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[1,2]

[1]Development Centre for Weather Forecasting, Norwegian Meteorological Institute
[2]Department of Geosciences, University of Oslo

February 15, 2023

Contents

1 Introduction

The Arctic sea ice extent has continuously decreased since the first satellite observations of the Arctic was obtained in 1978 (Serreze and Meier, 2019), with an average decrease of 4% per decade (Cavalieri and Parkinson, 2012). The summer months are experiencing the greatest loss of sea ice extent (Comiso et al., 2017), with models from the Coupled Model Intercomparison Project Phase 6 (CMIP6) projecting the first sea ice-free Arctic summer before 2050 (Notz and Community, 2020). As a consequence of the sea ice retreat during the summer months, previously inaccessible oceanic areas have opened up causing an increase in maritime operations in the Arctic waters (Ho, 2010; Eguíluz et al., 2016). The expected influx of operators to the Arctic regions due to the prolonged open water season call for user-centric sea ice products on different spatial scales and resolutions to ensure maritime safety in the region (Wagner et al., 2020; Veland et al., 2021).

Current information on Arctic sea ice concentration can be discerned into several types of products with different spatial and temporal resolutions. Sea ice products designed for climate applications such as OSI-450, SICCI-25km and SICCI-50km provide daily sea ice concentration by merging observations from multiple sensors to create a historical dataset. The purpose of a climatology is to provide accurate reference data (Lavergne et al., 2019a) which can be used for e.g. forecast validation or anomaly detection. Satellite observations are also supplied as daily products, with a timeliness of a few hours on the same day and posing higher spatial resolutions than climatologies. For example, OSI-401-b (Tonboe et al., 2017) and OSI-408 (Lavelle et al., 2016) provide single sensor daily averaged sea ice concentration covering the northern and southern hemisphere, and can be used to force numerical weather prediction systems which only resolve the atmosphere (Müller et al., 2017).

Sea ice models are physically based models resolving the growth and movement of sea ice forward in time. Standalone models such as CiCE (Hunke and Dukowicz, 1997) and neXtSIM (Williams et al., 2021) can be used independently or coupled with ocean models (Röhrs et al., 2022) to create sea ice forecasting systems for short lead times. Finally, sea ice charts drawn analogously by a sea ice specialist merge recent sea ice observations from different sensors and satellites into a single daily product. The Ice Service of the Norwegian Meteorological Institute (NIS) provides regional ice charts covering the European Arctic. The product consists of polygons which are drawn to match the current resolution of the available observations, which range from 50m to several kilometers, and are assumed to have a low uncertainty due to the quality control exerted by the sea ice specialist (Dinessen et al., 2020).

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 application scenario for maritime end users. While lower resolution products at larger

temporal time scales can be used in long term planning, regional high resolution products delivered at a high frequency can assist strategic decision making and short term route planning (Wagner et al., 2020). However, it is currently reported by end users that available operational passive microwave satellite products are of a too low resolution, partly due to their insufficient ability to resolve leads and other high-resolution information necessary for maritime safety. Moreover, it is also reported that sea ice forecasting systems lack desired verification, are inadequate for operational use as well as being difficult to integrate with a vessel where computational resources and data-bandwidth are limited (Veland et al., 2021). Though sea ice charts provide maritime end users 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 does not provide a description on the future outlook. Thus, the responsibility of interpreting the ice charts and other available sea ice information with a outlook on future development is delegated to the end-user and relies on their experience to ensure a continued safe navigation (Veland et al., 2021).

As such, a different approach to short-range sea ice forecasting may be necessary to deliver short-term sea ice information on a spatial scale that is relevant for end-users. Thus, this thesis proposes an alternative forecasting scheme that applies Convolutional deep learning in the form of a modified U-Net architecture (Ronneberger et al., 2015) to deliver a short lead time (1 - 3 days), 1km resolution forecasting product over a subsection of the European 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 (Veland et al., 2021; Melsom et al., 2019).

There have been made previous attempts to develop deep learning sea ice forecasting systems. Andersson et al. (2021) 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 (Andersson et al., 2021). 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 (Johnson et al., 2019) 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 (Lavergne et al., 2019a), as well as other climate variables obtained from the ERA5 reanalysis (Hersbach et al., 2020). Their model was validated against SEAS5, which is a seasonal forecasting system developed by the European Center for Medium-Range Weather Forecasts (ECMWF) (Johnson et al., 2019).

Similarly, Liu et al. (2021) 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 (Dee et al., 2011) and

ORAS4 (Balmaseda et al., 2013), 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 short-term sea ice forecast. The possibility of using higher resolution input data was explored by Fritzner et al. (2020), 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 (Kristensen et al., 2017). However, due to computational constraints of training the FCN, the subdomain was reduced to a resolution of 224 x 224 pixels which translates to 10 - 20km (Fritzner et al., 2020). Thus, the product has a limited accuracy for short term operational usage, similar to (Andersson et al., 2021) and (Liu et al., 2021).

Contrary to the authors above, Grigoryev et al. (2022) propose a 10 day lead time regional forecasting system with a 5km spatial resolution trained on a sequential (traditional) and recurrent U-Net architecture. The authors used 5km AMSR-2 sea ice concentration as the ground truth variable, and regrid atmospheric variables from the NCEP Global Forecast System (https://www.emc.ncep.noaa.gov/emc/pages/numerical_forecast_systems/gfs.php) to match the resolution of the ground truth. Their results showed that the recurrent setup slightly outperformed the sequential architecture for predictions with a lead time up to 3 days, with both architectures significantly outperforming persistence and the linear trend. However, the sequential architecture tended to outperform the recurrent architecture for 10 day forecasts, as the recurrent model was trained without weather data as it only had a lead time of 3 days.

As mentioned in (Andersson et al., 2021; Fritzner et al., 2020), 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 (Andersson et al., 2021). 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 works.

Moreover, the U-Net architecture is part of the supervised learning paradigm of machine learning, which require labelled samples in order to train the network (Ronneberger et al., 2015). 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. This allows for the inspection of how changes to the architecture as well as input data configurations affect the behavior of the forecasting system.

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 affecting 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 linear regression of the observed sea ice concentration. Due to the operational nature of the developed forecasting product, ice edge aware validation metrics such as the Integrated Ice Edge Error (Goessling et al., 2016) 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.

This thesis aims at exploring the following research questions:

- 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 U-Net 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 and final section presenting conclusions and future outlook.

2 Datasets

[Training and validating a deep learning system requires data, which can be categorized in two distinct groups. The first group is the data known by the system, which is used during training to increase or validate model performance. Additional to the data used during training is external data, which is needed to validate the generalizability of the model. I.e., how well does the model perform with unknown data, which is assumed to be drawn from the same distribution as the data used during training. It is standard practice to arbitrarily split by a given fraction into the three datasets (training, validation, testing), as outlined above. However, due to the variable seasonal dependency of meteorological data, a naive split of the data could result in seasonally unbalanced datasets. As such, the datasets constructed for this thesis's purpose cover at least a full year. Thus, no dataset is assumed to be skewed in the direction of any season.]

Dette passer bedre inn i en train-test-split **model development**

To facilitate the development and verification of a high resolution short-term deep learning sea ice forecasting system, several datasets from observations and physical model forecasting systems have been chosen. When selecting appropriate datasets, their spatial resolution as well as release frequency has been considered. Even though several observational sea ice concentration products which cover the region of interest exists, a lot of the satellite products based on passive microwave retrievals are of a too coarse resolution (e.g. Lavergne et al. (2019a) or Kern et al. (2019)) to be able to aid in short term decision making (Wagner et al., 2020). On the other hand Synthetic Aperture Radar (SAR) observations such as Sentinel 1A Interferometric Wide swath (5m $\times$ 20m) or Extra-Wide swath (20m $\times$ 40m) are on a sea ice structure resolving spatial resolution. However the daily SAR coverage is sparse in the Arctic (See Supporting Figure 15) and there are currently no sea ice concentration product based on retrieval algorithms of SAR observations which are known to the author.

Moreover, forecasts can be used as predictors for the deep learning system since they provide information regarding how the conditions should evolve in the period after the forecast has been initialized. Thus giving the deep learning system insight into the future state of the domain while still facilitating operational usage by not relying on e.g. future observations. Hence, atmospheric variables from a regional numerical weather prediction system will be included as input to the model. These variables (wind and temperature) have been chosen due to their physical impact on sea ice, and is assumed to encode information about the future state of sea ice concentration when seen in combination with past and present sea ice concentration by the deep learning system.

Finally, the highest resolution product with an appropriate temporal frequency available are the sea ice charts produced by the NIS (Dinessen et al., 2020). Moreover, the sea ice charts represents an interpretation of different sea ice observations delivered as a product directed towards operational users. Thus, the sea ice charts will serve as the ground truth

Table 1: List of the products used and their applications. The dashed line separates observational products (above) from forecast products (below)

Product	Variables	Training	Verification
Ice charts	SIC	Yes	Yes
OSI-SAF SSMIS	SIC trend	Yes	Yes
OSI-SAF CDR	Ice edge length	No	Yes
AMSR2	SIC	No	Yes
AROME-Arctic	T2M, X-wind, Y-wind	Yes	No
NeXtSIM	SIC	No	Yes
Barents-2.5	SIC	No	Yes

for the model. Furthermore, as a deep learning system can increase its skill by combining correlated variables as input, this thesis will explore the impact caused by including several datasets covering both current observations, past trends as well as forecasted variables on different spatial resolutions as input predictors.

The following section will describe the domain covered for this thesis, followed by a run-down of the satellite products as well as physical models used. Table 1 presents the different products used for this thesis, and whether the product is used to train or verify the model.

2.1 Region of interest

The domain covered by the deep learning system, covers part of the European Arctic. The region is an intersection between the domain covered by the Ice Charts (Dinessen et al., 2020) and AROME Arctic (Müller et al., 2017) as shown in Figure (1). The domain has a 1km spatial resolution, and contains 1972×1972 equidistant grid points. Compared to the AROME Arctic grid, the model domain has a reduced southern and eastern extent, which is manually subsectioned to conform to the square grid imposed by the U-Net architecture (Ronneberger et al., 2015). Another reason for the reduced domain extent was to limit the amount of memory needed when loading data during training. Both reasons will be thoroughly discussed in later sections. Moreover, the choice of limiting the southern and eastern extent of the domain was deliberate to reduce the amount of likely sea ice concentration containing grid cells lost in the process.

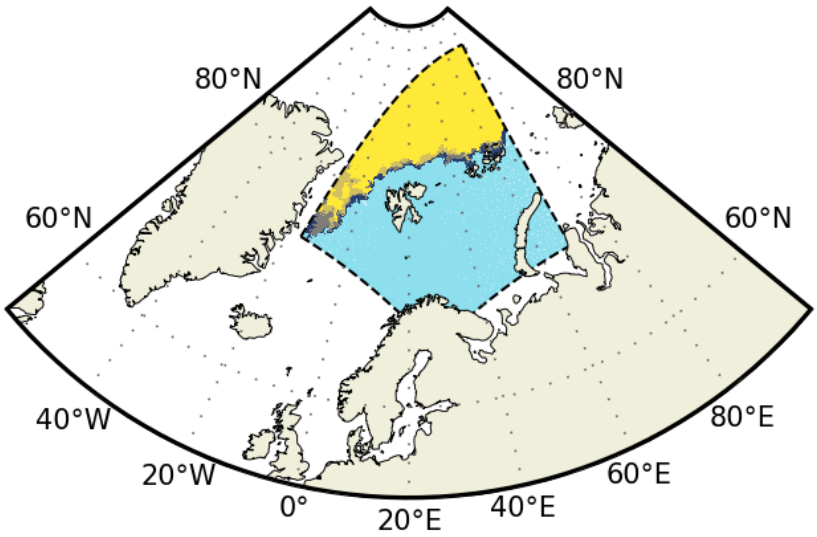


Figure 1: The model domain is shown by sea ice concentration contours retrieved from a sea ice chart (15 Sep 2022). No colorbar is shown, light blue is free open water and yellow is fast ice.

2.2 Observations

Observations are used to convey the current state of sea ice concentration. There is a lack of consistent in situ observations of sea ice concentration, due to the remoteness of the region. Thus, most independent observations are ship-based concentration estimates (Kern et al., 2019) or optical remote sensing, the latter is only available during summer. As a result, sea ice concentration is mainly observed automatically through passive microwave retrievals utilizing different sea ice retrieval algorithms (Lavergne et al., 2019b; Comiso et al., 1997; Spreen et al., 2008). Another source of sea ice observations are sea ice charts (https://usicecenter.gov/, Last Accessed 25 Jan 2023) (Dinessen et al., 2020), which are manually drawn interpretations combining available sea ice concentration observations such as SAR, passive microwave and optical imagery.

2.2.1 Sea Ice Charts

The sea ice charts utilized for this thesis are provided by the Norwegian Meteorological Institute through the National Ice Service. The product is manually drawn by a sea ice specialist, and is distributed every workday at 15:00 UTC. The Sea Ice specialist

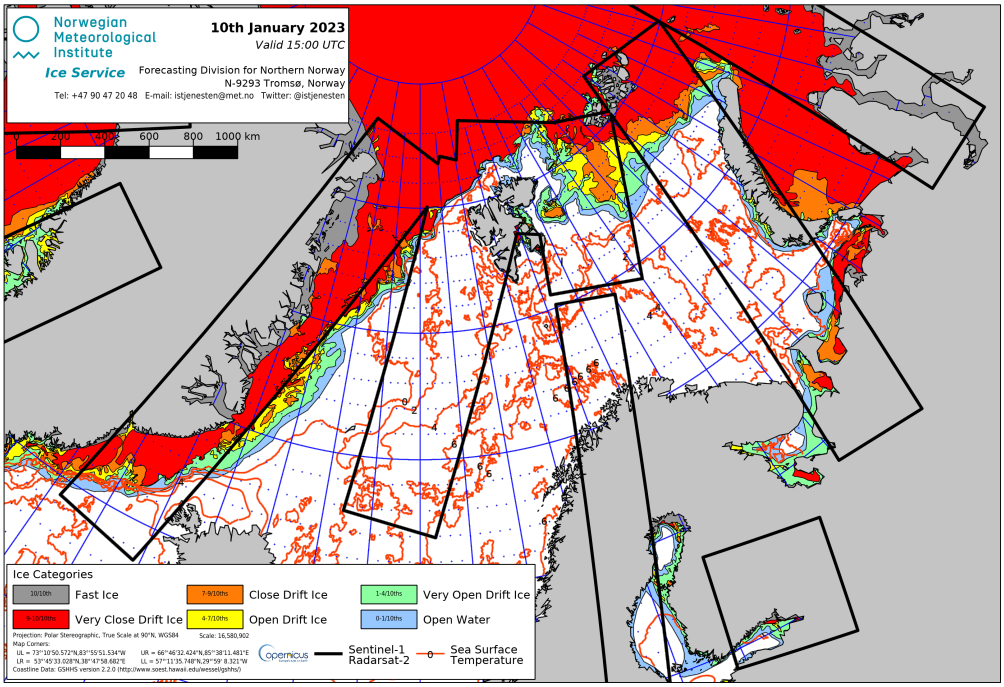


Figure 2: Sea Ice chart produced by the NIS covering 10 Jan 2023 at 15:00 UTC. Sea ice concentration categories are drawn as filled contours. The black lines indicate the available SAR data used to draw the sea ice chart.

assesses available SAR data from Sentinel 1 and Radarsat 2. However, due to the spatial variability in daily SAR coverage (See Supporting Figure (15)), visual, infrared and low resolution passive microwave observations are supplied to achieve a consistent spatial coverage (Dinessen et al., 2020). The sea ice charts are not drawn onto any resolution. Hence, a gridded representation of the ice charts is only a representation of the mean value of the polygons contained inside each grid cell. The sea ice charts used in this work has been interpolated onto a 1-km grid with the same projection as AROME Arctic (Müller et al., 2017).

With regards to consistency, it is noted that the current sea ice chart product have no easily identifiable way of noting which observations were used by the sea ice analyst to draw each segment of the chart. As the different satellite products used have different spatial scales, from meters to kilometers (Dinessen et al., 2020), the underlying uncertainty and ability to resolve structures varies both spatially and temporally. The published sea ice charts as seen in Figure (2) shows the available SAR coverage as black contours, which is the preferred data source for the ice analysts (Dinessen et al., 2020).

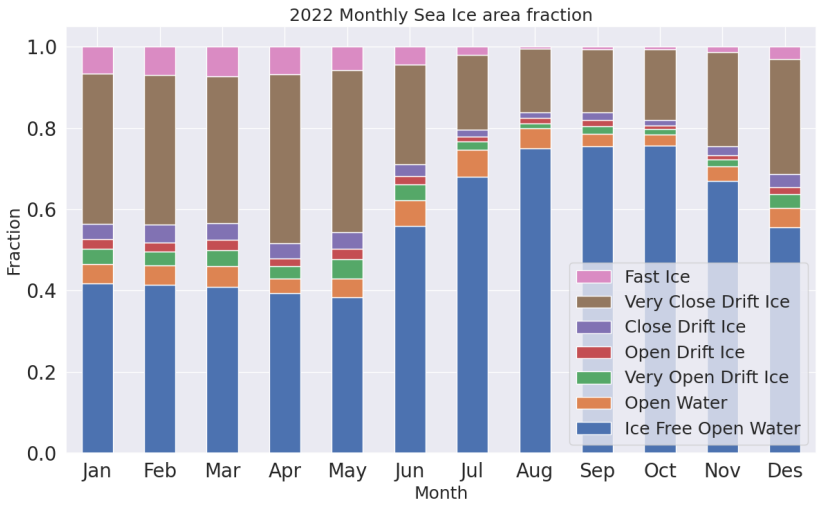


Figure 3: Monthly distribution of each concentration class as respective fraction of the total mean sea ice concentration for the sea ice charts covering 2022. **[Could extend to cover larger time period (e.g. from 2011), give a more climate perspective of the sea ice evolution], Add concentration ranges for each class**

Figure (3) shows the monthly distribution of sea ice concentration contours from the sea ice charts during the period of 2022. As can be seen from the figure, almost half of the region consists of ice free open water, with the other majority of an ice chart consisting of very close drift ice. Moreover, the figure shows the seasonal variability of the sea ice extent, with the ice free open water contributing between $\sim 38\%$ and $\sim 75\%$ of the entire domain depending on the month. The ice charts also resolve the intermediate sea ice concentration classes, which for the current region is mostly related to the marginal ice zone and the ice edge.

By inspecting Figure (4), it can be seen that the autocorrelation between two ice charts close in time is high. However, it can also be seen to steadily decline as the time lag increases. From the strong autocorrelation seen in Figure (4), it can be assumed that the persistence for short lead times (days) closely relate to the current sea ice concentration. Furthermore, the autocorrelation also renders previous sea ice concentration at short timescales as skillful at describing the current growth of the sea ice. The latter will be used as motivation to compute a sea ice concentration trend in a coming subsection.

The Sea Ice charts is an operational product aimed at marine end users. This influences the decision-making when creating the final operational product. Furthermore, though a single ice chart is assumed to be drawn by the same person, there are several sea ice specialists at the NIS whom draw ice charts. Hence, it can be assumed that there is an

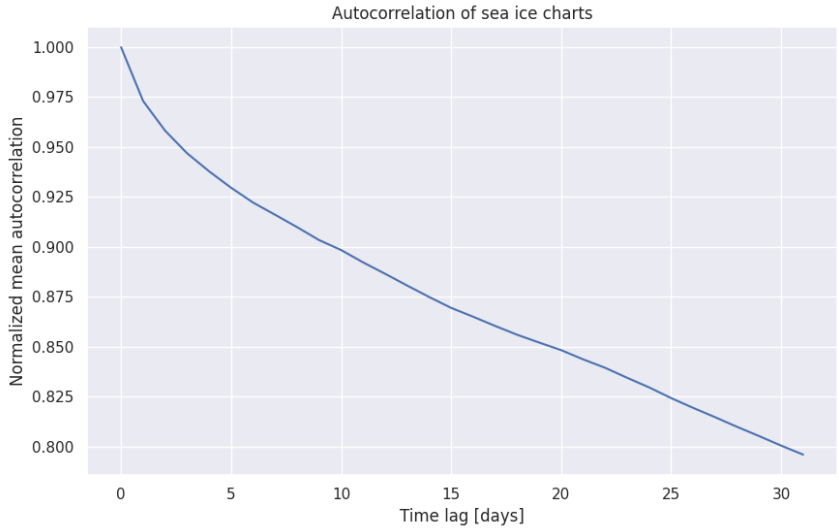


Figure 4: Autocorrelation of the sea ice charts from 2022. The x-axis is the time lag between to entries 2022 sea ice chart timeseries. The y-axis is the normalized autocorrelation, i.e. autocorrelation at a certain time lag divided by the autocorrelation at time lag 0. The autocorrelation is computed for a period covering 31 days.

unknown degree of personal bias to data. On the other hand, the human involvement may also introduce a degree of quality control not seen in automatic sea ice concentration retrieval algorithms. Thus, the ice charts are assumed to have a low uncertainty, though there are no uncertainty estimates included (Dinessen et al., 2020).

In spite of the uncertainties outlined above, the sea ice charts are assumed to be the most accurate sea ice concentration product available for the purpose of high resolution data tailored towards operational end users. Through utilizing the sea ice charts as the ground truth data when training the deep learning system, the developed model will fit towards the high resolution operational use case proposed.

2.2.2 OSI SAF

Two different sea-ice Concentration products are used from OSI SAF. OSI SAF Special Sensor Microwave Imager and Sounder (SSMIS) is an operational product delivering daily sea ice concentration on the northern (and southern) hemisphere. OSI SAF Climate Data Record (CDR) (Sørensen et al., 2021) delivers sea ice concentration beginning in 1979 (Lavergne et al., 2019a) The operational product will be used as a predictor for the model and for validation, whereas the CDR will be used only for validation purposes only.

OSI SAF SSMIS

OSI SAF SSMIS is a passive microwave product derived from the (SSMIS) instrument. To convert brightness temperature to estimated sea ice concentration, a hybrid approach combining the Bootstrap algorithm (Comiso et al., 1997) and the Bristol algorithm (Smith, 1996) where the prior is used over open water and the latter used for ice concentrations above 40% (Tonboe et al., 2017). The algorithm uses data from the 19GHz frequency channel (Vertically polarized) and 37GHz channel (Vertically and Horizontally polarized), which are the two lowest spectral resolution channels for the SSMIS Tonboe et al. (2017). Finally, atmospheric corrections are made using analyses from the European Center for Medium Range Weather Forecasts (ECMWF). The end product is delivered every day on a 10km polar stereographic grid.

With regards to uncertainty, OSI SAF SSMIS is validated against pan-arctic sea ice charts from the U.S. National Ice Center as well as regional sea ice charts covering the Svalbard region from the NIS. Moreover, the operational product is required to have a bias and standard deviation less then 10% ice concentration on an annual basis, when compared to the targets (https://osisaf-hl.met.no/sea-ice-conc-edge-validation, Last Accessed 24 Jan 2023) (Lavelle et al., 2017). This strengthens the assumption made at the end of Section (2.2.1) regarding the accuracy of the sea ice charts and their validity in terms of serving as an independent source for reference.

The operational OSI SAF SSMIS dataset is used to compute a coarse resolution (with respect to the ice charts) linear sea ice concentration trend in each grid cell, with a short term length covering a given amount of days backwards in time. The idea behind the computed trend is to encode multiple time-steps of sea ice concentration fields into a single 2d-array, in line with the lack of temporal awareness of the U-Net architecture. Moreover, the trend serve to limit the size of the training data, since the memory needed is equal to that of a single 2d-array regardless of the length of the trend. Furthermore, the ice concentration trend is computed from a separate sea ice product than the ice chart, with the intent to supply the model with correlated but not overlapping information, as the current day ice chart is already used as a predictor. However, it should also be noted that the lack of sea ice charts during the weekends (Dinessen et al., 2020) is also a contributing factor. As a sea ice concentration trend derived from Dinessen et al. (2020) would be limited to at most five days, which is not the case for OSI SAF SSMIS as there are no temporal gaps in the dataset. The coarser resolution also contributes to the OSI SAF trend serving as complementary information to the ice charts, as the coarse resolution makes the trend less resolvent of the local variability which is seen in the ice charts. As such, the trend serves as a indicator of where the sea ice growth / decline is occurring.

The temporal length used when deriving the trend will have an impact on how the trend

reflects the current growth and decline zones, especially with regards to the volatile position of the ice edge on a daily timescale but also due to the seasonal variability of the ice area (Holland and Kimura, 2016). Hence, a too large lookbehind would cause a decorrelation between the current sea ice concentration and computed trend. On the other hand, Figure (4) shows that there there is significant autocorrelation for sea ice concentration on a short time-range, as described previously. However, a trend computed from a sufficiently long temporal window could be assumed to better represent the spatial distribution of seasonal sea ice concentration growth and decline rather than representing the current growth and decline.

OSI SAF Climate Data Record

As briefly mentioned in Section (1), OSI SAF Climate Data record combines observations from different sensors (SMMR, SSM/I, SSMIS) as well as numerical weather prediction fields from the ERA Interim reanalysis (Dee et al., 2011). The latter are utilized to correct for the atmospheric conditions. Two versions of the dataset has been used, version 2 (OSI-450) which covers (2011 - 2015), and the interim version (OSI-430-b) which cover (2016 - 2020) (https://osisaf-hl.met.no/osi-450-430-b-desc)(Last Accessed 18 Jan 2023). Both products are processed using the same algorithms, ensuring consistency (Lavergne et al., 2019b). The Interim version is serving as an extension of the original scope of OSI-450 (1979 - 2015), with a difference being its use of ECMWF analyses compared to the reanalysis and different SSMIS input data (https://osisaf-hl.met.no/osi-450-430-b-desc, Last Accessed 24 Jan 2023). Regardless, both products will hereby be referred to in tandem as OSI SAF CDR

The OSI SAF retrieval algorithm has been shown to have strong correlation against ship based measurements (Kern et al., 2019) as well as optical satellite observations during the summer (Kern et al., 2020). Hence, OSI SAF CDR is expected to serve as a correct representation of the Arctic sea ice concentration, however it is noted that no retrieval algorithm is able to match the true state of the sea ice concentration.

OSI SAF CDR is provided with a 25km spatial resolution on a Lambert Azimuthal Grid projection (Sørensen et al., 2021). The sea ice concentration data retrieved has been used to compute a climatological ice edge length for each day of the year, applying a daily mean across the time period (2011 - 2020). The ice edge length has been computed according to Melsom et al. (2019), which will be derived in Section (not yet labelled). Note that though OSI SAF CDR provides a pan-arctic distribution of sea ice concentration, the data has been regridded onto the study region domain with the AROME Arctic projection and a 25km grid spacing before computing the ice edge length.

As can be seen in Figure (5), the Arctic sea ice edge experiences a strong seasonal variability. The computed climatological ice edge will be used as a normalization factor in order to use verification scores that are not seasonally dependent (Goessling et al., 2016;

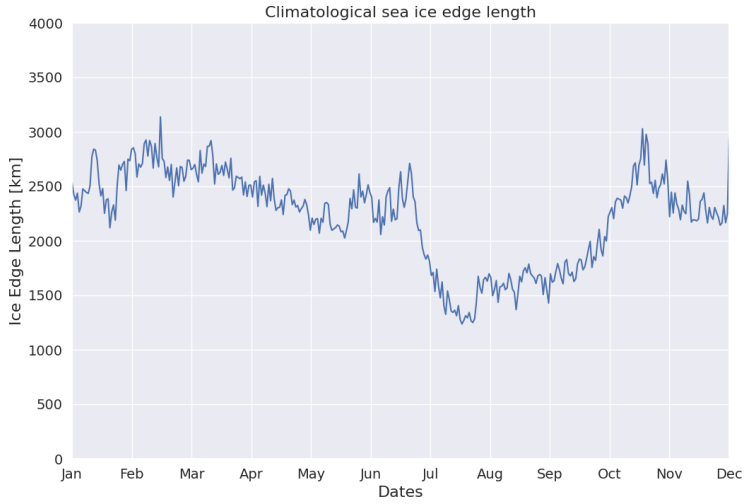


Figure 5: **FIX NONSENSICAL XTICKLABELS** Seasonal variability of the Arctic climatological ice edge length computed from satellite observations during the period 2011 - 2020

Zampieri et al., 2019; Palerme et al., 2019). Another benefit from utilizing a single ice edge length is to ensure that different sea ice products are normalized according to a common and independent sea ice length. Furthermore, it will be shown in a later section that the Integrated Ice Edge Error (Goessling et al., 2016) (Not yet derived) normalized by the ice edge length is correlated with the resolution of the ice edge length, proving the validity of normalizing using a common, coarse resolution ice edge length.

2.2.3 AMSR2

The Advanced Microwave Scanning Radiometer 2 (AMSR2) data utilized for this thesis is the sea ice concentration product from the University of Bremen (https://seaice.uni-bremen.de/sea-ice-concentration/amsre-amsr2/)(Last Accessed 18 Jan 2023) (Spreen et al., 2008). AMSR2 is a passive microwave sensor observing the microwaves emitted by the Earth, similar to **OSI SAF SSMIS**. AMSR2 is located on the JAXA GCOM-W1 satellite Melsheimer (2019), and the sea-ice concentration is retrieved using the ASI algorithm Spreen et al. (2008). The algorithm uses data from the 89GHz channel, which is the band with the highest spectral resolution, in both polarizations to determine the sea ice concentration. Bands at lower spectral resolutions are only used as weather filters, which can mask out false sea ice detected in the open ocean Spreen et al. (2008). The resulting data is a pan-arctic sea ice coverage with a spatial resolution of 6.25km.

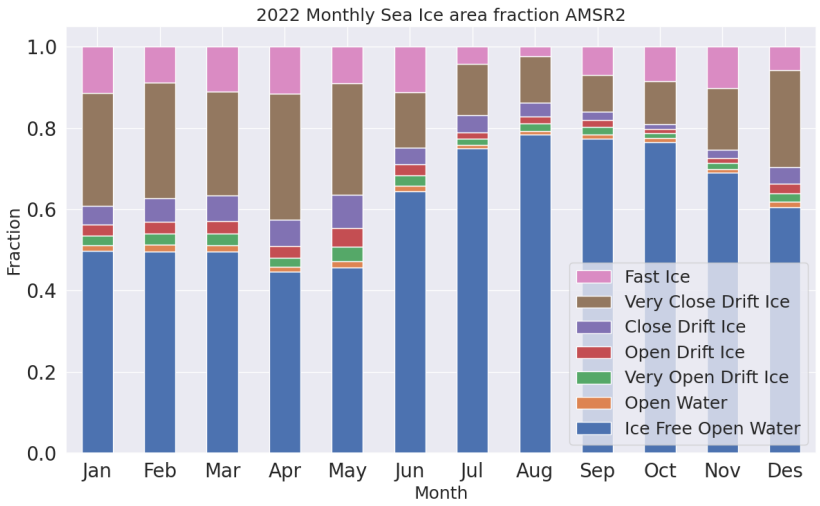


Figure 6: Monthly distribution of each sea ice concentration class fraction for the AMSR2 dataset covering 2022.

The current AMSR2 product was chosen as the ASI retrieval algorithm (Spreen et al., 2008) results in a higher spatial resolution product compared to similar AMSR2 products such as the AMSR2 product from OSI SAF (Lavelle et al., 2016), which is delivered on a 10km spatial resolution.

Figure (6) shows the monthly distribution of sea ice contours for the AMSR2 dataset. Similarly to Figure (3), a majority of the scenes are covered by Ice Free Open Water. However, Figure (6) shows the AMSR2 dataset has less very close drift ice, which may stem from an increased fast ice contour. Furthermore, AMSR2 has a less resolved open water contour compared to the sea-ice charts. This may be a result of AMSR2 being a algorithmically derived product, whereas the sea-ice charts are drawn for operational use such that regions of potential sea ice encounters are exaggerated to ensure maritime safety. On the other hand, Spreen et al. (2008) demonstrated that the ASI algorithm provide the most certainty at concentrations above 65%, with lower concentrations having higher deviations, mainly due to the error contributed by the atmosphere.

As the sea ice charts are treated as the ground truth during training of the deep learning model, it can be assumed that the model is best at predicting sea ice concentration distributions similar to those found in the training data. As such, the AMSR2 data will serve as an independent dataset, and will be used for validation only. Thus, the performance of the deep learning system can be inspected with regards to another dataset that is less similar than the sea ice charts, which measures the generalizability of the model.

DETTE MÅ NEVNES I DISKUSJON NÅR MAN SAMMENLIKNER MOT AMSR2

2.3 Forecasting systems

2.3.1 AROME Arctic

AROME Arctic is a non-hydrostatic, convection resolving high-resolution weather forecasting system which covers the European Arctic (Müller et al., 2017). The model covers the European Arctic similarly to Figure (1) which is the same domain though reduced, with a spatial resolution of 2.5km and 65 vertical levels. AROME Arctic uses different data assimilation techniques for the atmosphere and surface variables from the model background, with 3DVAR combining atmospheric background, HRES and observations and optimal interpolation on the surface background and observations to initialize the forecast analysis Müller et al. (2017). As previously mentioned, variables influencing the the sea ice concentration can aid in improving the predictive capabilities of a deep learning system. While observational products described above such as the ice charts (Dinessen et al., 2020) and OSI SAF SSMIS (Tonboe et al., 2017) describe the condition and dynamics of the sea ice concentration. Integrating weather forecast data as part of the model input can be used to describe the interaction between sea ice and atmospheric variables, thus providing relevant variables for predicting sea ice concentration. For the scope of this thesis, 2 meter temperature as well as 10 meter wind in the X and Y component have been selected.

Near surface winds influence the sea ice drift, with the sea ice in the European Arctic displaying a moderate to strong correlation between the sea ice drift speed and the wind speed during winter (Spreen et al., 2011). Moreover, sea ice drift speed is shown to be inverse proportional to the sea ice concentration (Yu et al., 2020). i.e. low concentration sea ice classes tend to have a higher drift speed than high concentration sea ice classes, though both classes display an increased drift speed given an increased near surface wind speed. Thus, including the X and Y component of the near surface wind from AROME Arctic provides the deep learning system with a high resolution proxy for the predicted sea ice drift.

Similarly, surface temperature influences the sea ice mass balance by melting or facilitating sea ice growth (Hibler, 1979), for example through the formation of melt ponds on top the sea ice. The 2 meter temperature from AROME Arctic is intended to serve as a proxy for the sea ice growth, by including a spatial distribution of temperature to the model. This may be correlated to areas in the model domain experiencing mean positive (melt) or negative (growth) temperatures during the forecast period.

AROME Arctic is shown to have lower RMSE for both 2-meter temperature and 10 meter zonal wind speed than both the deterministic (HRES) and ensemble (ENS) forecast as well as ERA-Interim from ECMWF, for all months when compared to measurements from 89 stations located in Finnmark, Svalbard as well as Jan Mayen and Bjørnøya (Müller

et al., 2017). Hence, it is reasonable to assume that extracting the wind and temperature fields from AROME Arctic will provide the most precise information with regards to the strength and spatial location, compared to global medium range numerical weather prediction systems such as the ECMWF Integrated Forecasting System (IFS) Cycle 47r3 (Haiden et al., 2022). However, it is noted that operational numerical weather prediction systems such as those described by Müller et al. (2017) and Haiden et al. (2022) are in constant development, with new improvements added without any retroactive effect for previous data. Firstly, the comparison made in Müller et al. (2017) was with HRES and ENS as of Cycle 38r2 Bauer et al. (2013) is not necessarily representative of the current state of both products. Secondly, significant advances in model development may cause data before and after the implementation date to be inconsistent, e.g. by introducing a permanent shift in bias for a variable. Problems regarding model updates could be avoided by using variables from a re-forecast or reanalysis product such as CARRA (Køltzow et al., 2022). However, CARRA similarly to other reanalysis products are not delivered with a daily frequency (see https://climate.copernicus.eu/copernicus-arctic-regional-reanalysis-service, Last Accessed 21 Jan 2023), which would inhibit the operational aspect of the developed deep learning system. It is also noted that CARRA specifically only pose a 30 hour lead time, which limits the desired "up to 3 day" lead time desired for the developed deep learning system.

With regards to model development, a major development in AROME Arctic in terms of temperature representation over sea ice ocurred 10 Oct 2018 (AROME Arctic Changelog, Last Access 21 Jan 2023), in the form of a *snow on ice* variable. As this change is expected to have changed the distribution of 2 meter temperature significantly, especially over sea ice covered grid cells (Batrak et al., 2018; Batrak and Müller, 2019), it has been opted to only consider near surface temperature data from AROME Arctic from 2019 and onwards. This decision is made to avoid having a shift in temperature distribution present in the data, which would exert a negative impact in training the deep learning model.

Though the different datasets in Table (1) have been chosen with the intention to serve as independent products without any intra coupling, it is noted that the sea ice observations used to compute the sea ice concentration trend (Tonboe et al., 2017) is also used to force AROME Arctic with sea ice concentration at the initial timestep (Müller et al., 2017). it is suboptimal to provide input parameters derived from other input parameters, as their correlation may render one of the input parameters obsolete in terms of additional information the deep learning system will infer from the "redundant" predictor. Nonetheless, it is assumed that the impact of the sea ice concentration forcing is low when combined with other surface forcings during the assimilation process. Furthermore, as the sea ice concentration is kept constant at all timesteps (Müller et al., 2017), the correlation between sea ice concentration and atmospheric variables can be assumed to be decaying with time. Thus, both products will be used as input variables, and their overlap is assumed

to tend towards zero.

2.3.2 NeXtSIM

The neXt generation Sea Ice Model (neXtSIM) is developed by the Nansen Environmental and Remote Sensing Center and performs the physical simulations for the neXtSIM-F deterministic forecasting platform (Williams et al., 2021). NeXtSIM-F assimilates sea ice concentration from operational OSI SAF sea ice concentration products (Tonboe et al., 2017; Lavelle et al., 2016) and forces the model with oceanic and atmospheric forecasts. Furthermore, the neXtSIM-F platform is not a coupled system, i.e. the neXtSIM sea ice model is not coupled to either and atmospheric or oceanic model. The version of neXtSIM-F data used for this thesis is supplied on a 3km polar stereographic grid on a pan-arctic domain.

NeXtSIM differentiates itself from comparative physical sea ice models as it does not apply a rheology based on the Viscous-Plastic scheme. 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 (Hibler, 1979). Instead, NeXtSIM applies a brittle sea ice rheology, specifically the brittle Bingham-Maxwell (BBM) rheology which treats the sea ice as a brittle material rather than a viscous fluid (Ólason et al., 2022). Due to the implementation of the BMM rheology, neXtSIM-F is the first sea ice forecasting system not to use a viscous-plastic scheme (Williams et al., 2021).

With a forecast range of 7 days, data from neXtSIM-F will be used to validate the deep learning system against current high resolution operational sea ice forecasts by serving as a comparable product.

2.3.3 Barents-2.5

Barents-2.5, (hereby Barents) is an operational coupled ocean and sea ice forecasting model under development at MET Norway (Röhrs et al., 2022). The model has been in operation since September 2021. Barents poses the same resolution and projection as AA, i.e. Lambert Conformal Conic with a 2.5km resolution (Röhrs et al., 2022; Müller et al., 2017). Furthermore, Barents also forecasts with a lead time up to 66 hours, which is the same as AROME Arctic. Since Barents covers the same spatial domain as the deep learning system and forecast with a lead time close to three days, its predicted sea ice concentration will be used for validation purposes.

The sea ice model used in Barents is the Los Alamos sea ice model (CICE) version 5.1, which uses an Elastic Viscous Plastic sea ice Rheology (Hunke et al., 2015). 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 (Hunke and Dukowicz, 1997).

Barents includes an Ensemble Prediction System with 6 members executed for each of the four model runs situated at (00, 06, 12 and 18) (Röhrs et al., 2022). As part its forcing routine, Barents 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 are forced using atmospheric data from ECMWF. As such, the members 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 (Johannes Röhrs, 2022, pers. commun.). However, there is generally little spread within one ensemble with regards to sea ice (Röhrs et al., 2022).

The data assimilation scheme applied for Barents is a Deterministic Ensemble Kalman filter, which solves for the analysis with a background error covariance matrix estimated as the variance of the ensemble of background members (Röhrs et al., 2022). Furthermore, it has been expressed by the developers of Barents that the model performance was unsatisfactory up until May / June 2022 due to spin up time of the data assimilation system (Johannes Röhrs, 2022, pers. commun.). As such, forecasts initiated prior to May 2022 will not be assessed for validational purposes due to the expected shift in performance as expressed by the model developers.

Similarly to the neXtSIM-F data in Section (2.3.2), Barents will also be used to validate the deep learning system. However, the forecast range of Barents is only 66 hours, which cuts it short of producing three full daily means. Furthermore, due to the ensemble setup of Barents, it is possible to present a forecast both through the ensemble mean as well as a pseudo deterministic run (single member). However, a forecast from a single Barents member would still be influenced by the other ensemble members during the assimilation stage.

ECMWF IFS is used to force both neXtSIM and Barents with atmospheric variables, whereas TOPAZ (Sakov et al., 2012) is used to force neXtSIM (Williams et al., 2021) while only nudging the boundaries of Barents (Röhrs et al., 2022). However, their differences in terms of ensemble setup, model coupling, sea ice rheology as well as domain coverage has led to both products being included for validational of the deep learning system. Moreover, both physical products are of a spatial and temporal scale for operational relevancy (Wagner et al., 2020), similar to the deep learning system.

Dette avsnittet ble kjelkete, men ønsker å si noe om at det er mer relevant å sammenlikne

3 Methodological framework

This Section will outline the theoretical background of convolutions from a deep learning point of view, as well as provide a brief overview of image segmentation as a computer vision task. Furthermore, the Methodological framework required to develop the U-Net architecture will be highlighted, followed by the description of a training loop and the central algorithms utilized. Finally, validation metrics used to asses the performance of the developed deep learning system will be derived.

3.1 Convolutional layers

Convolutional layers was initially proposed by (LeCun et al., 1989) to classify handwritten numbers and incorporated into a deep neural network through the backpropagation algorithm (Rumelhart et al., 1986). The layer LeCun et al. (1989) presented consists of an arbitrary amount of filters, which are small two dimensional matrices (e.g. (3×3) pixels) designed to capture a certain structure in the image such as lines or edges . Each filter contains trainable weights, which are learned from the data during backpropagation (LeCun et al., 1989). When a filter is convolved with all possible local neighborhoods from the input, it outputs a feature map which represent where the input image triggered a response from the filter (Zeiler et al., 2010). Moreover, inputting feature maps to a convolutional layer allows for the filters to respond to combinations of lower level structures, which trains the layer to detect more complicated patterns (Fukushima, 1980). Additionally, stacking convolutional layers in a network-architecture structure increases the field of view for each subsequent layer, which makes each layer observe an increasingly complex pattern of higher order feature maps at increasingly larger spatial scales (Fukushima, 1980). As a result, convolutional layers are able to discern between object and background by only perceiving a local neighborhood at a time, as well as being invariant to translation of the object.

Sitere boka til Goodfellow2016?

The number of trainable parameters for a convolutional layer is equal to the size of a filter times the number of filters. As a result, the number of trainable parameters is invariant to the spatial extent of the input images, and all units shares the same weights which causes the layer to detect the same feature at all locations (LeCun et al., 1989). On the other hand, fitting a fully connected layer to spatial gridded data consists of associating a separate hidden unit to each pixel. As such, the size of a fully connected layer scales with the size of the image, which increases the risk of overfitting the network. In the case of the convolutional layer, LeCun et al. (1989) notes that reducing the number of trainable parameters through weight sharing constrains the solution space such that overfitting is avoided while still fitting the layer to the data. Furthermore, the fully connected layer is not invariant to translation as each trainable parameter is exclusive to their respective

pixel. As such, the layer is unable to detect a similar object at a different position, reducing their usefulness for image-based prediction tasks.

Finally, Ciresan et al. (2012b) showed that the processing time of a convolutional layer is significantly shortened by utilizing a graphics processing unit (GPU), due to their large amount of compute cores compared to traditional Central Processing Units (CPUs). Furthermore, the authors of Krizhevsky et al. (2012) provided the first publicly available implementation of a CNN running on a GPU by utilizing the Nvidia Compute Unified Device Architecture api. Krizhevsky et al. (2012) also demonstrated that their results are tied to the performance of the GPU in terms of available memory as well as floating point operations, with the implications that a better GPU as well as larger datasets would improve their results. As such, the modern implementation of convolutional layers, and by extension deep learning architectures which heavily utilize the convolutional layer such as the CNN or U-Net, can be initiated with more trainable parameters to process greater datasets consisting of larger samples due to their implementation to run on the GPU.

The convolutional layer can be described mathematically by utilizing the previously described principle of allowing the filter to only perceive a local neighborhood of the input. Consider the value of a single point $y_{i,j} \subset Y \in \mathbb{R}^2$ where i, j denote the position in the x and y direction as a single output from a convolution. Let $X \in \mathbb{R}^3$ be an input image of size $(A \times B \times D)$ consisting of a single channel, and $W \in \mathbb{R}^3$ be a symmetric filter of size $(r \times r \times D)$. Then, the value at a single point $y_{i,j}$ is given as follows,

$$y_{i,j} = \sum_{a=1-\frac{r}{2}}^{\frac{r}{2}} \sum_{b=1-\frac{r}{2}}^{\frac{r}{2}} \sum_{d=1}^{D} W_{a+\frac{r}{2},b+\frac{r}{2},d} X_{i+a,j+b,d} \tag{1}$$

Where the subscript notation is used in W and X to denote indexes similarly to Y. Repeating Equation (1) across all points $x \subset X$ by applying a sliding window technique returns the convolution of X with filter W, which results in an output Y with size $(A-r+1) \times (B-r+1)$. Note that the above definition only applies for $X_{1 \leq i+a \leq A, 1 \leq j+b \leq B}$. The size of the output can be adjusted by padding the input X by a size P in each direction or increasing the stride S of the sliding window, which reformulates the output size of Y in a single dimension as a function

$$Y_{\text{dim}} = \lfloor \frac{A - r + 2P}{S} + 1 \rfloor \tag{2}$$

The convolutional layer adds the convolution described in Equation (1) with a bias term $B \in \mathbb{R}^2$ of same spatial shape as Y, as well as applying an activation function g to each

$y_{i,j}$ which introduces nonlinearity. In summary, the output of a convolutional layer can be described as

$$Y' = g(Y + B) = g(W^T X + B) \tag{3}$$

If the number of filters increases from 1 to N, Equation (3) is repeated N times, resulting in an output $Y \in \mathbb{R}^3$ of size $(Y_{\text{dim1}}, Y_{\text{dim2}}, N)$.

3.2 Image segmentation

Image segmentation is a computer vision task where pixels are assigned labels according to some predetermined rules. It is common to define an image segmentation task either as a study of countable *things* (Instance segmentation), or recognizing similarly textured *stuff* (Semantic segmentation) (Kirillov et al., 2018). The task for this thesis, which is labelling sea ice concentration according to its predicted concentration class, falls into the latter category following the definition of *stuff* in Adelson (2001). I.e. the current task is to assign each pixel in a predicted scene a class label.

Network architectures based on the CNN can be used to perform semantic segmentation given the right formulation. Ciresan et al. (2012a) presented an approach where a general CNN not tailored to the task itself (see the architecture of Ciresan et al. (2012b)) was used to label each pixel by using their surrounding neighborhood as input. However, due to only processing parts of the image at once, the segmentation algorithm in Ciresan et al. (2012a) is computationally expensive as well as limiting the context in which each pixel is classified in to a surrounding local context.

To capture the global context of a scene, network architectures such as Long et al. (2015); Noh et al. (2015); Ronneberger et al. (2015); Badrinarayanan et al. (2017) implement the Encoder-Decoder architecture (Long et al. (2015) skip the decoder), where the entire input scene is first processed by a CNN which is then followed by a Deconvolution network which upsamples the encoded features back to the resolution of the original image. A high level description of the Encoder-Decoder architecture is that the architecture first encodes what is in the image, before the decoder estimates where what was encoded is located.

This thesis will utilize the U-Net architecture proposed by Ronneberger et al. (2015). The U-Net was originally developed for medical image segmentation, however the architecture has shown promising results for both pan-arctic seasonal (Andersson et al., 2021) and regional short term (Grigoryev et al., 2022) sea ice concentration forecasting amongst other applications. Another aspect which makes the U-Net more suitable to the current task, compared to the previously described image-to-image architectures is that the network

converges quickly, which is ideal when working with a small dataset (Ronneberger et al., 2015).

3.3 Describing the U-Net architecture

Figure (7) shows the U-Net architecture. This section aims at describing the different components constituting the architecture from a technical point of view.

3.3.1 The convolutional block

A single convolutional block consists of two repeat convolutional layers, each followed by the Rectified Linear Unit (ReLU) nonlinear activation function, which is defined as $f(x) = \max(0, x)$ (Nair and Hinton, 2010). Each convolution is performed using a 3×3 window. The original formulation of the U-Net also does not apply padding to the input, resulting the convolutional filter only being applied to the entries of the input where the filter is never out of bounds. With a stride of $S = 1$, this results in each convolutional layer reducing the spatial extent by two pixels in each direction following Equation (2). It is also noted that the number of feature maps is doubled after each downsampling step, which is performed by the pooling layers.

3.3.2 Pooling layers

Pooling operations are used to reduce the spatial extent of the current feature maps, by downsampling the data in the spatial dimensions. As seen in Figure (7), the U-Net downsamples the data in the contracting path through 2×2 maximum pool layers with a stride of 2. This specific configuration causes the spatial resolution to be halved. In the max-pool layer, a filter runs through each input channel and and chooses the maximum value inside the neighborhood of the filter. As such, the extreme values in each feature map is retained at the expense of rejecting the rest of the data. See Figure (8) for a graphical description.

3.3.3 Transposed convolutions

Transposed convolution was proposed by Zeiler et al. (2010) (note the incorrect use of deconvolution, this is not the mathematical inverse of a convolution) to increase the resolution of a feature map. The method was utilized by Long et al. (2015) to connect the coarse output of an encoder with the image resolution of the target (it is referred to as both *backwards convolution* and *deconvolution* in the proceedings paper). Similar to the

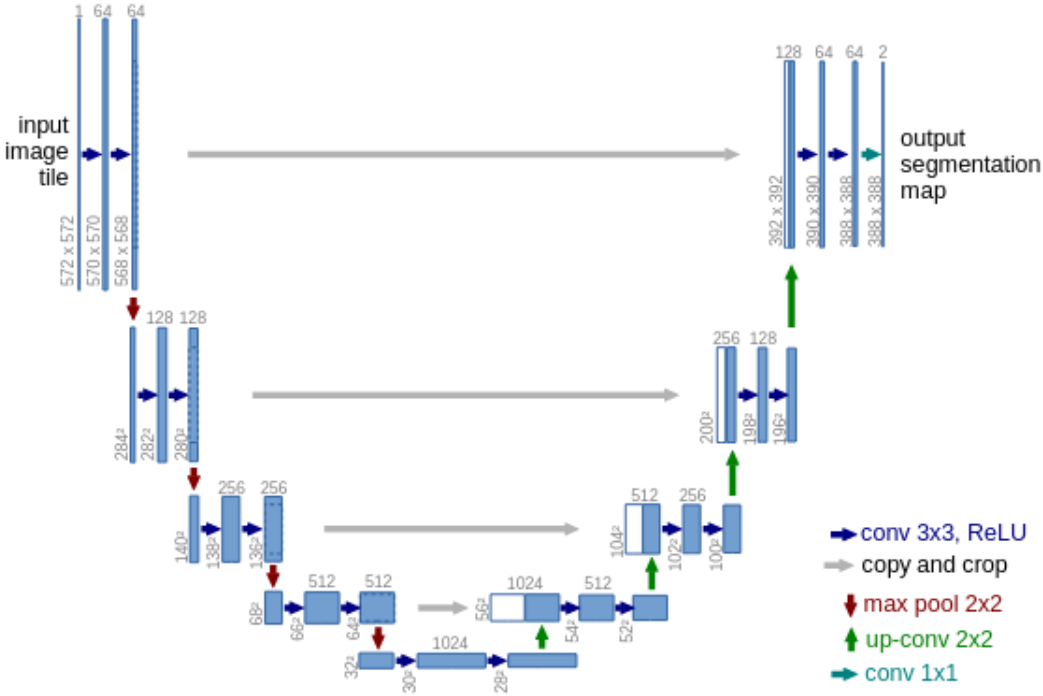


Figure 7: The U-Net architecture. The blue boxes represent feature maps, with the lower left numbers determining the spatial resolution and the top number the amount of feature maps. White boxes in the expansive path (right side / decoder) are the copied feature maps from the contractive path (left side / encoder). Arrows denote the different operations. Note that the original U-Net only performs *valid* convolutions, i.e. convolution without padding to match the input. This causes a convolutional layer to slightly decrease the spatial extent. As a result, the copied features from the contracting path are also cropped to match the dimensionality in the expansive path. Figure extracted from Ronneberger et al. (2015).

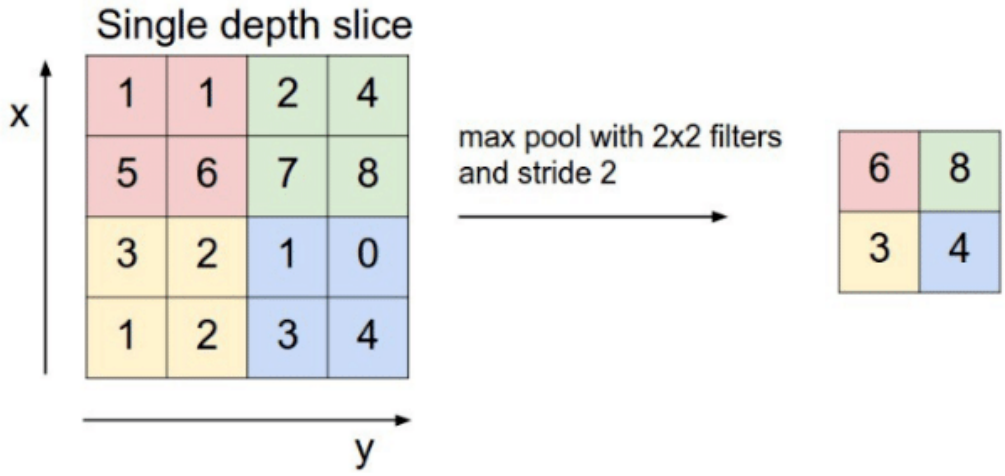


Figure 8: The max-pool operation for a 2×2 filter with a stride of 2. Figure taken from (RADU et al., 2020)

convolutional layer, the transposed convolutional layer involves striding a convolutional filter with trainable parameters across a feature map. However, the transposed convolutional layer projects a singular entry from the input through the convolutional kernel to produce an output that is larger than the input.

In the Encoder architecture, lower level feature maps provide spatial information regarding where stuff is located in a scene, whereas higher level feature maps contain information regarding what is in the scene at the expense of losing spatial information (Long et al., 2015). To circumvent this, Ronneberger et al. (2015) concatenate the features from the contracting path with the output from the transposed convolution at the same level of depth, i.e. where the number of feature maps are equal at the end of the convolutional block. The concatenation operation is possible in Ronneberger et al. (2015) since they crop the feature maps in the encoder in their spatial dimensions to match the spatial dimensionality of the feature maps in the decode. The operation can be seen in Figure(7) denoted by the gray arrow. The resulting convolutional layer is then trained to make a more precise prediction due to the concatenated input (Ronneberger et al., 2015).

3.3.4 Outputs

The output layer of the U-Net is denoted by the turquoise arrow at the right side of Figure (7). The arrow denote that the input processed by a convolutional layer with a number of filters equal to the number of output classes. Each filter is of size (1×1) and maps each layer in the input feature map to their respective class probability map of equal spatial

shape (Ronneberger et al., 2015).

3.4 Training procedure for the U-Net

This subsection aims to demonstrate how Ronneberger et al. (2015) trained the U-Net, and will consequently highlight some different hyperparameters and exemplify some functions which can be modified during training. Hyperparameters refer to model parameters which are not updated during training (Yu and Zhu, 2020), and may directly influence the model architecture or the training procedure.

Training the U-Net start by assigning random values to the weights of the network. Since the U-Net utilizes the ReLU activation function after each convolutional layer in the convolutional blocks (Ronneberger et al., 2015), it is standard for each layer to draw the weights from a normal distribution with $\mu = 0$ and standard deviation $\sigma = \sqrt{\frac{2}{n_l}}$, where n_l is the number of inputs to the layer He et al. (2015). This weight initialization scheme ensures that variance of the feature maps are approximately equal, i.e. avoids varying the activation of input signals between layers He et al. (2015); Ronneberger et al. (2015).

The process of training a neural network involves making predictions on all training data. For each sample, what is predicted is compared against a ground truth label associated with the sample. To quantify the prediction error, a loss function is defined. The overall goal of training a neural network is to minimize the loss function. For classification problems, the cross-entropy is a common loss function, and is defined as $L(p, y) = -\log(p)$ where p is the predicted probability of the ground truth class and y is the true probability.

The error computed by the loss function is then sent backwards throughout the network according to the backpropagation algorithm (Rumelhart et al., 1986), which effectively computes the gradient of the loss function with regards to the trainable parameters

$$\frac{\partial L}{\partial w_l} = \frac{\partial L}{\partial p} \frac{\partial p}{\partial w_l} \tag{4}$$

where w_l is the trainable parameters associated with the l-th layer. The gradient of the loss for a weight at a given layer shown in Equation (4) is used by an optimizer to adjust the weights such that the loss is minimized with respect to the weights (gradient descent).

A single iteration trough all available training data is known as an epoch. During training, multiple data samples are fed to the model at the same time in batches, the size of which is predetermined by the batch_size hyperparameter.

Finally, each trainable parameter are updated by an optimizer which is a function of the gradient of the loss at each trainable parameter. The purpose of the optimizer is to find the global minima of the loss function, thus the gradient of the error at each trainable parameter tells the optimizer in which direction to nudge the weight in order to achieve this minima.

3.5 Forecast verification metrics

Verification schemes provide insight into how a forecasting system performs. For this thesis, verification metrics serve a dual purpose. From a model development point of view, verification metrics will be used to increase the skill of the model. However, the same metrics will also be utilized to assess the quality of a prediction as well as explain the physical interpretation of the model (Casati et al., 2008). The model developed for this thesis predicts a scene consisting of labelled pixels, as described in Section (3.2). It was mentioned in Section (1) that the developed model is aimed towards operational end users, which is partly achieved by validating the model against metrics of end user relevance. Furthermore, it can be assumed that the model and target observation wont differ much outside of the marginal ice zone (Fritzner et al., 2020). Thus, this section will introduce metrics which are aware of the sea ice edge, as the sea ice edge is a relevant quantity for maritime end users operating in the Arctic (Melsom et al., 2019) as well as providing skill scores stemming from where the change in sea ice concentration is occurring. The following subsections will describe how to determine the position of the sea ice edge, as well as its length according to Melsom et al. (2019), and derive the Integrated Ice Edge Error (Goessling et al., 2016), with regards to a spatially gridded dataset of deterministic sea ice concentration values.

The Integrated Ice Edge Error is chosen among similar sea ice edge metrics (Melsom et al., 2019; Dukhovskoy et al., 2015) as it has been shown to be less sensitive to isolated ice patches (Palerme et al., 2019). Furthermore, the work of Melsom et al. (2019) recommend the Integrated Ice Edge Error amongst other metrics for its intuitive interpretation as well as for the possibility to provide the spatial distribution of IIEE areas.

3.5.1 Defining the Ice Edge

The sea ice edge for a given spatial distribution of sea ice concentration values is derived on a per pixel basis, and defined as the grid cells which meets the following condition,

$$c_{i,j} \geq c_e \wedge \min(c_{i-1,j}, c_{i+1,j}, c_{i,j-1}, c_{i,j+1}) < c_e \tag{5}$$

In Condition (5), $c \subset C$ are gridded sea ice concentration values, with i, j denoting indexes. c_e is a given concentration threshold. The entries $c_{i,j}$ which adhere to the condition in Condition (5) form the set E where $\exists e_{i,j}, \forall c_{i,j}$ where Condition (5) holds (Melsom et al., 2019).

Moreover, all the entries in E each contribute to the total length of the sea ice edge, with each entries' length contribution determined based on that entries' 4-connected neighborhood. Using this formulation, the different combination of entry neighborhood in E can result in three different length contributions. For the following contributions, s is the spatial resolution of the grid.

- A neighborless pixel is assumed to yield a contribution the length of the diagonal of a grid cell ($l = \sqrt{2}s$). Here it is assumed that the grid cell only have diagonal neighbors.
- A pixel with two or more 4-neighbors contributes with its spatial resolution (length of the grid cell) $l = s$.
- A pixel with one 4-neighbor contributes the mean value between the length of the grid cell and length og the diagonal of the grid cell $l = \frac{s+\sqrt{2}s}{2}$. It is assumed that the grid cell also have a diagonal neighbor.

The final length of the sea ice edge length then becomes

$$L = \sum_{\text{e in E}} l^e \tag{6}$$

where the superscript l^e denotes the length associated with the entry e according to the algorithm listed above. I.e. the sum of all contributions.

3.5.2 Integrated Ice Edge Error

The IIEE is an error metric which compares a forecast f to a predefined ground truth target t Goessling et al. (2016). The metric is defined as

$$\text{IIEE} = \text{O} + \text{U} \tag{7}$$

where

$$\text{O} = \int_A \max(C_f - C_t, 0) dA \tag{8}$$

and

$$\mathrm{U} = \int_A \max(C_t - C_f, 0) dA \tag{9}$$

with $A \in \mathbb{R}^2$ being the area of interest, and is of similar size as C. Subscript f, t denotes whether C contains forecasted or target sea ice concentration values. In Equations (8 and 9), C is binary and is 1 if its concentration value is above some predefined threshold, and 0 elsewhere (Goessling et al., 2016). 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.

Additionally, the IIEE can also be represented as a spatial metric by removing the integral with respect to A in Equation (8 and 9). In this way, the metric is 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 spatial distribution of False Positive and False Negatives of the forecast respectively.

The length of the ice edge has a strong influence on the IIEE (Goessling and Jung, 2018; Palerme et al., 2019). Hence, to ensure that forecast errors are comparable across seasons, IIEE is normalized with the length of the ice edge, as mentioned in Section (2.2.2)

4 Model development

This section will cover the implementation of the U-Net architecture, as well as related processes such as data preparation and a custom dataloader. Furthermore, this section will present intermediate results obtained during development to highlight technical decisions made as well as their consequence for model performance. Decisions made will be highlighted from both a MachineLearning paradigm point of view, although when relevant they will also be explored in a context of the underlying physics.

4.1 Data preparations

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 subsection will describe the process from raw data to ready sample.

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 (1).

4.2 Data sources

Data sources used are Sea Ice charts from Nick initiated at 15:00 as well as Arome Arctic initiated at 18:00 Dinessen et al. (2020); Müller et al. (2017). 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.

4.2.1 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 JCOMM Expert Team on Sea Ice (2014).

label sections

Say thanks in acknowledgements

4.3 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.

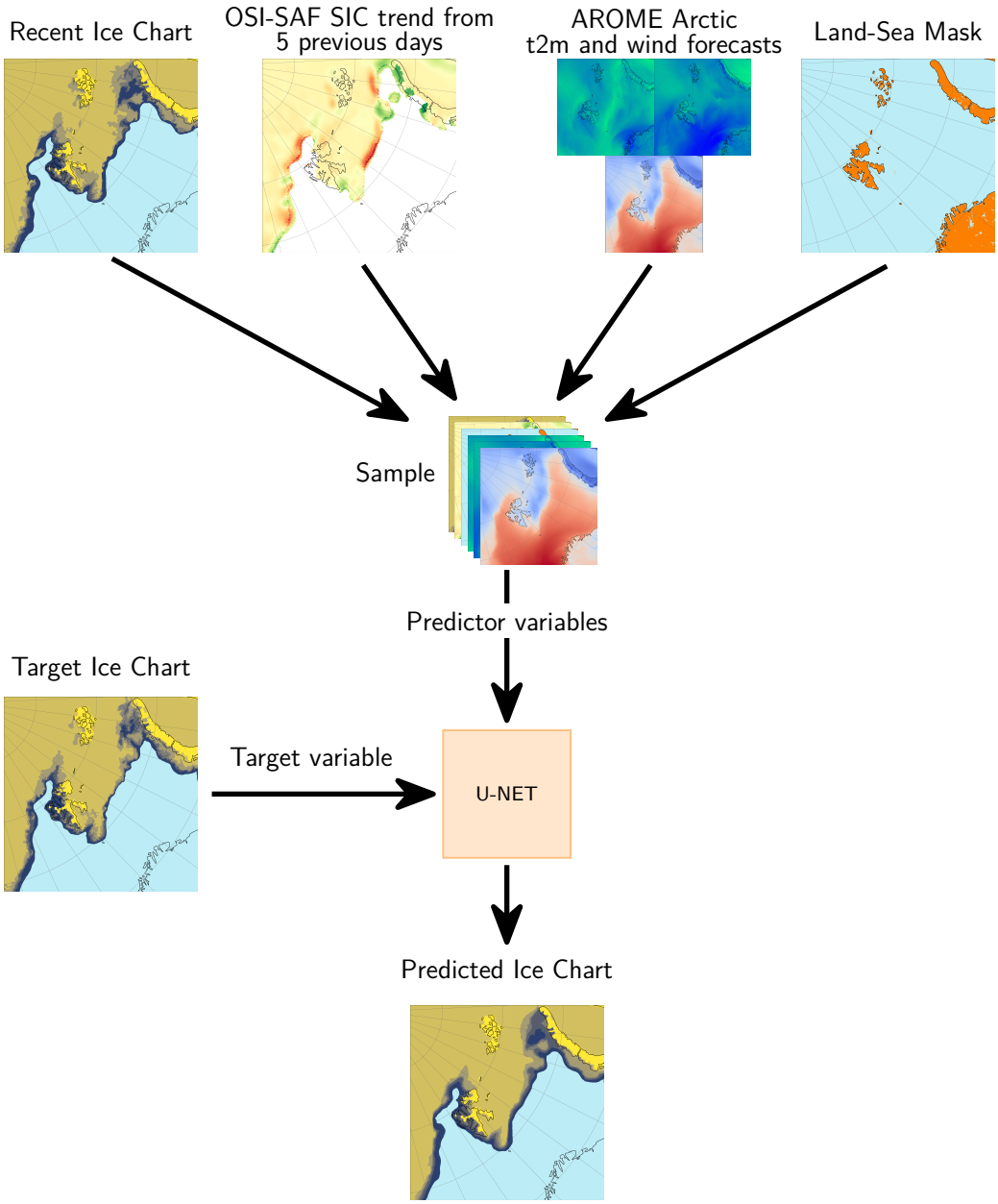


Figure 9: Workflow figure providing a overview of the data pipeline. Data is sampled from four sources (Recent Sea Ice Chart, OSI SAF, AROME Arctic and a Land Sea Mask), preprocessed and merged into a single sample. The sample is fed into the network together with an associated ground truth target sea ice chart. The predicted sea ice chart is compared against the ground truth sea ice chart, and their binary cross entropy error is propagated backwards throughout the network, which constitutes a step in the training loop.

4.4 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. .

OSI-SAF SSMIS is a continously developed operational product, where changes are not required to act retroactively on the data. As such, the Sea Ice Concentration used for few samples with t2m runs and many data samples no t2m differ due to the introduction of a filtered ice concentration variable 10/05-2017 Tonboe et al. (2017). Thus, the filtered ice concentration will be used when the training data spans 2019-2020, and the unfiltered ice concentration will be used when the training data spans 2011 - 2018 to assert that there is no sudden shift in the ice concentration trend which can negatively impact the training period.

4.5 Deviations from the 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 (10). 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.

Mention how when using Osi Saf trend as predictor, the trend up to but not including the forecast start date is used. This is to make the model ready for operational use, as the Osi Saf daily product is not ready on lustre when the Ice Charts are poublished

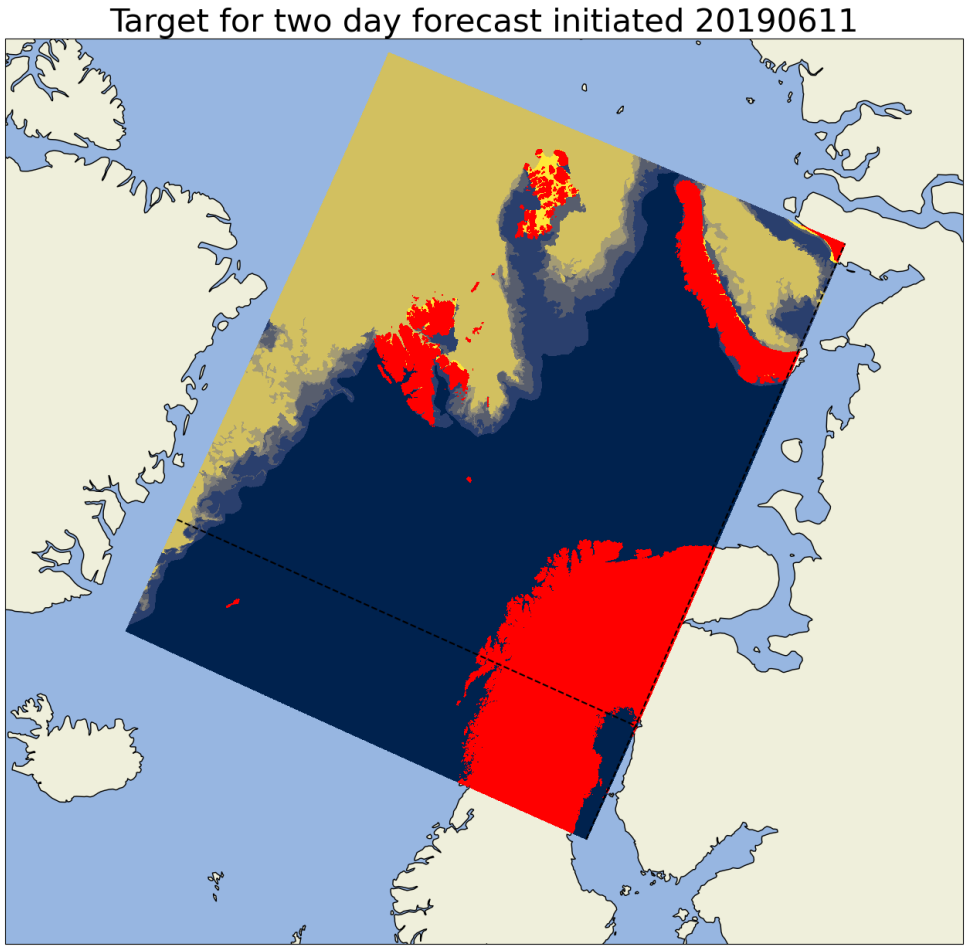


Figure 10: Example sample displaying an Ice Chart on a 1km Arome Arctic projection. Note the horizontal and vertical dashed black line which indicate the domain subsection used by the UNET

5 Model Architecture

The model architecture follows an encoder - decoder structure, commonly referred to as a U-NET Ronneberger et al. (2015) 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 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 He et al. (2015), as the activation function used throughout the network is the ReLU function Nair and Hinton (2010). The final output of the model is a (1920, 1840, 7) tensor containing softmaxed probabilities along its final axis.

5.1 CategoricalCrossEntropy-Loss

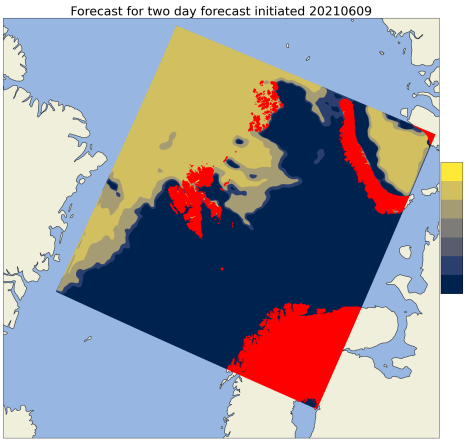
As the title suggests, these runs of the model involved using CategoricalCrossEntropy as the loss function for multi-class image segmentation. Categorical Cross Entropy loss is defined as

$$CE = -\sum_{i}^{C} y_i \log\left(\widehat{y}_i\right) \tag{10}$$

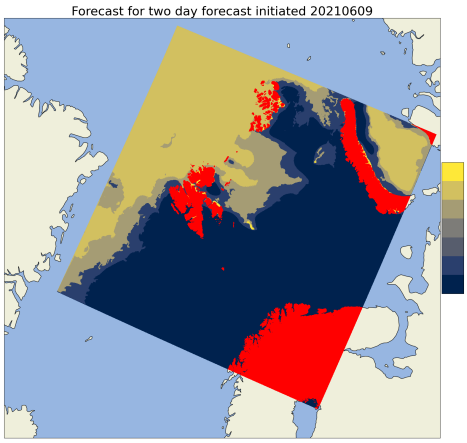
where C denotes the number of available classes, y the ground truth and $\widehat{y}$ a prediction of y. Note that as y is onehot-encoded, the formulated function only contributes to the overall loss with the log of the predicted probability of the correct class according to the ground truth.

Two variants of the previously described model have been trained with the Categorical-CrossEntropy described in equation (10). 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 (11).

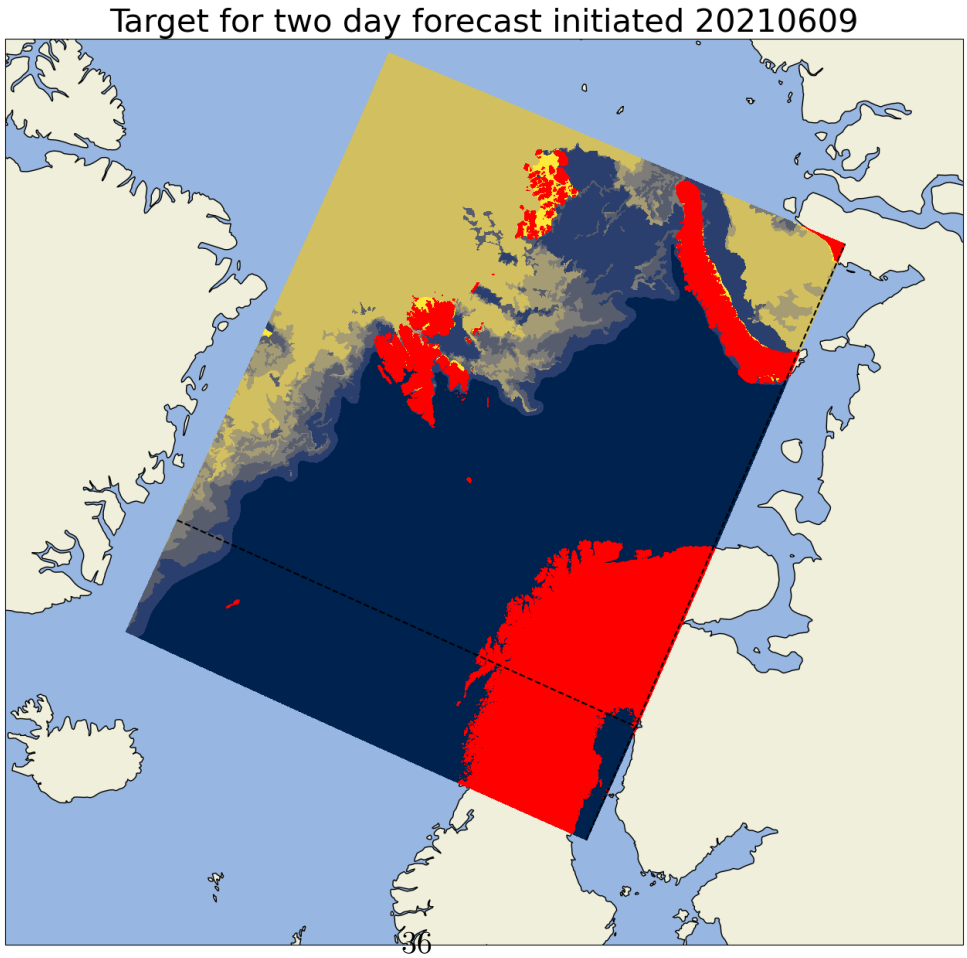
By inspecting Figure (11), two observations can be made. The first observation is regarding how the model complexity affects how it fit to the data. By comparing Figure (11a) with (11b), it can be seen that the latter is resolving the finer structures of the ice edge to larger extent than the prior. 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



(a) Forecast with two day lead time with model_512 architecture



(b) Forecast with two day lead time with model_1024 architecture



(c) Target for forecast with two day lead time

Figure 11: Example forecast attempt made by model_512 and model_1024 09-06-2021

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.

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).

This may have a source

They should be

5.2 FocalLoss

The focal loss is derived as a generalization of the Cross Entropy Loss listed in Equation (10). 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 Lin et al. (2017). Mathematically, focal loss is defined as

$$\mathrm{FL} = -\sum_{i}^{C} \alpha_i (1 - \hat{y}_i)^{\gamma} y_i \log(\hat{y}_i) \tag{11}$$

where α is a balancing parameter, γ is the focusing parameter ($\gamma = 0 \rightarrow CE$), with the rest similar as Equation (10).

By inspecting Equation (11), 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 (11). 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

5.3 Cumulative probability distribution model

5.3.1 Separate convolutionial layers as output

Discuss difference in dataloader, same dataset is used differently

5.4 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 Abadi et al. (2015). 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 Goessling et al. (2016) could be monitored instead.

Data exists, start writing

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 2 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.

Write about the climatological Ice Edge dataset, ref section from here

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 Palerme et al. (2019) . 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

This citation is actually for SPS_{length} but SPS is reduced

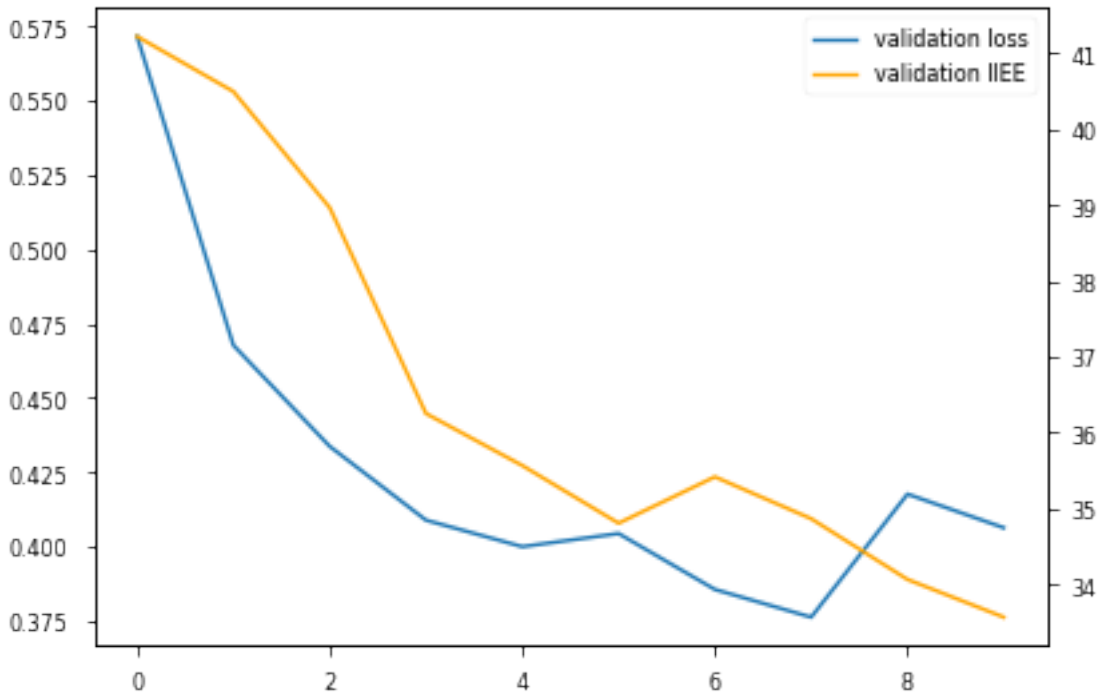


Figure 12: validation loss and Normalized IIEE computed as mean of validation set for each epoch during training

done in the above callback increases training-time of ten epochs from ≈ two hours without the IIEE callback to ≈ 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.

This may change

On the other hand, it can be seen by inspecting Figure (12) that the Normalized IIEE tend to evolve conjunctionally 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 (12) have a correlation of 0.82 with regards to epoch. Note that this has been calculated only using the numbers present in Figure (12). 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.

tmp figure, redo with inspect data notebook in ForecastVerification

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.

6 physical connections

6.1 Variograms

6.2 Case study

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

6.3 Synthetic AA forcing

7 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.

7.1 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.

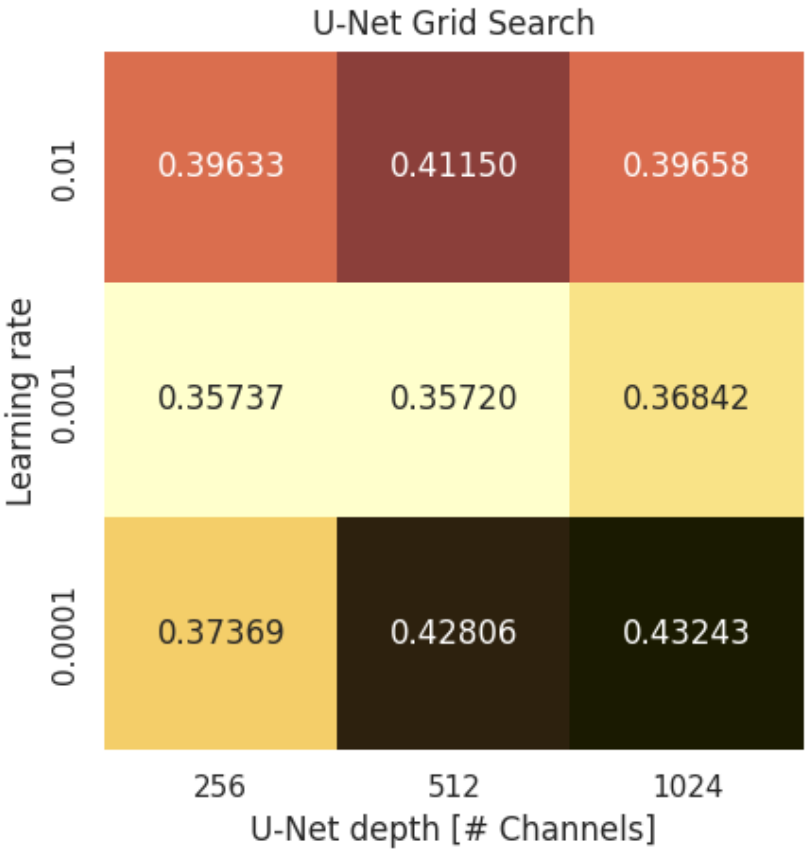


Figure 13: Grid search performed over variations of the learning rate as well as an increasing U-Net depth (represented by the number of feature maps at the at the final convolutional block). Each cell contain the minimum obtained validation loss of its respective combination which is associated with the best validation performance during training.

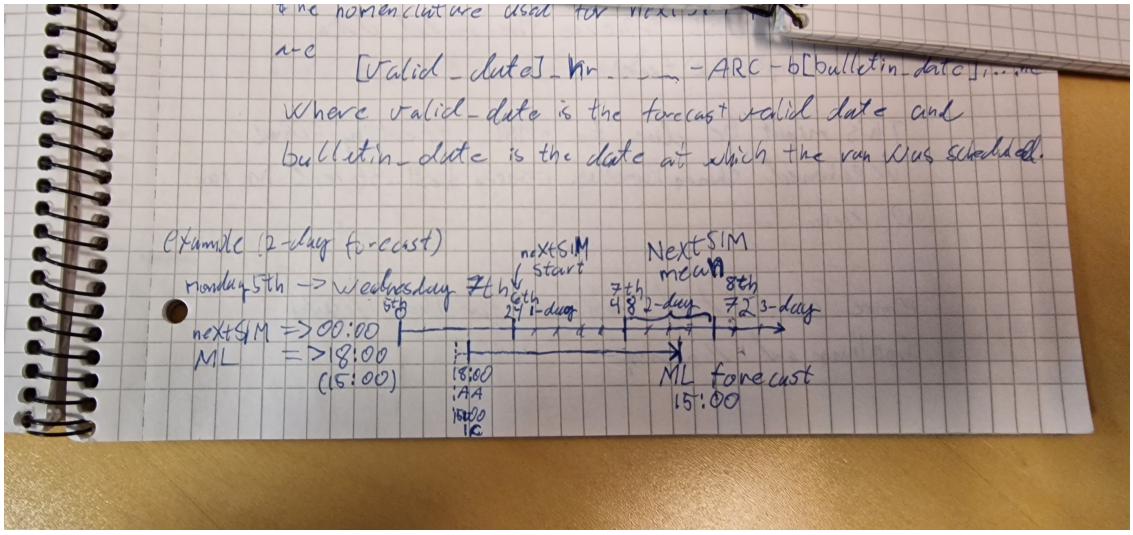


Figure 14: Sketch presenting how physical model forecasts are compared against machine learning forecasts. The axis represents time after 00:00 bulletin date of the machine learning forecast. The machine learning forecast is initiated 6 hours prior to the start of the physical model. The sketch exemplifies 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).

Figure (14) to be made professional, using e.g. TiX

8 Conclusion and future outlook

A consequence of the operational aspect is the possibility to force decoupled NWP-systems with updated Sea Ice Concentration.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X.: TensorFlow: Large-

Scale Machine Learning on Heterogeneous Systems, URL https://www.tensorflow.org/, software available from tensorflow.org, 2015.

Adelson, E. H.: On seeing stuff: the perception of materials by humans and machines, in: SPIE Proceedings, edited by Rogowitz, B. E. and Pappas, T. N., SPIE, https://doi.org/10.1117/12.429489, 2001.

Andersson, T. R., Hosking, J. S., Pérez-Ortiz, M., Paige, B., Elliott, A., Russell, C., Law, S., Jones, D. C., Wilkinson, J., Phillips, T., Byrne, J., Tietsche, S., Sarojini, B. B., Blanchard-Wrigglesworth, E., Aksenov, Y., Downie, R., and Shuckburgh, E.: Seasonal Arctic sea ice forecasting with probabilistic deep learning, Nature Communications, 12, https://doi.org/10.1038/s41467-021-25257-4, 2021.

Badrinarayanan, V., Kendall, A., and Cipolla, R.: SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 39, 2481–2495, https://doi.org/10.1109/tpami.2016.2644615, 2017.

Balmaseda, M. A., Mogensen, K., and Weaver, A. T.: Evaluation of the ECMWF ocean reanalysis system ORAS4, Quarterly Journal of the Royal Meteorological Society, 139, 1132–1161, https://doi.org/https://doi.org/10.1002/qj.2063, URL https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.2063, 2013.

Batrak, Y. and Müller, M.: On the warm bias in atmospheric reanalyses induced by the missing snow over Arctic sea-ice, Nature Communications, 10, https://doi.org/10.1038/s41467-019-11975-3, 2019.

Batrak, Y., Kourzeneva, E., and Homleid, M.: Implementation of a simple thermodynamic sea ice scheme, SICE version 1.0-38h1, within the ALADIN–HIRLAM numerical weather prediction system version 38h1, Geoscientific Model Development, 11, 3347–3368, https://doi.org/10.5194/gmd-11-3347-2018, 2018.

Bauer, P., Beljaars, A., Ahlgrimm, M., Bechtold, P., Bidlot, J.-R., Bonavita, M., Bozzo, A., Forbes, R., Hólm, E., Leutbecher, M., Lopez, P., Magnusson, L., Prates, F., Rodwell, M., Sandu, I., Untch, A., and Vitart, F.: Model Cycle 38r2: Components and Performance, https://doi.org/10.21957/XC1R0LJ6L, 2013.

Casati, B., Wilson, L. J., Stephenson, D. B., Nurmi, P., Ghelli, A., Pocernich, M., Damrath, U., Ebert, E. E., Brown, B. G., and Mason, S.: Forecast verification: current status and future directions, Meteorological Applications, 15, 3–18, https://doi.org/10.1002/met.52, 2008.

Cavalieri, D. J. and Parkinson, C. L.: Arctic sea ice variability and trends, 1979–2010, The Cryosphere, 6, 881–889, https://doi.org/10.5194/tc-6-881-2012, 2012.

Ciresan, D., Giusti, A., Gambardella, L., and Schmidhuber, J.: Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images, in: Advances in Neural Information Processing Systems, edited by Pereira, F., Burges, C., Bottou, L., and Weinberger, K., vol. 25, Curran Associates, Inc., URL https://proceedings.neurips.cc/paper/2012/file/459a4ddcb586f24efd9395aa7662bc7c-Paper.pdf, 2012a.

Ciresan, D., Meier, U., and Schmidhuber, J.: Multi-column deep neural networks for image classification, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, https://doi.org/10.1109/cvpr.2012.6248110, 2012b.

Comiso, J. C., Cavalieri, D. J., Parkinson, C. L., and Gloersen, P.: Passive microwave algorithms for sea ice concentration: A comparison of two techniques, Remote Sensing of Environment, 60, 357–384, https://doi.org/10.1016/s0034-4257(96)00220-9, 1997.

Comiso, J. C., Meier, W. N., and Gersten, R.: Variability and trends in the Arctic Sea ice cover: Results from different techniques, Journal of Geophysical Research: Oceans, 122, 6883–6900, https://doi.org/10.1002/2017jc012768, 2017.

Dee, D. P., Uppala, S. M., Simmons, A. J., Berrisford, P., Poli, P., Kobayashi, S., Andrae, U., Balmaseda, M. A., Balsamo, G., Bauer, P., Bechtold, P., Beljaars, A. C. M., van de Berg, L., Bidlot, J., Bormann, N., Delsol, C., Dragani, R., Fuentes, M., Geer, A. J., Haimberger, L., Healy, S. B., Hersbach, H., Hólm, E. V., Isaksen, L., Kållberg, P., Köhler, M., Matricardi, M., McNally, A. P., Monge-Sanz, B. M., Morcrette, J.-J., Park, B.-K., Peubey, C., de Rosnay, P., Tavolato, C., Thépaut, J.-N., and Vitart, F.: The ERA-Interim reanalysis: configuration and performance of the data assimilation system, Quarterly Journal of the Royal Meteorological Society, 137, 553–597, https://doi.org/https://doi.org/10.1002/qj.828, URL https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.828, 2011.

Dinessen, F., Hackett, B., and Kreiner, M. B.: Product User Manual For Regional High Resolution Sea Ice Charts Svalbard and Greenland Region, Tech. rep., Norwegian Meteorological Institute, 2020.

Dukhovskoy, D. S., Ubnoske, J., Blanchard-Wrigglesworth, E., Hiester, H. R., and Proshutinsky, A.: Skill metrics for evaluation and comparison of sea ice models, Journal of Geophysical Research: Oceans, 120, 5910–5931, https://doi.org/10.1002/2015jc010989, 2015.

Eguíluz, V. M., Fernández-Gracia, J., Irigoien, X., and Duarte, C. M.: A quantitative assessment of Arctic shipping in 2010–2014, Scientific Reports, 6, https://doi.org/10.1038/srep30682, 2016.

Fritzner, S., Graversen, R., and Christensen, K. H.: Assessment of High-Resolution Dynamical and Machine Learning Models for Prediction of Sea Ice Concentration in a Regional Application, 125, https://doi.org/10.1029/2020jc016277, neural Networks for predicting Sea-Ice concentration are only slightly more accurate than persistence forecasting for short-term predictions., 2020.

Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position, Biological Cybernetics, 36, 193–202, https://doi.org/10.1007/bf00344251, 1980.

Goessling, H. F. and Jung, T.: A probabilistic verification score for contours: Methodology and application to Arctic ice-edge forecasts, Quarterly Journal of the Royal Meteorological Society, 144, 735–743, https://doi.org/10.1002/qj.3242, 2018.

Goessling, H. F., Tietsche, S., Day, J. J., Hawkins, E., and Jung, T.: Predictability of the Arctic sea ice edge, Geophysical Research Letters, 43, 1642–1650, https://doi.org/10.1002/2015gl067232, 2016.

Grigoryev, T., Verezemskaya, P., Krinitskiy, M., Anikin, N., Gavrikov, A., Trofimov, I., Balabin, N., Shpilman, A., Eremchenko, A., Gulev, S., Burnaev, E., and Vanovskiy, V.: Data-Driven Short-Term Daily Operational Sea Ice Regional Forecasting, Remote Sensing, 14, https://doi.org/10.3390/rs14225837, URL https://www.mdpi.com/2072-4292/14/22/5837, 2022.

Haiden, T., Janousek, M., Vitart, F., Ben-Bouallegue, Z., Ferranti, L., Prates, F., and Richardson, D.: Evaluation of ECMWF forecasts, including the 2021 upgrade, https://doi.org/10.21957/XQNU5O3P, 2022.

He, K., Zhang, X., Ren, S., and Sun, J.: Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, https://doi.org/10.48550/ARXIV.1502.01852, 2015.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., Simmons, A., Soci, C., Abdalla, S., Abellan, X., Balsamo, G., Bechtold, P., Biavati, G., Bidlot, J., Bonavita, M., De Chiara, G., Dahlgren, P., Dee, D., Diamantakis, M., Dragani, R., Flemming, J., Forbes, R., Fuentes, M., Geer, A., Haimberger, L., Healy, S., Hogan, R. J., Hólm, E., Janisková, M., Keeley, S., Laloyaux, P., Lopez, P., Lupu, C., Radnoti, G., de Rosnay, P., Rozum, I., Vamborg, F., Villaume, S., and Thépaut, J.-N.: The ERA5 global reanalysis, Quarterly Journal of the Royal Meteorological Society, 146, 1999–2049, https://doi.org/https://doi.org/10.1002/qj.3803, URL https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3803, 2020.

Hibler, W. D.: A Dynamic Thermodynamic Sea Ice Model, Journal of Physical Oceanography, 9, 815–846, https://doi.org/10.1175/1520-0485(1979)009<0815:adtsim>2.0.co;2, 1979.

Ho, J.: The implications of Arctic sea ice decline on shipping, Marine Policy, 34, 713–715, https://doi.org/10.1016/j.marpol.2009.10.009, 2010.

Holland, P. R. and Kimura, N.: Observed Concentration Budgets of Arctic and Antarctic Sea Ice, Journal of Climate, 29, 5241–5249, https://doi.org/10.1175/jcli-d-16-0121.1, 2016.

Hunke, E. C. and Dukowicz, J. K.: An Elastic–Viscous–Plastic Model for Sea Ice Dynamics, Journal of Physical Oceanography, 27, 1849–1867, https://doi.org/10.1175/1520-0485(1997)027<1849:aevpmf>2.0.co;2, 1997.

Hunke, E. C., Lipscomb, W. H., Turner, A. K., Jeffery, N., and Elliott, S.: CICE: the Los Alamos Sea Ice Model Documentation and Software User's Manual Version 5.1 LA-CC-06-012, techreport, Los Alamos National Laboratory, Los Alamos NM 87545, 2015.

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) ., https://doi.org/10.25607/OBP-1515, 2014.

Johnson, S. J., Stockdale, T. N., Ferranti, L., Balmaseda, M. A., Molteni, F., Magnusson, L., Tietsche, S., Decremer, D., Weisheimer, A., Balsamo, G., Keeley, S. P. E., Mogensen, K., Zuo, H., and Monge-Sanz, B. M.: SEAS5: the new ECMWF seasonal forecast system, Geoscientific Model Development, 12, 1087–1117, https://doi.org/10.5194/gmd-12-1087-2019, URL https://gmd.copernicus.org/articles/12/1087/2019/, 2019.

Kern, S., Lavergne, T., Notz, D., Pedersen, L. T., Tonboe, R. T., Saldo, R., and Sørensen, A. M.: Satellite passive microwave sea-ice concentration data set intercomparison: closed ice and ship-based observations, The Cryosphere, 13, 3261–3307, https://doi.org/10.5194/tc-13-3261-2019, 2019.

Kern, S., Lavergne, T., Notz, D., Pedersen, L. T., and Tonboe, R.: Satellite passive microwave sea-ice concentration data set inter-comparison for Arctic summer conditions, The Cryosphere, 14, 2469–2493, https://doi.org/10.5194/tc-14-2469-2020, 2020.

Kirillov, A., He, K., Girshick, R., Rother, C., and Dollár, P.: Panoptic Segmentation, https://doi.org/10.48550/ARXIV.1801.00868, 2018.

Køltzow, M., Schyberg, H., Støylen, E., and Yang, X.: Value of the Copernicus Arctic Regional Reanalysis (CARRA) in representing near-surface temperature and wind speed in the north-east European Arctic, Polar Research, 41, https://doi.org/10.33265/polar.v41.8002, 2022.

Kristensen, N. M., JensBDebernard, SebastianMaartensson, Keguang Wang, and Hedstrom, K.: Metno/Metroms: Version 0.3 - Before Merge, https://doi.org/10.5281/ZENODO.1046114, 2017.

Krizhevsky, A., Sutskever, I., and Hinton, G. E.: ImageNet Classification with Deep Convolutional Neural Networks, in: Advances in Neural Information Processing Systems, edited by Pereira, F., Burges, C., Bottou, L., and Weinberger, K., vol. 25, Curran Associates, Inc., URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf, 2012.

Lavelle, J., Tonboe, R., Tian, T., Pfeiffer, R.-H., and Howe, E.: Product User Manual for the OSI SAF AMSR-2 Global Sea Ice Concentration Product OSI-408, Tech. Rep. 1.1, Danish Meteorological Institute, 2016.

Lavelle, J., Tonboe, R., Jensen, M. B., and Howe, E.: Validation Report for OSI SAF Global Sea Ice Concentration Product OSI-401-b, Tech. Rep. 1.2, Danish Meteorological Institute, 2017.

Lavergne, T., Sørensen, A. M., Kern, S., Tonboe, R., Notz, D., Aaboe, S., Bell, L., Dybkjær, G., Eastwood, S., Gabarro, C., Heygster, G., Killie, M. A., Brandt Kreiner, M., Lavelle, J., Saldo, R., Sandven, S., and Pedersen, L. T.: Version 2 of the EUMETSAT OSI SAF and ESA CCI sea-ice concentration climate data records, The Cryosphere,

13, 49–78, https://doi.org/10.5194/tc-13-49-2019, URL https://tc.copernicus.org/articles/13/49/2019/, 2019a.
Lavergne, T., Tonboe, R., Lavelle, J., and Eastwood, S.: Algorithm Theoretical Basis Document for the OSI SAF Global Sea Ice Concentration Climate Data Record OSI-450, OSI-430-b, techreport 1.2, 2019b.
LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D.: Backpropagation Applied to Handwritten Zip Code Recognition, Neural Computation, 1, 541–551, https://doi.org/10.1162/neco.1989.1.4.541, 1989.
Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P.: Focal Loss for Dense Object Detection, 2017.
Liu, Y., Bogaardt, L., Attema, J., and Hazeleger, W.: Extended Range Arctic Sea Ice Forecast with Convolutional Long-Short Term Memory Networks, Monthly Weather Review, https://doi.org/10.1175/mwr-d-20-0113.1, 2021.
Long, J., Shelhamer, E., and Darrell, T.: Fully convolutional networks for semantic segmentation, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, https://doi.org/10.1109/cvpr.2015.7298965, 2015.
Melsheimer, C.: ASI Version 5 Sea Ice Concentration User Guide, Tech. rep., Institute of Environmental Physics, University of Bremen, 2019.
Melsom, A., Palerme, C., and Müller, M.: Validation metrics for ice edge position forecasts, Ocean Science, 15, 615–630, https://doi.org/10.5194/os-15-615-2019, 2019.
Müller, M., Batrak, Y., Kristiansen, J., Køltzow, M. A. Ø., Noer, G., and Korosov, A.: Characteristics of a Convective-Scale Weather Forecasting System for the European Arctic, Monthly Weather Review, 145, 4771–4787, https://doi.org/10.1175/mwr-d-17-0194.1, 2017.
Nair, V. and Hinton, G.: Rectified Linear Units Improve Restricted Boltzmann Machines Vinod Nair, vol. 27, pp. 807–814, 2010.
Noh, H., Hong, S., and Han, B.: Learning Deconvolution Network for Semantic Segmentation, in: 2015 IEEE International Conference on Computer Vision (ICCV), IEEE, https://doi.org/10.1109/iccv.2015.178, 2015.
Notz, D. and Community, S.: Arctic Sea Ice in CMIP6, Geophysical Research Letters, 47, https://doi.org/10.1029/2019gl086749, 2020.
Ólason, E., Boutin, G., Korosov, A., Rampal, P., Williams, T., Kimmritz, M., Dansereau, V., and Samaké, A.: A New Brittle Rheology and Numerical Framework for Large-Scale Sea-Ice Models, Journal of Advances in Modeling Earth Systems, 14, https://doi.org/10.1029/2021ms002685, 2022.
Palerme, C., Müller, M., and Melsom, A.: An Intercomparison of Verification Scores for Evaluating the Sea Ice Edge Position in Seasonal Forecasts, Geophysical Research Letters, 46, 4757–4763, https://doi.org/10.1029/2019gl082482, 2019.
RADU, M. D., COSTEA, I. M., and STAN, V. A.: Automatic Traffic Sign Recognition Artificial Inteligence - Deep Learning Algorithm, in: 2020 12th International Conference

on Electronics, Computers and Artificial Intelligence (ECAI), IEEE, https://doi.org/10.1109/ecai50035.2020.9223186, 2020.

Ronneberger, O., Fischer, P., and Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation, in: Lecture Notes in Computer Science, pp. 234–241, Springer International Publishing, https://doi.org/10.1007/978-3-319-24574-4_28, 2015.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J.: Learning representations by back-propagating errors, Nature, 323, 533–536, https://doi.org/10.1038/323533a0, 1986.

Röhrs, J., Gusdal, Y., Rikardsen, E., Moro, M. D., Brændshøi, J., Kristensen, N. M., Fritzner, S., Wang, K., Sperrevik, A. K., Idžanović, M., Lavergne, T., Debernard, J., and Christensen, K. H.: "in prep for GMD" An operational data-assimilative coupled ocean and sea ice ensembleprediction model for the Barents Sea and Svalbard, p. 20, 2022.

Sakov, P., Counillon, F., Bertino, L., Lisæter, K. A., Oke, P. R., and Korablev, A.: TOPAZ4: an ocean-sea ice data assimilation system for the North Atlantic and Arctic, Ocean Science, 8, 633–656, https://doi.org/10.5194/os-8-633-2012, 2012.

Serreze, M. C. and Meier, W. N.: The Arctic's sea ice cover: trends, variability, predictability, and comparisons to the Antarctic, Annals of the New York Academy of Sciences, 1436, 36–53, https://doi.org/https://doi.org/10.1111/nyas.13856, 2019.

Smith, D. M.: Extraction of winter total sea-ice concentration in the Greenland and Barents Seas from SSM/I data, International Journal of Remote Sensing, 17, 2625–2646, https://doi.org/10.1080/01431169608949096, 1996.

Spreen, G., Kaleschke, L., and Heygster, G.: Sea ice remote sensing using AMSR-E 89-GHz channels, Journal of Geophysical Research, 113, https://doi.org/10.1029/2005jc003384, 2008.

Spreen, G., Kwok, R., and Menemenlis, D.: Trends in Arctic sea ice drift and role of wind forcing: 1992-2009, Geophysical Research Letters, 38, n/a–n/a, https://doi.org/10.1029/2011gl048970, 2011.

Sørensen, A. M., Lavergne, T., and Eastwood, S.: Global Sea Ice Concentration Climate Data Record Product Uses Manual Product OSI-450 & OSI-430-b, Tech. Rep. 2.1, Norwegian Meteorological Institute, 2021.

Tonboe, R., Lavelle, J., Pfeiffer, R.-H., and Howe, E.: Product User Manual for OSI SAF Global Sea Ice Concentration, Tech. Rep. 1.6, Danish Meteorological Institute, 2017.

Veland, S., Wagner, P., Bailey, D., Everett, A., Goldstein, M., Hermann, R., Hjort-Larsen, T., Hovelsrud, G., Hughes, N., Kjøl, A., Li, X., Lynch, A., Müller, M., Olsen, J., Palerme, C., Pedersen, J. L., Rinaldo, ., Stephenson, S., and Storelvmo, T.: 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.

Wagner, P. M., Hughes, N., Bourbonnais, P., Stroeve, J., Rabenstein, L., Bhatt, U., Little, J., Wiggins, H., and Fleming, A.: Sea-ice information and forecast needs

for industry maritime stakeholders, Polar Geography, 43, 160–187, https://doi.org/10.1080/1088937x.2020.1766592, 2020.
Williams, T., Korosov, A., Rampal, P., and Ólason, E.: Presentation and evaluation of the Arctic sea ice forecasting system neXtSIM-F, The Cryosphere, 15, 3207–3227, https://doi.org/10.5194/tc-15-3207-2021, 2021.
Yu, T. and Zhu, H.: Hyper-Parameter Optimization: A Review of Algorithms and Applications, https://doi.org/10.48550/ARXIV.2003.05689, 2020.
Yu, X., Rinke, A., Dorn, W., Spreen, G., Lüpkes, C., Sumata, H., and Gryanik, V. M.: 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, The Cryosphere, 14, 1727–1746, https://doi.org/10.5194/tc-14-1727-2020, 2020.
Zampieri, L., Goessling, H. F., and Jung, T.: Predictability of Antarctic Sea Ice Edge on Subseasonal Time Scales, Geophysical Research Letters, 46, 9719–9727, https://doi.org/10.1029/2019gl084096, 2019.
Zeiler, M. D., Krishnan, D., Taylor, G. W., and Fergus, R.: Deconvolutional networks, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, https://doi.org/10.1109/cvpr.2010.5539957, 2010.

9 Supporting Figures

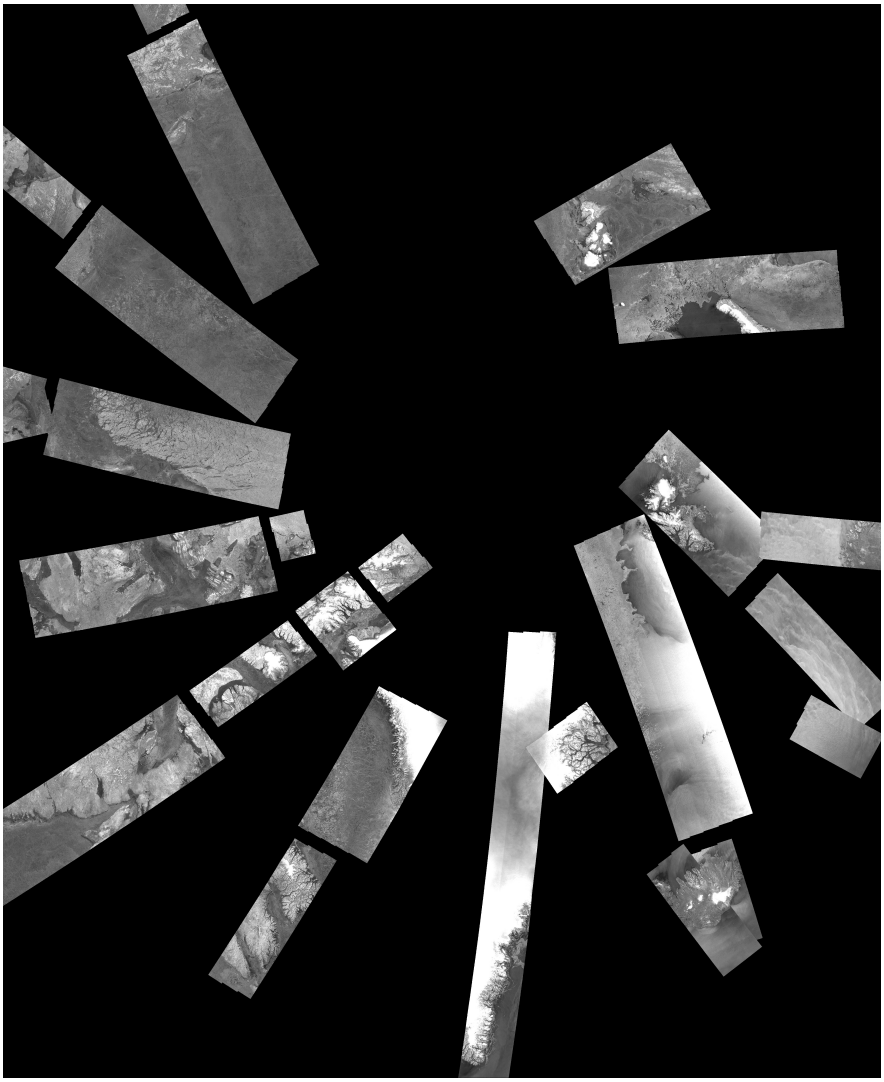


Figure 15: Daily SAR observations of the Arctic from Sentinel 1A 23 Jan 2023