Deep Graph Neural Networks for Spatiotemporal Forecasting of Sub-Seasonal Sea Ice: A Case Study in Hudson Bay

Zacharie Gousseau[1], K Andrea Scott[2], Philippe Lamontagne[3], and Mohammad Sina Jahangir[4]

[1]Department of Systems Design Engineering, University of Waterloo
[2]Department of Mechanical and Mechatronics Engineering, University of Waterloo
[3]Coastal and River Engineering Research Centre, National Research Council Canada
[4]Department of Civil Engineering, University of Waterloo

December 27, 2023

Deep Graph Neural Networks for Spatiotemporal Forecasting of Sub-Seasonal Sea Ice: A Case Study in Hudson Bay

Zacharie Gousseau[1], Philippe Lamontagne[2], and Mohammad Sina Jahangir[3], K. Andrea Scott[4]

[1]Department of Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada
[2]Ocean, Coastal and River Engineering Research Centre, National Research Council Canada, Ottawa, Ontario, Canada
[3]Department of Civil Engineering, University of Waterloo, Waterloo, Ontario, Canada
[4]Department of Mechanical and Mechatronics Engineering, University of Waterloo, Waterloo, Ontario, Canada

Key Points:

- GraphSIFNet employs a sequence-to-sequence deep learning framework based on the Graph Long-Short Term Memory (GCLSTM) framework for sub-seasonal sea ice forecasting.
- The model demonstrates superior performance over a statistical baseline in Hudson Bay, particularly in short- to medium-term predictions of sea ice concentration.
- GraphSIFNet's graph-based approach provides a more natural representation of sea ice dynamics, more closely resembling physically-based models than those based on two-dimensional kernel convolutions.

Corresponding author: Zacharie Gousseau, `zgoussea@uwaterloo.ca`

Abstract

This study introduces GraphSIFNet, a novel graph-based deep learning framework for spatiotemporal sea ice forecasting. GraphSIFNet employs a specialized Graph Long-Short Term Memory (GCLSTM) module within a sequence-to-sequence architecture to predict daily sea ice concentration (SIC) and sea ice presence (SIP) in Hudson Bay over a 90-day time horizon. The use of graph networks allows the domain to be discretized into arbitrarily specified meshes. This study demonstrates the model's ability to forecast over an irregular mesh with higher spatial resolution near shorelines, and lower resolution otherwise. Utilizing atmospheric data from ERA5 and oceanographic data from GLORYS12, the model is trained to model complex spatial relationships pertinent to sea ice dynamics. Results demonstrate the model's superior skill over a linear combination of persistence and climatology as a statistical baseline. The model showed skill particularly in short- to medium-term (up to 35 days) SIC forecasts, with a noted reduction in root mean squared error (RMSE) by up to 10% over the statistical baseline during the break-up season, and up to 5% in the freeze-up season. Long-term (up to 90 days) SIP forecasts also showed significant improvements over the baseline, with increases in accuracy of around 10% even at a lead time of 90 days. Variable importance analysis via feature ablation was conducted which highlighted current sea ice concentration and thickness as critical predictors. Thickness was shown to be important at longer lead times during the melting season suggesting its importance as an indicator of ice longevity, while concentration was shown to be more critical at shorter lead times which suggests it may act as an indicator of immediate ice integrity. The study lays the groundwork for future exploration into dynamic mesh-based forecasting, the use of more complex graph structures, and mesh-based forecasting of climate phenomena beyond sea ice.

1 Introduction

The drastic loss of Arctic sea ice volume is one of the most visible and immediate impacts of climate change (J. Stroeve & Notz, 2018). The Arctic is the fastest-warming region on Earth, and this warming is affecting the sea ice cover more than any other component of the climate system (Vihma, 2014; J. C. Stroeve et al., 2012; Cavalieri & Parkinson, 2012). According to the National Snow and Ice Data Center (NSIDC), Arctic sea ice extent (SIE)—the total area of the Arctic Ocean with at least 15% ice cover—is seeing a steady decline. This is especially prominent in September when sea ice extent is at its minimum (Serreze & Meier, 2019). Declining sea cover is connected to increasing air temperatures, changes in atmospheric and oceanic circulation, the albedo feedback loop, and the concentration of greenhouse gases in the atmosphere (J. C. Stroeve et al., 2012). The Arctic ice cover is of particular importance as it helps regulate the Earth's climate, and the decline in sea ice and subsequent loss of reflectivity directly contribute to the acceleration of climate change (Moon et al., 2019). Changes in Arctic sea ice cover also disturb marine and terrestrial ecological dynamics (Post et al., 2013); create challenges for Northern communities (Meier et al., 2014); and influence human activity as new trade routes become available through the Arctic (Mudryk et al., 2021). Forecasting sea ice conditions is therefore becoming increasingly important as accurate knowledge of these changes would allow for more effective preparation.

In this study, we introduce a deep learning based sea ice forecasting model that employs Graph Neural Networks (GNNs) integrated within a Long Short-Term Memory (LSTM) module to predict daily sea ice concentration (SIC) and sea ice presence (SIP) in Hudson Bay up to 90 days in advance. The choice of Hudson Bay as our study area is driven by its important role as a shipping hub, the presence of communities living within the region relying on maritime re-supply, and its unique characteristics as an in-land sea largely isolated from the wider Arctic. The 90-day forecasting horizon addresses the needs for planning and decision-making in industries such as shipping operations as well as the planning requirements of local communities residing in the region. This time horizon covers short-term (up to 7 days), medium-term (up to a month) and long-term (up to 3 months) planning needs. The study highlights the effectiveness of GNNs in handling irregular spatial domains by dividing Hudson Bay into a spatially irregular mesh with a higher resolution along shorelines. We evaluate the performance of two types of spatial graph convolutions within the model: the basic Graph Convolutional Network (GCN) and an attention-based transformer convolution. The model was trained using sea ice and oceanographic data from a coupled ice-ocean reanalysis product (GLORYS12 (Jean-Michel et al., 2021)), as well as atmospheric data from the ECMWF Reanalysis v5 (ERA5 (Hersbach et al., 2020)). We validate the model's accuracy by comparing its predictions to a statistical baseline and comparing forecasted and observed freeze-up and break-up dates at ports on Hudson Bay.

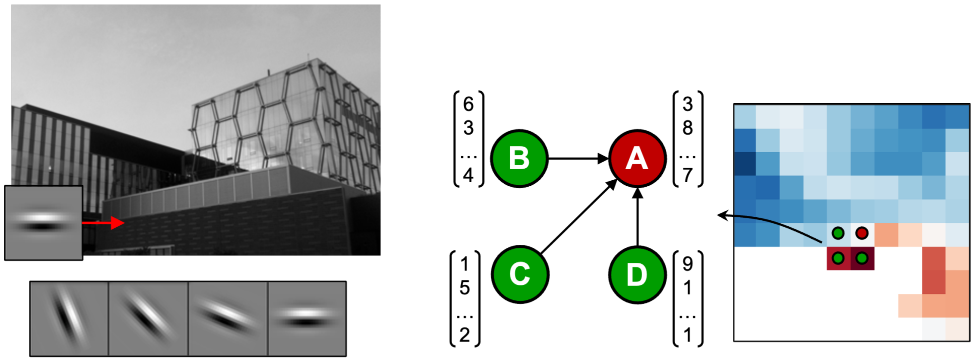
2 Background

Sea ice forecasting is a spatiotemporal forecasting task which can be formulated as a next-frame prediction problem. Given a sequence of frames $\mathbf{X} = (\mathbf{X}_{t-n}, ..., \mathbf{X}_{t-1}, \mathbf{X}_t)$ with $\mathbf{X_t} \in \mathbb{R}^{w \times h \times c}$ where n is the number of frames in the sequence, w and h are the spatial dimensions of the frames, and c is the number of channels, the objective is to predict the next T frames in the sequence, $X_{t+1}, ..., X_{t+T}$.

While traditional time series modeling techniques such as ARIMA have been widely used for forecasting, they are less effective for spatiotemporal forecasting due to their inherent limitations in handling spatial dependencies and complex temporal dynamics. ARIMA models, primarily designed for univariate time series, lack the capacity to effectively model spatial relationships and multi-dimensional data structures, which are critical in spatiotemporal forecasting. To address these limitations, methods like Vector Autoregression (VAR) (Sims, 1980) and Spatial Autoregressive (SAR) (Anselin, 1988) models were developed, offering improved handling of multivariate data and spatial dependencies, respectively. However, these models still struggled with dynamic spatial relationships and non-linear interactions. Space-Time Autoregressive Integrated Moving Average (STARIMA) models (Pfeifer & Deutsch, 1980) were introduced to better integrate spatial dependencies with temporal dynamics. Dynamic Linear Models (DLMs) and State Space Models (Kalman, 1960) offered a framework for handling evolving temporal dynamics but were limited in their spatial modeling capabilities.

With the advent of deep learning, many neural network methods were developed for spatiotemporal problems, largely based on spatial convolutions with fixed-size two- or three-dimensional kernels (Oprea et al., 2022). These convolutional models are particularly well-suited for image data with a gridded structure such as images or video frames and allow for learning rich features that are present in real-world image sequences.

Graph Neural Networks (GNNs) offer a compelling alternative to Convolutional Neural Networks (CNNs) for emulating models of physical processes, such as ice dynamics, for several reasons. One of the primary advantages of GNNs in this context is their inherent ability to capture the spatial relationships between neighboring nodes through graph edges, which can be arbitrarily specified. This is particularly crucial in applications like sea ice dynamics, where the spatial relationships are fundamental in determining heat and momentum exchanges, and other factors influencing ice processes. In GNNs, both nodes and edges can encode information about the system, and graph convolutions update these encodings by applying some non-linear function. This allows GNNs to effectively model the exchange of physical quantities such as heat or ice volume at a given location in space and time while accounting for the directionality of processes, which is represented by directed edges. In contrast, CNNs operate on a fundamentally different principle. They extract features such as edges or gradients from an input image by tuning kernel filters. This process involves convolving these filters over the input image to identify patterns and features at various scales and orientations. While this approach



(a) CNN: Kernel filters (bottom figures) are learned to extract patterns in an image of a building.

(b) GNN: Non-linear functions are learned to model the relationship between neighboring nodes in a graph.

Figure 1: Conceptual comparison of the mechanisms of convolutional neural networks (CNN) and graph neural networks (GNN). (a) CNNs learn kernel filters which slide across the image to identify patterns in the image, such as edges or gradients. (b) GNNs learn a function to update a target node's state vector (A) by non-linearly combining the state vectors of its neighbours (B, C, D).

is highly effective for tasks like image recognition, where identifying and categorizing visual patterns is key, it may not be as well-suited for learning the underlying physical laws that govern interactions between points in space. CNNs typically lack the ability to explicitly model directional relationships and complex dependencies between disparate points in a spatial domain, which are critical in understanding and predicting physical phenomena like ice dynamics. A high-level visual representation of these two neural network types, highlighting their structural and functional differences, is shown in Figure 1. CNNs leverage spatial locality and translation invariance inherent in images through convolutional layers with fixed-size filters that extract local features across the image. Techniques such as the use of pooling operators, stride convolutions, or dilated filters can be used to capture longer-range patterns and hierarchical information(K. He et al., 2016; Yu & Koltun, 2016). In contrast, message-passing GNNs can natively capture long-range patterns through edge propagation, potentially reaching across the entire graph structure given a sufficiently deep network. Although in most cases the underlying graphs are too large for information to be propagated globally, limited information propagation across can help models gain a holistic view of the spatial domain and learn complex spatial patterns (Wu et al., 2022). Additionally, most types of GNNs exhibit both translation and rotation invariance as convolutions are applied indiscriminately to all nodes and the aggregation operators are most often permutation invariant. Note that this is not always the case; operators based on recurrent units such as the LSTM variant of GraphSAGE (Hamilton et al., 2017) or sorting units such as the SortPooling aggregator (M. Zhang et al., 2018) do not exhibit rotation invariance. Another noteworthy advantage of GNNs over CNNs is their scalability due to the inherent parallelism in their architecture, allowing for efficient processing of data over large regions or with fine resolution. This parallelism however comes at the cost of higher memory usage which may become limiting, though this can be circumvented by partitioning the graph and processing the subgraphs independently before combining the outputs. Overlapping subgraphs can be used to ensure no spatial artifacts or discontinuities arise from the partitioning.

3 Related work

Prior to the advent of deep learning techniques in sea ice forecasting, traditional physics-based and statistical models were the mainstay for both short-term and long-term predictions. Dynamic models, often integrated within data assimilation systems, such as the Pan-Arctic Ice-Ocean Modeling and Assimilation System (PIOMAS) (J. Zhang & Rothrock, 2003), rely on solving physical equations to simulate the interactions between sea ice, atmosphere, and ocean. These models are computationally intensive and require extensive calibration, but are considered fairly reliable due to their capacity to incorporate well-understood physical processes and parameters. On the other hand, statistical models such as multiple linear regression (MLR) and autoregressive integrated moving average (ARIMA) have been used for their simplicity and computational efficiency relative to physical-based models (Petty et al., 2017). These models often utilize historical sea ice concentration, temperature, and other meteorological variables to make short-term forecasts. However, they lack the ability to adequately capture the complex spatial and temporal patterns inherent in sea ice dynamics needed to forecast over longer timeframes.

The application of deep learning techniques to sea ice forecasting has gained increasing attention in recent years due to their computational efficiency and generalizability, particularly in the face of a changing climate and increased availability of large training datasets. Early studies applying deep learning to sea ice forecasting were limited to either spatial or temporal modelling. For instance, Chi and Kim (2017) used a long-short term memory (LSTM) module to forecast sea ice on a per-pixel level but did not consider spatial patterns. Kim et al. (2019) later used a deep neural network (DNN) with two fully-connected layers to forecast sea ice concentration considering interactions between pixels through dense layers but did not explicitly account for spatial autocorrelation. Later models based on the convolutional neural network (CNN) were able to leverage spatial patterns. Andersson et al. (2021) used a U-net trained on both climate simulation and observation data to forecast monthly sea ice concentration and was found to out-perform the SEAS5 dynamical model, but did not explicitly model in the temporal dimensions. Spatiotemporal models were then proposed that unify spatial and temporal models. Liu et al. (2021) proposed a model based on the convolutional long-short term memory (ConvLSTM) (X. Shi et al., 2015) to perform one-step ahead forecasting of sea ice in the Barents sea which showed promise by outperforming statistical baselines. Asadi et al. (2022) built on this work by proposing a sequence-to-sequence model based on the ConvLSTM to forecast sea ice presence in Hudson Bay. The model generally outperformed the European Centre for Medium-Range Weather Forecasts's (ECMWF) subseasonal-to-seasonal (S2S) ensemble predictions (Vitart & Robertson, 2018).

GNN-based approaches have recently seen some attention in global climate modelling, motivated in part by successes in GNN-based physics simulation models such as MeshGraphNets (Pfaff et al., 2020) or graph network simulators (Sanchez-Gonzalez et al., 2020; Rubanova et al., 2022). Keisler (2022) first proposed a GNN for forecasting the global climate using an autoregressive encoder-processor-decoder architecture. Gridded reanalysis data was encoded onto an icosohedron graph structure on which a message-passing neural network performed several steps of processing before being decoded back onto the latitude-longitude grid. Results showed that the model is competitive in comparison with state-of-the-art physical models when forecasting geopotential height and temperature over a 6-day rollout with a 6-hour temporal step. Lam et al. (2022) built upon this work with GraphCast, a similar model structure with the most notable difference being the use of multiple icosahedron grids at varying spatial resolution. They demonstrated greater skill than operational state-of-the-art physical models when forecasting global temperature, precipitation, and wind patterns over a 10-day rollout at a 6-hour temporal step.

4 Methodology

4.1 Data

In this study, ERA5 reanalysis data is used as atmospheric forcing data to train the models along with oceanographic variables from the GLORYS12 reanalysis product. Sea ice concentration estimates from GLORYS12 are used as the target variable and a proxy for the ground truth.

4.1.1 ERA5

ERA5 (Hersbach et al., 2020) is a climate reanalysis dataset produced by ECMWF that offers hourly estimates of climatic variables at a spatial resolution of 0.25° from 1979 to present. It is based on the IFS Cycle 41r2 4D-Var data assimilation system and includes a wide range of climatic variables at different pressure levels of the atmosphere. The IFS system assimilates observations from dozens of satellite missions and ground stations to create a physically consistent best representation of atmospheric conditions. Although the model does not have a coupled ocean-atmosphere component, it uses daily passive microwave-derived sea ice concentration estimates from the Ocean and Sea Ice Satellite Application Facilities (OSI-SAF) as boundary conditions (Hersbach et al., 2020). In this study, we follow previous studies (Asadi et al., 2022; Andersson et al., 2021) and use 2-meter temperature, 10-meter wind speeds, and surface sensible heat fluxes from ERA5 as input features to our model (see Table 1)

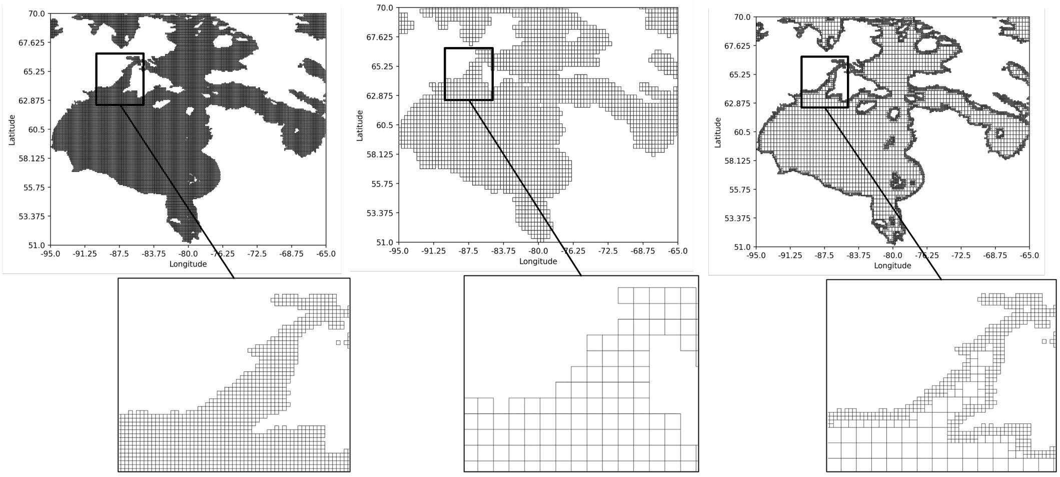
4.1.2 GLORYS12

GLORYS12 (Jean-Michel et al., 2021) is a global ocean and sea ice reanalysis data product developed by the Copernicus Marine Environment Monitoring Service (CMEMS), utilizing the LIM2 EVP NEMO 3.1 platform (Madec, n.d.) in the ORCA025 configuration designed by the DRAKKAR consortium. This configuration includes a global sea-ice model with a 1/4° Mercator grid. Atmospheric forcing for the ocean surface model is provided by ECMWF's ERA-Interim (Dee et al., 2011) reanalysis data until 2019, and ERA5 data thereafter. The spatial resolution of the ocean and ice models is 1/12°. The data assimilation component of GLORYS12 includes in-situ temperature and salinity (T&S) profiles, satellite sea surface temperature (SST), and along track sea-level anomalies derived from satellite altimetry. The assimilation of oceanic observations occurs using a reduced-order Kalman filter, which is based on a singular evolutive extended Kalman (SEEK) filter. The SEEK filter utilizes a three-dimensional multivariate background error covariance matrix and operates on a 7-day assimilation cycle. The system also integrates sea ice concentration observations from IFREMER/CERSAT. Historical records are available from 1993 to present. This study uses GLORYS12 sea ice concentration, thickness, velocities and sea surface temperatures.

4.2 Meshing

Meshes allow for greater flexibility in defining the model's spatial basis. Unlike two-dimensional convolutional approaches, which require defining a regular two-dimensional grid of pixels over a region, meshes are comprised of cells of abitrary sizes, allowing the modeler to control which areas are modelled in higher resolution (e.g., around ports or passages of interest). Since cells are only defined in regions of interest we also avoid the need to apply a land mask as a post-processing step, unlike in CNN-based approaches which most often model over the whole region before applying a mask to exclude land pixels from the output.

Figure 2 shows possible meshes for Hudson Bay using a 1/12 degree grid as the base resolution when trying to balance resolution and computational requirements. The mesh



(a) Full resolution regular mesh (b) Coarsened regular mesh (c) Irregular mesh

Figure 2: Comparison of different mesh definitions for modeling Hudson Bay. (a) A high-resolution regular mesh with 32,856 cells, computationally intensive but highly detailed; (b) a four-times coarsened regular mesh with 2,425 cells lacking sufficient detail along land interfaces; (c) irregular mesh with 9,422 cells, a compromise for both computational efficiency and high resolution at land interfaces. This approach ensures no cell overlaps land while providing high-resolution data for critical regions like ports, passages, and areas of meteorological interest such as the Kivalliq latent heat polynya.

shown in (a) uses the base resolution as a regular mesh, which is computationally heavier with its 32,856 cells, while the mesh in (b) uses a regular four-times coarsened version of the same mesh with 2,425 cells, which may not have sufficient definition. At the shoreline, this coarse mesh overlaps land but the model does not have the ability to acknowledge this overlapping. A 4×4 cell with only one non-land pixel assigns the sea ice concentration value to the entire cell, possibly undermining the model's ability to reason about volumetric continuity. As a compromise between resolution and computational efficiency, an irregular mesh can be defined with the same four-times coarsened resolution refined near shorelines such that no cell overlaps land. This is shown in (c). This can be done by recursively splitting the cells of the base (coarsened) mesh in four equal parts until no cell overlaps land. The result is a mesh with 9,422 cells. A secondary advantage of this technique is that modelling around shorelines at a higher resolution may be of interest to port operators or local communities. For shipping and freight purposes in Hudson Bay, there is a keen interest in knowing the state of the ice near shipping ports since some operations might required ice free conditions. However, large areas of navigable waters do not require the same high degree of spatial resolution since vessels have the possibility to slightly change their routes, thus a coarser resolution is sufficient.

To convert gridded data from a grid representation $X \in \mathbb{R}^{W \times H \times C}$ for data with C channels and $W \times H$ spatial dimensions to a mesh representation $G \in \mathbb{R}^{C \times N}$ with N cells, we first construct a sparse mapping tensor $M \in \mathbb{R}^{N \times W * H}$ where entry (n, p) is assigned 1 if the p^{th} pixel of the flattened grid $Y \in \mathbb{R}^{C \times W * H}$ should be mapped to cell n. We also construct a tensor $P \in \mathbb{R}^{N}$ which stores the number of pixels which are mapped to each cell. Then, to convert a sample from a grid to a mesh representation, for each node we find the mean value of each of its constituent pixels with

$$G = YM^T \oslash P \tag{1}$$

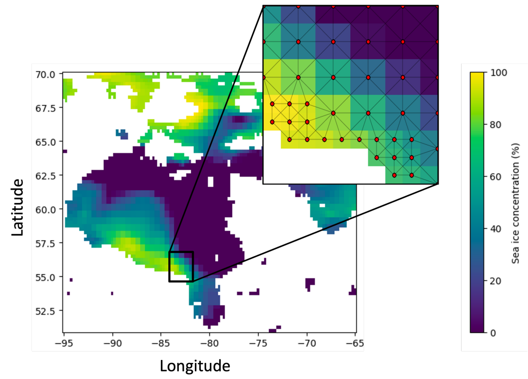


Figure 3: Input images are represented as graphs by relating each neighbouring pixel with edges. In this figure, a spatially irregular mesh is used to represent SIC in Hudson Bay, where red dots represent graph nodes and black lines represent edges.

where $\oslash$ represents an element-wise or Hadamard division. G can be converted back to a grid representation by splitting the cells back into its constituent pixels as

$$\hat{Y} = GM. \tag{2}$$

Since Equation 1 takes the mean of the constituent pixels of each cell, it cannot be perfectly reverted, instead Equation 2 simply assigns the cell value to each of its constituent pixels. Formulating these transformations as matrix multiplications allows for greater GPU acceleration which is important if the input meshes are re-meshed dynamically during training, although this is not done in this study.

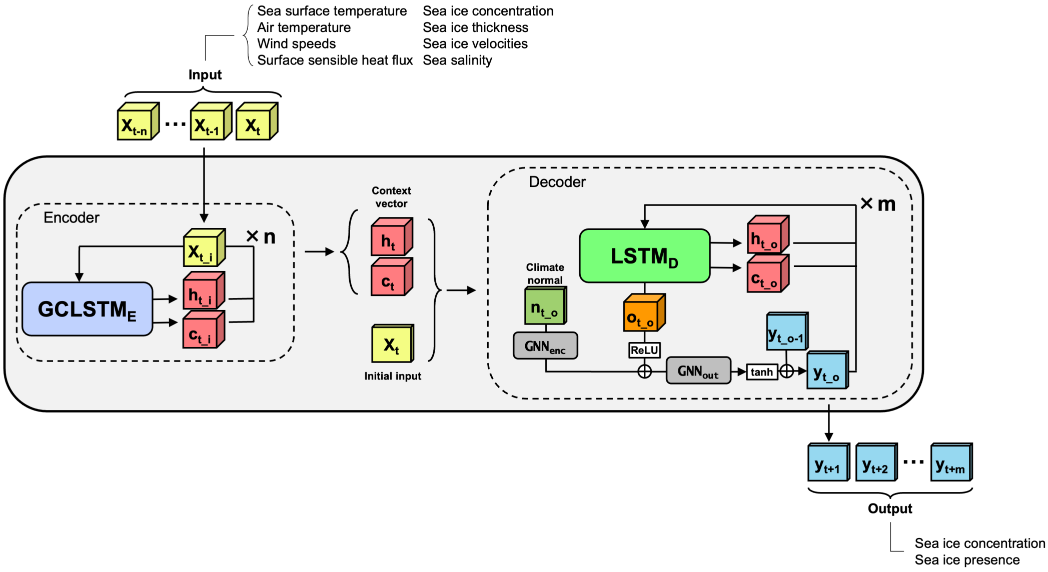
A graph can then be defined based on this mesh by assigning a node to each cell and placing edges between any two neighboring cell as in Figure 3. To preserve spatial awareness, the positions of each node and size of each cell are added as node features, and the length and angle of the edges are stored as edge features. The edges are therefore considered to be directed edges as the edge features are direction-dependent, that is, for two nodes x_i and x_j, the edge from x_i to x_j (e_{ij})) is not equivalent to the edge from x_j to x_i (e_{ji}))

4.3 Model Architecture

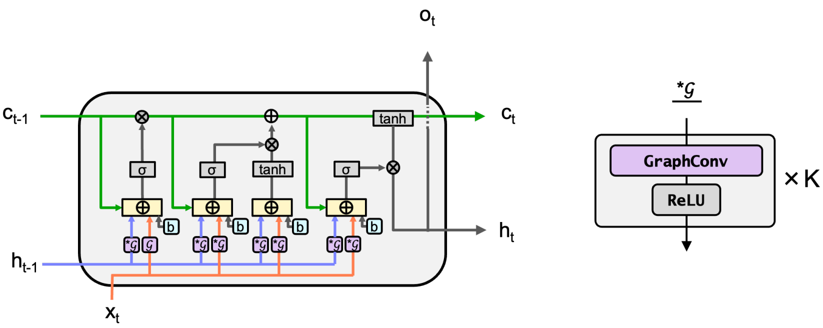
The proposed model uses graph convolutional long-short term memory (GCLSTM) modules within a sequence-to-sequence architecture. The GCLSTM module and the overall architecture are shown in Figure 4, and described in the subsections below.

4.3.1 GCLSTM

The graph convolutional long-short term memory (GCLSTM) module used in this work is a modified version of the model from Seo et al. (2018), which is in turn inspired by the ConvLSTM first proposed by X. Shi et al. (2015). The module closely resembles the peephole LSTM introduced by (Gers et al., 2002), with the only modification being the addition of graph convolution operators over the hidden and input states at each of the input, forget, cell and output gates in the place of weight matrices. This is represented as the $*\mathcal{G}$ block in Figure 4b. The graph convolution operators allow information exchange between nodes through the directed edges. The model proposed by Seo et al. (2018) uses a single Chebyshev graph convolution (M. He et al., n.d.) which has limited spatial expressivity since a single convolution can only exchange information between



(a) Overall model architecture. The last hidden (h_t) and cell (c_t) states of the encoder act as the context vectors and are used as the initial states of the decoder. The encoder learns features from the n input timesteps, and the last hidden (h_t) and cell (c_t) states are retained as the context vector used to initiate the decoder, which unrolls over the fixed m desired output timesteps. The initial input to the decoder X_t is the ice channel of the last input timestep. GNN_{enc} and GNN_{out}, used to encode climatology at each output timestep (n_{t_o}) and reduce the dimensionality of the output (o_{t_o}), respectively, are stacked spatial convolutions with leaky ReLU activations.



(b) Graph convolutional long-short term memory (GCLSTM) module. The module is based on the peephole LSTM (Gers et al., 2002), with the addition of K stacked graph convolutions applied to both the hidden states and input.

Figure 4: Model architecture showing (a) overall encoder-decoder architecture, and (b) a single graph convolutional long-short term memory (GConvLSTM) cell. $\bigoplus$ represents element-wise addition, and $\bigotimes$ represents element-wise multiplication.

immediate neighbors. Since the processes dominating ice formation and break-up are physical processes occurring across space, we wish to increase the model's ability to recognize spatial patterns, and therefore use K stacked convolutions followed by leaky ReLU activations, which provides information exchange over K hops. The peephole variant of the LSTM is used here as it has been shown to outperform the vanilla LSTM (Joshi et al., 2022), particularly for video understanding (Srivastava et al., 2015). The convolution operator taking the place of GraphConv in Figure 4b can be arbitrarily selected from the myriad graph convolution operators that have been proposed. In this work, we evaluate both the graph transformer convolution from Y. Shi et al. (2021), and the more basic Graph Convolutional Network (GCN) first proposed by Kipf and Welling (Kipf & Welling, 2017).

In the graph transformer convolution, the feature vector of a given node i, x_i, is updated by aggregating information from its neighbors $j \in \mathcal{N}(i)$, and the node itself, using edge features from i to j, e_{ij}. The governing equation for the graph transformer convolution is

$$x_i' = W_1 x_i + \sum_{j \in \mathcal{N}(i) \cup i} \alpha_{ij} (W_2 x_j + W_3 e_{ij}) \tag{3}$$

where $\mathcal{N}(i)$ denotes the neighbors of node i, W are weight matrices that project the inputs to their latent representation where the attention coefficients α_{ij} are given by

$$\alpha_{ij} = softmax\left(\frac{(W_4 x_i)^T (W_4 x_j + W_3 e_{ij})}{\sqrt{d}}\right) \tag{4}$$

.

The attention weights allow the model to selectively attend to a given node's neighbors based on their node and edge feature vectors. The inclusion of edge features and an edge specific weight matrix allows the model to learn to relate the edge features to better reflect anisotropic evolution of the model state.

We compare the transformer convolution with the Graph Convolutional Network (GCN) proposed by Kipf and Welling (2017), as it is a commonly used and simpler convolution operator. The GCN operator is defined by the equation

$$x_i' = W^T \sum_{j \in \mathcal{N}(i) \cup i} \frac{e_{ij}}{\sqrt{\hat{d}_j \hat{d}_i}} x_j \tag{5}$$

where X is a weight matrix, $\hat{d}_i = 1 + \sum_{j \in \mathcal{N}(i)} e_{ij}$ and e_{ij} are the edge weights from i to j. Since e_{ij} must be a scalar, here we use the normalized distance between nodes as the edge weights. Note that this limits the spatial awareness of the model as it does not receive information about the nodes' relative positions, unlike the transformer convolution.

4.3.2 Sequence-to-Sequence Architecture

The GCLSTM module is used within a sequence-to-sequence encoder-decoder structure to learn features from the inputs and evolve the sea ice state forward in time. The overall architecture is shown in Figure 4a. Since navigation and offshore operations are affected at various degree by the presence and concentration of sea ice, our model forecasts both SIC and SIP as a multi-task learning approach. Although sea ice presence—defined as any pixel where SIC is greater than 15%—can be derived from the forecasted SIC values, a model trained without the secondary SIP forecasts would not be optimized for this 15% threshold. It was also found through experimentation that including SIP as a secondary task improved SIC forecasts in the break-up and freeze-up seasons.

The encoder is responsible for learning rich spatiotemporal features from the input sequence while the decoder is responsible for evolving the state forward in time from these learned features. The encoder therefore acts as an information bottleneck, meaning it is crucial that the encoder is sophisticated enough to distill the inputs into a context vector with sufficient information for the decoder to use in the unrolling process. Given a sufficiently rich context vector, the decoder does not necessarily need to learn additional spatial features within the context vector, nor during the unrolling process. Therefore, in this work we use a spatiotemporal GCLSTM module in the encoder block, and a simple LSTM in the decoder block. Although the decoder block also contains graph convolutions (e.g., in GNN_{out}), the distinction between the two is that the GCLSTM in the encoder block integrates graph convolutions within the temporal model allowing for simultaneous spatial and temporal modelling, while the decoder block models temporal and spatial dynamics separately, with GNN_{out} being used mainly for dimensionality reduction. Using an LSTM rather than a GCLSTM module in the decoder block also greatly reduces training time in the case where there are fewer input timesteps than output timesteps. Note that experiments with a GCLSTM in the decoder were also run but showed no improvements over using an LSTM.

The encoder processes each input timestep sequentially, updating the hidden and cell states at each timestep with layer normalization (Ba et al., 2016) applied to the hidden and cell states after each timestep to increase model stability. The final hidden and cell states are the high-dimensional vectors that are taken as the context vectors that contain the learned features from the input and are used to initialize the hidden and cell state of the decoder. The last input ice state is used as the initial input to the decoder (or start token) since we wish to evolve the state forward from this initial state. The decoder is run recurrently for the desired number of output timesteps in a similar fashion to the encoder but using the last step's prediction (y_{t-1}) as the input for the current step (y_t).

Since sea ice is highly seasonal, the model is susceptible to a form of modal collapse wherein the model converges to a local minimum, predicting only the average sea ice conditions for a particular day of the year. These daily averages are known as the climate normals or climatology. For long-term forecasting of climatological variables, climatology can perform reasonably well compared to dynamic or statistical models due to strong seasonality. Since we wish to outperform climatology and expect the model to learn to use it as a heuristic, we choose to include it as an input such that model can focus on learning departures from normal conditions. This was shown to be beneficial for sea ice forecasting in a previous study (Asadi et al., 2022). Climate normals are calculated as the mean ice concentration values for each day of the year over the entire training set and are encoded into latent space using a shallow multi-layer GNN before being combined with the decoder output by element-wise addition. The result is then fed through a multi-layer GNN with leaky ReLU activations to reduce the dimensionality to two, and finally through a hyperbolic tangent activation to map the values between -1 and 1. This output represents the change in sea ice conditions and is added to the last timestep's prediction. Since both SIC and SIP should be bound between 0 and 1, the output is passed through a sigmoid layer that produces the final predictions.

4.4 Experimental set-up

4.4.1 Mesh Definition

To illustrate the advantage of using graph networks, experiments were designed to demonstrate the ability to produce forecasts over an irregular mesh. To this end, experiments were run on an irregular mesh as well as the coarsened regular mesh described in Section 4.2 and shown in Figure 2b and Figure 2c. The irregular mesh is refined to a higher resolution at the land edges by splitting the base 1/3° mesh if a cell intersects

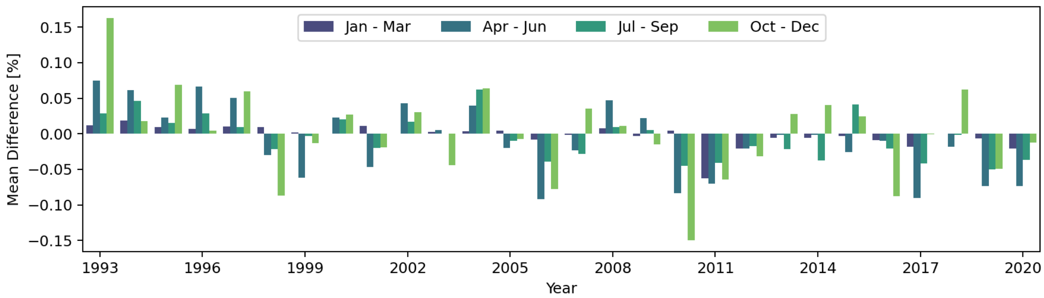


Figure 5: Monthly sea ice concentration anomalies in Hudson Bay from 1993-2020. Highlights periods of higher and lower-than-average sea ice concentrations.

a one-cell buffer around land. This buffer is used since near-shore dynamics can be particularly complex. By extending high-resolution meshing slightly beyond the immediate land-water interface, the model may be better equipped to capture these complex dynamics occurring in these more critical regions. The resulting irregular mesh contains 1/12°, 1/6° and 1/3° sized cells. To show that the complexities introduced by this irregular mesh is not a detriment to the model, a separate experiment is conducted by training the same model over the regular 1/3° mesh. This should be an easier task than the irregular mesh, therefore showing similar performance over either meshes is sufficient to demonstrate that the model is resolution-agnostic.

4.4.2 Data Partitioning

The Hudson Bay region, including Hudson Strait, James Bay and Foxe Basin, undergoes a cyclical transformation in its ice cover characterized by complete freezing during the winter months and total melt in the summer, with some multi-year ice possible in Foxe Basin. This seasonal cycle is subject to considerable inter-annual variability, both in terms of the rate at which these processes occur and the timing of these transitions. Figure 5 illustrates this variability by showing monthly SIC anomalies between 1993 and 2020. These anomalies are computed as the mean differences between observed SIC and the long-term average concentration for each corresponding month. The data reveals distinct periods of anomalous behavior in SIC. Specifically, the years 1993 to 1997 were marked by higher-than-average SIC, indicating that during these years, Hudson Bay experienced an earlier freeze-up and a delayed break-up season. In contrast, the period from 2010 to 2012 exhibited anomalously low SIC, characterized by a late onset of freeze-up and an earlier melting season. Including data from both these anomalous periods along with years that exhibit more typical ice conditions is critical for enhancing model robustness in the face of varying environmental conditions. This is particularly important in the context of climate change, where shifts in temperature and weather patterns could further exacerbate the variability in sea ice conditions. The data is therefore partitioned into a sequential 20-year, 3-year, 3-year split, wherein data from 1993-2013 is used for training, 2013-2016 is used for validation, and 2016-2019 is used for testing. Note however that the test period only includes years with normal or lower-than-usual ice conditions. Although this bias may not be optimal, lower-than-normal ice conditions may be more representative of future ice conditions in the Hudson region (J. Stroeve & Notz, 2018)

One model is trained for each month of the year, each denoted as a 'monthly model'. Each monthly model was trained using data from the respective month with a 15-day buffer before and after the beginning and end of the month respectively. For example, the April model is trained with input data for each day between March 16 and May 15 over all training years. A longer buffer of one month was tested but did not lead to significant improvements in model performance. In inference mode, each model is used only to produce a forecast with inputs from its respective month. For example, to generate 90 day forecasts for April, a 90 day forecast is launched for each day between April 1 and April 30. Training separate model for each month of the year was done since we expect the dynamics that must be learned for one time of the year to be sufficiently different from other times of the year such that each model will have greater accuracy by concentrating efforts in learning specific ice dynamics (Asadi et al., 2022). As a secondary benefit, this also allows training to be carried out more efficiently as each monthly model can be trained in parallel.

4.4.3 Input Features

Sea ice concentration data from GLORYS12 serve as the target variable, while atmospheric variables from ERA5, combined with oceanographic variables from GLORYS12, are used as input features. Sea ice dynamics are primarily influenced by factors such as air and sea temperature (Wang et al., 2019), wind (Stammerjohn et al., 2003), heat fluxes (Ivanov et al., 2012), and ocean salinity (Yao et al., 2000), thus we include these variables as input features. The 10 chosen input variables are listed in Table 1, along with the rationale for their selection. It should be noted that ERA5 hourly variables are regridded from their original 0.25° grid to match the GLORYS12 1/12° grid, and resampled to match the GLORYS12 daily temporal resolution. This is achieved through spatial linear interpolation and aggregation from an hourly to a daily resolution using a simple mean. The input sequence length is 10 days and the spatial domain as a grid is 229×361. Since the model operates over the mesh domain rather than the grid domain, the dimensionality of the inputs to the encoder as (input steps, number of nodes, input features) is $10 \times 9,422 \times 10$ for the irregular mesh and $10 \times 2,425 \times 10$ for the regular mesh. The input to the decoder is the context vectors provided by the encoder as well as the climatology for each forecast day. The output dimensionality is $90 \times 9,422 \times 2$ for the irregular mesh, and $90 \times 2,425 \times 2$ for the regular mesh.

4.4.4 Baseline Model

As a baseline model with which to compare the model, we use a combination of two common statistical baselines: persistence and climatology. Persistence refers to persisting the most recent sea ice conditions and tends to perform well at very short forecast lengths particularly outside of the freezing and melting seasons. Climatology refers to the pixel-wise average SIC for each day of the year where the average is taken over the historical period of interest. Climatology tends to perform best relative to forecast models at longer lead times. For forecasts produced over a seasonal scale, a stronger baseline than either persistence and climatology can be derived by combining the two using a weighted average with the relative weights varying by lead time, where more weight is given to persistence than climatology at short lead times and more weight is given to climatology than persistence at long lead times. The form chosen for the baseline model is

$$F = (1-\gamma)P + \gamma C, \tag{6}$$

where

$$\gamma(t) = \gamma_0 \times e^{-\lambda t}. \tag{7}$$

γ_0 is set to 1 since we know persistence to be a strong predictor at short lead times, and λ is optimized by minimizing the mean squared error over the training dataset for each month. The resulting weights are shown as a heatmap in Figure 6.

Table 1: Selected input variables to the encoder, data source and rationale for inclusion.

Short Name	**Full Name**	**Source**	**Rationale for Inclusion**
`sic`	Sea ice concentration	GLORYS12	Direct measure of what is being forecasted; crucial for temporal dynamics and initial conditions.
`sit`	Sea ice thickness	GLORYS12	Provides insights into the resiliency and robustness of the ice, affecting its likelihood to melt or deform.
`siuv`	Sea ice velocities	GLORYS12	Indicates the direction and speed of sea ice movement.
`so`	Sea water salinity	GLORYS12	Salinity affects the freezing point of sea water and is crucial in the dynamics of ice formation and melt.
`sst`	Sea surface temperature	GLORYS12	The temperature of surrounding sea water directly affects ice melt and formation rates.
`t2m`	2-meter temperature	ERA5	Air temperature can provide additional context for the thermal conditions affecting the sea ice surface.
`u10/v10`	10-meter wind velocity	ERA5	Influences the motion and deformation of sea ice.
`sshf`	Surface sensible heat flux	ERA5	Surface sensible heat flux is an indicator of the heat exchange between the atmosphere and the sea surface, affecting ice melt and formation.
`x`	x-position of each node	—	Provides the latitudinal spatial context for each data point.
`y`	y-position of each node	—	Provides the longitudinal spatial context for each data point.
`doy`	Day of the year	—	Provides temporal context.
`csize`	Cell size	—	Provides the relative size of the area covered by each cell for additional spatial context.

4.4.5 Model Hyperparameter Configurations and Implementation

This study evaluates three distinct models, listed in Table 2. Our primary focus is the GraphSIFNet-Att model, which incorporates three TransformerConv spatial convolutions in the GCLSTM block and is trained on the irregular mesh described in Section 4.2 for 35 epochs. That is, in Figure 4b, $*\mathcal{G}$ uses the TransformerConv as the GraphConv block with $K = 3$. For comparison, we examine the GraphSIFNet-Att-Reg model which is identical in architecture but trained on the coarsened regular mesh from Section 4.2 for 35 epochs. Additionally, we compare with the GraphSIFNet-GCN model, which employs six GCN convolutions within the GCLSTM module, that is, the GraphConv block is the GCN with $K = 6$. GraphSIFNet-GCN is trained over the irregular mesh for 45 epochs. Each of these models have the same number of parameters (approximately 123,000).

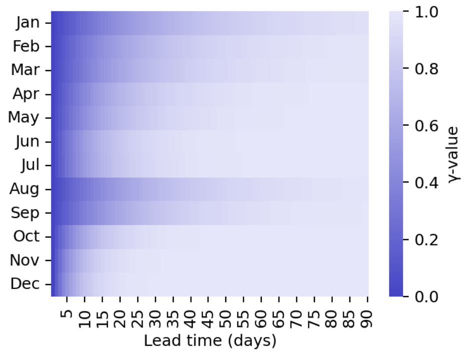


Figure 6: Gamma values for the baseline model (Equation 6) showing the balance between persistence and climatology by the month of the launch date and lead time. Gamma values near 0 favor persistence while values near 1 favor climatology. Less variable ice seasons such as January/February and August/September rely more on persistence for longer lead times.

Each model uses a 10-day input sequence to predict the subsequent 90 days. A hidden dimension size of 32 is used for each of the hidden state and cell state of the encoder and decoder LSTMs, as well as in all graph convolutional layers. The GNN used to encode climatology (GNN_{enc}) is comprised of a single graph convolution layer, and the output GNN (GNN_{out}) is comprised of 3 stacked convolution layers with leaky ReLU activations. The hidden size, number of spatial convolutions and number of GCLSTM/LSTM layers were chosen based on small-scale experiments which aimed to keep the model simple yet effective. The optimizer is the Adam optimizer with an initial learning rate of 0.001 reducing by 10% every 5 epochs. An L2 regularization value of 0.01 is applied to the weights reduce the risk of overfitting, and gradient clipping with a value of 1.0 is applied to mitigate the risk of gradient explosion due to the extended forecast length. Early stopping was used if no improvement in the validation loss was observed for 10 epochs. Since the model produces two outputs, a custom loss function was used that combines a mean square error (MSE) loss from the continuous SIC prediction and binary cross-entropy (BCE) loss from the probabilistic SIP prediction. The BCE loss is scaled by a factor of 0.1 and added to the MSE loss before back-propagation. Since losses are calculated over a mesh with cells of varying physical sizes, the losses are also scaled by the size of each cell. This prevents the model from over-valuing correct predictions in areas of higher spatial resolution. The models are implemented in Pytorch using the pytorch-geometric (Fey & Lenssen, 2019) package and trained on a single Tesla V100 GPU hosted by the Digital Research Alliance of Canada. A summary of models tested and training times is given in Table 2.

5 Results

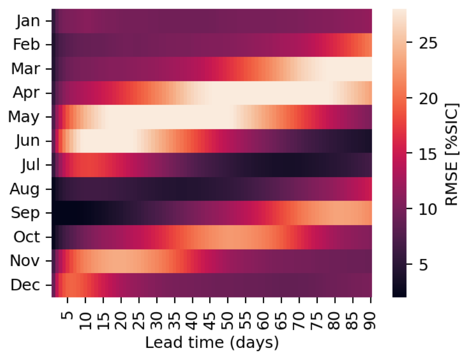
In this section, the GraphSIFNet-Att model is evaluated by comparing its performance with the statistical baseline and contrasting with the two other configurations: GraphSIFNet-Att-Reg and GraphSIFNet-GCN. Using GraphSIFNet-Att, insights from the attention weights, the results of a variable importance experiment, and an evaluation of its ability to predict break-up and freeze-up dates are also presented.

Table 2: Summary of developed model configurations. The models differ in their spatial convolutions and their underlying meshes, with the aim of contrasting the attention-based transformer convolution with the graph convolutional network, as well as demonstrating the model's ability to model over an irregular mesh.

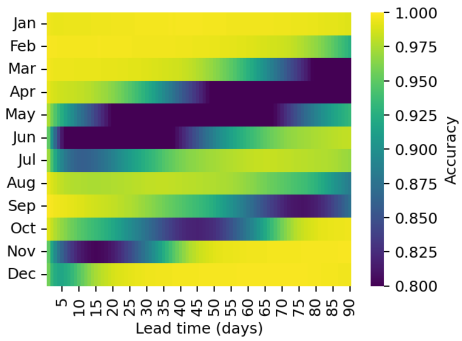
Name	**Convolution (# stacked layers)**	**Mesh**	**Approximate training time**
`GraphSIFNet-Att`	TransformerConv (3)	Irregular (1/12° - 1/3°)	10h (30 epochs)
`GraphSIFNet-Att-Reg`	TransformerConv (3)	Regular (1/3°)	8h (30 epochs)
`GraphSIFNet-GCN`	GCN (6)	Irregular (1/12° - 1/3°)	10h (45 epochs)
`Baseline`	N/A	N/A	N/A

5.1 Baseline Performance

The performance of the baseline statistical model defined by Equation 6 for both the SIC and SIP forecasting task is shown in Figure 7a and Figure 7b, respectively. These heatmaps are generated by calculating the spatial average of the root mean squared error (RMSE) over the domain using only the test years (2016-2019). The errors are grouped by the month of the launch dates and lead times. For instance, the value in the top right corner of the error heatmaps (January, 90-day lead time) indicates the mean RMSE for all 90-day forecasts launched in January, that is, forecasts for dates spanning April 1st to May 1st. The two clearly visible bands of higher RMSE values correspond to the break-up and freeze-up seasons, the former normally spanning from the beginning of May to mid-July and the latter normally spanning from the beginning of November to the end of December. These seasons are the most difficult to forecast as the timing and pattern of the break-up and freeze-up vary between years. Conversely, August to beginning of October are largely ice-free, thus the errors are near zero. In the winter months, that is, mid-December to the beginning of April, ice is present throughout the Hudson Bay system though some open water can sporadically be found around shorelines, for example due to offshore winds, thus SIC RMSE values during the winter months are small but not zero.



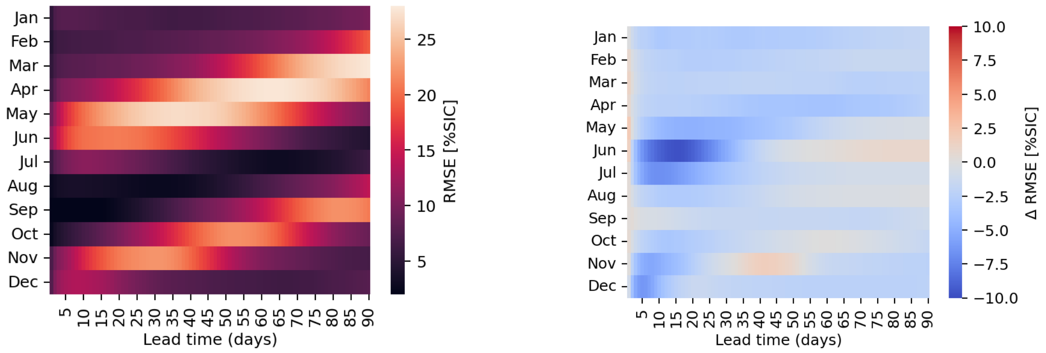
(a) Baseline SIC performance



(b) Baseline SIP performance

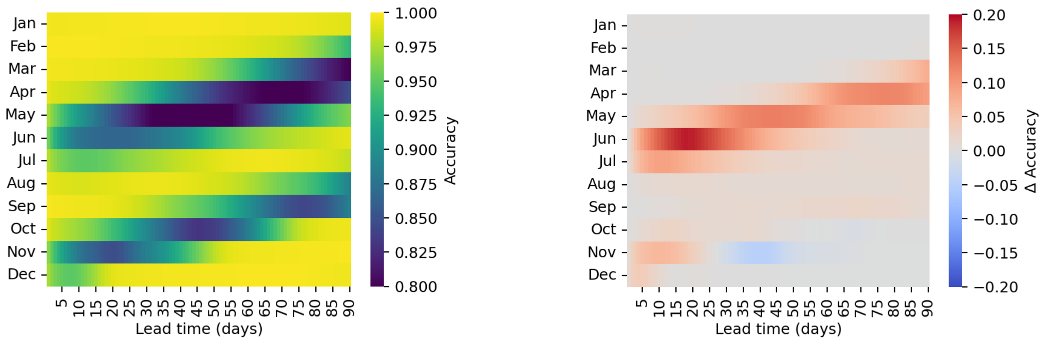
Figure 7: Performance of the baseline statistical model on SIC (a) and SIP (b) over the test years aggregated by the month of the launch date and lead time.

5.2 GraphSIFNet-Att Performance



(a) GraphSIFNet-Att (b) Δ(GraphSIFNet-Att, Baseline)

Figure 8: RMSE heatmaps for the SIC forecasting task by month and lead time for the GraphSIFNet-Att model (a), and the RMSE differences between GraphSIFNet-Att and the baseline (b) where negative values (blue) indicate a reduction in model error relative to the baseline.



(a) GraphSIFNet-Att (b) Δ(GraphSIFNet-Att, Baseline)

Figure 9: Accuracy heatmaps for the SIP forecasting task by month and lead time for the GraphSIFNet-Att model (a), and the difference between GraphSIFNet-Att and the baseline (b) where positive values (red) in the difference plots indicate an increase in model accuracy relative to the baseline.

The performance of GraphSIFNet-Att model and the difference in performance relative to the baseline model is shown in Figure 8 and Figure 9 for SIC and SIP forecasts, respectively. Since persistence and climatology are usually used as baselines seperately, the difference in performance relative to both are shown in Section Appendix A. Models are evaluated against GLORYS12 SIC and SIP on the full-resolution 1/12° GLORYS12 grid.

For the majority of the months and lead times, the GraphSIFNet-Att model exhibits improvements in SIC forecasts over the baseline, with minor exceptions. The model

Task	Model	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Avg.
SIC	GraphSIFNet-GCN	0.29	0.19	0.02	0.11	0.67	-0.33	0.18	0.22	0.00	0.11	-0.37	0.40	0.12
	GraphSIFNet-Att-Reg	0.43	0.12	-0.16	0.30	-0.19	-0.71	0.22	0.03	-0.08	1.19	-0.79	0.51	0.07
SIP	GraphSIFNet-GCN	0.00	-0.01	0.05	0.14	-0.54	0.08	-0.06	-0.07	0.37	0.13	-0.38	-0.03	-0.03
	GraphSIFNet-Att-Reg	0.00	-0.03	0.08	-0.22	-0.10	0.25	-0.10	0.02	0.18	-0.21	0.83	-0.15	0.05

Figure 10: Difference in monthly SIC RMSE [%SIC] and SIP [%]accuracy between GraphSIFNet-Att-Reg and GraphSIFNet-GCN relative to GraphSIFNet-Att averaged over all 90 forecast days. Negative RMSE differences and positive accuracy differences indicate better performance on the part of GraphSIFNet-Att relative to the other models.

exhibits the largest improvements over the baseline in its short- to medium-term forecasts of the break-up season (lead times 5 to 45 launched in May to July). These show up to a 10% improvement over the baseline. At longer timesteps, the improvements over the baseline during the break-up period (launched in March and April) are less pronounced, hovering around 2-3%. However at these long lead times even small improvements demonstrate forecast skill and can provide value to users of the system. During the winter months when the region is almost entirely frozen, the model still exhibits a 2-3% improvement over the baseline at all lead times. This suggests that the model may be able to better capture the effects of off-shore winds mechanically creating open water regions along the shoreline. During freeze-up, the model only shows skill over the baseline at short lead times from 0 to 25 days. Longer forecasts beyond 25 days perform on par with the baseline or only marginally better. Forecasts launched in November with a 30 to 55 day lead time perform worse than the baseline, indicating difficulty in capturing the final stages of ice formation.

The SIP accuracy heatmaps in Figure 9 show similar patterns, with increases in accuracy of up to 20% from the GraphSIFNet-Att model over the baseline during the break-up process, and more modest increases during the freeze-up process. Notably, however, GraphSIFNet-Att outperforms quite significantly ($> 10\%$) even at long lead times. This indicates that although the model may struggle to forecast the precise SIC at these lead times, it still has skill in forecasting the point at which the ice will completely melt or break up.

5.3 Comparison Between Model Configurations

Differences in both SIC RMSE and SIP accuracy between the GraphSIFNet model configurations, averaged for all timesteps for each month, are shown in Figure 10. GraphSIFNet-GCN and GraphSIFNet-Att-Reg demonstrate comparable performance relative to GraphSIFNet-Att, with differences being largely insignificant when aggregated across the entire region. To better understand the differences in their capabilities, spatial monthly SIC RMSE maps for the 15-, 30-, and 60-day lead times for forecasts launched in May and November are presented in Figure 11. These correspond to parts of the break-up and freeze-up periods, respectively. Panels a) and c) show the impact of the convolution operator, while panels b) and d) show the impact of the mesh resolution.

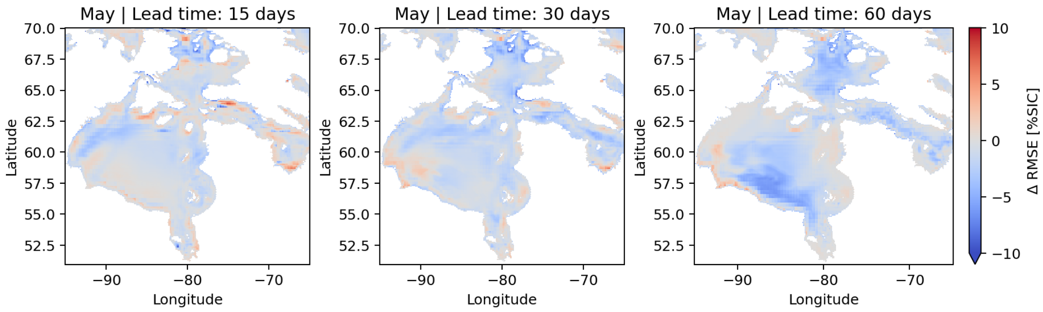
Early (15-day) forecasts in the Northwest region of Hudson Bay, launched in May, are best captured by GraphSIFNet-Att-Reg. This region is characterized by a latent heat polynya, suggesting that the coarser uniform resolution mesh may aid the model in forecasting the formation and behavior of the polynya. Using a finer resolution mesh in this region might cause the model to overemphasize local variations in sea ice concentration and thickness, potentially obscuring the broader spatial patterns crucial for accurate polynya forecasting. Both GraphSIFNet-GCN and GraphSIFNet-Att-Reg outperform GraphSIFNet-

Att in the 15- and 30-day forecasts launched in November in Hudson Strait. The freeze-up in Hudson Strait is characterized by rapid changes in ice formation and movement influenced by strong ocean currents. These conditions create a highly dynamic and challenging environment for sea ice prediction. Since all three models exhibit similar performance, the additional interpretability granted by the attention weights in GraphSIFNet-Att motivates the use of GraphSIFNet-Att over the others.

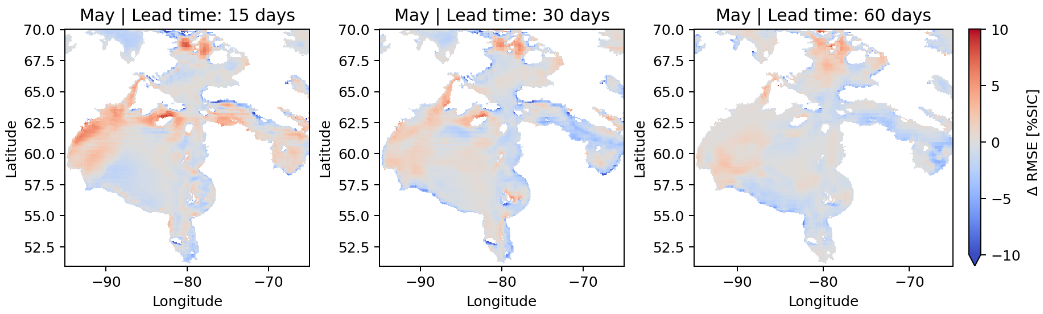
5.4 Attention Maps

The use of transformer convolution in the model enhances its interpretability. By examining the attention weights in the encoder's first layer of graph convolutions, insights can be gleaned into how the model encodes the input data. According to Equation 3 and Equation 4, each node is assigned attention weights for its neighboring nodes based on learned weight matrices in each transformer layer. The softmax function ensures that the sum of all attention weights for a given node's neighbors equals 1. Consequently, the node is updated using a weighted average of its neighbors' features, which are projected into a latent space. Due to the large number of edges, visualizing these weights on a simple map is challenging. A simpler approach for visualization involves calculating the primary direction from which each node is updated. This can be done by summing the attention weights as vectors (α values in Equation 3 with the direction of their respective edges) for each node. These can be represented by arrows, the magnitude of which is proportional to the difference in weights. For example, a node with evenly distributed attention weights among eight neighbors would be represented as a single dot, whereas a node with a dominant westward neighbor would have a large arrow pointing westward. These arrows can be interpreted as indicating the direction of information flow through the graph as the model processes the input maps.

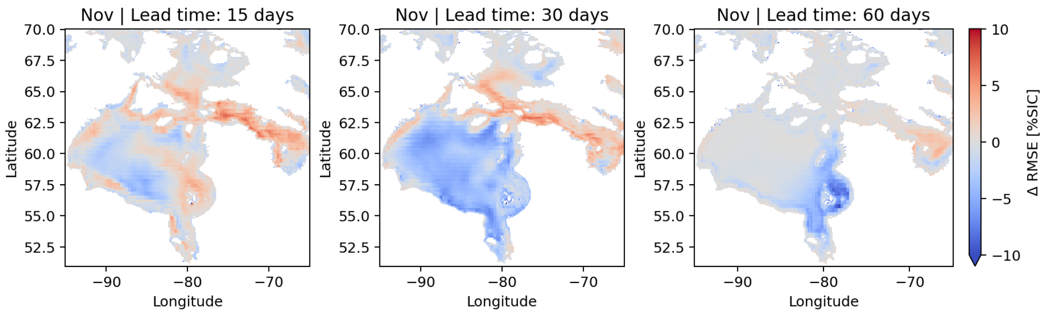
Figure 12 provides examples of attention weights of the input gate for a single input image during both freeze-up (Figure 13a) and melting (Figure 13b) seasons. Although the attention mechanism is applied to the hidden and input tensors at each of the LSTM gates, it is most informative to visualize the weights that are applied to the inputs since the inputs are physically interpretable. Note that attention weights at land interfaces are omitted for visual clarity, as they are numerous and the lack of nodes on land means the dominant direction is always away from the shore. In the freeze-up condition, the model directs information flow generally from the southeast to the northwest. This suggests that the model learns the importance of understanding the sea ice and atmospheric conditions of nodes to the northwest, aligning with the direction of freezing. It it is logical that a node that contains water should know the condition of its 3-hop neighbor to the northwest, as if this neighbor is frozen, it is likely that this node will freeze in the near future. Conversely, during the melting season, arrows point towards open water, indicating that nodes with icy conditions but with water-containing neighbors should consider these neighbors important as they indicate the node is likely to melt soon. Notably, the magnitude of the arrows is larger at the ice edge and nearly zero in the consolidated ice region, which could reflect the localized nature of the break-up process compared to the more gradual freeze-up. That is, the break-up process is largely confined to the ice edge, while freeze-up gradually occurs across the region, as seen by changes in sea ice concentration. Nodes in the open water region during the melting season are less likely to change and, therefore, do not require attention to specific neighbors. Note that although the weights are visualized on the sea ice concentration inputs, they apply indiscriminately to all input features. Interestingly, the model appears to prioritize sea ice thickness over concentration, evidenced by the larger attention weights where thickness drops more dramatically than concentration in Figure 13b. This is logical given the importance of thickness in determining the rate at which the ice will melt or break up. Additionally, the attention weights in the open-water region during the freeze-up condition appear to be influenced by surface sensible heat flux, suggesting its significance as an input feature.



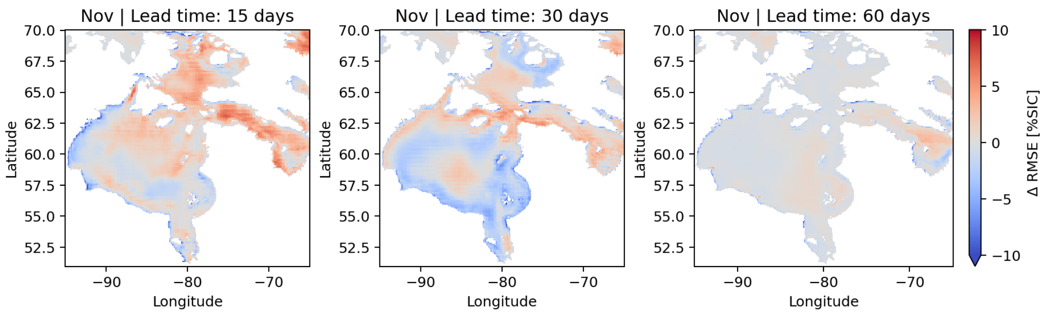
(a) May — Δ(GraphSIFNet-Att, GraphSIFNet-GCN)



(b) May — Δ(GraphSIFNet-Att, GraphSIFNet-Att-Reg)

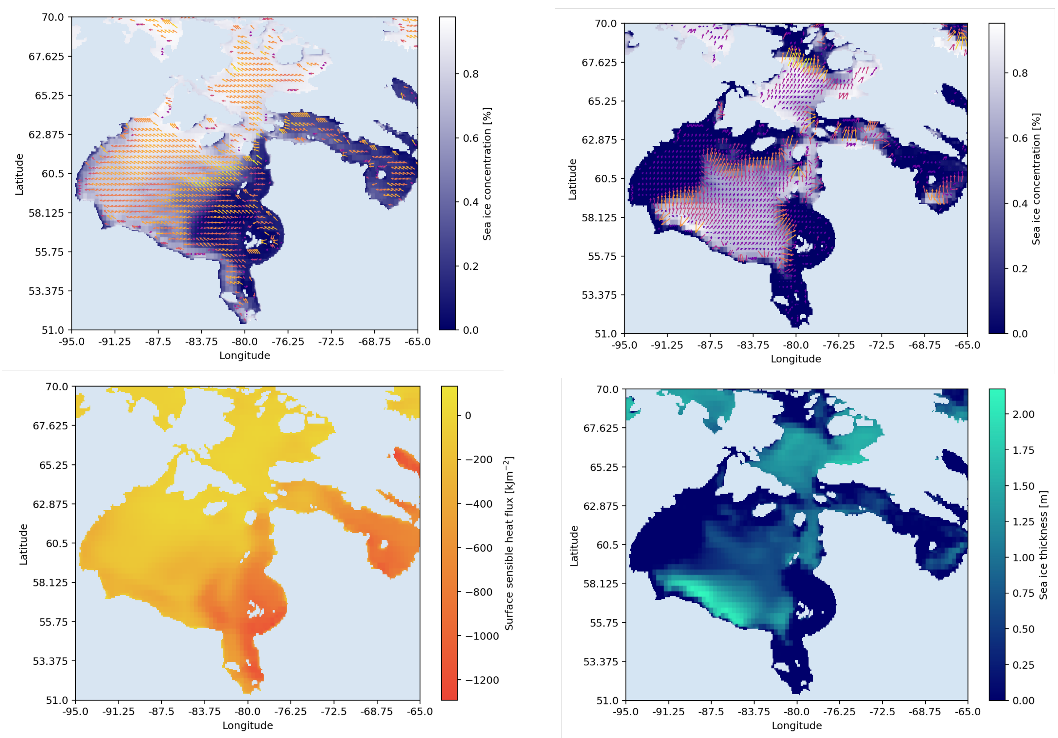


(c) November — Δ(GraphSIFNet-Att, GraphSIFNet-GCN)



(d) November — Δ(GraphSIFNet-Att, GraphSIFNet-Att-Reg)

Figure 11: Comparison of SIC RMSE for GraphSIFNet-Att, GraphSIFNet-Att-Reg, and GraphSIFNet-GCN models at 15-, 30-, and 60-day forecast lead times, initiated in May and November. The figure shows the difference in RMSE between GraphSIFNet-Att and both GraphSIFNet-Att-Reg and GraphSIFNet-GCN. Negative values indicate a reduction in error in the GraphSIFNet-Att relative to the other indicated model.



(a) December 1, 2014

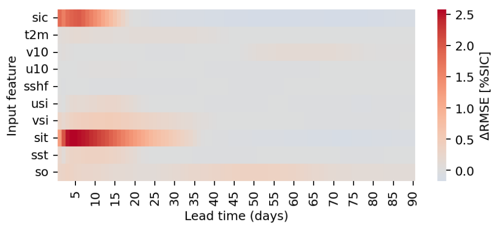
(b) July 1, 2014

Figure 12: Visualization of attention weights of the input gate applied to the input tensors during the freeze-up (a) and melting (b) seasons overlaid on the sea ice concentration input. Arrows indicate the primary direction and magnitude of information flow based on the learned attention weights. Attention weights at the land interfaces are omitted for clarity. The attention weights appear to be largely influenced by sea ice concentration, but other input variables also influence the weights, for example surface sensible heat flux in (a), and sea ice thickness in (b).

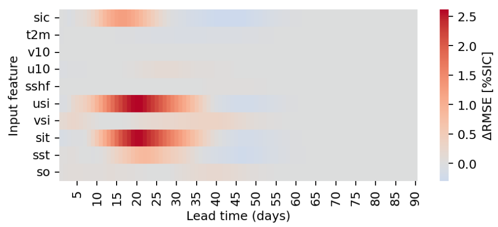
5.5 Variable Importance

The models are trained with a number of input variables (refer to Table 1), which we anticipated the model might utilize to make its predictions. However, these variables may not contribute equally to the resulting predictions. In this section, we explore the significance of each feature by feature ablation through omission (Fong & Vedaldi, 2017). Specifically, we produce forecasts using the trained GraphSIFNet-Att model by substituting each input variable, one at a time, with white Gaussian noise generated using the mean and standard deviation of the real inputs. Figure 13 shows the resulting difference in RMSE when re-generating predictions on the test years using the June and December models when each variable is replaced with noise.

During the break-up process (June model), the model largely relies on the input sea ice concentration and sea ice thickness to make its predictions, but also considers the ice velocities, sea surface temperature and sea salinity to a smaller degree. Other variables do not significantly affect the resulting predictions. The model appears to use sea ice concentrations to inform near-term forecasts (days 0 through 20), and sea ice thickness to inform its medium-term forecasts (days 0 through 35). This makes intuitive sense as thickness is an indicator of the ice cover's longevity making it relevant at longer fore-



(a) June model feature importance



(b) November model feature importance

Figure 13: Feature ablation with noise injection for the June and November GraphSIFNet-Att models. Positive values indicate an increase in RMSE when each respective variable is replaced with noise.

cast steps, while sea ice concentration is more important for immediate predictions since lower ice concentrations are normally associated with ice parcels that are already breaking up. Note that at forecast steps larger than 35 days, forecasts launched in June are largely forecasting periods where Hudson Bay is fully open water, thus none of the input features contribute to the resulting forecasts.

Similarly, during the freeze-up process, the model relies on sea ice thickness, sea ice concentration, sea ice velocity and sea surface temperature to make its predictions. Again, the model largely considers sea ice concentration to make its shorter term forecasts (days 10 through 25), while considering ice velocity and thickness for medium-term forecasts (days 15 through 40). Ice velocity may be indicating areas where ice migrates, thereby creating space for new ice formation. The difference between the vertical and horizontal ice velocity component (*usi* and *vsi*) may indicate that they offer redundant information, thus it is sufficient for the model to consider one of the components. Again, November forecasts at larger than 40 days are largely forecasting periods of full ice cover, therefore omitting input features does not impact the scores. It is also worth noting that in both cases, the model does not appear to consider the variables originating from ERA5. This could point to a mismatch between ERA5 and GLORYS12, which would be unsurprising as GLORYS12 uses ERA-Interim as model forcing at the surface. Since the target variables are derived from GLORYS12, the models therefore prioritize input features originating from GLORYS12.

To illustrate the impact of these variables on the resulting predictions, a sample GraphSIFNet-Att forecast is shown in Figure 14, along with the same forecast when replacing sea ice concentration and sea ice thickness (SIT) with noise as described above. Replacing either SIC or SIT with noise does not significantly affect the 1-day forecast, suggesting the model uses persistence as a heuristic at very short lead times. Beyond the 10-day forecast, predictions are affected by the noise injections, with the model forecasting a quicker melt when sea ice thickness is replaced with noise, consistent with the theory that thickness is used as a signal of ice longevity. When SIC is replaced with noise, the model persists more of the ice in the 20-day forecast, suggesting that SIC is also important for ice integrity.

Although this technique offers some insight into feature importance, it should be noted that since the models are not re-trained, the observed changes in performance due to feature omission may not perfectly reflect the true importance of each feature. This is because the model has been optimized to make predictions based on the full set of features, therefore the omission of any one feature changes the input space in a way that the model was not specifically trained to handle. Moreover, the interdependencies be-

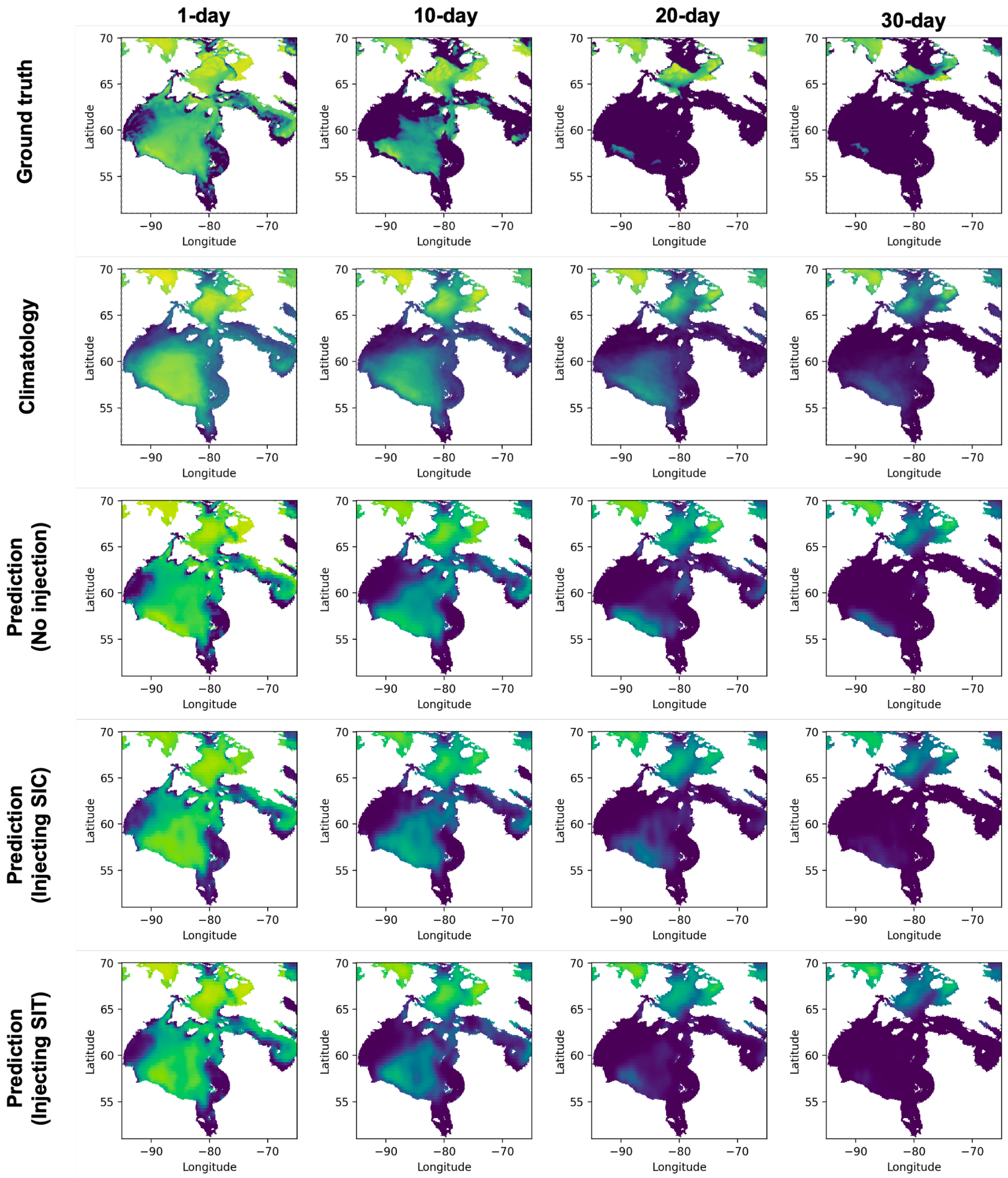


Figure 14: Sample 1-, 10-, 20-, and 30-day forecasts from GraphSIFNet-Att launched on June 15, 2014. The climatology for each forecast day is shown for reference, and the results of running inference after replacing sea ice concentration (SIC) and sea ice thickness (SIT) with noise is shown.

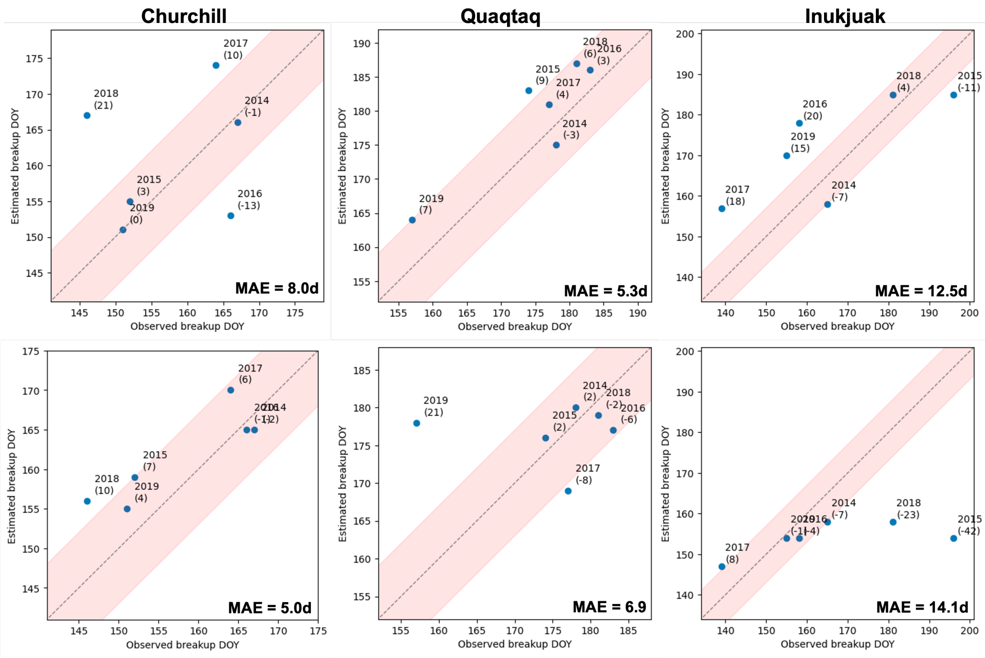
tween features are not accounted for in this single-feature ablation approach. Variables in the dataset may interact in complex, non-linear ways that are not captured by examining each variable in isolation. Despite these limitations, this feature ablation technique provides useful insights into the relative importance of the different input features used in these particular trained models (Fong & Vedaldi, 2017). Since we know which features the models are using, we know which input variables should be more closely monitored.

5.6 Estimating Break-up and Freeze-up Dates

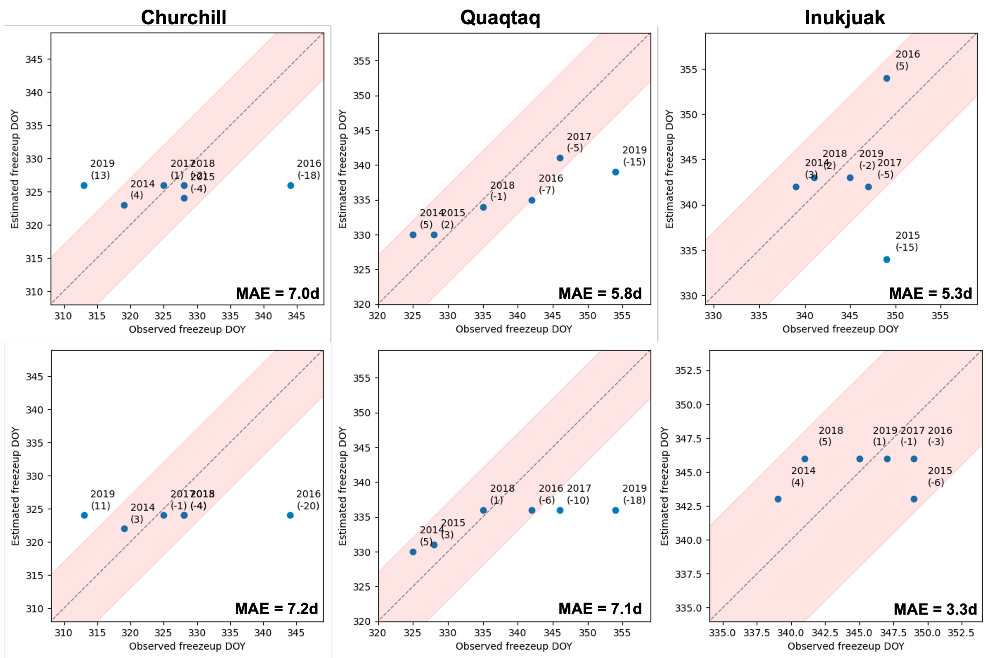
A potential use-case for sea ice prediction in Hudson Bay is the estimation of break-up and freeze-up dates in key locations, as these dates have significant implications for maritime navigation and local communities. We evaluate the GraphSIFNet-Att model's performance in estimating the freeze-up date at three key ports in Hudson Bay: the ports of Churchill, Quaqtaq and Inukjuak. The port of Churchill is mostly used to export grain while the ports of Quaqtaq and Inukjuak are regularly used for community resupply. These three ports were chosen as their locations are representative of the varying sea ice conditions found in the Hudson Bay region. In this study, the validation and test year (2014 to 2019) serve as the period for assessing the predicted break-up and freeze-up dates. These dates are determined using the same criteria as the previous study, which follows the definition given by the Canadian Ice Service (CIS). That is, the freeze-up date at a given site is defined as the initial day when open water persists for 15 consecutive days, with open water being defined as a SIC of less than 15%. Conversely, the break-up date is defined as first day at which SIC exceeds 15% for 15 consecutive days. The 30-day and 60-day predicted break-up and freeze-up dates are determined using the same criteria, but with open water and ice conditions being defined as a sea ice presence probability less than and greater than 50%, respectively. For each port, we take the mean pixel value of a 3×3 window around the nearest pixel to the port locations.

Figure 15 displays the predicted dates of freeze-up/breakup at the three ports with 30 and 60 days of lead time compared to the actual observed dates for the validation and test years along with the mean absolute error. Predicted dates falling within 7 days of the observed dates are considered correct, visualized by the pink shaded area. This definition of a correct forecast is in line with a previous study (Asadi et al., 2022). The 30-day forecasted break-up and freeze-up dates for Churchill are noticeably inferior to the other two ports, likely due to challenges presented by the latent heat polynya in the Northwest of Hudson Bay. The uniform forecasts of freeze-up dates at Churchill can be interpreted as an admission that the model does not have skill here and resorts to forecasting the same mean day every year. Break-up predictions at Inukjuak also pose a challenge for the model, likely due to freshwater inflows from the James Bay area affecting the timing and rate of melt. Quaqtaq sees the most successful predictions, with all freeze-up dates falling within 7 days of the observed date.

In Figure 16, the break-up and freeze-up accuracies are shown spatially for the entire region. These accuracies are calculated as the proportion of years with predicted break-up or freeze-up dates within 7 days of the observed date. These are compared to predictions made using the climate normals. The model performs equally or better than climatology for most of the region in predicting break-up dates at both 30-days and 60-days of lead time. However, there is a strong pattern in the freeze-up maps where the model performs worse than climatology in the western half of the bay but still outperforms climatology in the eastern half and in Hudson Strait. This is unsurprising as Hudson Bay begins its freeze-up process in the northwest corner of the bay, thus the onset of that initial freezing is difficult to predict. Once the bay has begun freezing over, the model can better predict the timing of the rest of the bay. Although we might expect the model to use atmospheric conditions such as temperature to predict the onset of freeze-up, the model only has access those atmospheric conditions 30 or 60 days prior to the



(a) Break-up date estimates



(b) Freeze-up date estimates

Figure 15: Break-up and freeze-up dates predicted by GraphSIFNet-Att at Churchill, Inukjuak, and Quaqtaq ports for lead times of 30 and 60 days for the years 2014 to 2019 compared to the observed dates from GLORYS12. The pink shaded area represents a 7-day buffer around a perfect forecast. Samples which fall within this buffer are deemed correct forecasts. The annotated numbers in parentheses are the error for each year.

forecast date. There may not be a strong enough signal in those initial conditions to allow the model to accurately predict how quickly the temperatures will drop.

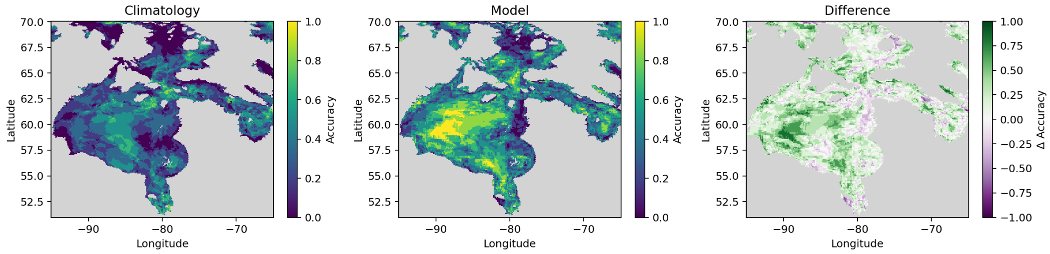
6 Conclusion

The study presented in this paper demonstrated the effectiveness of using a GNN-based spatiotemporal forecasting model for predicting daily sea ice concentration and sea ice presence in Hudson Bay over a 90-day time horizon. To demonstrate the ability of GNNs to handle spatially irregular meshes, models were trained on both a uniform regular mesh and an irregular mesh with higher resolution near shorelines. The proposed model uses an attention-based transformer spatial convolution to learn spatial features from the input, which was shown to have similar performance compared to the more basic graph convolutional network. The attention-based convolution however has the additional benefit of increasing the model's interpretability, motivating its use.

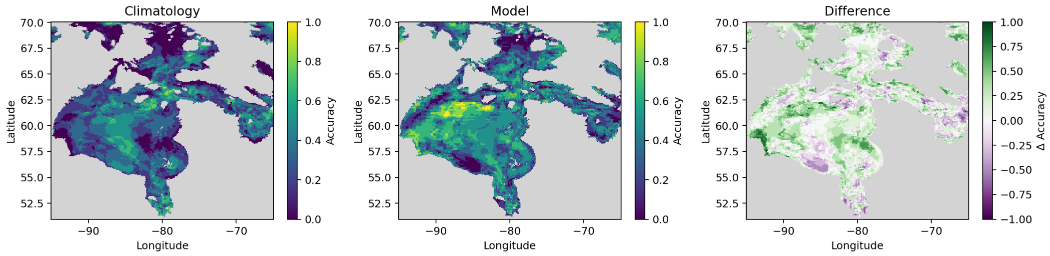
Results from this study highlighted the model's skill in predicting sea ice dynamics, with particular success noted in short- to medium-term forecasts during the break-up season when compared to a linear combination of persistence and climatology as a statistical baseline. The model performed as well or better on the irregular mesh as on the regular mesh, with the exception of some difficulty capturing the initial freeze-up in the Northwest region of Hudson Bay as well as the polynya formation at longer lead times. This suggests that improvements could be made in refining the model's sensitivity to complex spatial features associated with irregular meshes, particularly in areas where ice dynamics are highly variable. This could involve more sophisticated positional and spatial encoding, perhaps by projecting the positional, cell size, distance and angle encodings into higher dimensional latent space. The model showed similar overall performance between the model using the transformer convolution and the GCN within the GCLSTM module, with some differences in performance in certain regions such as Hudson Strait. This suggested potential overfitting in the model using the spatial transformer convolution.

The attention mechanism within the transformer convolution offered interpretability by highlighting the primary direction and magnitude of information flow in the encoder, which aligned with known physical processes such as the direction of freezing and melting. A feature ablation experiment indicated the trained model's reliance on sea ice concentration, thickness and velocities to inform its predictions. Other variables did not contribute significantlty to the resulting forecasts, which could explain the model's poor performance in forecasting the Kivalliq latent heat polynya. A evaluation of the model's ability to predict freeze-up and break-up dates was conducted, revealing the model's limited ability to forecast the onset of freeze-up in Hudson Bay, as well as the onset of break-up in the Northwest region which is influenced by the polynya. The model however still showed skill over the statistical baseline in these tasks.

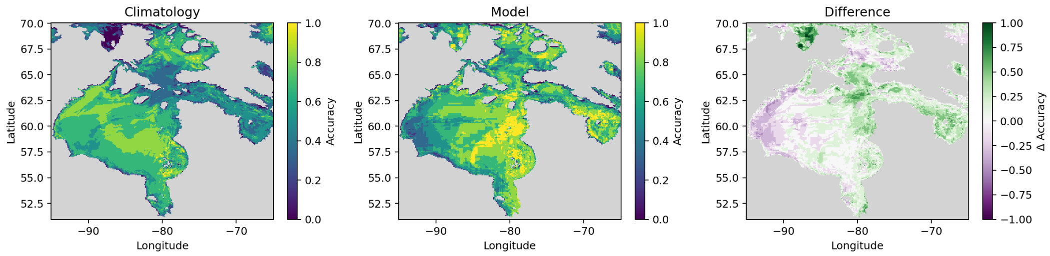
Several potential avenues for future work exist. In a GNN, each node is processed as a separate sample by the network. This has two major implications. First, one input image $X \in \mathbb{R}^{W \times H \times C}$ does not necessarily need to be processed fully at once, instead, nodes could also be sampled in batches sequentially until the full sample has been processed. This would be helpful in the case where the region is large and modelling it in its entirety would be infeasible due to memory constraints. Second, since each node has its own hidden and cell states, cells can be combined by averaging the states or split by duplicating the states. This means that the underlying mesh could be dynamic in time, evolving as the underlying data changes (e.g. as the ice conditions evolves). For example, one could define a dynamic mesh which has a higher resolution at the ice edge where the ice conditions are known to be more dynamic. As the ice conditions evolve, so too would the underlying mesh. The advantages are two-fold. First, it allows for a reduction in data volume with minimal information loss, contrary to the static mesh used in



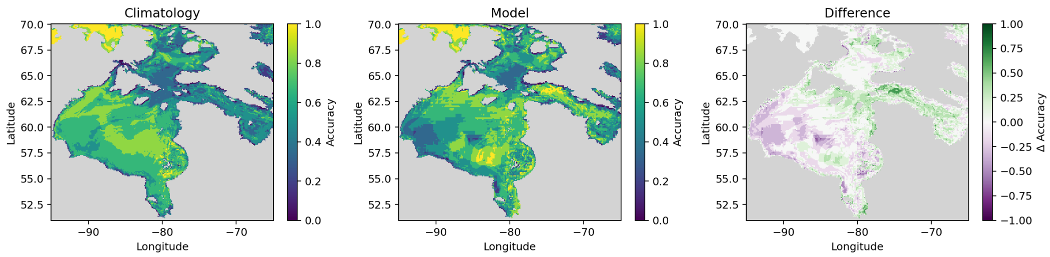
(a) 30-day break-up date estimate map



(b) 60-day break-up date estimate map



(c) 30-day freeze-up date estimate map

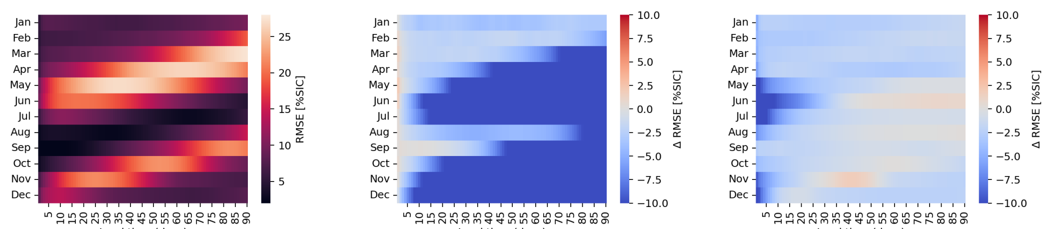


(d) 60-day freeze-up date estimate map

Figure 16: Break-up and freeze-up date estimate maps from the climatological baseline (a), GraphSIFNet-Att model predictions (b), and the difference between the two (c). Positive values in the difference plots indicate an increase in accuracy from the model relative to the baseline, where accuracy is defined as the proportion of predictions falling within 7 days of the observed date for the years 2014 to 2019.

this work which has information loss where the data has high spatial variance. Second, the dynamic mesh could help the model learn more sophisticated dynamics and is more consistent with physical simulation software. This idea was explored in (Pfaff et al., 2020). Another avenue for future work could be a deeper investigation of the adjacency matrix. In this study, edges were placed between any two directly spatially adjacent cells. However, edges could also be placed between distant cells thereby widening the receptive field without adding convolutions. This could be investigated by transforming the adjacency matrix into a learnable matrix optimized during training. Furthermore, node sampling strategies could also be used to reduce training time. Specifically, adaptive sampling techniques could be employed where nodes in dynamic regions, such as the ice edges known for their fluctuating conditions, are sampled with higher frequency compared to the more static areas. Incorporating long-term weather forecasts from third party sources such as the Canadian Global Ice Ocean Prediction System (GIOPS) could also be beneficial, particularly in forecasting freeze-up. Lastly, multi-resolution modelling either through an ensemble of models operating over meshes of different resolution or using multiple meshes of varying resolutions within a single model could be explored. This may help the model better capture both large-scale and small-scale phenomena.

Appendix A Additional RMSE Heatmaps



(a) Model (b) Δ(Model, Persistence) (c) Δ(Model, Climatology)

Figure A1: RMSE heatmaps for the SIC forecasting task by month and lead time for the GraphSIFNet-Att model (a), and the RMSE differences between GraphSIFNet-Att and persistence (b) and climatology (c) where negative values (blue) indicate a reduction in model error relative to the baseline.

Data Availability Statement

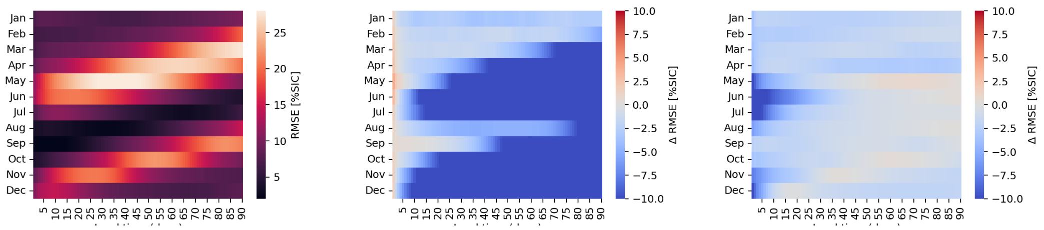
ERA5 atmospheric reanalysis data (Hersbach et al., 2020) are available at `https://doi.org/10.24381/cds.adbb2d47`, and GLORYS12 ocean reanalysis data (Jean-Michel et al., 2021) are available at `https://doi.org/10.48670/moi-00021`.

Acknowledgments

The authors would like to acknowledge funding from the National Research Council Canada through the AI4Logistics and Ocean programs and computing resources provided by the Digital Research Alliance of Canada.

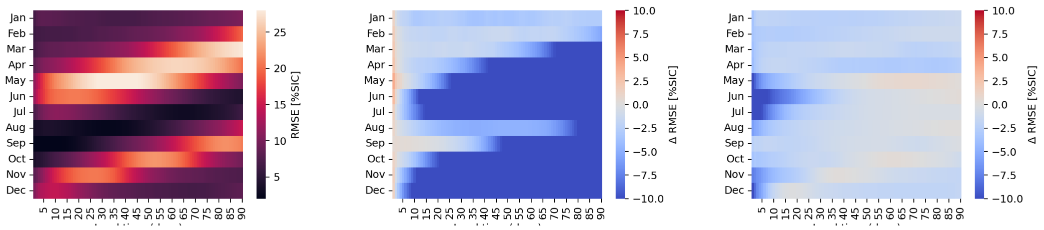
References

Andersson, T. R., Hosking, J. S., Pérez-Ortiz, M., Paige, B., Elliott, A., Russell, C., ... Shuckburgh, E. (2021, August). Seasonal Arctic sea ice fore-



(a) Model (b) Δ(Model, Persistence) (c) Δ(Model, Climatology)

Figure A2: RMSE heatmaps for the SIC forecasting task by month and lead time for the GraphSIFNet-Att-Reg model (a), and the RMSE differences between GraphSIFNet-Att-Reg and persistence (b) and climatology (c) where negative values (blue) indicate a reduction in model error relative to the baseline.



(a) Model (b) Δ(Model, Persistence) (c) Δ(Model, Climatology)

Figure A3: RMSE heatmaps for the SIC forecasting task by month and lead time for the GraphSIFNet-Att-Reg model (a), and the RMSE differences between GraphSIFNet-GCN and persistence (b) and climatology (c) where negative values (blue) indicate a reduction in model error relative to the baseline.

casting with probabilistic deep learning. *Nature Communications*, *12*(1), 5124. Retrieved 2022-09-28, from `https://www.nature.com/articles/s41467-021-25257-4` (Number: 1 Publisher: Nature Publishing Group) doi: 10.1038/s41467-021-25257-4

Anselin, L. (1988). A Typology of Spatial Econometric Models. In L. Anselin (Ed.), *Spatial Econometrics: Methods and Models* (pp. 32–40). Dordrecht: Springer Netherlands. Retrieved 2023-11-16, from `https://doi.org/10.1007/978-94-015-7799-1_4` doi: 10.1007/978-94-015-7799-1_4

Asadi, N., Lamontagne, P., King, M., Richard, M., & Scott, K. A. (2022, September). Probabilistic spatiotemporal seasonal sea ice presence forecasting using sequence-to-sequence learning and ERA5 data in the Hudson Bay region. *The Cryosphere*, *16*(9), 3753–3773. Retrieved 2022-10-16, from `https://tc.copernicus.org/articles/16/3753/2022/` doi: 10.5194/tc-16-3753-2022

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016, July). *Layer Normalization.* arXiv. Retrieved 2023-09-19, from `http://arxiv.org/abs/1607.06450` (arXiv:1607.06450 [cs, stat])

Cavalieri, D. J., & Parkinson, C. L. (2012, August). Arctic sea ice variability and trends, 1979–2010. *The Cryosphere*, *6*(4), 881–889. Retrieved 2022-04-18, from `https://tc.copernicus.org/articles/6/881/2012/` (Publisher: Copernicus GmbH) doi: 10.5194/tc-6-881-2012

Chi, J., & Kim, H.-c. (2017, December). Prediction of Arctic Sea Ice Concentration Using a Fully Data Driven Deep Neural Network. *Remote Sensing*, *9*(12), 1305. Retrieved 2023-07-28, from https://www.mdpi.com/2072-4292/9/12/1305 (Number: 12 Publisher: Multidisciplinary Digital Publishing Institute) doi: 10.3390/rs9121305

Dee, D. P., Uppala, S. M., Simmons, A. J., Berrisford, P., Poli, P., Kobayashi, S., ... Vitart, F. (2011). The ERA-Interim reanalysis: configuration and performance of the data assimilation system. *Quarterly Journal of the Royal Meteorological Society*, *137*(656), 553–597. Retrieved 2023-06-01, from https://onlinelibrary.wiley.com/doi/abs/10.1002/qj.828 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/qj.828) doi: 10.1002/qj.828

Fey, M., & Lenssen, J. E. (2019, April). *Fast Graph Representation Learning with PyTorch Geometric.* arXiv. Retrieved 2022-11-27, from http://arxiv.org/abs/1903.02428 (arXiv:1903.02428 [cs, stat]) doi: 10.48550/arXiv.1903.02428

Fong, R. C., & Vedaldi, A. (2017, October). Interpretable Explanations of Black Boxes by Meaningful Perturbation. In *2017 IEEE International Conference on Computer Vision (ICCV)* (pp. 3449–3457). Venice: IEEE. Retrieved 2023-11-09, from http://ieeexplore.ieee.org/document/8237633/ doi: 10.1109/ICCV.2017.371

Gers, F., Schraudolph, N., & Schmidhuber, J. (2002, January). Learning Precise Timing with LSTM Recurrent Networks. *Journal of Machine Learning Research*, *3*, 115–143. doi: 10.1162/153244303768966139

Hamilton, W., Ying, Z., & Leskovec, J. (2017). Inductive Representation Learning on Large Graphs. In *Advances in Neural Information Processing Systems* (Vol. 30). Curran Associates, Inc. Retrieved 2023-11-09, from https://proceedings.neurips.cc/paper_files/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html

He, K., Zhang, X., Ren, S., & Sun, J. (2016, June). Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 770–778). Las Vegas, NV, USA: IEEE. Retrieved 2023-11-09, from http://ieeexplore.ieee.org/document/7780459/ doi: 10.1109/CVPR.2016.90

He, M., Wei, Z., & Wen, J.-R. (n.d.). Convolutional Neural Networks on Graphs with Chebyshev Approximation, Revisited.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz Sabater, J., ... Thépaut, J. N. (2020, July). The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, *146*(730), 1999–2049. Retrieved 2022-09-28, from https://onlinelibrary.wiley.com/doi/10.1002/qj.3803 doi: 10.1002/qj.3803

Ivanov, V. V., Alexeev, V. A., Repina, I., Koldunov, N. V., & Smirnov, A. (2012, August). Tracing Atlantic Water Signature in the Arctic Sea Ice Cover East of Svalbard. *Advances in Meteorology*, *2012*, e201818. Retrieved 2023-09-14, from https://www.hindawi.com/journals/amete/2012/201818/ (Publisher: Hindawi) doi: 10.1155/2012/201818

Jean-Michel, L., Eric, G., Romain, B.-B., Gilles, G., Angélique, M., Marie, D., ... Pierre-Yves, L. T. (2021). The Copernicus Global 1/12° Oceanic and Sea Ice GLORYS12 Reanalysis. *Frontiers in Earth Science*, *9*. Retrieved 2022-09-28, from https://www.frontiersin.org/articles/10.3389/feart.2021.698876

Joshi, A., Deshmukh, P. K., & Lohokare, J. (2022, June). Comparative analysis of Vanilla LSTM and Peephole LSTM for stock market price prediction. In *2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS)* (pp. 1–6). doi: 10.1109/IC3SIS54991.2022.9885528

Kalman, R. E. (1960, March). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, *82*(1), 35–45. Retrieved 2023-11-16, from https://doi.org/10.1115/1.3662552 doi: 10.1115/1.3662552

Keisler, R. (2022, February). *Forecasting Global Weather with Graph Neural Networks.* arXiv. Retrieved 2022-07-05, from http://arxiv.org/abs/2202.07575 (Number: arXiv:2202.07575 arXiv:2202.07575 [physics])

Kim, J., Kim, K., Cho, J., Kang, Y. Q., Yoon, H.-J., & Lee, Y.-W. (2019, January). Satellite-Based Prediction of Arctic Sea Ice Concentration Using a Deep Neural Network with Multi-Model Ensemble. *Remote Sensing*, *11*(1), 19. Retrieved 2023-07-28, from https://www.mdpi.com/2072-4292/11/1/19 (Number: 1 Publisher: Multidisciplinary Digital Publishing Institute) doi: 10.3390/rs11010019

Kipf, T. N., & Welling, M. (2017, February). *Semi-Supervised Classification with Graph Convolutional Networks.* arXiv. Retrieved 2022-11-27, from http://arxiv.org/abs/1609.02907 (arXiv:1609.02907 [cs, stat]) doi: 10.48550/arXiv.1609.02907

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Pritzel, A., … Battaglia, P. (2022, December). *GraphCast: Learning skillful medium-range global weather forecasting.* arXiv. Retrieved 2023-01-30, from http://arxiv.org/abs/2212.12794 (arXiv:2212.12794 [physics])

Liu, Y., Bogaardt, L., Attema, J., & Hazeleger, W. (2021, March). Extended Range Arctic Sea Ice Forecast with Convolutional Long-Short Term Memory Networks. *Monthly Weather Review*, *149*. doi: 10.1175/MWR-D-20-0113.1

Madec, G. (n.d.). NEMO ocean engine.

Meier, W. N., Hovelsrud, G. K., van Oort, B. E., Key, J. R., Kovacs, K. M., Michel, C., … Reist, J. D. (2014). Arctic sea ice in transformation: A review of recent observed changes and impacts on biology and human activity. *Reviews of Geophysics*, *52*(3), 185–217. Retrieved 2022-04-18, from https://onlinelibrary.wiley.com/doi/abs/10.1002/2013RG000431 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2013RG000431) doi: 10.1002/2013RG000431

Moon, T. A., Overeem, I., Druckenmiller, M., Holland, M., Huntington, H., Kling, G., … Wong, G. (2019). The Expanding Footprint of Rapid Arctic Change. *Earth's Future*, *7*(3), 212–218. Retrieved 2022-04-18, from https://onlinelibrary.wiley.com/doi/abs/10.1029/2018EF001088 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2018EF001088) doi: 10.1029/2018EF001088

Mudryk, L., Dawson, J., Howell, S., Derksen, C., Zagon, T., & Brady, M. (2021, August). Impact of 1, 2 and 4 °C of global warming on ship navigation in the Canadian Arctic. *Nature Climate Change*, *11*, 1–7. doi: 10.1038/s41558-021-01087-6

Oprea, S., Martinez-Gonzalez, P., Garcia-Garcia, A., Castro-Vargas, J. A., Orts-Escolano, S., Garcia-Rodriguez, J., & Argyros, A. (2022, June). A Review on Deep Learning Techniques for Video Prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*(6), 2806–2826. (Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence) doi: 10.1109/TPAMI.2020.3045007

Petty, A. A., Schröder, D., Stroeve, J. C., Markus, T., Miller, J., Kurtz, N. T., … Flocco, D. (2017). Skillful spring forecasts of September Arctic sea ice extent using passive microwave sea ice observations. *Earth's Future*, *5*(2), 254–263. Retrieved 2023-10-02, from https://onlinelibrary.wiley.com/doi/abs/10.1002/2016EF000495 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2016EF000495) doi: 10.1002/2016EF000495

Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., & Battaglia, P. (2020, October).

Learning Mesh-Based Simulation with Graph Networks.. Retrieved 2023-11-09, from https://openreview.net/forum?id=roNqYL0_XP

Pfeifer, P. E., & Deutsch, S. J. (1980). A STARIMA Model-Building Procedure with Application to Description and Regional Forecasting. *Transactions of the Institute of British Geographers*, *5*(3), 330. Retrieved 2023-11-21, from https://www.jstor.org/stable/621846?origin=crossref doi: 10.2307/621846

Post, E., Bhatt, U. S., Bitz, C. M., Brodie, J. F., Fulton, T. L., Hebblewhite, M., ... Walker, D. A. (2013, August). Ecological Consequences of Sea-Ice Decline. *Science*, *341*(6145), 519–524. Retrieved 2022-04-18, from https://www.science.org/doi/full/10.1126/science.1235225 (Publisher: American Association for the Advancement of Science) doi: 10.1126/science.1235225

Rubanova, Y., Sanchez-Gonzalez, A., Pfaff, T., & Battaglia, P. (2022, June). Constraint-based graph network simulator. In *Proceedings of the 39th International Conference on Machine Learning* (pp. 18844–18870). PMLR. Retrieved 2023-11-09, from https://proceedings.mlr.press/v162/rubanova22a.html (ISSN: 2640-3498)

Sanchez-Gonzalez, A., Godwin, J., Pfaff, T., Ying, R., Leskovec, J., & Battaglia, P. (2020, November). Learning to Simulate Complex Physics with Graph Networks. In *Proceedings of the 37th International Conference on Machine Learning* (pp. 8459–8468). PMLR. Retrieved 2023-11-09, from https://proceedings.mlr.press/v119/sanchez-gonzalez20a.html (ISSN: 2640-3498)

Seo, Y., Defferrard, M., Vandergheynst, P., & Bresson, X. (2018). Structured Sequence Modeling with Graph Convolutional Recurrent Networks. In L. Cheng, A. C. S. Leung, & S. Ozawa (Eds.), *Neural Information Processing* (pp. 362–373). Cham: Springer International Publishing. doi: 10.1007/978-3-030-04167-0_33

Serreze, M. C., & Meier, W. N. (2019). The Arctic's sea ice cover: trends, variability, predictability, and comparisons to the Antarctic. *Annals of the New York Academy of Sciences*, *1436*(1), 36–53. Retrieved 2023-11-28, from https://onlinelibrary.wiley.com/doi/abs/10.1111/nyas.13856 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/nyas.13856) doi: 10.1111/nyas.13856

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-k., & WOO, W.-c. (2015). Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. In *Advances in Neural Information Processing Systems* (Vol. 28). Curran Associates, Inc. Retrieved 2023-11-09, from https://proceedings.neurips.cc/paper/2015/hash/07563a3fe3bbe7e3ba84431ad9d055af-Abstract.html

Shi, Y., Huang, Z., Feng, S., Zhong, H., Wang, W., & Sun, Y. (2021, August). Masked Label Prediction: Unified Message Passing Model for Semi-Supervised Classification. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence* (pp. 1548–1554). Montreal, Canada: International Joint Conferences on Artificial Intelligence Organization. Retrieved 2023-05-01, from https://www.ijcai.org/proceedings/2021/214 doi: 10.24963/ijcai.2021/214

Sims, C. A. (1980). Macroeconomics and Reality. *Econometrica*, *48*(1), 1–48. Retrieved 2023-11-16, from https://www.jstor.org/stable/1912017 (Publisher: [Wiley, Econometric Society]) doi: 10.2307/1912017

Srivastava, N., Mansimov, E., & Salakhudinov, R. (2015, June). Unsupervised Learning of Video Representations using LSTMs. In *Proceedings of the 32nd International Conference on Machine Learning* (pp. 843–852). PMLR. Retrieved 2023-11-09, from https://proceedings.mlr.press/v37/srivastava15.html (ISSN: 1938-7228)

Stammerjohn, S. E., Drinkwater, M. R., Smith, R. C., & Liu, X. (2003). Ice-atmosphere interactions during sea-ice advance and retreat in the western Antarctic Peninsula region. *Journal of Geophysical Research: Oceans*, *108*(C10). Retrieved 2023-09-14, from https://onlinelibrary.wiley.com/doi/abs/10.1029/2002JC001543 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2002JC001543) doi: 10.1029/2002JC001543

Stroeve, J., & Notz, D. (2018, September). Changing state of Arctic sea ice across all seasons. *Environmental Research Letters*, *13*(10), 103001. Retrieved 2023-11-28, from https://dx.doi.org/10.1088/1748-9326/aade56 (Publisher: IOP Publishing) doi: 10.1088/1748-9326/aade56

Stroeve, J. C., Serreze, M. C., Holland, M. M., Kay, J. E., Malanik, J., & Barrett, A. P. (2012, February). The Arctic's rapidly shrinking sea ice cover: a research synthesis. *Climatic Change*, *110*(3), 1005–1027. Retrieved 2022-04-18, from https://doi.org/10.1007/s10584-011-0101-1 doi: 10.1007/s10584-011-0101-1

Vihma, T. (2014, September). Effects of Arctic Sea Ice Decline on Weather and Climate: A Review. *Surveys in Geophysics*, *35*(5), 1175–1214. Retrieved 2023-02-09, from https://doi.org/10.1007/s10712-014-9284-0 doi: 10.1007/s10712-014-9284-0

Vitart, F., & Robertson, A. W. (2018, March). The sub-seasonal to seasonal prediction project (S2S) and the prediction of extreme events. *npj Climate and Atmospheric Science*, *1*(1), 1–7. Retrieved 2023-07-28, from https://www.nature.com/articles/s41612-018-0013-0 (Number: 1 Publisher: Nature Publishing Group) doi: 10.1038/s41612-018-0013-0

Wang, C., Graham, R. M., Wang, K., Gerland, S., & Granskog, M. A. (2019, June). Comparison of ERA5 and ERA-Interim near-surface air temperature, snowfall and precipitation over Arctic sea ice: effects on sea ice thermodynamics and evolution. *The Cryosphere*, *13*(6), 1661–1679. Retrieved 2023-09-14, from https://tc.copernicus.org/articles/13/1661/2019/ (Publisher: Copernicus GmbH) doi: 10.5194/tc-13-1661-2019

Wu, Z., Jain, P., Wright, M. A., Mirhoseini, A., Gonzalez, J. E., & Stoica, I. (2022, January). *Representing Long-Range Context for Graph Neural Networks with Global Attention.* arXiv. Retrieved 2023-09-28, from http://arxiv.org/abs/2201.08821 (arXiv:2201.08821 [cs])

Yao, T., Tang, C. L., & Peterson, I. K. (2000). Modeling the seasonal variation of sea ice in the Labrador Sea with a coupled multicategory ice model and the Princeton ocean model. *Journal of Geophysical Research: Oceans*, *105*(C1), 1153–1165. Retrieved 2023-09-14, from https://onlinelibrary.wiley.com/doi/abs/10.1029/1999JC900264 (_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/1999JC900264) doi: 10.1029/1999JC900264

Yu, F., & Koltun, V. (2016, April). *Multi-Scale Context Aggregation by Dilated Convolutions.* arXiv. Retrieved 2023-11-16, from http://arxiv.org/abs/1511.07122 (arXiv:1511.07122 [cs]) doi: 10.48550/arXiv.1511.07122

Zhang, J., & Rothrock, D. (2003, May). Modeling Global Sea Ice with a Thickness and Enthalpy Distribution Model in Generalized Curvilinear Coordinates. *Monthly Weather Review - MON WEATHER REV*, *131*. doi: 10.1175/1520-0493(2003)131⟨0845:MGSIWA⟩2.0.CO;2

Zhang, M., Cui, Z., Neumann, M., & Chen, Y. (2018, April). An End-to-End Deep Learning Architecture for Graph Classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). Retrieved 2023-07-28, from https://ojs.aaai.org/index.php/AAAI/article/view/11782 doi: 10.1609/aaai.v32i1.11782