectively. Hence,  $E_f$  is the set of pixels which constitutes the forecasted ice edge, with  $E_t$  defined analogously. With this definition, the euclidean displacement of the target ice edge  $E_t$  from all forecasted ice grid cells  $E_f$  has been defined as the following in [30]

$$d_t^n = \min \left( \forall e_f \in E_f : [(x_f - x_t^n)^2 + (y_f - y_t^n)^2]^{1/2} \right) \quad (6)$$

where x and y represent the coordinates of the grid cells and n the index of the pixel inspected in  $E_t$ . The opposite definition, i.e. the euclidean displacement of the forecasted ice edge in terms of the target ice edge is defined equally.

Equation (6) can also be generalized to hold for other use cases, not only comparing the

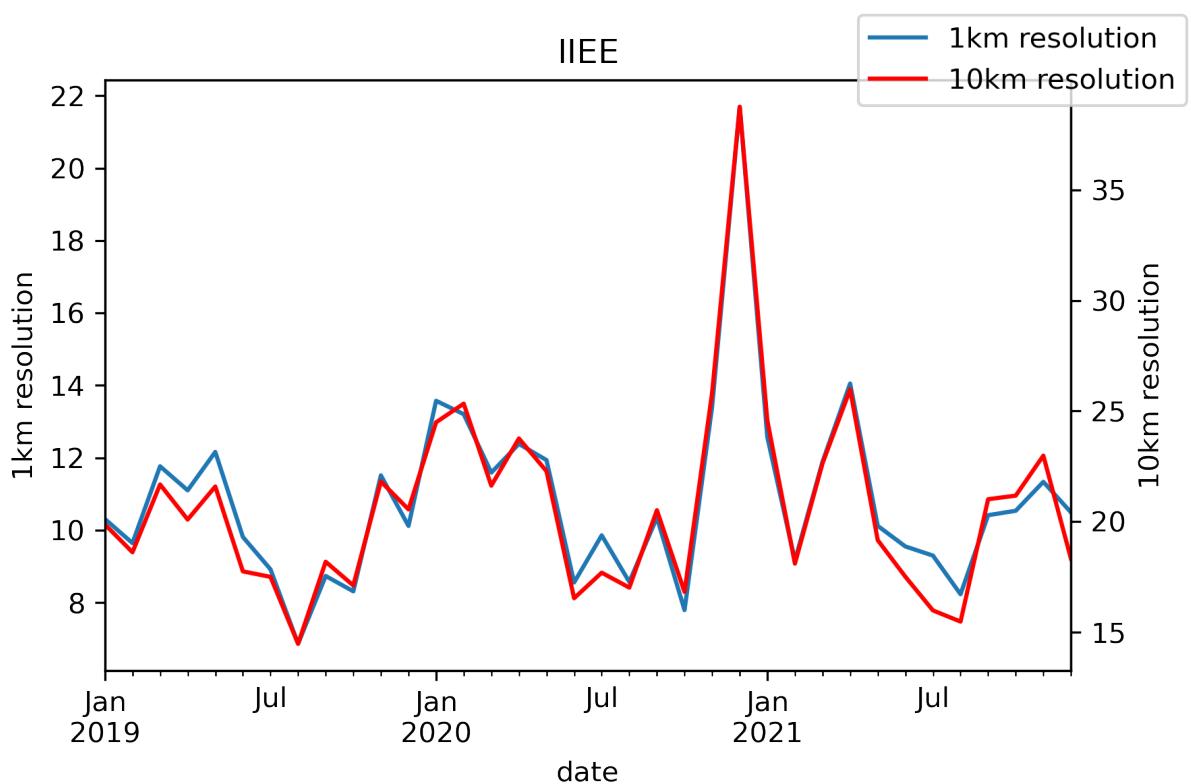


Figure 1: Monthly mean of Normalized IIEE spanning 2019 - 2021

displacement of two ice edges. Given an IIEE which has been separated into  $A^+$  and  $A^-$ , and a target ice edge  $E_t$ , the distance between the nearest ice edge pixel  $e_t$  two a misclassified pixel ( $a^+$  or  $a^-$ ) can be defined as, (using  $a_+$  as an example)

$$d_n = \min (\forall e_t \in E_t : \text{distance}(e_t, a_n^+)) \quad (7)$$

where distance is used to define som arbitrary distance metric, n denotes the pixel index in  $A^+$ .

## 7 Data pipeline

The deep learning system can be disassembled into two parts working in tangent. The deep learning architecture which propagates fields containing information through its weights, and the dataloader which structures the dataset into trainable samples. This section will describe the process from raw data to ready sample, with the following Section (8) containing a rundown and results of the model architecture.

The data pipeline is made such that it constitutes models of three different lead times (one, two and three day lead time). A quick overview of the pipeline is as such. The raw data used are Sea Ice Charts, OSI-SAF and AA. For the Sea Ice Charts, ice charts from the bulletin date and valid date are selected. From AA, relevant meteorological fields are selected and daily means are computed (more details in following sections). Finally, from OSI-SAF a sea ice trend is computed. For a given bulletin date, the data fetched above is stored in a .hdf5 file, such that each sample (bulletin date) is represented by its own .hdf5 file. Furthermore, a dataloader object is initialized with a list of .hdf5 files, with the list containing filenames of the samples constituting a data subset such as train, validation or test data. This processes is visualized in Figure (2).

### 7.1 Data sources

Data sources used are Sea Ice charts from Nick initiated at 15:00 as well as Arome Arctic initiated at 18:00 [8, 31]. For a given date, the current Ice Chart is used as a predictor for the model, while the Ice Chart drawn two days later is supplied as the model target.

This should be represented by a figure where individual sources are merged into one stack which is then treated as an entry

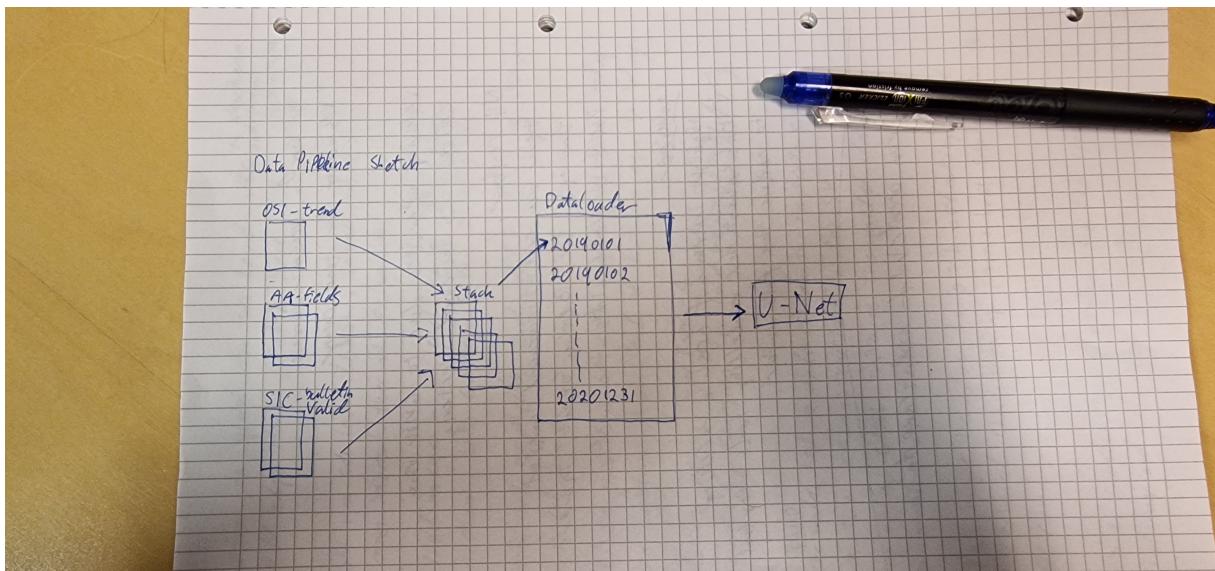


Figure 2: Sketch showing an idealized data pipeline, where three sources are merged into one sample, kept track of by a dataloader which feeds the samples into the U-Net

### 7.1.1 Sea Ice Charts

The Sea Ice Charts used are a derived dataset of the Sea Ice Charts presented in a previous section. The present Ice Chart dataset has been postprocessed by Nick Hughes of the National Ice Service, such that they are presented on a 1km Arome Arctic grid. Furthermore, the Ice Charts does not feature a land-mask, which has been replaced with interpolated values resulting in a spatially consistent dataset where all values present are according to the WMO Sea Ice Concentration intervals [20].

## 7.2 AROME-Arctic

The Arome Arctic data is structured such that the period between forecast initialization and machine learning forecast lead time is stored as a mean product in the temporal dimension at intervals [0 - 18, 42, 66]. This ensures that temporal AA information is encoded into a single field up until 12:00 UTC of the publishing date of the target ice chart. The 4d variables used from AA are T2M, uwind and vwind. Finally, the land sea mask present in AA is fetched and used as a predictor, though this land sea mask is also used for validation purposes given the case where no other SIC-product is considered.

label sections

Say thanks in acknowledgements

### 7.3 OSI-SAF

A linear SIC trend of variable temporal length is computed from 12.5km OSI-SAF data . In the case of OSI-SAF, the product is scheduled to be published daily at 15:00 UTC . However, given operational concerns of the developed forecasting system, where the availability of data is essential for the model to run, the previous day OSI-SAF trend is utilized. .

Cite  
OS-  
ISAF

double  
check

Det er  
ikke  
sikkert  
det er  
nød-  
vendig,  
hvis  
AA  
18:00  
pub-  
liseres  
19 - 20  
burde  
OSI-  
SAF  
være  
publis-  
ert

## 8 Developing a U-Net

The model developed for the two day prediction is based on the SimpleUNET architecture, though with a different sized Input layer to accommodate for the changed dataloader. The dataloader has subsequently been changed to appropriately select the correct fields from the .hdf5 samples and appoint them as input or target variables. As a result of using three variables of two days mean AA forecast, as well as sst, land-sea-mask and current time-step ice chart, the total number of predictors fed into the model is 9. Moreover, the resolution of all fields are kept at 1km, though their spatial extent is limited to (1920 x 1840). This resolution and spatial size conserves (almost) the entirety of the west-east axis of the AA domain. However, the southern border is raised by 450km compared to the AA domain. There are two main motivations behind readjusting the spatial extent of the predictors and targets.

1. The spatial extent of the input domain has to be divisible by the reducing factor enforced by the MaxPooling operation performed in the encoding component of the UNET.
2. The southern latitudes covered by AA has a proportionally skewed Sea Ice / Ice Free open water ratio, as exemplified in Figure (3). Increasing the southern bounding latitude of the subdomain thus decreases the number of guaranteed ice free pixels, which in turn decreases the skewness towards the ice free open water class for the UNET.

## 9 Model Architecture

The model architecture follows an encoder - decoder structure, commonly referred to as a U-NET [35] due to its shape funnelling the spatial data to coarser resolution, which resembles the letter "U". The current U-NET implementation follows that of Ronneberger

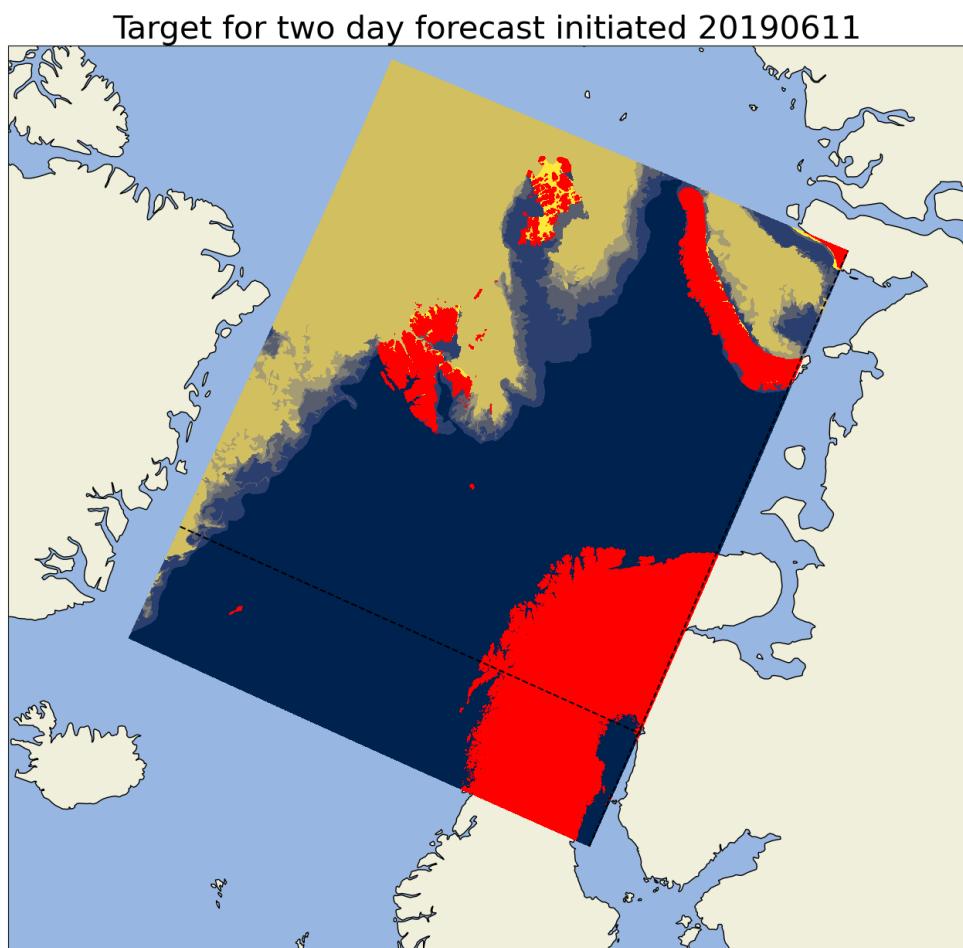


Figure 3: Example sample displaying an Ice Chart on a 1km Arome Arctic projection. Note the horizontal and vertical dashed black line which indicate the domain subsection used by the UNET

et.al, though it has been modified with batch normalization after each convolution operation to ensure a more stable gradient flow. The weights of the model are Kaiming-He initialized [14], as the activation function used throughout the network is the ReLU function [32]. The final output of the model is a (1920, 1840, 7) tensor containing softmaxed probabilities along its final axis.

## 9.1 CategoricalCrossEntropy-Loss

As the title suggests, these runs of the model involved using CategoricalCrossEntropy as the loss function for multi-class image segmentation. Categorical Cross Entropy loss is defined as

$$CE = - \sum_i^C y_i \log (\hat{y}_i) \quad (8)$$

where  $C$  denotes the number of available classes,  $y$  the ground truth and  $\hat{y}$  a prediction of  $y$ . Note that as  $y$  is onehot-encoded, the formulated function only contributes to the overall loss with the log of the predicted probability of the correct class according to the ground truth.

Two variants of the previously described model have been trained with the CategoricalCrossEntropy described in equation (8). The first model was trained with an encoder consisting of 4 convolutional blocks with channel dimensions (64, 128, 256, 512). The second model consisted of 5 convolutional blocks, with an identical architecture except for the last convolutional block increasing the channel dimension to (1024). Example outputs as well as target can be seen in Figure (4).

By inspecting Figure (4), two observations can be made. The first observation is regarding how the model complexity affects how it fit to the data. By comparing Figure (4a) with (4b), it can be seen that the latter is resolving the finer structures of the ice edge to larger extent than the prior. Though the overall correctness is left to be discovered, this shows that increasing the depth of the encoder (increasing the trainable parameter count from 7 million to 31 million) is reflected by the model preserving the details of the ice edge structure. Though it is non-trivial to say why the 1024-model preserves the details to a larger extent than the 512-model, it does follow from the U-Net architecture that a deeper encoder (higher channel count and more convolutional blocks) is better at describing "WHAT" is in the image compared to the shallow-layers, which include a larger amount of spatial information and tells the model to a larger extent "WHERE" things are in the model.

This  
may  
have a  
source

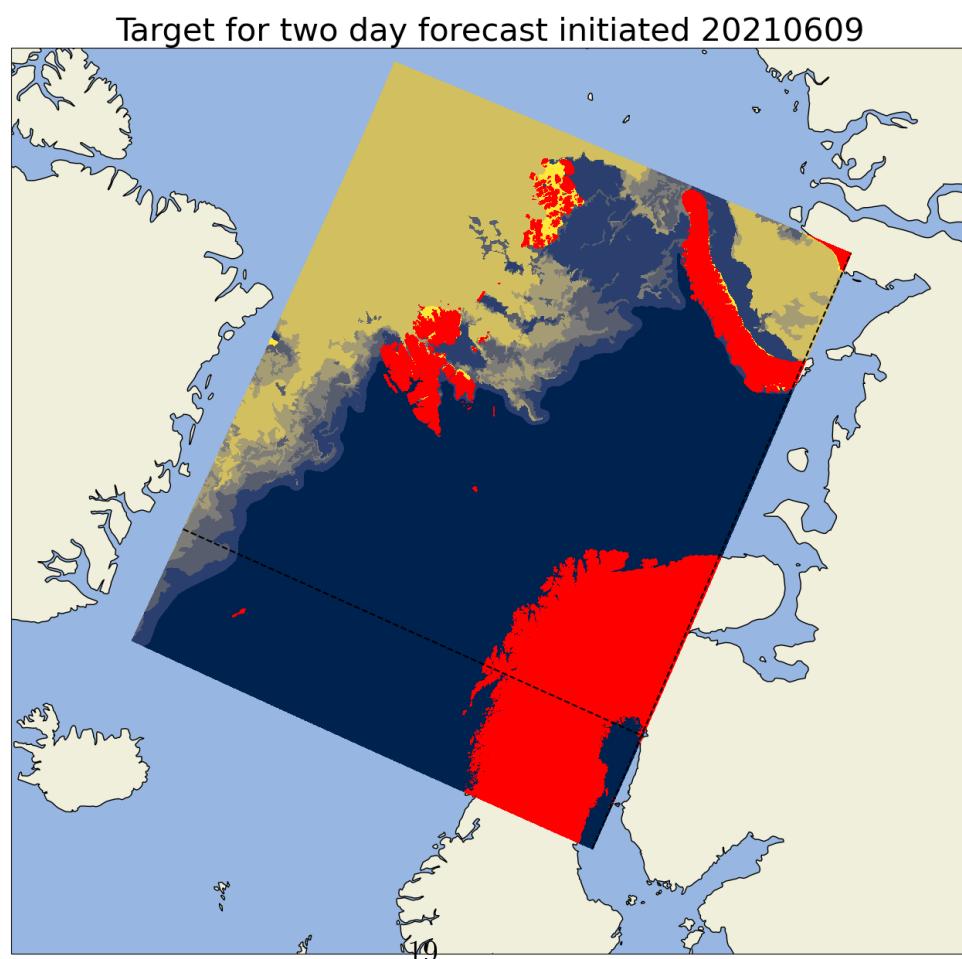
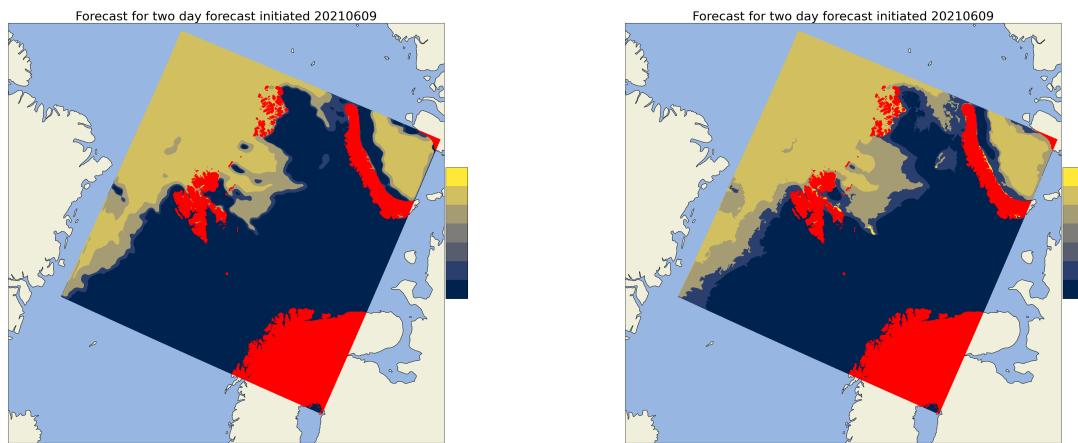


Figure 4: Example forecast attempt made by model\_512 and model\_1024 09-06-2021

The second observation made from inspecting both forecasts is their inability to represent classes 2 and 3. This likely arises from the general movement-pattern of the sea ice, where the intermediate classes are much less likely to appear than the edge-most classes. Furthermore, the sea ice is much more likely to represent a wider range of concentration classes in the intermediate ice edge region over time, making it more difficult for the network to confidently predict those classes compared to the more probable classes. As can be seen by the network immediately predicting class 4 after class 1, creating an artificial cut-off region. However, to what extent the intermediate classes are predicted has not been inspected directly, though it is likely to assume that they are predicted though with a lower confidence than that of class 4 (which is consequently why it is visualized, as the most probable class is chosen regardless).

They  
should  
be

## 9.2 FocalLoss

The focal loss is derived as a generalization of the Cross Entropy Loss listed in Equation (8). The intent of the loss function is to downweight the easy to predict samples, while focusing on the hard to predict samples by allowing their gradient to have a higher impact on the network [25]. Mathematically, focal loss is defined as

$$\text{FL} = - \sum_i^C \alpha_i (1 - \hat{y}_i)^\gamma y_i \log (\hat{y}_i) \quad (9)$$

where  $\alpha$  is a balancing parameter,  $\gamma$  is the focusing parameter ( $\gamma = 0 \rightarrow CE$ ), with the rest similar as Equation (8).

By inspecting Equation (9), it can be seen that predictions that the model is quite confident in making, i.e.  $\hat{y}_i \rightarrow 1$  send the Focal Loss towards zero. For the current application, the assumptive motivation is that this affects (by reducing) the contribution made by the Ice Free Open Water pixels as well as the Very Close Drift Ice (class 6), which are the most represented classes in the CE loss model seen in Figure (4). Consequently, as the loss contributions of the most likely (and most represented classes) is reduced, the harder to predict (both due to being less represented and due to sea ice movement) have a larger impact on the overall loss propagating backwards throughout the model. As a result, these intermediate classes should be predicted as the most likely class, resulting in a less sharp ice edge which closer represent the Ice Charts.

Include figure showing focal loss output, discuss implications of using this loss function

## 9.3 Cumulative probability distribution model

Discuss difference in dataloader, same dataset is used differently

### 9.3.1 Separate convolutional layers as output

## 9.4 Model Selection

During the training of a deep learning system, there exists several different ways to save a state of the model during training. A naive approach would be to let the model train all predetermined epochs, and save the weights of the model at the end of the final epoch. However, this approach would be indifferent to whether the model has converged, generalized or overfitted and is thus an inadequate way to save the weights. The Tensorflow Keras API supplies functions which can be used customize the training loop in the form of [callbacks](#), with the EarlyStopping and ModelCheckpoint callbacks relevant for model selection [28]. EarlyStopping is a technique which ends the training loop when it detects that a monitored values has stopped decreasing. On the other hand, ModelCheckpoint continuously saves the model if a certain condition is met, without terminating the training loop. Both callbacks support monitoring the validation loss as the metric in which to optimize the model. However, a custom metric such as yearly mean IIEE [13] could be monitored instead.

To aid in model selection, I developed a custom callback which computed the Normalized IIEE with respect to a climatological Ice Edge length derived from ten years of OsiSaf data , following the observation in 6 that IIEE is correlated across spatial resolutions. The callback computes said metric for all samples and reduces them to a yearly mean of the validation set. Similar to the aforementioned callbacks, the developed callback is executed at the end of an epoch where it computes the mean Normalized IIEE for all predicted samples from the validation set, which it appends to the *logs* dictionary used by Tensorflow to keep track of other computed metrics, such as loss and validation\_loss for the current case. Thus, the newly developed callback would allow for model selection based on Normalized IIEE, as well as the already computed validation loss.

When comparing different models to asses their performance, this project will frequently compare their Normalized IIEE as the metric is Normalized by the ice edge, thus reducing the seasonal variability of the Metric [34] . As such, it would be beneficial to select a model based on its Normalized IIEE validation performance. With the above callback, such a selection is possible. However, including the IIEE verification metric as is done in the above callback increases training-time of ten epochs from  $\approx$  two hours without the IIEE callback to  $\approx$  24 hours with the IIEE callback. As 20 epochs is currently an adequate number of epochs at the time of writing , it would be too computationally costly to select a model based in its validation Normalized IIEE performance.

Data exists,  
start writing

Write about  
the  
climatolog-  
ical Ice  
Edge  
dataset,  
ref sec-  
tion  
from  
here

This  
cita-  
tion  
is ac-  
tually  
for  
 $SPS_{length}$   
but  
 $SPS$   
is re-  
duced  
to  
IIEE  
for a  
deter-  
minis-

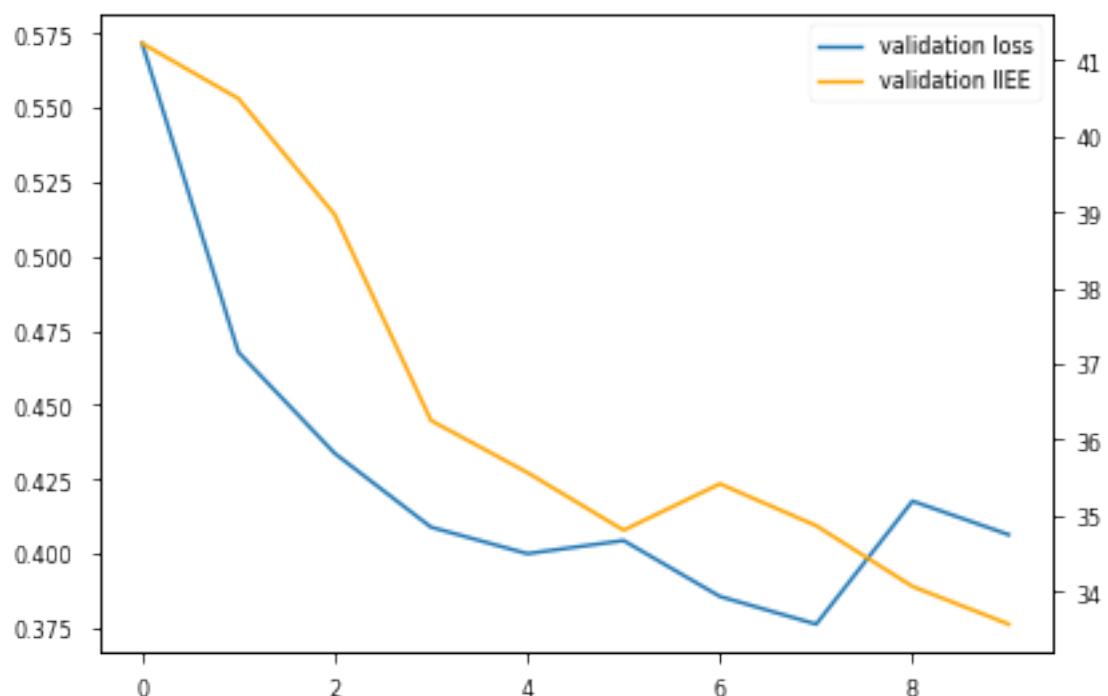


Figure 5: validation loss and Normalized IIEE computed as mean of validation set for each epoch during training

tmp  
figure,  
redo  
with  
in-  
spect  
data  
note-  
book  
in  
Fore-  
castVer-  
ifica-  
tion

On the other hand, it can be seen by inspecting Figure (5) that the Normalized IIEE tend to evolve conjunctually with the validation loss, in the current case defined as the mean cross entropy of all validation samples. Furthermore, the validation loss and Normalized IIEE in Figure (5) have a correlation of 0.82 with regards to epoch. Note that this has been calculated only using the numbers present in Figure (5). As such, there is reason to believe that selecting a model based on its validation loss, which is quick to compute, would result in a generalized model which may also excel at lowering its Normalized IIEE.

When selecting the best model, this project will apply the ModelCheckpoint callback with regards to validation loss as outlined above. ModelCheckpoint is preferred compared to EarlyStopping, as interrupting the training loop early may result in an "undercooked" model. E.g. the weights in earlier model layers are adjusting slower than later weights, giving the impression that the model training has reached a plateau which causes the model to stop. Whereas if the model where to continue training, the later adjustment of earlier weights would cause a later spur in increased model performance. ModelCheckpoint was chosen since behavior such as what was just exemplified is possible with the callback.

## 10 physical connections

### 10.1 Variograms

### 10.2 Case study

A case study is conducted where the highest reported IIEE value by the machine learning model.

### 10.3 Synthetic AA forcing

## 11 Comparing against physical models

The purpose of this section is twofold. Firstly, it aims at describing the process of preparing samples from the Barents-2.5 and NeXtSIM forecasting systems which are comparable to the Machine Learning forecasts at lead times of one, two and three days. Secondly, the performance of the forecasting systems will be assessed against the Sea Ice Charts, which are assumed to be the ground truth.

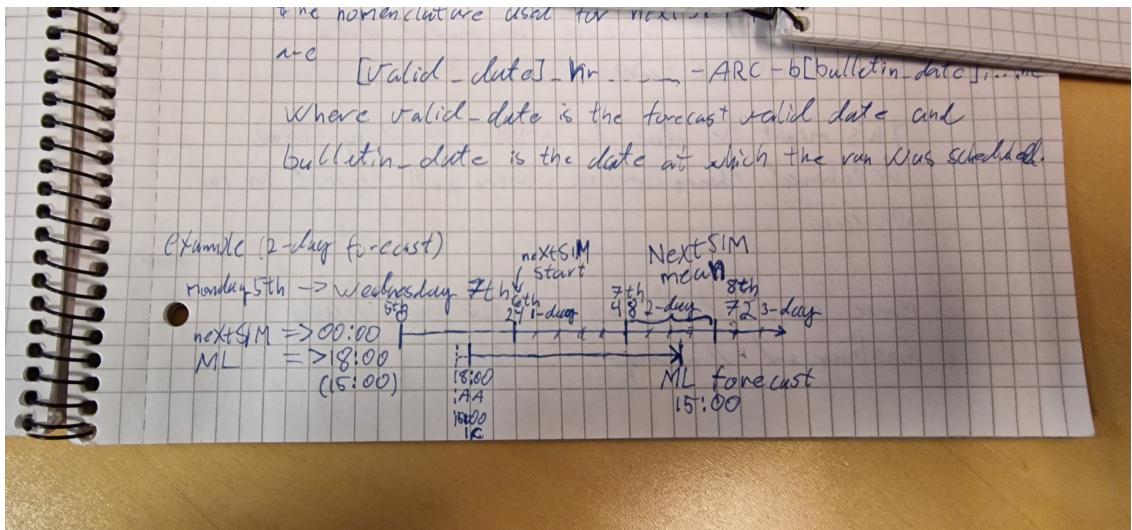


Figure 6: Sketch presenting how physical model forecasts are compared against machine learning forecasts. The axis represents time after 00:00 bulletin date of the machine learning forecast. The machine learning forecast is initiated 6 hours prior to the start of the physical model. The sketch exemplifies how the 2-day lead time machine learning forecast at 15:00 (reality 45 hours) is compared against an entire second day of a physical forecast (lead times 24 - 47).

## 11.1 Preparing data

The logic behind sample creation is similar for both physical models. The idea is that the bulletin date of the physical forecasting system is +1 the bulletin date of the machine learning forecast. Furthermore, a daily mean is computed from the forecast based on the lead time of the forecast. I.e., a 1 day lead time for the machine learning forecast would constitute a daily mean of the first 24 hours forecasted by a physical forecasting system starting at 00 the following day of the machine learning bulletin date.

## References

- [1] Tom R. Andersson et al. “Seasonal Arctic sea ice forecasting with probabilistic deep learning”. In: *Nature Communications* 12.1 (Aug. 2021). DOI: [10.1038/s41467-021-25257-4](https://doi.org/10.1038/s41467-021-25257-4).

Figure (6)  
to be  
made  
profes-  
sional,  
using  
e.g.  
TiX

- [2] Magdalena Alonso Balmaseda, Kristian Mogensen, and Anthony T. Weaver. “Evaluation of the ECMWF ocean reanalysis system ORAS4”. In: *Quarterly Journal of the Royal Meteorological Society* 139.674 (2013), pp. 1132–1161. DOI: <https://doi.org/10.1002/qj.2063>. eprint: <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.2063>. URL: <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2063>.
- [3] B. Casati et al. “Forecast verification: current status and future directions”. In: *Meteorological Applications* 15.1 (2008), pp. 3–18. DOI: [10.1002/met.52](https://doi.org/10.1002/met.52).
- [4] D. J. Cavalieri and C. L. Parkinson. “Arctic sea ice variability and trends, 1979–2010”. In: *The Cryosphere* 6.4 (Aug. 2012), pp. 881–889. DOI: [10.5194/tc-6-881-2012](https://doi.org/10.5194/tc-6-881-2012).
- [5] Josefino C. Comiso, Walter N. Meier, and Robert Gersten. “Variability and trends in the <scp>A</scp> rctic <scp>S</scp> ea ice cover: Results from different techniques”. In: *Journal of Geophysical Research: Oceans* 122.8 (Aug. 2017), pp. 6883–6900. DOI: [10.1002/2017jc012768](https://doi.org/10.1002/2017jc012768).
- [6] Véronique Dansereau et al. “A Maxwell elasto-brittle rheology for sea ice modelling”. In: *The Cryosphere* 10.3 (July 2016), pp. 1339–1359. DOI: [10.5194/tc-10-1339-2016](https://doi.org/10.5194/tc-10-1339-2016).
- [7] D. P. Dee et al. “The ERA-Interim reanalysis: configuration and performance of the data assimilation system”. In: *Quarterly Journal of the Royal Meteorological Society* 137.656 (2011), pp. 553–597. DOI: <https://doi.org/10.1002/qj.828>. eprint: <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.828>. URL: <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.828>.
- [8] Frode Dinessen, Bruce Hackett, and Matilde Brandt Kreiner. *Arctic Ocean - Sea Ice Concentration Charts - Svalbard and Greenland*. en. 2015. DOI: [10.48670/MOI-00128](https://doi.org/10.48670/MOI-00128).
- [9] Dmitry S. Dukhovskoy et al. “Skill metrics for evaluation and comparison of sea ice models”. In: *Journal of Geophysical Research: Oceans* 120.9 (Sept. 2015), pp. 5910–5931. DOI: [10.1002/2015jc010989](https://doi.org/10.1002/2015jc010989).
- [10] Victor M. Eguíluz et al. “A quantitative assessment of Arctic shipping in 2010–2014”. In: *Scientific Reports* 6.1 (Aug. 2016). DOI: [10.1038/srep30682](https://doi.org/10.1038/srep30682).
- [11] Sindre Fritzner, Rune Graversen, and Kai H. Christensen. “Assessment of High-Resolution Dynamical and Machine Learning Models for Prediction of Sea Ice Concentration in a Regional Application”. In: 125.11 (Oct. 2020). Neural Networks for predicting Sea-Ice concentration are only slightly more accurate than persistence forecasting for short-term predictions. DOI: [10.1029/2020jc016277](https://doi.org/10.1029/2020jc016277).

- [12] H. F. Goessling and T. Jung. “A probabilistic verification score for contours: Methodology and application to Arctic ice-edge forecasts”. In: *Quarterly Journal of the Royal Meteorological Society* 144.712 (Apr. 2018), pp. 735–743. DOI: [10.1002/qj.3242](https://doi.org/10.1002/qj.3242).
- [13] H. F. Goessling et al. “Predictability of the Arctic sea ice edge”. In: *Geophysical Research Letters* 43.4 (Feb. 2016), pp. 1642–1650. DOI: [10.1002/2015gl067232](https://doi.org/10.1002/2015gl067232).
- [14] Kaiming He et al. *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*. 2015. DOI: [10.48550/ARXIV.1502.01852](https://doi.org/10.48550/ARXIV.1502.01852).
- [15] Hans Hersbach et al. “The ERA5 global reanalysis”. In: *Quarterly Journal of the Royal Meteorological Society* 146.730 (2020), pp. 1999–2049. DOI: <https://doi.org/10.1002/qj.3803>. eprint: <https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3803>. URL: <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3803>.
- [16] W. D. Hibler. “A Dynamic Thermodynamic Sea Ice Model”. In: *Journal of Physical Oceanography* 9.4 (July 1979), pp. 815–846. DOI: [10.1175/1520-0485\(1979\)009<0815:adtsim>2.0.co;2](https://doi.org/10.1175/1520-0485(1979)009<0815:adtsim>2.0.co;2).
- [17] Joshua Ho. “The implications of Arctic sea ice decline on shipping”. In: *Marine Policy* 34.3 (May 2010), pp. 713–715. DOI: [10.1016/j.marpol.2009.10.009](https://doi.org/10.1016/j.marpol.2009.10.009).
- [18] E. C. Hunke and J. K. Dukowicz. “An Elastic–Viscous–Plastic Model for Sea Ice Dynamics”. In: *Journal of Physical Oceanography* 27.9 (Sept. 1997), pp. 1849–1867. DOI: [10.1175/1520-0485\(1997\)027<1849:aevpmf>2.0.co;2](https://doi.org/10.1175/1520-0485(1997)027<1849:aevpmf>2.0.co;2).
- [19] Elizabeth C. Hunke et al. *CICE: the Los Alamos Sea Ice Model Documentation and Software User’s Manual Version 5.1 LA-CC-06-012*. Tech. rep. Los Alamos NM 87545: Los Alamos National Laboratory, July 2015.
- [20] JCOMM Expert Team on Sea Ice. *Sea-Ice Nomenclature: snapshot of the WMO Sea Ice Nomenclature WMO No. 259, volume 1 – Terminology and Codes; Volume II – Illustrated Glossary and III – International System of Sea-Ice Symbols* . 2014. DOI: [10.25607/0BP-1515](https://doi.org/10.25607/0BP-1515).
- [21] S. J. Johnson et al. “SEAS5: the new ECMWF seasonal forecast system”. In: *Geoscientific Model Development* 12.3 (2019), pp. 1087–1117. DOI: [10.5194/gmd-12-1087-2019](https://doi.org/10.5194/gmd-12-1087-2019). URL: <https://gmd.copernicus.org/articles/12/1087/2019/>.
- [22] Nils M Kristensen et al. *Metno/Metroms: Version 0.3 - Before Merge*. 2017. DOI: [10.5281/ZENODO.1046114](https://doi.org/10.5281/ZENODO.1046114).

- [23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012. URL: <https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>.
- [24] T. Lavergne et al. “Version 2 of the EUMETSAT OSI SAF and ESA CCI sea-ice concentration climate data records”. In: *The Cryosphere* 13.1 (2019), pp. 49–78. DOI: [10.5194/tc-13-49-2019](https://doi.org/10.5194/tc-13-49-2019). URL: <https://tc.copernicus.org/articles/13/49/2019/>.
- [25] Tsung-Yi Lin et al. “Focal Loss for Dense Object Detection”. In: (Aug. 2017). arXiv: [1708.02002 \[cs.CV\]](https://arxiv.org/abs/1708.02002).
- [26] Yang Liu et al. “Extended Range Arctic Sea Ice Forecast with Convolutional Long-Short Term Memory Networks”. In: *Monthly Weather Review* (Mar. 2021). DOI: [10.1175/mwr-d-20-0113.1](https://doi.org/10.1175/mwr-d-20-0113.1).
- [27] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully Convolutional Networks for Semantic Segmentation”. In: (Nov. 2014). arXiv: [1411.4038 \[cs.CV\]](https://arxiv.org/abs/1411.4038).
- [28] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: <https://www.tensorflow.org/>.
- [29] Wes McKinney. “Data Structures for Statistical Computing in Python”. In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman. 2010, pp. 56–61. DOI: [10.25080/Majora-92bf1922-00a](https://doi.org/10.25080/Majora-92bf1922-00a).
- [30] Arne Melsom, Cyril Palerme, and Malte Müller. “Validation metrics for ice edge position forecasts”. In: *Ocean Science* 15.3 (May 2019), pp. 615–630. DOI: [10.5194/os-15-615-2019](https://doi.org/10.5194/os-15-615-2019).
- [31] Malte Müller et al. “Characteristics of a Convective-Scale Weather Forecasting System for the European Arctic”. In: *Monthly Weather Review* 145.12 (Dec. 2017), pp. 4771–4787. DOI: [10.1175/mwr-d-17-0194.1](https://doi.org/10.1175/mwr-d-17-0194.1).
- [32] Vinod Nair and Geoffrey E. Hinton. “Rectified Linear Units Improve Restricted Boltzmann Machines”. In: *ICML*. 2010, pp. 807–814. URL: <https://icml.cc-Conferences/2010/papers/432.pdf>.
- [33] Dirk Notz and SIMIP Community. “Arctic Sea Ice in CMIP6”. In: *Geophysical Research Letters* 47.10 (May 2020). DOI: [10.1029/2019gl086749](https://doi.org/10.1029/2019gl086749).
- [34] Cyril Palerme, Malte Müller, and Arne Melsom. “An Intercomparison of Verification Scores for Evaluating the Sea Ice Edge Position in Seasonal Forecasts”. In: *Geophysical Research Letters* 46.9 (May 2019), pp. 4757–4763. DOI: [10.1029/2019gl082482](https://doi.org/10.1029/2019gl082482).

- [35] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: (May 2015). arXiv: [1505 . 04597 \[cs.CV\]](https://arxiv.org/abs/1505.04597).
- [36] Johannes Röhrs et al. “”in prep for GMD” An operational data-assimilative coupled ocean and sea ice ensembleprediction model for the Barents Sea and Svalbard”. In: (2022), p. 20.
- [37] Pavel Sakov and Peter R. Oke. “A deterministic formulation of the ensemble Kalman filter: an alternative to ensemble square root filters”. In: *Tellus A: Dynamic Meteorology and Oceanography* 60.2 (Jan. 2008), p. 361. doi: [10.1111/j.1600-0870.2007.00299.x](https://doi.org/10.1111/j.1600-0870.2007.00299.x).
- [38] Mark C. Serreze and Walter N. Meier. “The Arctic’s sea ice cover: trends, variability, predictability, and comparisons to the Antarctic”. In: *Annals of the New York Academy of Sciences* 1436.1 (2019), pp. 36–53. doi: <https://doi.org/10.1111/nyas.13856>.
- [39] Gunnar Spreen, Ron Kwok, and Dimitris Menemenlis. “Trends in Arctic sea ice drift and role of wind forcing: 1992-2009”. In: *Geophysical Research Letters* 38.19 (Oct. 2011), n/a–n/a. doi: [10.1029/2011gl048970](https://doi.org/10.1029/2011gl048970).
- [40] The pandas development team. *pandas-dev/pandas: Pandas*. Version latest. Feb. 2020. URL: <https://doi.org/10.5281/zenodo.3509134>.
- [41] S Veland et al. *Knowledge needs in sea ice forecasting for navigation in Svalbard and the High Arctic*. Tech. rep. NF-rapport 4/2021. Svalbard Strategic Grant, Svalbard Science Forum, 2021.
- [42] Penelope Mae Wagner et al. “Sea-ice information and forecast needs for industry maritime stakeholders”. In: *Polar Geography* 43.2-3 (June 2020), pp. 160–187. doi: [10.1080/1088937x.2020.1766592](https://doi.org/10.1080/1088937x.2020.1766592).
- [43] Timothy Williams et al. “Presentation and evaluation of the Arctic sea ice forecasting system neXtSIM-F”. In: *The Cryosphere* 15.7 (July 2021), pp. 3207–3227. doi: [10.5194/tc-15-3207-2021](https://doi.org/10.5194/tc-15-3207-2021).
- [44] Xiaoyong Yu et al. “Evaluation of Arctic sea ice drift and its dependency on near-surface wind and sea ice conditions in the coupled regional climate model HIRHAM-NAOSIM”. In: *The Cryosphere* 14.5 (May 2020), pp. 1727–1746. doi: [10.5194/tc-14-1727-2020](https://doi.org/10.5194/tc-14-1727-2020).