Evaluating Long Short-Term Memory Networks for Modeling Land Cover Change

by

Alysha van Duynhoven

BSc., Simon Fraser University, 2018

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
Department of Geography
Faculty of Environment

SIMON FRASER UNIVERSITY
Summer 2019

Approval

Name:	**Alysha van Duynhoven**
Degree:	**Master of Science**
Title:	**Evaluating Long Short-Term Memory Networks for Modeling Land Cover Change**
Examining Committee:	**Chair:** Eugene McCann Professor

Suzana Dragicevic
Senior Supervisor
Professor

Martin Ester
Supervisor
Professor
School of Computing Science

Brian Klinkenberg
External Examiner
Professor
Department of Geography
University of British Columbia

Date Defended/Approved: August 14, 2019

Abstract

Land cover change (LCC) can be viewed as dynamic complex systems which require relevant relationships to be encoded when represented within various modeling approaches. Recurrent Neural Networks (RNNs), specifically the Long Short-Term Memory (LSTM) variant, belong to a category of Deep Learning (DL) approaches best suited for sequential and timeseries data analysis, thus suitable for representing LCC. The primary objective of this study is to examine the capacity and effectiveness of LSTM networks for forecasting LCC given varying geospatial input datasets with feature impurities. Using synthetic and MODIS land cover datasets for British Columbia, Canada, results demonstrate the sensitivity of LSTM models to varying geospatial input dataset characteristics. Geospatial datasets with finer temporal resolutions and increased timesteps yielded favourable results while coarser temporal resolutions and fewer timesteps were affiliated with less successful outcomes. This thesis research contributes to the advancement of automated, data-driven DL methodologies for forecasting LCC.

Keywords: Recurrent Neural Networks; Sensitivity Analysis; Long Short-Term Memory; Land Cover Change Modeling; Geographic Information Science; Deep Learning

To my resilient, selfless, loving parents who provide me endless support and encouragement.

Acknowledgements

I would like to thank the Natural Sciences and Engineering Research Council (NSERC) of Canada for fully supporting this study through the Discovery Grant awarded to Dr. Suzana Dragicevic.

I am grateful for the invaluable advice, direction, guidance, support, and patience provided by my senior supervisor, Dr. Suzana Dragicevic. I appreciate her aid in my navigation of graduate studies and in keeping my work focussed. I would also like to thank Dr. Martin Ester, who has provided advice and guided the direction of my research.

I would also like to thank my colleagues in the Spatial Analysis and Modeling Lab for their support and encouragement throughout my studies. Finally, thank-you to my parents and brothers who have seen me through all my successes and failures.

Table of Contents

List of Tables

List of Figures

Chapter 1.

Introduction

1.1. Introduction

Land Cover Change (LCC) dynamics can be viewed as complex systems processes with spatial and temporal dependencies. Land Cover (LC) refers to Earth's surface cover characteristics, delineated as features including forests, water, built environments, and barren lands (Brown et al. 2014). Emerging patterns of LCC at global scales are driven by human and environmental interactions occurring at local levels (Friedl et al. 2002). LCC studies are important to many disciplines such as geography, urban planning, agriculture, forestry, and resource management (Lyu, Lu, and Mou 2016; Mayfield et al. 2017). For example, anthropogenic disturbances including deforestation directly and indirectly lead to elevated CO_2 levels and disturb local weather patterns, making LCC significant to consider as global temperatures rise (Findell et al. 2017; Turner, Lambin, and Reenberg 2007; Mayfield et al. 2017).

LCC has been previously assessed at local, regional, and global extents (Hansen and Loveland 2012). Assessments focused on biodiversity and habitat fragmentation have concerned LCC processes occurring within localized study areas (Lambin, Geist, and Lepers 2004). LCC has also been linked to changes in precipitation, air temperature, and ecology at regional scales (Patil et al. 2017; Findell et al. 2017). Global scale studies have also been conducted to assess the cumulative implications of land change processes such as urban growth and deforestation (Seto, Guneralp, and Hutyra 2012). Addressing LCC from a top-down perspective, data-driven modeling tactics enable the extraction and detections of patterns that have resulted from local interactions (Fu et al. 2019). Top-down approaches are primarily focussed on overall patterns that result from processes, utilizing facets of satellite and aggregated data sources such as Census data to obtain rates of change over time (Verburg et al. 2008; Ren et al. 2019). Given the complexity of LCC, automated statistical learning approaches have been increasingly considered for classification and predictive tasks in this domain (Otukei and Blaschke 2010; Ienco et al. 2017).

Throughout the past decade, data-driven modeling methods have become increasingly considered with the unprecedented availability of data, the Big Data paradigm shift, and the growing capacity of computational resources (Samardžić-Petrović et al. 2017; Li et al. 2016). Machine Learning (ML) refers to a collection of algorithms that automatically improve at various tasks when provided more data. ML methods aim to automate pattern recognition and minimize manual inputs required of a researcher or developer. These algorithms are developed to learn patterns subsisting in input datasets, thus it is important to acknowledge these techniques are deterministic approaches for modeling non-linear spatial processes. This means that after a model is developed, providing the same input will result in the same output each time. While the creation of the model is a stochastic procedure, the models are trained to fit a single set of parameters that minimize an error function with respect to a training set (Bishop 2006).

In the presence of high dimensionality and volume of modern datasets, ML algorithms typically perform with great efficacy. Previous ML approaches for land change forecasting and classification procedures have included Decision Trees (DTs), Random Forests (RFs), Neural Networks (NNs), and Support Vector Machines (SVMs) (Otukei and Blaschke 2010; Patil et al. 2017; Boulila et al. 2011; B. Huang et al. 2009). However, the overall infrequency of LCC and scarcity of apt datasets have proven challenging in this domain (Karpatne et al. 2016). Likewise, for ML algorithms to work effectively, preprocessing steps including feature extraction are important to consider. The researcher or practitioner utilizes domain knowledge to perform preprocessing steps before providing input to the model. These preliminary procedures are time-consuming and increase the expense of ML workflows (Bengio, Courville, and Vincent 2013). Neural Networks (NNs) are particularly adept in capturing complicated relationships obscured in input datasets as model depth (the number of layers) and breadth (the number of neurons per layer) increases (Sauter, Weitzenkamp, and Schneider 2010; G. Huang et al. 2017). Characterizing a subfield of ML called Deep Learning (DL), NNs featuring two or more layers are referred to as Deep NNs. DL techniques facilitate automated learning of feature representations and have demonstrated ability to capture intricate, hierarchical relationships from a dataset. Therefore, these methods depart from modeling approaches that necessitate expert knowledge (Rußwurm and Körner 2018). A Recurrent Neural Network (RNN) is a variation of the traditional NN approach that utilizes internal memory

for each node, proving useful for capturing dynamic dependencies over time from sequential inputs (Hochreiter and Schmidhuber 1997).

1.2. Theoretical Background and Research Problem

RNNs are a type of DL model best suited for sequential datasets. That is, RNNs have the capacity to recognize and extrapolate patterns occurring in the temporal dimension of a given dataset. These networks are composed of neurons, weights, and biases, with internal parameters updated using gradient-based learning procedures (Bishop 2006; Lecun, Bengio, and Hinton 2015). However, the traditional RNN modeling approach is hindered by the “Vanishing Gradient” problem, which impacts the capacity of these networks to learn patterns from long input sequences (Hochreiter 1998). To solve this problem, an advanced RNN architecture called Long Short-Term Memory (LSTM) was later introduced (Hochreiter and Schmidhuber 1997). These improved renditions of traditional RNN models have enabled learning of long-term dependencies from sequential data while reducing the effects of the Vanishing Gradient problem (Hochreiter and Schmidhuber 1997). LSTM architectures vary from traditional RNNs via the critical addition of “gating” functions that permit or block the propagation of information through a cell. Hidden states and internal memory are updated with respect to previous elements in a timeseries, thus allowing information from early in a sequence to be connected to elements occurring later in the given sequence.

To capture prevalent temporal dependencies in geospatial datasets, Recurrent Neural Networks (RNNs) have been employed for prediction and classification tasks involving geospatial data (Zhu et al. 2017). Sauter et al. (2010) utilized snow cover derived from MODIS satellite data with meteorological data to develop a snow cover depth forecasting system using an early RNN approach (Sauter, Weitzenkamp, and Schneider 2010). More recent research endeavors have utilized LSTM to recognize relationships occurring over time within geospatial datasets (Lyu, Lu, and Mou 2016; Chi and Kim 2017; Ienco et al. 2017; Zhang et al. 2017; Liu et al. 2018; Kong et al. 2018). Proposing an LSTM model called REFEREE, Lyu et al. (2016) demonstrate the abilities of pixel-based RNN for binary and multi-class LCC detection in a transfer learning scenario involving various study areas featuring urban, water, soil, and agricultural land (Lyu, Lu, and Mou 2016). LSTMs have also been previously used to forecast sea ice concentration, with input sequences extracted along the temporal dimension for each pixel (Chi and Kim 2017).

RNNs have also proven effective in LC classification applications. An LSTM model was provided 3- and 23-step timeseries to perform pixel-based classification, showing improved performance over traditional RF and SVM methods (Ienco et al. 2017). Others have explored the effectiveness of cell-based forecasting where grids of separate LSTM cells are employed to capture temporal dependencies (Zhang et al. 2017). Utilizing a grid of LSTM nodes, each was fit to sea surface temperature values at distinct locations (Zhang et al. 2017). LSTM has also been successfully applied in the prediction of influenza propagation in the state of Georgia in the United States, considering geographic datasets pertaining to weather, pollution, and influenza spread (Liu et al. 2018). Additionally, further optimized LSTM networks have been effectively utilized for forecasting bus passenger flow at respective bus stations by capturing temporal dependencies (Han et al. 2019). Finally, LSTM has also been used to detect natural disturbances such as fires or floods, or human disturbances such as deforestation or urban growth, considering satellite image timeseries datasets obtained from a moderate-resolution imaging spectroradiometer (MODIS) data product (Kong et al. 2018).

Though it is acknowledged that a multitude of RNN architectures and variants exist, in this thesis, LSTM has been selected as the primary architecture to be evaluated for its capacity to model LCC (Jia et al. 2017). A prior study compared the performance of LSTM and its variants, demonstrating insignificant improvements obtained over the traditional LSTM architecture in three applications, including “acoustic modeling, handwriting recognition, and polyphonic music modeling” (Greff et al. 2017). It was also demonstrated that traditional LSTM networks have the capacity to reliably obtain improved performance versus its simplified variant, the Gated Recurrent Unit (GRU) (Chung et al. 2015), in large-scale “neural machine translation” tasks (Britz et al. 2017). Bidirectional RNNs are another type of RNN that permit information from the past and subsequent timesteps to influence the network’s learned representation, doing so by replicating the input sequence in reverse to provide as an additional input (Schuster and Paliwal 1997). Such networks are conducive to tasks such as speech recognition (Arisoy et al. 2015) and classification of human activities (Edel and Köppe 2016), where providing additional context from the previous and forthcoming inputs in a sequence is essential. However, for phenomena such as LCC where processes propagate changes over space and time that are linked to spatial dependences, it is deemed inappropriate to consider the temporal dimension in reverse. Additionally, hybrid approaches integrating convolutional neural

networks (CNNs) and LSTMs have been employed to forecast traffic conditions (Liu et al. 2017; Yu et al. 2017). To represent spatial relationships, Liu et al. (2017) generate one-dimensional vectors containing measures obtained from surrounding locations for each input record. Likewise, Yu et al. (2017) include information for each input entry pertaining to “longitude, latitude, timestamp, direction, and vehicle speed.” Methods reviewed here are deterministic techniques used in a variety of spatial and non-spatial modeling applications.

While yielding successful outcomes in classification and predictive tasks involving geospatial datasets, a recurring aspect predominant in the aforementioned studies is the lack of consensus pertaining to which characteristics of geospatial datasets are conducive to the success of these sequential DL methods. It is undetermined whether the method is apt for scenarios in which geospatial data is limited in the temporal dimension. Likewise, the spatial extent, LC class cardinality, and number of years required for sequential DL methods such as LSTM to be of benefit remains unexplored.

To address these limitations existing in previous data-driven LCC modeling methodologies, and the wider domain of data-driven modeling methods applied to geospatial data, this thesis aims to address the following research questions:

1) What geospatial data characteristics are favourable for use with sequential DL modeling approaches?

2) How effective are data-driven modeling approaches such as LSTM for real-world geospatial applications where changes are typically slow or rare, and the number of timesteps or temporal resolution is limited?

1.3. Research Objectives

To address the research questions posed, the objectives of this study are to:

1) Develop a sensitivity analysis approach to assess the implications of varying geospatial input data properties and LSTM modeling scenarios;

2) Apply the proposed sensitivity analysis approach to hypothetical and real-world LC datasets.

A progression of modeling approaches has been developed and applied to synthetic and real-world datasets to determine the suitability of LSTM for LCC modeling. Likewise, this thesis research aims to guide future endeavours considering sequential DL methods for modeling LCC. This includes guiding the selection of geospatial datasets for use with LSTM modeling approaches in future works.

1.4. Datasets and Study Areas

Hypothetical or synthetic datasets are often used when working with simulation models (Ligmann-Zielinska and Sun 2010; Bone and Dragićević 2009). To demonstrate concepts presented, synthetic and real-world geospatial datasets were used to represent study areas featuring increasing/large numbers of unique LC classes and varying rates of change. Each synthetic dataset established for the research work presented in chapter two features an identical temporal resolution and number of timesteps to control model input parameters. Each dataset features a 25-meter spatial resolution, with 45 timesteps representing annual LC in the localized study area. Subsequent evaluations of modeling approaches presented in chapter three utilize real-world datasets for the province of British Columbia, Canada, as a study area. The synthetic datasets created for this work feature four, eight, and 16 LC classes, respectively. The synthetic datasets have been generated as *GeoTiffs* using Esri's ArcGIS Pro (v2.4.0) (Esri 2017).

Additionally, the global "MODIS Terra+Aqua Combined Land Cover product" has been obtained, featuring global annual land cover data from 2001 to 2017 (Friedl, M., Sulla-Menashe 2015). This dataset features 500-meter spatial resolution, with LC data available at one-year temporal resolution for the period of 2001 to 2017. This LC data is also made available with a classification confidence layer featuring percentages for each cell at each timestep. Both the categorical LC dataset and continuous assessment layers have been extracted to the digital provincial boundary file for British Columbia, Canada ("Boundary Files, 2016 Census" 2016). Despite this data product featuring coarse spatial resolution, the temporal resolution and the number of timesteps were the strongest motivators in the selection of the MODIS LC data product. With finer temporal resolutions and increased timesteps prioritized, acquisition of suitable LC datasets at smaller scales proved challenging to obtain. Likewise, the extensible spatial extent and large number of geospatial layers included in this LC dataset make it favourable for the implementation of RNN methods, thus proving suitable for conducting this thesis research.

1.5. Thesis Overview

This thesis is comprised of four chapters. Following the introduction, chapter two addresses the Sensitivity Analysis approach for localized cell-based modeling approaches. The objective is to assess the implications of varying (1) temporal resolutions, (2) number of timesteps, (3) number of land cover classes, and (4) rates of change existing in the input dataset. DL methods are initially evaluated using the three small-scale synthetic datasets created. This enables localized analyses and reduced computation time to perform the repeated evaluations (Hermes and Poulsen 2012). Aiming to evaluate assumptions pertaining to sequential DL model's response to varying input data characteristics, such datasets support the exploration of particular scenarios (Burlacu, O'Donoghue, and Sologon 2014). The proposed analysis intends to reveal the repercussions on method performance across all implemented modeling scenarios by altering training dataset characteristics. LSTM modeling scenarios are developed to optimize one facet of a stacked LSTM model at a time to observe common trends in method response, despite the repeated application of standard optimization tactics to a baseline model configuration.

Chapter three incorporates the SA approach and findings from chapter two in a real-world application utilized to showcase whether the method is propitious for LCC applications. Geospatial input datasets are selected, and data-preprocessing procedures are applied to match the favourable characteristics determined in the preceding chapter. LSTM models are trained on four datasets encapsulating all or subsets of timesteps available from the real-world land cover data obtained for the province of British Columbia. The optimal number of land cover classes is selected based on previous findings from chapter two. Using annual land cover data available from 2001 to 2017, the sensitivity of the method is assessed using varying temporal resolutions to determine if method response trends subsist as spatial extent and heterogeneity increases. An additional modeling scenario is also designed to assess the implications of static LC classes surrounded by real-world changes. Finally, the LC classification confidence layer is integrated to examine potential improvements that may be afforded by indicating potentially erroneous values.

Chapter four concludes this thesis, synthesizing results of the modeling scenarios developed in chapters two and three. Limitations of these methods are then presented. Finally, this chapter includes various trajectories for future work.

1.6. References

Arisoy, E, Sethy, A, Ramabhadran, B, and Chen, S. 2015. “Bidirectional Recurrent Neural Network Language Models for Automatic Speech Recognition.” In ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 2015-Augus:5421–25. IEEE.

Bengio, Y, Courville, A, and Vincent, P. 2013. “Representation Learning: A Review and New Perspectives.” IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (8): 1798–1828.

Bishop, CM. 2006. Pattern Recognition and Machine Learning. Springer-Verlag New York.

Bone, C and Dragićević, S. 2009. “Defining Transition Rules with Reinforcement Learning for Modeling Land Cover Change.” Simulation 85 (5): 291–305.

Boulila, W, Farah, IR, Saheb Ettabaa, K, Solaiman, B, and Ghézala, H Ben. 2011. “A Data Mining Based Approach to Predict Spatiotemporal Changes in Satellite Images.” International Journal of Applied Earth Observation and Geoinformation 13 (3): 386–95.

“Boundary Files, 2016 Census.” 2016. Statistics Canada Catalogue no. 92-160-X.

Britz, D, Goldie, A, Luong, M-T, and Le, Q. 2017. “Massive Exploration of Neural Machine Translation Architectures.” In Conference on Empirical Methods in Natural Language Processing, 1442–51. Copenhagen, Denmark: Association for Computational Linguistics.

Brown, DG, Polsky, C, Bolstad, P, Brody, SD, Hulse, D, Kroh, R, Loveland, TR, and Thomson, A. 2014. “Land Use and Land Cover Change.” In Climate Change Impacts in the United States: The Third National Climate Assessment, 13:318–32. J. M. Melillo, Terese (T.C.) Richmond, and G. W. Yohe, Eds., U.S. Global Change Research Program.

Burlacu, I, O’Donoghue, C, and Sologon, DM. 2014. “Hypothetical Models.” In Handbook of Microsimulation Modelling (Contributions to Economic Analysis, Volume 293), 23–46. Emerald Group Publishing Limited.

Chi, J and Kim, H. 2017. “Prediction of Arctic Sea Ice Concentration Using a Fully Data Driven Deep Neural Network.” Remote Sensing 9 (12): 1305.

Chung, J, Gulcehre, C, Cho, K, and Bengio, Y. 2015. “Gated Feedback Recurrent Neural Networks.” In ICML’15 Proceedings of the 32nd International Conference on International Conference on Machine Learning, 2067–75. Lille, France.

Edel, M and Köppe, E. 2016. “Binarized-BLSTM-RNN Based Human Activity Recognition.” In 2016 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2016, 1–7. IEEE.

Esri. 2017. “ArcGIS Pro: 2.0.” Redlands, CA: Environmental Systems Research Institute.

Findell, KL, Berg, A, Gentine, P, Krasting, JP, Lintner, BR, Malyshev, S, Santanello, JA, and Shevliakova, E. 2017. “The Impact of Anthropogenic Land Use and Land Cover Change on Regional Climate Extremes.” Nature Communications 8 (1): 989.

Friedl, M., Sulla-Menashe, D. 2015. “MCD12Q1 MODIS/Terra+Aqua Land Cover Type Yearly L3 Global 500m SIN Grid V006 [Data Set].” NASA EOSDIS Land Processes DAAC. 2015.

Friedl, MA, McIver, DK, Hodges, JCF, Zhang, XY, Muchoney, D, Strahler, AH, Woodcock, CE, et al. 2002. “Global Land Cover Mapping from MODIS: Algorithms and Early Results.” Remote Sensing of Environment 83 (1–2): 287–302.

Fu, B, Wu, L, Harris, P, Lü, Y, Comber, A, and Ren, Y. 2019. “Spatially Explicit Simulation of Land Use/Land Cover Changes: Current Coverage and Future Prospects.” Earth-Science Reviews 190: 398–415.

Greff, K, Srivastava, RK, Koutnik, J, Steunebrink, BR, and Schmidhuber, J. 2017. “LSTM: A Search Space Odyssey.” IEEE Transactions on Neural Networks and Learning Systems 28 (10): 2222–32.

Han, Y, Wang, C, Ren, Y, Wang, S, Zheng, H, and Chen, G. 2019. “Short-Term Prediction of Bus Passenger Flow Based on a Hybrid Optimized LSTM Network.” ISPRS International Journal of Geo-Information 8 (9): 366.

Hansen, MC and Loveland, TR. 2012. “A Review of Large Area Monitoring of Land Cover Change Using Landsat Data.” Remote Sensing of Environment 122 (July): 66–74.

Hermes, K and Poulsen, M. 2012. “A Review of Current Methods to Generate Synthetic Spatial Microdata Using Reweighting and Future Directions.” Computers, Environment and Urban Systems 36 (4): 281–90.

Hochreiter, S. 1998. “The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions.” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 06 (02): 107–16.

Hochreiter, S and Schmidhuber, J. 1997. “Long Short-Term Memory.” Neural Computation 9 (8): 1735–80.

Huang, B, Xie, C, Tay, R, and Wu, B. 2009. “Land-Use-Change Modeling Using Unbalanced Support-Vector Machines.” Environment and Planning B: Planning and Design 36 (3): 398–416.

Huang, G, Liu, Z, Maaten, L Van Der, and Weinberger, KQ. 2017. “Densely Connected Convolutional Networks.” In Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-Janua:2261–69. IEEE.

Ienco, Di, Gaetano, R, Dupaquier, C, and Maurel, P. 2017. “Land Cover Classification via Multitemporal Spatial Data by Deep Recurrent Neural Networks.” IEEE Geoscience and Remote Sensing Letters 14 (10): 1685–89.

Jia, X, Khandelwal, A, Nayak, G, Gerber, J, Carlson, K, West, P, and Kumar, V. 2017. “Incremental Dual-Memory LSTM in Land Cover Prediction.” In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD ’17, 867–76. New York, New York, USA: ACM Press.

Karpatne, A, Jiang, Z, Vatsavai, RR, Shekhar, S, and Kumar, V. 2016. “Monitoring Land-Cover Changes: A Machine-Learning Perspective.” IEEE Geoscience and Remote Sensing Magazine 4 (2): 8–21.

Kong, Y-LL, Huang, Q, Wang, C, Chen, JJ, Chen, JJ, and He, D. 2018. “Long Short-Term Memory Neural Networks for Online Disturbance Detection in Satellite Image Time Series.” Remote Sensing 10 (3): 452.

Lambin, EF, Geist, HJ, and Lepers, E. 2004. “Dynamics of Land-Use and Land-Cover Change in Tropical Regions.” Annual Review of Environment and Resources 28 (1): 205–41.

Lecun, Y, Bengio, Y, and Hinton, G. 2015. “Deep Learning.” Nature 521 (7553): 436–44.

Li, S, Dragicevic, S, Castro, FA, Sester, M, Winter, S, Coltekin, A, Pettit, C, et al. 2016. “Geospatial Big Data Handling Theory and Methods: A Review and Research Challenges.” ISPRS Journal of Photogrammetry and Remote Sensing 115 (May): 119–33.

Ligmann-Zielinska, A and Sun, L. 2010. “Applying Time-Dependent Variance-Based Global Sensitivity Analysis to Represent the Dynamics of an Agent-Based Model of Land Use Change.” International Journal of Geographical Information Science 24 (12): 1829–50.

Liu, L, Han, M, Zhou, Y, and Wang, Y. 2018. “LSTM Recurrent Neural Networks for Influenza Trends Prediction.” In , 259–64. Springer, Cham.

Liu, Y, Zheng, H, Feng, X, and Chen, Z. 2017. “Short-Term Traffic Flow Prediction with Conv-LSTM.” In 2017 9th International Conference on Wireless Communications and Signal Processing, WCSP 2017 - Proceedings, 2017-Janua:1–6. IEEE.

Lyu, H, Lu, H, and Mou, L. 2016. “Learning a Transferable Change Rule from a Recurrent Neural Network for Land Cover Change Detection.” Remote Sensing 8 (6): 506.

Mayfield, H, Smith, C, Gallagher, M, and Hockings, M. 2017. “Use of Freely Available Datasets and Machine Learning Methods in Predicting Deforestation.” Environmental Modelling and Software 87 (January): 17–28.

Otukei, JR and Blaschke, T. 2010. “Land Cover Change Assessment Using Decision Trees, Support Vector Machines and Maximum Likelihood Classification Algorithms.” International Journal of Applied Earth Observation and Geoinformation 12 (SUPPL. 1): S27–31.

Patil, SD, Gu, Y, Dias, FSA, Stieglitz, M, and Turk, G. 2017. “Predicting the Spectral Information of Future Land Cover Using Machine Learning.” International Journal of Remote Sensing 38 (20): 5592–5607.

Ren, Y, Lü, Y, Comber, A, Fu, B, Harris, P, and Wu, L. 2019. “Spatially Explicit Simulation of Land Use/Land Cover Changes: Current Coverage and Future Prospects.” Earth-Science Reviews 190 (March): 398–415.

Rußwurm, M and Körner, M. 2018. “Multi-Temporal Land Cover Classification with Sequential Recurrent Encoders.” ISPRS International Journal of Geo-Information 7 (4): 129.

Samardžić-Petrović, M, Kovačević, M, Bajat, B, and Dragićević, S. 2017. “Machine Learning Techniques for Modelling Short Term Land-Use Change.” ISPRS International Journal of Geo-Information 6 (12): 387.

Sauter, T, Weitzenkamp, B, and Schneider, C. 2010. “Spatio-Temporal Prediction of Snow Cover in the Black Forest Mountain Range Using Remote Sensing and a Recurrent Neural Network.” International Journal of Climatology 30 (15): 2330–41.

Schuster, M and Paliwal, KK. 1997. “Bidirectional Recurrent Neural Networks.” In IEEE TRANSACTIONS ON SIGNAL PROCESSING, 45:1053–10587.

Seto, KC, Guneralp, B, and Hutyra, LR. 2012. “Global Forecasts of Urban Expansion to 2030 and Direct Impacts on Biodiversity and Carbon Pools.” Proceedings of the National Academy of Sciences 109 (40): 16083–88.

Turner, BL, Lambin, EF, and Reenberg, A. 2007. “The Emergence of Land Change Science for Global Environmental Change and Sustainability.” In Proceedings of the National Academy of Sciences. Vol. 104. National Academy of Sciences.

Verburg, PH, Kok, K, Pontius, RG, and Veldkamp, A. 2008. “Modeling Land-Use and Land-Cover Change.” In Land-Use and Land-Cover Change, 117–35. Berlin, Heidelberg: Springer Berlin Heidelberg.

Yu, H, Wu, Z, Wang, S, Wang, Y, and Ma, X. 2017. “Spatiotemporal Recurrent Convolutional Networks for Traffic Prediction in Transportation Networks.” Sensors (Switzerland) 17 (7).

Zhang, Q, Wang, H, Dong, J, Zhong, G, and Sun, X. 2017. “Prediction of Sea Surface Temperature Using Long Short-Term Memory.” IEEE Geoscience and Remote Sensing Letters 14 (10): 1745–49.

Zhu, XX, Tuia, D, Mou, L, Xia, GS, Zhang, L, Xu, F, and Fraundorfer, F. 2017. “Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources.” IEEE Geoscience and Remote Sensing Magazine 5 (4): 8–36.

Chapter 2.

A Sensitivity Analysis of Recurrent Neural Network Models for Forecasting Land Cover Change

2.1. Abstract

Land cover change (LCC) is a dynamic process characterized by gradual changes across Earth's surface. Data-driven methods for LCC analyses have included Deep Learning (DL) approaches. Recurrent Neural Networks (RNNs) are a type of DL method that utilize a variation of traditional neural network units, yielding successful outcomes in timeseries analysis tasks. While previous LCC analyses employing RNN architectures have demonstrated favourable performance, few research studies have reported on the assessment of model behaviour when variable input data characteristics have been used. The main objective of this study is to evaluate the performance of RNN models, namely the Long Short-Term Memory (LSTM) architecture, given varying temporal resolution and attributes of geospatial input data. This assessment is conducted using Sensitivity Analysis applied to various modeling scenarios involving a selection of spatial metrics. The approach aims to assess the implications of varying (1) temporal resolutions, (2) number of timesteps, (3) number of land cover classes, and (4) rates of change existing in the input dataset. The obtained results indicate that varying these data properties have important repercussions on method performance across all tested scenarios, despite adjustments to training time or method optimization techniques applied.

2.2. Introduction

Land cover change (LCC) is a dynamic process with spatial and temporal dependencies that can be characterized as complex systems. The interactions within and between human and natural environmental systems propagate LCCs over space and time (Meyer and Turner 1996). LCC research is significant to many disciplines such as geography, urban planning, environmental science, forestry, agriculture, and resource management (Zadbagher, Becek, and Berberoglu 2018; Rahman et al. 2017; Zhu et al. 2017). For instance, anthropogenic disturbances such as deforestation both directly and indirectly induces increased CO_2 levels and affect local weather changes, making LCC important to consider as global temperatures rise (Findell et al. 2017; Turner, Lambin, and Reenberg 2007; Mayfield et al. 2017; Novo-Fernández et al. 2018). Given the complexity of LCC, automated statistical learning approaches have been applied for analyses in this domain (Otukei and Blaschke 2010; Ienco et al. 2017).

Throughout the past decade, data-driven modeling methods have been increasingly considered with the unprecedented availability of data, the Big Data paradigm shift, and the escalating capacity of computational resources (Li et al. 2016). Machine Learning (ML) algorithms automate pattern recognition with minimal manual interference, making it important to acknowledge these techniques are deterministic approaches to modeling non-linear spatial processes. Given high dimensionality and volume of modern datasets, ML algorithms typically perform with great efficacy. Previous ML approaches for LC classification and forecasting have included Neural Networks (NNs) (Maithani 2015), Decision Trees (W. Boulila et al. 2011; Otukei and Blaschke 2010), Random Forests (Patil et al. 2017; Otukei and Blaschke 2010), and Support Vector Machines (Otukei and Blaschke 2010).

Deep Learning (DL) is a subfield of ML, characterized by NNs of increasing depth and breadth. DL techniques facilitate automated learning of feature representations and have demonstrated ability to capture intricate, hierarchical relationships from a dataset (Kong et al. 2018). A Recurrent Neural Network (RNN) is a type of NN suitable for sequential data. With internal memory allocated to each neuron, RNNs are useful for capturing dynamic, temporal dependencies (Hochreiter and Schmidhuber 1997). RNNs, specifically the Long Short-Term Memory (LSTM) variation, have exhibited propitious performance in previous LCC analyses (Rußwurm and Körner 2017; Ienco et al. 2017;

Lyu, Lu, and Mou 2016). However, geospatial input data characteristics conducive to the success of LSTMs have not been fully studied. This motivates an investigation of the method's response to varying geospatial data qualities by means of Sensitivity Analysis (SA).

SA is valuable for identifying favourable geospatial input data properties and how they affect model outputs (Lilburne and Tarantola 2009) and has been used in geospatial applications to evaluate the implications of changing inputs and parameters on the outputs produced (Kocabas and Dragicevic 2006; Ligmann-Zielinska and Sun 2010). SA approaches were also employed for super-resolution land cover mapping (Li et al. 2019) and LCC modeling (Boulila, Ayadi, and Farah 2017). These SA methods have measured a model's forecasting ability by comparing forecasted maps with reference maps and through measures such as Kappa indices or ROC statistics (Bone et al. 2014). Prior studies involving ML techniques have also emphasized measuring a model's capacity to forecast changed cells correctly (Samardžić-Petrović et al. 2017).

Therefore, the main objective of this study is to assess the implications of changing geospatial input data properties on the performance of LSTM models and to evaluate their ability to forecast localized LCCs using a SA. This research considers LCC sequences extracted along the temporal dimension for each cell within various synthetic datasets. A SA is evaluated with a collection of incremental modeling scenarios involving a selection of Kappa metrics. The aim is to measure the ramifications of varying (1) temporal resolution, (2) sequence length, (3) cardinality, and (4) rates of LCC. It is hypothesized that performance of these sequential DL models will be impeded by low temporal resolutions and shorter sequence lengths. Additionally, it is speculated that fewer unique classes and greater LCC occurrences will be favourable for all scenarios.

2.3. Theoretical Background of Recurrent Neural Networks

Traditional feed-forward NN models utilize gradient-based learning methods to update model parameters as new inputs are provided. The goal is to determine an optimal set of network weights that allow the model to generalize to new data. Inputs are fed through the network, a cost function is evaluated, and an error term is computed using the result obtained from the output layer. Derivatives are computed with respect to each weight in the network using a process called backpropagation (Bishop 2006; Lecun,

Bengio, and Hinton 2015). Backpropagation informs adjustments of network weight parameters with the intent of minimizing the error term.

RNNs are a type of DNN that are best suited for problems involving sequential data. By introducing a recurrence relation to the standard feed-forward neuron, information from previous timesteps is propagated to inform future cell state changes and outputs. Figure 2.1 depicts a typical RNN model, featuring a hidden state ht. Each hidden state at time t depends on the input xt and the previous state ht-1 (Figure 2.1). Network weights are shared across all timesteps. However, RNN structures are impeded by a phenomenon called a “Vanishing Gradient” (Hochreiter 1998). This occurrence inhibits the propagation of previous information and is caused by the inability to maintain gradients to backpropagate updates with respect to the error term. Gradients that are too small (vanishing) or too large (exploding) prevent any meaningful adjustments of network weights. This required alterations to the RNN architecture to handle long-term dependencies.

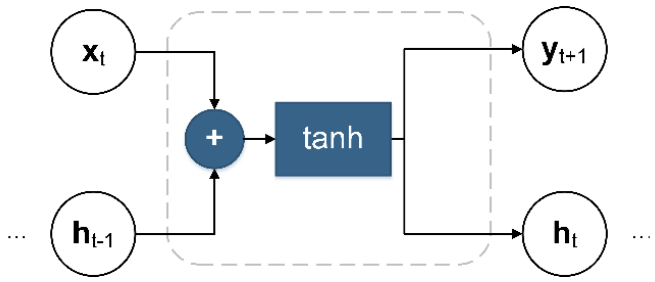


Figure 2.1. A basic RNN cell.

2.3.1. Long Short-Term Memory

A notable advancement of the RNN architecture was Long Short-Term Memory (LSTM), developed to solve the “Vanishing Gradient” problem (Hochreiter and Schmidhuber 1997). Internal components called “gates” permit LSTM cells to allow a range of information learned early in a timeseries to inform states in later stages of the sequence. Gates in an LSTM cell include an *input gate* (i_t), *forget gate* (f_t), *output gate* (o_t), and *input modulation gate* (g_t), which control how much information is permitted to propagate through the cell (Figure 2.2) (Lyu, Lu, and Mou 2016; Ienco et al. 2017).

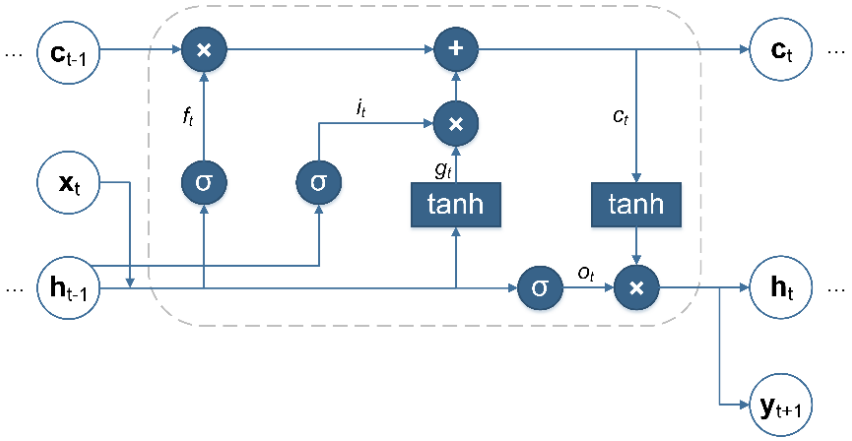


Figure 2.2. An LSTM unit (Figure adapted from Rußwurm and Körner (2017) and "Understanding LSTM Networks" (2016)).

Updates of an LSTM cell at timestep *t* when provided an input x_t are formulated based on Donahue et al. (2015). A hidden state passed from the previous timestep is denoted as h_{t-1}, and the internal memory state from the previous timestep is referred to as c_{t-1} (Donahue et al. 2015). The hidden state h_t is passed as an input parameter with the next input value x_{t+1}. Network weight matrices and bias vectors obtained during training procedures are denoted by W_{**} and b_* , respectively (Ienco et al. 2017). The sigmoid (σ) function constrains output to an interval of values from 0 to 1 and is used to control the flow of information from the *input gate* (i_t), *forget gate* (f_t), and *output gate* (o_t) (Equations 1-3). The hyperbolic tangent (*tanh*) function produces a value within the interval of -1 to 1, imposing a scaling operation to the result produced for the *input modulation gate* (g_t) (Equation 4). Element-wise multiplication is symbolized by $\odot$. The following equations (1)-(6) define the LSTM cell behavior as described in (Donahue et al. 2015):

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \quad (2)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (3)$$

$$g_t = \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (4)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (5)$$

$$h_t = o_t \odot \tanh(c_t) \quad (6)$$

2.3.2. Flow of Information through a Long Short-Term Memory Unit

Gates (i_t , f_t , o_t , and g_t) allow or obstruct information propagation through an LSTM unit. When the next item in the input sequence x_t is provided, the *input gate* (1) provides an indicator of how much information from x_t and of the hidden state from the previous timestep, h_{t-1}, should remain when updating the internal memory cell (c_t). The *forget gate* (2) produces a value indicating whether the previous cell state (c_{t-1}) should be saved, or if the network should "forget" some or all of the previous cell state. The *output gate* described in (3) controls how much of the current state committed to the cell's internal memory (c_t) should be passed as the next hidden state (h_t). The *input modulation gate* (4) scales the input values x_t and h_t prior to considering it for the update to the cell's memory in (5). In equations (5)-(6), both the cell's updated internal memory (c_t) and hidden state (h_t) are updated with consideration to constraints of gates (1)-(3) and the input modulation (4). Finally, the cell's internal memory and hidden state at timestep *t* are provided as input with the next item of the input sequence (x_{t+1}), repeating the process of updating the cell's internal memory and hidden state vectors.

2.3.3. Geospatial Applications of Recurrent Neural Networks

Geospatial applications of RNNs have included classification and forecasting (Zhu et al. 2017). Sauter et al. (2010) utilized snow cover derived from MODIS satellite data with meteorological data to develop a snow cover depth forecasting system using an early RNN approach (Sauter, Weitzenkamp, and Schneider 2010). More recently, an LSTM model called REFEREE, Lyu et al. (2016) demonstrate the abilities of RNN for binary and multi-class LCC detection. The transfer learning scenario involved various study areas featuring urban, water, soil, and agriculture. LSTMs have also been used to forecast sea ice concentration, with input sequences extracted along the temporal dimension for each cell in the raster dataset (Chi and Kim 2017). RNNs have also proven effective in LC classification. Provided 3- and 23-step timeseries, an LSTM model showed improved performance over Random Forest and Support Vector Machine methods (Ienco et al. 2017).

While previous studies have demonstrated the effectiveness of RNNs and its variants for geospatial applications, the performance of these methods when working with variable geospatial data properties has not been assessed. Using SA applied to three

modeling scenarios, the goal of this study is to explore the response of RNNs to varying geospatial data inputs.

2.4. Methods

The systematic assessment of RNN, specifically LSTM, involves modeling scenarios established to represent a typical development progression. The model response to changing geospatial input is assessed in each scenario. The aim is to observe if there are trends in model response despite model optimizations. An overview of the end-to-end methodology used to train respective models in each scenario is shown in Figure 2.3. An overview of the end-to-end methodology used to generate forecasted maps in each scenario is shown in Figure 2.4.

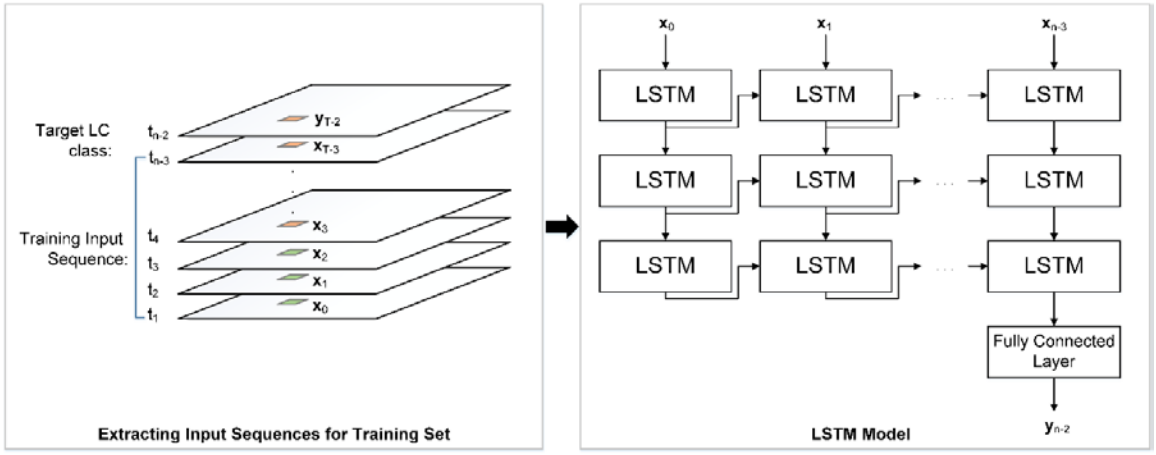


Figure 2.3. End-to-end methodology for training in each modeling scenario.

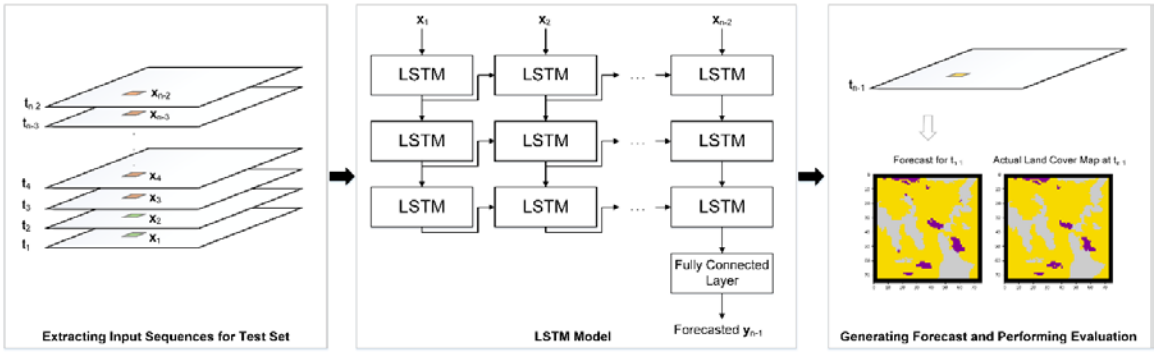


Figure 2.4. End-to-end methodology for generating forecasted maps in each modeling scenario.

A baseline RNN model undergoes optimizations to create new scenarios. Stacked DL models have proven ability to learn increasingly complex relationships (Pascanu et al. 2014; Hermans and Schrauwen 2013). Stacked LSTM models proposed in this work are composed of three layers, featuring 32 neurons per layer. Neurons refer to the units composing the network. Each are characterized by a chosen function, working in cohesion to produce a "statistical generalization" of the given dataset (Goodfellow, Bengio, and Courville 2016). The input layer is compatible with the one-hot encoded sequences. One-hot encoding involves converting LC class labels to vectors containing zeroes and a single non-zero value (1) at the index corresponding with the class ID (Pai and Potdar 2017). Each input sequence is a matrix of N × M dimensions corresponding to respective cells across the study area, where N denotes the number of timesteps and M denotes the number of possible categories represented by the one-hot encoded vector. The output layer of the respective models produces a M × 1 vector containing the probabilities of each class the model forecasts to occur at the cell at the next time step. The position in the output vector featuring the highest probability is selected, with the position in the vector corresponding to the forecasted class label.

Model parameter configuration includes pre-set components as well as iteratively selected hyperparameters (parameters that are set a priori to aid the model in achieving the "statistical generalization"). Hyperparameters set before initiating training procedures affect optimization of the model's internal parameters, thus influencing the quality of the model. The number of internal model parameters that are affected by chosen hyperparameters selected prior to model training is 24,567. Using a grid search approach, two model hyperparameters, the number of epochs and batch size, were determined for the respective models for each dataset (Chi and Kim 2017). The number of epochs refers to how many times the entire dataset is passed through the network while batch size refers to how many data points are considered when computing the gradient prior to each update of the model's internal parameters (weights and biases). The Adam optimization algorithm (Kingma and Lei Ba 2015) is used instead of the traditional stochastic gradient descent approach due to its proven success and its robustness to model hyperparameters. Categorical cross-entropy is utilized as the objective function to accommodate the multi-class data sequences. The Softmax activation function (Bishop 2006) was employed for the output layer to produce a vector of probabilities corresponding to each class label (Ienco et al. 2017). This activation function is commonly used with multi-class classification

and predictive models (Bishop 2006). Models were implemented using the Python programming language (v3.6.5) (van Rossum 2016) and the Keras API (v2.2.0) (Chollet 2015). The Keras API assists developers in prototyping ML and DL models while providing an interface to the extensive functionality of Google's TensorFlow (v1.8.0) (Chollet 2015). TensorFlow is an open-source ML framework that provides advanced features to construct and fine-tune the data-driven models (Abadi et al. 2016).

2.4.1. Modeling Scenarios

Modeling scenarios were created to assess method sensitivity to varying input data characteristics. These include (1) a deterministic baseline, (2) a stochastic scenario, and (3) a regularized stochastic scenario. These will be referred to as Model A, B, and C, respectively. By emulating a conventional development pathway, it is intended to reveal whether resulting metrics obtained exhibit trends that subsist as models are optimized. The scenarios ensure results are not unique to a specific configuration, as there are infinite arrangements and many model optimization techniques that have proven to improve results various applications (Pham et al. 2014).

The deterministic baseline modeling scenario (Model A) provides a foundation for subsequent models. This scenario features three LSTM layers and uses a random seed instead of allowing different network weight initializations for each run. This ensures results are reproducible given the same set of input parameters over repeated tests.

The stochastic scenario (Model B) uses "true" random weight initialization with the removal of the random seed. Model B iterates upon Model A, allowing initial weights for each run to be different given the same set of input parameters. Reintroducing stochastic weight initialization potentially leads to improved performance by determining a more suitable set of internal weights (Goodfellow, Bengio, and Courville 2016). This allows potentially better local optima to be found during the batch gradient descent procedure.

Finally, a regularized stochastic scenario (Model C) is developed to improve generalization and to prevent overfitting. This is done by applying dropout regularization between each of the LSTM layers and the final output layer to improve upon Model B. Dropout forces a percentage of neurons to be ignored (Srivastava et al. 2014). Dropout has been used in previous work involving geospatial data inputs extracted along the

temporal dimension, informing a dropout factor of 0.5 (Kong et al. 2018). This means that the probability that a neuron is “dropped” is 50% (Sutskever et al. 2014). When a neuron is “dropped,” input and output connections to the neuron are also ignored.

2.4.2. Sensitivity Analysis

To ensure each modeling scenario was evaluated equally, a collection of test cases was designed. This research study utilizes a local SA technique in which a single input parameter or variable is adjusted at a time (Boulila, Ayadi, and Farah 2017). For each synthetic dataset, for each temporal resolution, and for each modeling scenario, a grid search of hyperparameter space was conducted, producing new models and results at each iteration. Trained models were then tested, generating a forecasted map output for the next unobserved timestep. While outputs of all test case combinations were logged, the best performing models were selected based on their ability to forecast changed cells using the test set.

Forecasted and actual maps were compared using a variety of metrics to assess the method’s sensitivity to varying input data characteristics. Given that only the generated patterns of LCC were observed, overall accuracy and traditional Kappa metrics have been used (Hagen 2002). Previous geospatial applications of RNNs have also relied on traditional Kappa statistics (Lyu, Lu, and Mou 2016; Rußwurm and Körner 2018; Ienco et al. 2017), including Kappa, $K_{Histogram}$, and $K_{Location}$ (Hagen 2002). The Kappa metric provides a measurement of agreement between two maps being compared, while $K_{Histogram}$ considers the quantity of similarities between the two maps being compared (van Vliet, Bregt, and Hagen-Zanker 2011). $K_{Location}$ provides a measurement of agreement based on similarity of location for each class between two maps compared (Hagen 2002). To evaluate the method’s ability to forecast changes, $K_{Simulation}$, $K_{Transition}$, and $K_{Translocation}$ have also been selected (Hagen 2002; van Vliet, Bregt, and Hagen-Zanker 2011). $K_{Simulation}$, $K_{Transition}$, and $K_{Translocation}$ dismiss effects of persistent cells, enhancing assessment of the method’s ability to simulate changed cells (van Vliet, Bregt, and Hagen-Zanker 2011). $K_{Simulation}$ is a measure of agreement for changed cells present in a reference and simulated map. $K_{Transition}$ is a measure of agreement between the number of transitions occurring for each class between a reference and simulated map. Lastly, $K_{Translocation}$ provides a measure of agreement based on the similarity of location for transitioned cells in each LC class. While overall

accuracy of the model provides a general measure of its performance, typical metrics do not suffice for this study when the focus is to understand how well the method forecasts LCC. The number of changed cells with respective accuracies were utilized for comparison instead of the overall accuracy metric for ML modeling approaches (Samardžić-Petrović et al. 2017).

2.5. Data

Hypothetical data is used to regulate geospatial input characteristics to assess method response to changing inputs (Burlacu, O'Donoghue, and Sologon 2014). Using synthetic small-area datasets enables localized analyses and reduced computation time to perform evaluations (Hermes and Poulsen 2012) and are commonly used in geosimulation modeling (Ligmann-Zielinska and Sun 2010; Bone and Dragićević 2009). In this study, synthetic LC datasets were created to control the number of unique LC classes and rates of change.

The three datasets developed feature four, eight, and 16 LC classes, respectively (Table 2.1). The LC classes composing the respective datasets have been named as per classes featured in Homer et al. (2004) and Friedl, M., Sulla-Menashe (2015). The synthetic datasets have been generated using Esri's ArcGIS Pro (v2.4.0) in order to control LC changes to emulate or exaggerate real-world scenarios (Esri 2017). That is, LC classes have been specified to emerge, grow, or dissipate over time. For instance, in the four-class dataset, forest and cropland are shown to transition to low and high intensity developments (Figure 2.5).

Each dataset has identical dimensions, temporal resolution, and number of timesteps. Datasets feature 76×76 cells with 25-meter spatial resolution. A 3-cell buffer is considered around the entire study area to mitigate edge effects. This results in a working study area of 70×70 cells. These dimensions were chosen to expedite the evaluation processes, as small models (where breadth and depth are relatively small) can be typically expected to fit to small-scale datasets (Goodfellow, Bengio, and Courville 2016). Each full dataset features 45 years with one-year temporal resolution.

The number of cells belonging to each LC class at each timestep are shown in Figure 2.5. Test cases considering all 45 years, including timesteps t_0 to t_{44}, are referred

to as *full sequence* tests. Five *full sequence* tests feature five different temporal resolutions (1, 2, 4, 11, and 22-year) that ensured t_0 and t_{44} were included in all cases. This guaranteed equal comparisons of forecasts for t_{44}. Shorter sequence lengths were tested using data subsets of the 45 years available (Figure 2.5). Subsets contain 15 timesteps each, where *Subset A* includes t_0 to t_{14}, *Subset B* includes t_{15} to t_{29}, and *Subset C* includes t_{30} to t_{44}. The three 15-year test cases defined feature one, two, and seven-year temporal resolutions.

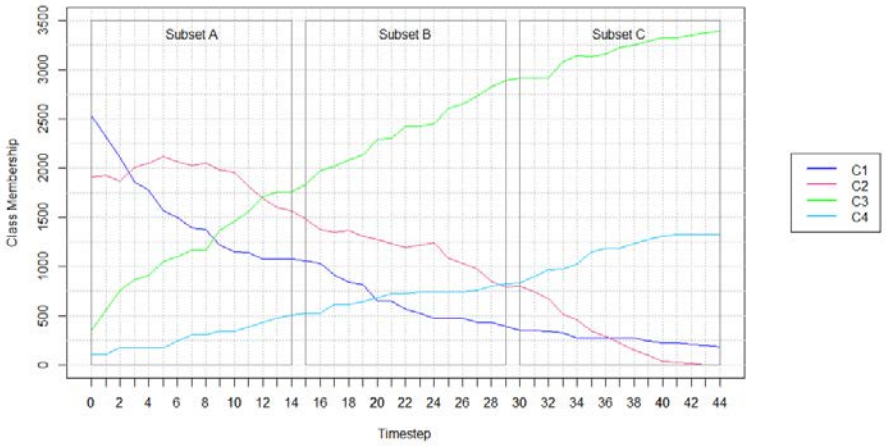
Table 2.1. LC Classes featured in the 4, 8, and 16-class datasets

4 Class Dataset	8 Class Dataset	16 Class Dataset
C1 – Forest Land	C1 – High Intensity Development	C1 – Cropland
C2 – Cropland	C2 – Pasture	C2 – Pasture
C3 – Low Intensity Development	C3 – Forest Land	C3 – Deciduous Forest
C4 – High Intensity Development	C4 – Barren Land	C4 – Evergreen Forest
	C5 – Grasslands	C5 – Mixed Forest
	C6 – Cropland	C6 – High Intensity Development
	C7 – Low Intensity Development	C7 – Low Intensity Development
	C8 – Water	C8 – Shrubland
		C9 – Grasslands
		C10 – Road Surfaces
		C11 – Barren Land
		C12 – Lakes
		C13 – Streams
		C14 – Wetland
		C15 – Beaches
		C16 – Bare Exposed Rock

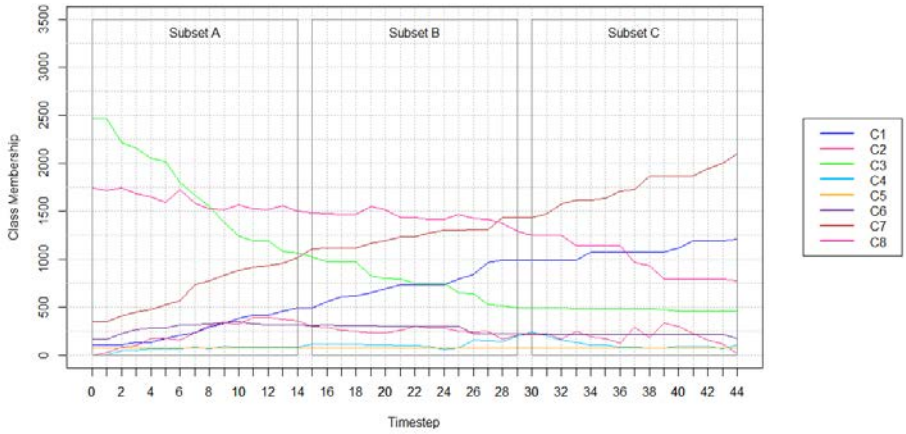
2.5.1. Data Pre-Processing

Input sequences are extracted from the synthetic datasets at each cell along the temporal dimension according to the specified temporal resolution. An input sequence in the training set is denoted as ($\mathbf{x}_0$, $\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_{T-3}$), while the target LC class is denoted by ($\mathbf{y}_{T-2}$). Input sequences in the test set are denoted as ($\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, …, $\mathbf{x}_{T-2}$), while the target LC class is denoted by ($\mathbf{y}_{T-1}$). The sliding temporal window approach for establishing training and test sets is used (Kong et al. 2018). Each ($\mathbf{x}_{T-N-1}$) and ($\mathbf{y}_{T-N}$), where N=1 or N=2, are one-hot encoded vectors representing the classes at each cell at each timestep.

(a)



(b)



(c)

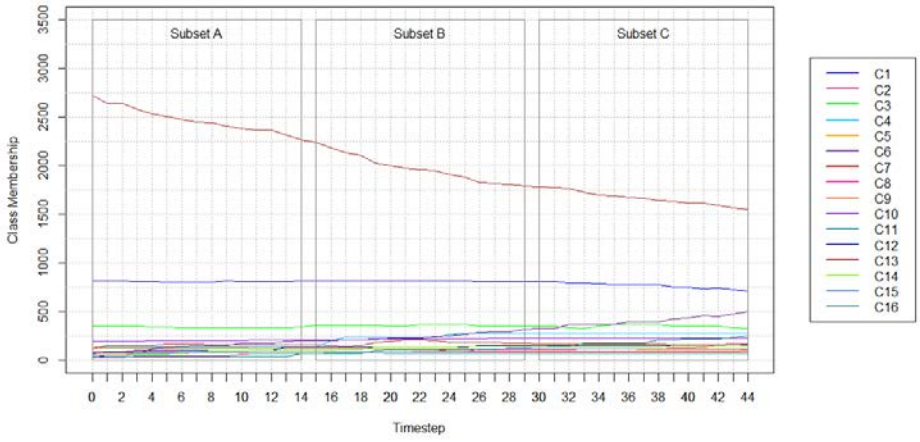


Figure 2.5. Number of cells belonging to each class at each timestep in the (a) 4 class, (b) 8 class, and (c) 16 class LC datasets.

Given the disproportionate percentages of persistent versus changed cells in LC datasets, a "balanced sampling strategy" was used (Samardžić-Petrović et al. 2017). To balance the training sets, it was first determined if a change occurred in an input sequence. If a change occurred, the sequence was marked as “changed” (otherwise “persistent”). If the number of sequences marked “changed” exceeded those marked as “persistent,” the entire training set was considered. Conversely, if more sequences were “persistent,” then equal counts of “changed” and “persistent” cells were sampled at random. This means that the number of training samples available for the four, eight, and 16-class datasets may differ according to the number of changes that occur in order to uphold the balanced sampling strategy.

2.6. Results

All modeling scenarios (A, B, and C) were considered for the generation of results. One run of a modeling scenario considering one dataset, the grid-search hyperparameter tuning approach, and varying temporal resolutions consisted of 180 models fit and evaluated. This is repeated across each modeling scenario (A, B, and C), with each of the LC datasets. The number of tests run becomes 180×3×3=1,620. Additional to map comparison metrics, various qualitative outputs were also output from each model evaluation, including simulation maps, and maps featuring misses produced in the forecast.

The results for *full sequence* tests featuring 45 years indicate models perform better with finer temporal resolutions (Table 2.2, Figure 2.6). Kappa, $K_{Histogram}$, $K_{Simulation}$, and $K_{Transition}$ measures exhibited distinct decreases as temporal resolution becomes coarser (ie. fewer data layers) across all scenarios. This decline is also apparent as the number of classes increases, observed as a repercussion of the decreasing sample size per class. As the number of classes present in the study area increases, the number of errors in the forecasted map increase, demonstrated in Figure 2.7. The $K_{Simulation}$ measures decrease as temporal resolution reduces and cardinality increases. For instance, $K_{Simulation}$ obtained from Model C outputs using one-year temporal resolution with four, eight, and 16 LC classes are 0.99, 0.74, and 0.73, respectively. An exception to these trends is shown in the results from Model B using the eight-class dataset (Table 2.2). Persistent cells were forecast with high accuracies in almost all configurations.

Simulation maps using the *full sequence* datasets are depicted in Figure 2.8 for Model C.

Table 2.2. Metrics obtained using 45-year LC datasets featuring 4, 8, and 16 classes. A, B, and C refer to modeling scenarios A, B, and C, respectively.

		4 Classes			8 Classes			16 Classes		
Performance Metric	Temporal Resolution	A	B	C	A	B	C	A	B	C
Changed Cell Forecasting Accuracy	1	0.993	0.996	0.995	0.728	0.703	0.762	0.668	0.699	0.715
	2	0.991	0.991	0.991	0.738	0.692	0.763	0.678	0.683	0.686
	4	0.988	0.986	0.986	0.680	0.687	0.676	0.657	0.659	0.661
	11	0.901	0.897	0.901	0.778	0.813	0.792	0.559	0.551	0.573
	22	0.785	0.785	0.785	0.619	0.583	0.649	0.164	0.164	0.164
Kappa	1	0.986	0.992	0.991	0.744	0.725	0.775	0.820	0.828	0.837
	2	0.982	0.982	0.982	0.755	0.712	0.777	0.824	0.824	0.826
	4	0.977	0.973	0.973	0.699	0.703	0.701	0.816	0.817	0.817
	11	0.816	0.808	0.812	0.783	0.815	0.797	0.777	0.775	0.783
	22	0.580	0.580	0.580	0.519	0.490	0.538	0.605	0.605	0.605
$K_{Histogram}$	1	0.989	0.992	0.992	0.788	0.756	0.813	0.845	0.850	0.846
	2	0.983	0.983	0.983	0.779	0.734	0.793	0.877	0.840	0.837
	4	0.977	0.979	0.979	0.772	0.780	0.729	0.850	0.845	0.839
	11	0.890	0.890	0.918	0.824	0.870	0.825	0.844	0.847	0.829
	22	0.721	0.721	0.721	0.567	0.490	0.583	0.624	0.622	0.624
$K_{Location}$	1	0.997	1.000	0.999	0.945	0.959	0.953	0.970	0.975	0.989
	2	0.999	0.999	0.999	0.969	0.970	0.980	0.940	0.981	0.987
	4	1.000	0.994	0.994	0.905	0.902	0.961	0.960	0.966	0.975
	11	0.917	0.908	0.885	0.950	0.936	0.966	0.921	0.915	0.945
	22	0.804	0.804	0.804	0.916	0.999	0.923	0.970	0.972	0.970
$K_{Simulation}$	1	0.984	0.990	0.989	0.710	0.689	0.744	0.703	0.710	0.731
	2	0.979	0.979	0.979	0.723	0.674	0.746	0.707	0.705	0.713
	4	0.973	0.968	0.968	0.660	0.666	0.661	0.691	0.694	0.694
	11	0.784	0.774	0.778	0.752	0.790	0.768	0.615	0.609	0.627
	22	0.526	0.526	0.526	0.453	0.420	0.477	0.307	0.306	0.307
$K_{Transition}$	1	0.986	0.990	0.989	0.753	0.724	0.773	0.724	0.722	0.733
	2	0.979	0.979	0.979	0.750	0.698	0.764	0.726	0.721	0.721
	4	0.973	0.976	0.976	0.735	0.751	0.693	0.713	0.724	0.718
	11	0.849	0.848	0.880	0.795	0.829	0.800	0.674	0.663	0.682
	22	0.628	0.628	0.628	0.506	0.421	0.528	0.326	0.322	0.326
$K_{Translocation}$	1	0.998	1.000	1.000	0.944	0.952	0.963	0.971	0.984	0.998
	2	1.000	1.000	1.000	0.964	0.965	0.977	0.973	0.978	0.989
	4	1.000	0.992	0.992	0.898	0.887	0.954	0.969	0.958	0.966
	11	0.924	0.913	0.884	0.946	0.952	0.960	0.913	0.918	0.919
	22	0.837	0.837	0.837	0.896	0.999	0.903	0.942	0.949	0.942

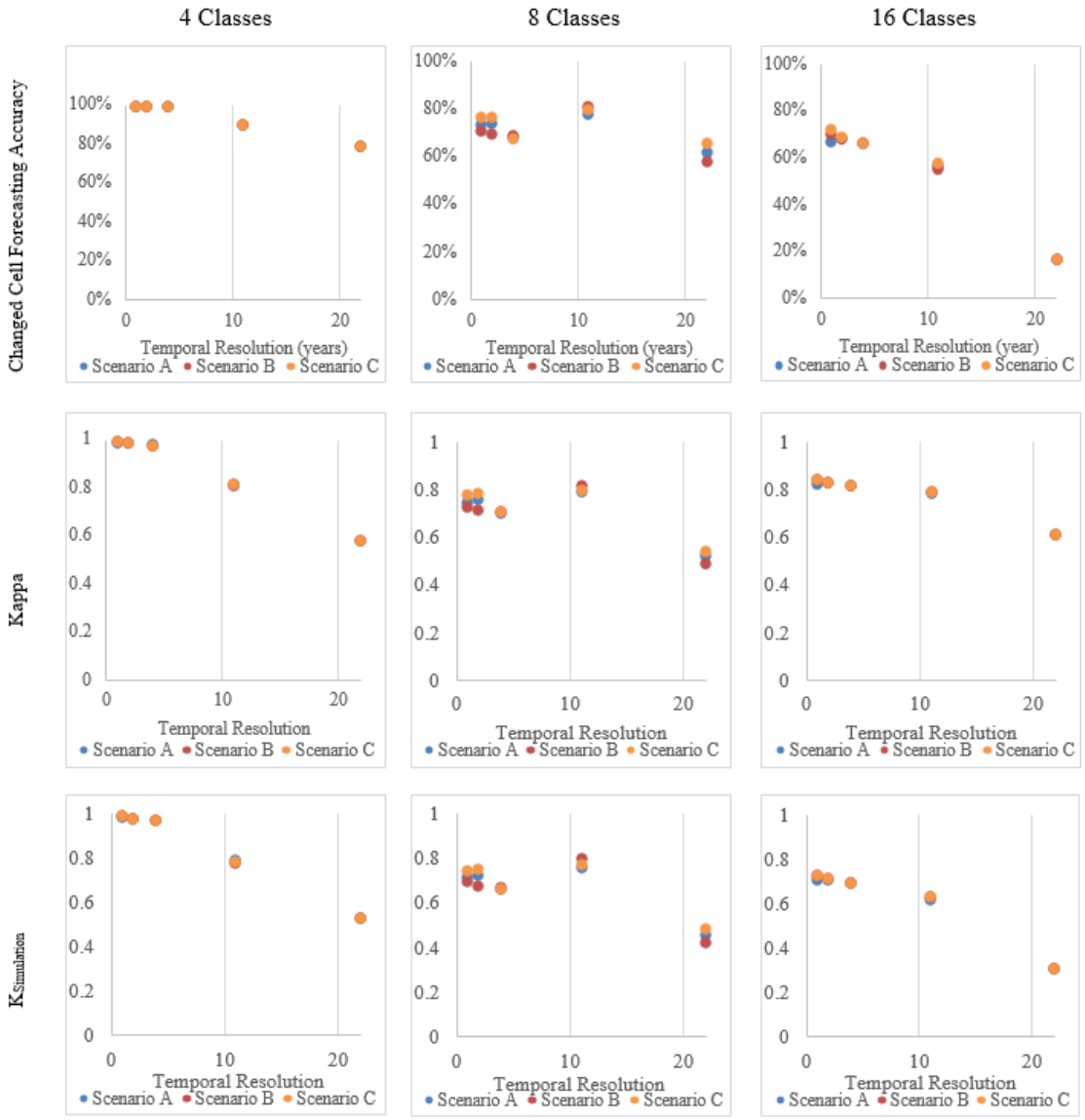
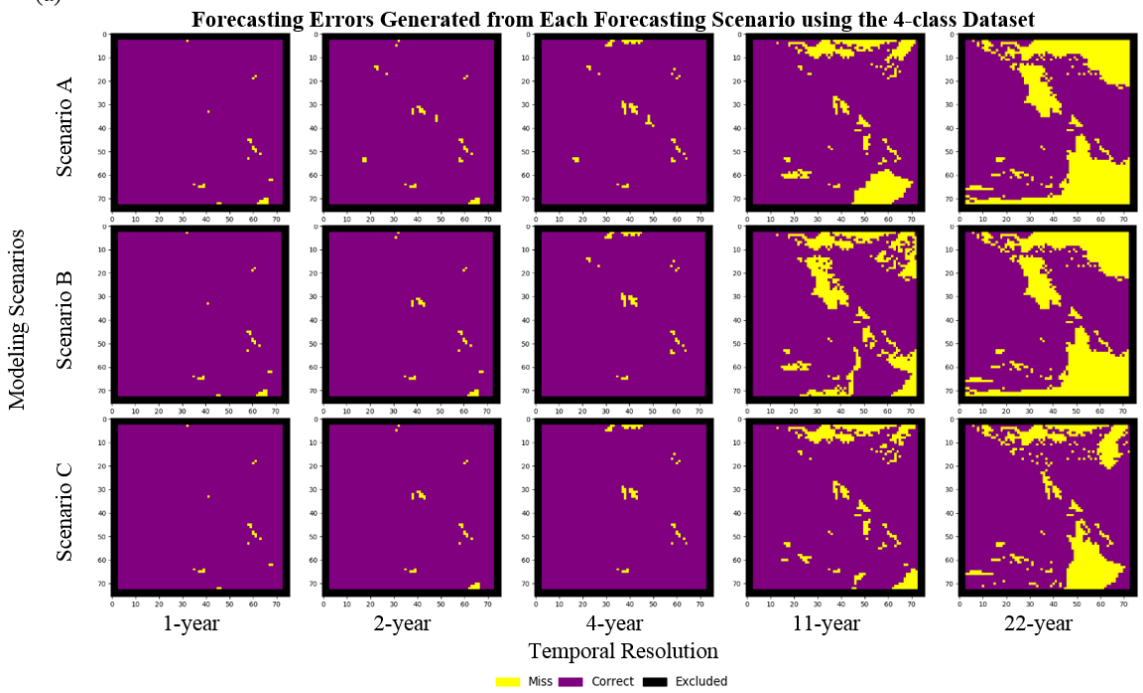
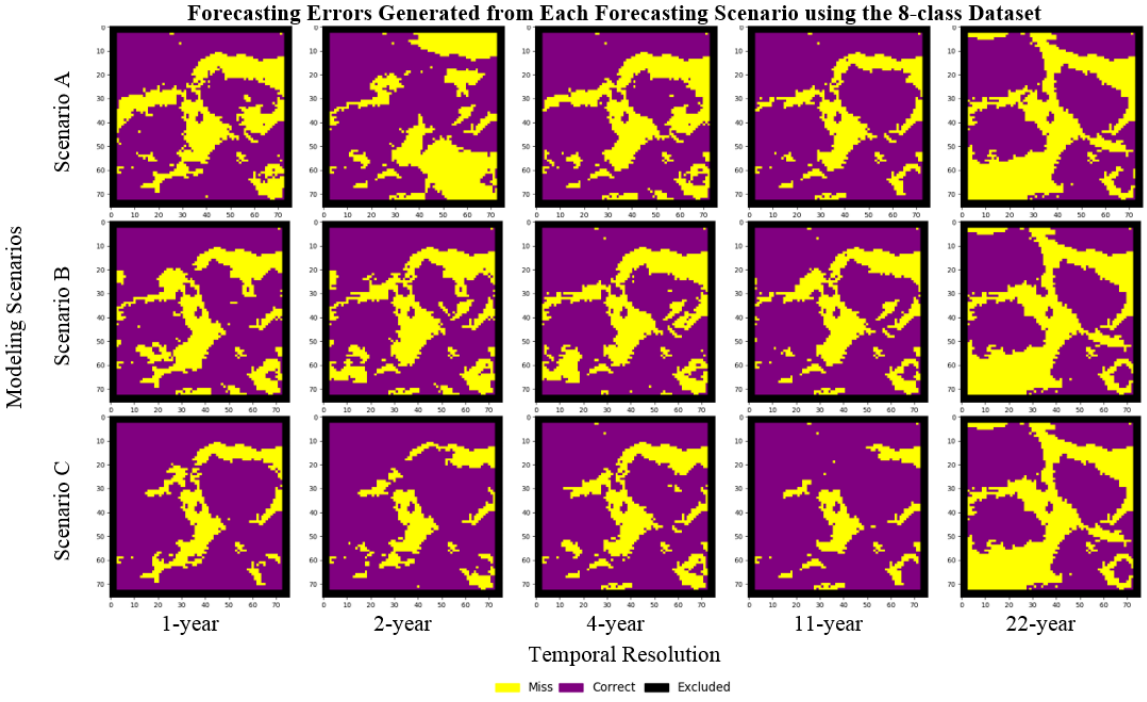


Figure 2.6. Trends in performance metrics versus increasing temporal resolution.

(a)


Forecasting Errors Generated from Each Forecasting Scenario using the 4-class Dataset
Modeling Scenarios
Scenario A
Scenario B
Scenario C
1-year
2-year
4-year
11-year
22-year
Temporal Resolution
Miss
Correct
Excluded

(b)


Forecasting Errors Generated from Each Forecasting Scenario using the 8-class Dataset
Modeling Scenarios
Scenario A
Scenario B
Scenario C
1-year
2-year
4-year
11-year
22-year
Temporal Resolution
Miss
Correct
Excluded

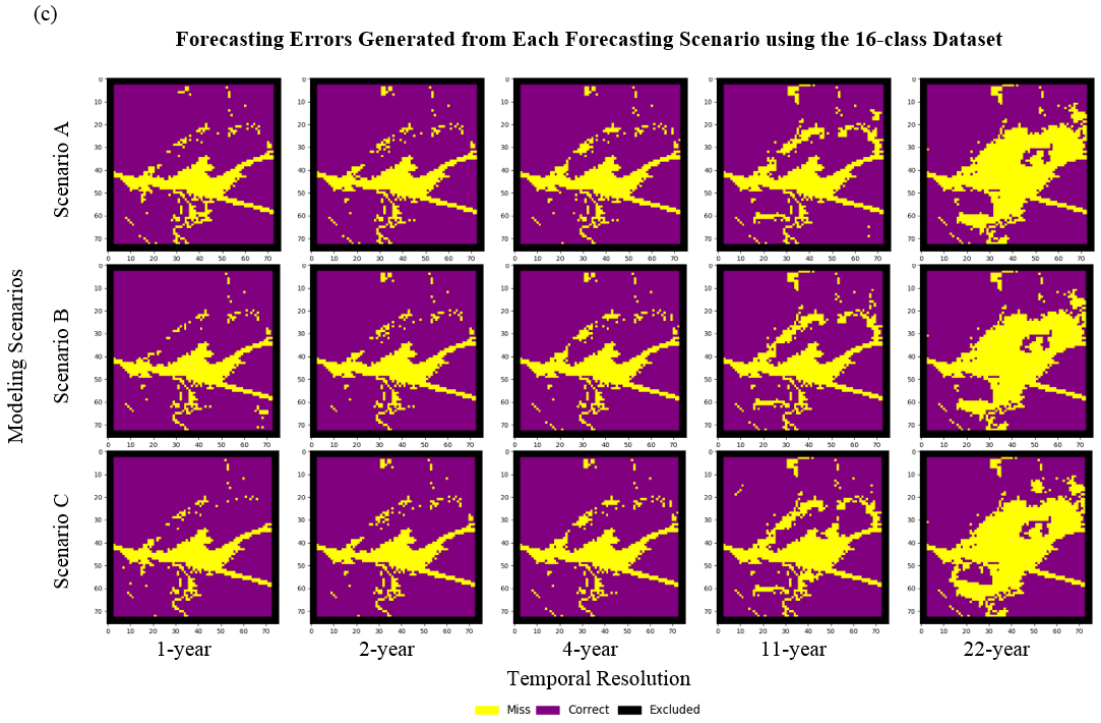


Figure 2.7. Forecasting Errors using 45-year dataset with (a) 4 class, (b) 8 class, and (c)16 class LC Datasets

The results for 15-year sequence tests indicate that across all modeling scenarios, $K_{Simulation}$ and $K_{Transition}$ metrics exhibit sharp decreases as temporal resolution becomes finer. The overall map comparison metrics including Kappa, $K_{Histogram}$, and $K_{Location}$ are typically higher when the number of LC changes is lower. As the number of changed cells increases, these measures of map agreement decrease. It is also observed that these shorter sequence lengths used for input to the sequential models result in lower performance metrics than in the 45-year sequence tests in most cases. Results obtained using the 15-year sequences have been shown in Tables 2.3-2.5.

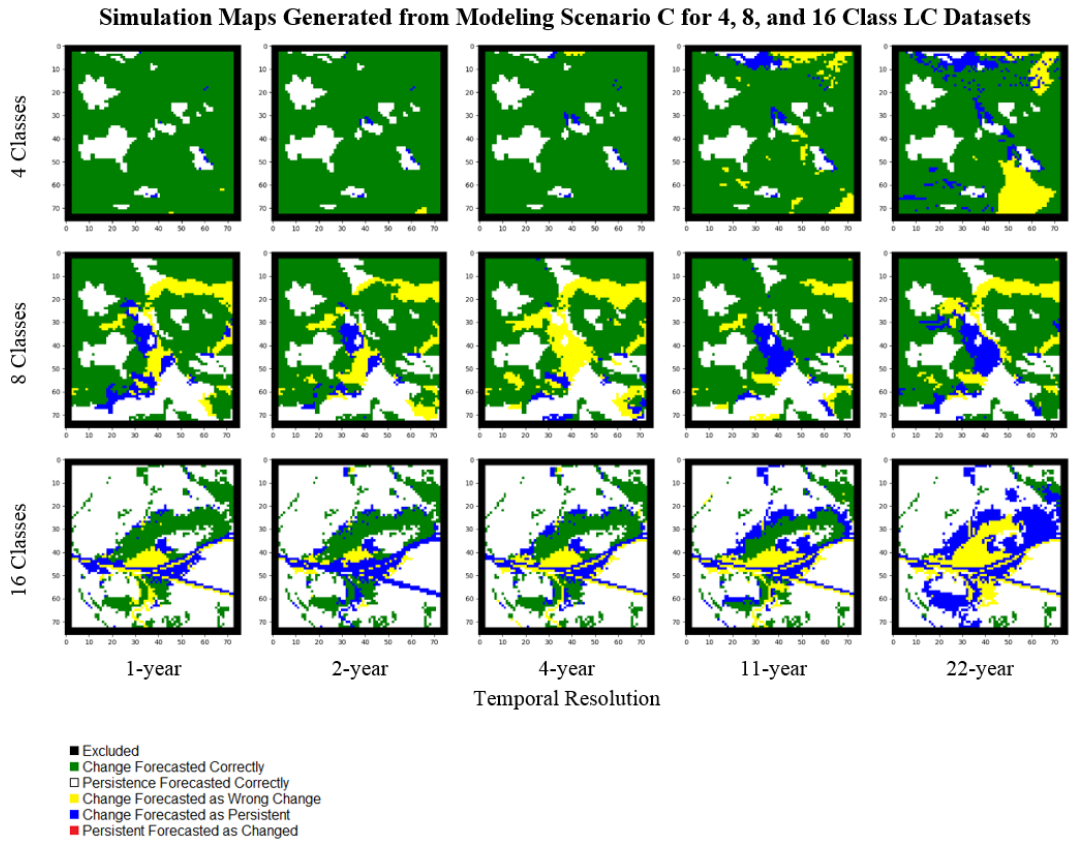


Figure 2.8. Simulation Maps generated from Modeling Scenario C using 45-year dataset with 4, 8, and 16 class LC Datasets.

2.7. Discussion

When analyzing results obtained from the SA, it was observed that the methods are highly affected by temporal resolution. Models performed better with finer temporal resolutions in nearly all test cases except Model B applied to the eight-class, 45-year dataset (Table 2.2, Figure 2.6). Changes that appear over coarser resolutions may appear more abrupt, impacting performance across all scenarios. This is expected to have affected performance of modeling scenarios applied to the eight-class dataset. LCC processes typically occur gradually over long periods of time and finer temporal resolutions preserve more detail. Thus, finer temporal resolutions are typically associated with higher changed cell forecasting accuracy.

Table 2.3. Performance Metrics obtained using Subset A (including timesteps 0 to 14), featuring 15-year LC datasets with 4, 8, and 16 classes.

		4 Classes			8 Classes			16 Classes		
Performance Metric	Temporal Resolution	A	B	C	A	B	C	A	B	C
Changed Cell Forecasting Accuracy	1	89.24%	84.98%	89.37%	78.40%	75.61%	77.26%	80.46%	80.46%	83.13%
	2	94.75%	94.91%	94.91%	74.68%	73.71%	75.61%	78.51%	78.51%	78.51%
	7	45.85%	45.85%	45.85%	35.50%	35.50%	35.50%	57.73%	57.73%	57.73%
Kappa	1	0.925	0.895	0.927	0.867	0.849	0.859	0.969	0.969	0.973
	2	0.963	0.964	0.964	0.845	0.839	0.850	0.966	0.966	0.966
	7	0.291	0.291	0.300	0.567	0.567	0.567	0.933	0.933	0.933
$K_{Histogram}$	1	0.940	0.935	0.938	0.893	0.867	0.887	0.972	0.972	0.977
	2	0.963	0.964	0.964	0.925	0.946	0.910	0.969	0.971	0.969
	7	0.379	0.379	0.387	0.752	0.752	0.752	0.944	0.944	0.944
$K_{Location}$	1	0.983	0.957	0.988	0.970	0.980	0.969	0.997	0.997	0.997
	2	1.000	1.000	1.000	0.914	0.887	0.934	0.997	0.995	0.997
	7	0.768	0.768	0.776	0.754	0.754	0.754	0.988	0.988	0.988
$K_{Simulation}$	1	0.908	0.872	0.910	0.824	0.802	0.813	0.881	0.881	0.898
	2	0.955	0.956	0.956	0.796	0.788	0.801	0.867	0.869	0.868
	7	0.207	0.207	0.222	0.423	0.423	0.423	0.712	0.712	0.712
$K_{Transition}$	1	0.927	0.913	0.921	0.848	0.812	0.838	0.881	0.881	0.898
	2	0.955	0.956	0.956	0.858	0.876	0.844	0.868	0.874	0.868
	7	0.305	0.305	0.318	0.544	0.544	0.544	0.712	0.712	0.712
$K_{Translocation}$	1	0.979	0.955	0.989	0.972	0.987	0.971	1.000	1.000	1.000
	2	1.000	1.000	1.000	0.928	0.899	0.948	0.999	0.994	1.000
	7	0.678	0.678	0.697	0.776	0.776	0.776	1.000	1.000	1.000

Using one input feature with one input label for training and testing sequences, respectively, produced poor performing models across all scenarios. This response to using the coarsest resolution indicates this type of model may not be appropriate for geospatial input datasets featuring only three timesteps available for training and testing. Using finer temporal resolutions with more timesteps produced the best results. Datasets with higher rates of change or variability also appeared to benefit from finer temporal resolutions. An exception to this trend is shown in Table 2.3 considering the four-class, 15-year dataset including timesteps 0 to 14. Abrupt changes occurring between each timestep from timesteps 0 to 14 are expected to have impacted performance metrics. That is, considering the four-class synthetic dataset (see Appendix) at two-year temporal resolution, cell transition rates are observed as less erratic. This further exemplifies the method's sensitivity to increased or inconsistent rates of LCC.

Table 2.4. Performance Metrics obtained using Subset B (including timesteps 15 to 29), featuring 15-year LC datasets with 4, 8, and 16 classes.

		4 Classes			8 Classes			16 Classes		
Performance Metric	Temporal Resolution	A	B	C	A	B	C	A	B	C
Changed Cell Forecasting Accuracy	1	84.58%	89.26%	89.83%	70.45%	76.37%	72.93%	92.68%	92.50%	95.31%
	2	83.65%	84.71%	84.71%	63.50%	61.52%	67.01%	81.80%	81.99%	82.18%
	7	61.74%	61.74%	61.74%	37.23%	37.23%	37.23%	47.28%	54.97%	54.97%
Kappa	1	0.916	0.941	0.944	0.854	0.875	0.862	0.990	0.990	0.994
	2	0.912	0.917	0.917	0.807	0.845	0.864	0.976	0.976	0.976
	7	0.803	0.803	0.803	0.714	0.714	0.713	0.896	0.939	0.939
$K_{Histogram}$	1	0.916	0.954	0.944	0.875	0.891	0.878	0.990	0.990	0.994
	2	0.912	0.917	0.917	0.876	0.903	0.921	0.976	0.976	0.976
	7	0.815	0.815	0.815	0.845	0.846	0.845	0.905	0.939	0.939
$K_{Location}$	1	1.000	0.986	1.000	0.976	0.982	0.982	1.000	1.000	1.000
	2	1.000	1.000	1.000	0.922	0.936	0.938	1.000	1.000	1.000
	7	0.985	0.985	0.985	0.844	0.845	0.844	0.990	1.000	1.000
$K_{Simulation}$	1	0.848	0.894	0.900	0.672	0.726	0.697	0.958	0.957	0.973
	2	0.840	0.850	0.850	0.589	0.653	0.703	0.889	0.890	0.891
	7	0.629	0.629	0.629	0.365	0.365	0.365	0.527	0.693	0.693
$K_{Transition}$	1	0.848	0.917	0.900	0.688	0.747	0.699	0.958	0.957	0.973
	2	0.840	0.850	0.850	0.610	0.677	0.724	0.889	0.891	0.893
	7	0.652	0.652	0.652	0.416	0.416	0.416	0.567	0.693	0.693
$K_{Translocation}$	1	1.000	0.974	1.000	0.976	0.972	0.998	1.000	1.000	1.000
	2	1.000	1.000	1.000	0.966	0.964	0.972	1.000	0.999	0.999
	7	0.964	0.964	0.964	0.877	0.877	0.877	0.929	1.000	1.000

Results showed that increasing sequence length improved a model's capacity to forecast changes. As the input sequence length decreased, models exhibited poorer performance indicated by the overall map comparison metrics, including Kappa, $K_{Histogram}$, and $K_{Location}$. In cases where smallest sequence lengths are used in both 45-year and 15-year test cases, $K_{Simulation}$ and $K_{Transition}$ indicate that models are hindered from learning LCCs. No model trained on the 15-year, four-class datasets exceeded performance measures of models trained on all 45 timesteps with one- and two-year resolution, indicating improved performance is associated with greater sequence length.

Table 2.5. Performance Metrics obtained using Subset C (including timesteps 30 to 45), featuring 15-year LC datasets with 4, 8, and 16 classes.

		4 Classes			8 Classes			16 Classes		
Performance Metric	Temporal Resolution	A	B	C	A	B	C	A	B	C
Changed Cell Forecasting Accuracy	1	97.12%	98.25%	98.25%	81.18%	81.91%	87.17%	82.38%	85.90%	86.34%
	2	95.68%	95.88%	95.88%	67.72%	70.77%	72.56%	67.18%	69.82%	73.79%
	7	82.30%	82.30%	82.30%	71.92%	71.92%	71.92%	38.33%	38.33%	38.33%
Kappa	1	0.987	0.992	0.992	0.913	0.918	0.928	0.981	0.985	0.985
	2	0.981	0.982	0.982	0.878	0.884	0.888	0.964	0.967	0.971
	7	0.924	0.924	0.924	0.887	0.887	0.887	0.932	0.932	0.932
$K_{Histogram}$	1	0.988	0.992	0.992	0.921	0.938	0.948	0.981	0.985	0.986
	2	0.984	0.983	0.983	0.907	0.934	0.949	0.972	0.972	0.975
	7	0.926	0.926	0.926	0.909	0.909	0.909	0.942	0.942	0.942
$K_{Location}$	1	0.999	1.000	1.000	0.992	0.978	0.979	0.999	0.999	0.999
	2	0.997	0.999	0.999	0.968	0.946	0.936	0.992	0.995	0.996
	7	0.999	0.999	0.999	0.975	0.975	0.975	0.990	0.990	0.990
$K_{Simulation}$	1	0.951	0.970	0.970	0.715	0.734	0.760	0.894	0.917	0.919
	2	0.926	0.930	0.930	0.607	0.628	0.627	0.787	0.806	0.835
	7	0.688	0.688	0.688	0.574	0.574	0.574	0.533	0.533	0.533
$K_{Transition}$	1	0.951	0.970	0.970	0.715	0.734	0.772	0.894	0.917	0.919
	2	0.926	0.930	0.930	0.607	0.721	0.727	0.797	0.816	0.845
	7	0.688	0.688	0.688	0.574	0.574	0.574	0.545	0.545	0.545
$K_{Translocation}$	1	1.000	1.000	1.000	1.000	1.000	0.985	1.000	1.000	1.000
	2	1.000	1.000	1.000	1.000	0.871	0.862	0.988	0.988	0.989
	7	1.000	1.000	1.000	1.000	1.000	1.000	0.979	0.979	0.979

Overall, the four-class dataset was associated with the best performance metrics of all the tests conducted. The SA indicates that dataset cardinality also affects method performance. The 16-class dataset proved most challenging for all modeling scenarios to forecast changed cells. The more optimized models (Model C) performed slightly better as the number of LC classes increased. Finer temporal resolutions and increased sequence lengths also improved model performance as LC cardinality increased. The four-class dataset allowed all modeling scenarios to forecast cell transitions with high performance measures.

Contrary to initial speculations, the SA demonstrated that models were most effective when rates of LCC were more gradual. The 16-class subsets where few changes occurred enabled modeling scenarios to obtain improved performance measures despite high cardinality. As the number of changes increased, model performance suffered especially when using coarse temporal resolutions and when considering the 16-class dataset. Overall, cells featuring persistent LC were typically forecasted correctly with fine temporal resolutions, indicating that modifications or an alternate approach to the

sampling strategy procedure used to compose training datasets in this research study should be considered to obtain improved results.

2.8. Conclusion

This research study aimed to evaluate the repercussions of varying (1) temporal resolution, (2) sequence length, (3) LC class count, and (4) rates of LCC present in the input dataset. The systematic SA identified similar responses across Models A, B, and C. Results indicated that LSTM models perform poorly when datasets feature coarser temporal resolution, fewer timesteps, increased cardinality, and greater LCC rates over time. LSTM models perform best when datasets feature finer temporal resolution, longer sequence lengths, lower cardinality, and more gradual LCCs.

This study considered changes occurring at each cell over the temporal dimension, neglecting explicit spatial dependencies. There is a need to investigate how spatiotemporal dependencies may affect the performance and spatial autocorrelation should be considered explicitly in future studies. Experiments considering convolutional LSTM (ConvLSTM) have begun considering one-year temporal resolution data, exhibiting similar sensitivity to the number of LC classes. The size of filter being considered should be further assessed using a SA. Larger filter sizes used with this method produced forecasts with worse agreement to the real map in all scenarios tested thus far. A preliminary study has been conducted and results show that the ConvLSTM method forecasts the four-class dataset nearly as well as the LSTM method. As cardinality is increased, ConvLSTM demonstrations potential to be more robust to this data property than LSTM. It has not yet been assessed if the method produces lower quality forecasts as temporal resolution becomes coarser, sequence length increases, and rates of LCC present in the considered dataset increase. Future work should continue to explore the capacity of ConvLSTM for modeling LCC. Likewise, a comparison of LSTM and ConvLSTM should be conducted to compare response to the varying geospatial data properties assessed in this study.

The models designed for this study was intended to focus on local changes and control input variations to assess model response while reducing computation time. Producing categorical forecasts also have implications on results. It would be possible to highlight areas of uncertainty using the probabilistic outputs produced for each cell, which

could help direct attention to more specific characteristics of geospatial datasets that prove challenging for models to forecast. Likewise, an assessment of how well the modeling scenarios were able to forecast emerging phenomena or dissipating phenomena should be investigated.

As cell-based LSTM models are fit to accommodate variation occurring across an entire study area, it would prove useful to determine the implications on method performance when using actual datasets with increased spatial heterogeneity. The large number of optimizations and configurations of LSTM models also provides opportunities for future exploration. The SA indicated the capacity of LSTM for modeling gradual change as exhibited by LCC. Similarly, this study shows that RNNs become even more suitable for forecasting LCC as more timesteps become available.

2.9. References

Abadi, M, Agarwal, A, Barham, P, Brevdo, E, Chen, Z, Citro, C, Corrado, GS, et al. 2016. “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems.”

Bishop, CM. 2006. Pattern Recognition and Machine Learning. Springer-Verlag New York.

Bone, C and Dragićević, S. 2009. “Defining Transition Rules with Reinforcement Learning for Modeling Land Cover Change.” Simulation 85 (5): 291–305.

Bone, C, Johnson, B, Nielsen-Pincus, M, Sproles, E, and Bolte, J. 2014. “A Temporal Variant-Invariant Validation Approach for Agent-Based Models of Landscape Dynamics.” Transactions in GIS 18 (2): 161–82.

Boulila, W., Farah, IR, Saheb Ettabaa, K, Solaiman, B, and Ghézala, H Ben. 2011. “A Data Mining Based Approach to Predict Spatiotemporal Changes in Satellite Images.” International Journal of Applied Earth Observation and Geoinformation 13 (3): 386–95.

Boulila, W, Ayadi, Z, and Farah, IR. 2017. “Sensitivity Analysis Approach to Model Epistemic and Aleatory Imperfection: Application to Land Cover Change Prediction Model.” Journal of Computational Science 23 (November): 58–70.

Burlacu, I, O’Donoghue, C, and Sologon, DM. 2014a. “Hypothetical Models.” In Handbook of Microsimulation Modelling (Contributions to Economic Analysis, Volume 293), 23–46. Emerald Group Publishing Limited.

Chi, J and Kim, H. 2017. “Prediction of Arctic Sea Ice Concentration Using a Fully Data Driven Deep Neural Network.” Remote Sensing 9 (12): 1305.

Chollet, F. 2015. “Keras: The Python Deep Learning Library.” Keras.Io. 2015.

Donahue, J, Hendricks, LA, Guadarrama, S, Rohrbach, M, Venugopalan, S, Darrell, T, and Saenko, K. 2015. “Long-Term Recurrent Convolutional Networks for Visual Recognition and Description.” In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 07-12-June:2625–34.

Esri. 2017. “ArcGIS Pro: 2.0.” Redlands, CA: Environmental Systems Research Institute.

Findell, KL, Berg, A, Gentine, P, Krasting, JP, Lintner, BR, Malyshev, S, Santanello, JA, and Shevliakova, E. 2017. “The Impact of Anthropogenic Land Use and Land Cover Change on Regional Climate Extremes.” Nature Communications 8 (1): 989.

Friedl, M., Sulla-Menashe, D. 2015. “MCD12Q1 MODIS/Terra+Aqua Land Cover Type Yearly L3 Global 500m SIN Grid V006 [Data Set].” NASA EOSDIS Land Processes DAAC. 2015.

Goodfellow, I, Bengio, Y, and Courville, A. 2016. Deep Learning Book. MIT Press.

Hagen, A. 2002. “Multi-Method Assessment of Map Similarity.” 5th AGILE Conference on Geographic Information Sceicne, 1–8.

Hermans, M and Schrauwen, B. 2013. “Training and Analysing Deep Recurrent Neural Networks.” Advances in Neural Information Processing Systems 26 (NIPS 2013), 190–98.

Hermes, K and Poulsen, M. 2012. “A Review of Current Methods to Generate Synthetic Spatial Microdata Using Reweighting and Future Directions.” Computers, Environment and Urban Systems 36 (4): 281–90.

Hochreiter, S. 1998. “The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions.” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 06 (02): 107–16.

Hochreiter, S and Schmidhuber, J. 1997. “Long Short-Term Memory.” Neural Computation 9 (8): 1735–80.

Homer, C, Huang, C, Yang, L, Wylie, B, and Coan, M. 2004. “Development of a 2001 National Land-Cover Database for the United States.” Photogrammetric Engineering and Remote Sensing 70 (7): 829–40.

Ienco, Di, Gaetano, R, Dupaquier, C, and Maurel, P. 2017. “Land Cover Classification via Multitemporal Spatial Data by Deep Recurrent Neural Networks.” IEEE Geoscience and Remote Sensing Letters 14 (10): 1685–89.

Kingma, DP and Lei Ba, J. 2015. “Adam: A Method for Stochastic Optimization.” In Proceedings of the 3rd International Conference on Learning Representations (ICLR). San Diego.

Kocabas, V and Dragicevic, S. 2006. “Assessing Cellular Automata Model Behaviour Using a Sensitivity Analysis Approach.” Computers, Environment and Urban Systems 30 (6): 921–53.

Kong, Y-LL, Huang, Q, Wang, C, Chen, JJ, Chen, JJ, and He, D. 2018. “Long Short-Term Memory Neural Networks for Online Disturbance Detection in Satellite Image Time Series.” Remote Sensing 10 (3): 452.

Lecun, Y, Bengio, Y, and Hinton, G. 2015. “Deep Learning.” Nature 521 (7553): 436–44.

Li, S, Dragicevic, S, Castro, FA, Sester, M, Winter, S, Coltekin, A, Pettit, C, et al. 2016. “Geospatial Big Data Handling Theory and Methods: A Review and Research Challenges.” ISPRS Journal of Photogrammetry and Remote Sensing 115 (May): 119–33.

Li, X, Ling, F, Foody, GM, Ge, Y, Zhang, Y, Wang, L, Shi, L, Li, X, and Du, Y. 2019. “Spatial–Temporal Super-Resolution Land Cover Mapping With a Local Spatial–Temporal Dependence Model.” IEEE Transactions on Geoscience and Remote Sensing 57 (7): 4951–66.

Ligmann-Zielinska, A and Sun, L. 2010. “Applying Time-Dependent Variance-Based Global Sensitivity Analysis to Represent the Dynamics of an Agent-Based Model of Land Use Change.” International Journal of Geographical Information Science 24 (12): 1829–50.

Lilburne, L and Tarantola, S. 2009. “Sensitivity Analysis of Spatial Models.” International Journal of Geographical Information Science 23 (2): 151–68.

Lyu, H, Lu, H, and Mou, L. 2016. “Learning a Transferable Change Rule from a Recurrent Neural Network for Land Cover Change Detection.” Remote Sensing 8 (6): 506.

Maithani, S. 2015. “Neural Networks-Based Simulation of Land Cover Scenarios in Doon Valley, India.” Geocarto International 30 (2): 163–85.

Mayfield, H, Smith, C, Gallagher, M, and Hockings, M. 2017. “Use of Freely Available Datasets and Machine Learning Methods in Predicting Deforestation.” Environmental Modelling and Software 87 (January): 17–28.

Meyer, WB and Turner, BL. 1996. “Land-Use/Land-Cover Change: Challenges for Geographers.” GeoJournal 39 (3): 237–40.

Novo-Fernández, A, Franks, S, Wehenkel, C, López-Serrano, PM, Molinier, M, and López-Sánchez, CA. 2018. “Landsat Time Series Analysis for Temperate Forest Cover Change Detection in the Sierra Madre Occidental, Durango, Mexico.” International Journal of Applied Earth Observation and Geoinformation 73 (December): 230–44.

Otukei, JR and Blaschke, T. 2010. “Land Cover Change Assessment Using Decision Trees, Support Vector Machines and Maximum Likelihood Classification Algorithms.” International Journal of Applied Earth Observation and Geoinformation 12 (SUPPL. 1): S27–31.

Pai, C and Potdar, K. 2017. “A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers.” Article in International Journal of Computer Applications 175 (4): 975–8887.

Pascanu, R, Gulcehre, C, Cho, K, and Bengio, Y. 2014. “How to Construct Deep Recurrent Neural Networks.” In Proceedings of the 2nd International Conference on Learning Representations (ICLR). Banff, Canada.

Patil, SD, Gu, Y, Dias, FSA, Stieglitz, M, and Turk, G. 2017. “Predicting the Spectral Information of Future Land Cover Using Machine Learning.” International Journal of Remote Sensing 38 (20): 5592–5607.

Pham, V, Bluche, T, Kermorvant, C, and Louradour, J. 2014. “Dropout Improves Recurrent Neural Networks for Handwriting Recognition.” In Proceedings of International Conference on Frontiers in Handwriting Recognition (ICFHR), 285–90. Crete, Greece.

Rahman, MTU, Tabassum, F, Rasheduzzaman, M, Saba, H, Sarkar, L, Ferdous, J, Uddin, SZ, and Zahedul Islam, AZM. 2017. “Temporal Dynamics of Land Use/Land Cover Change and Its Prediction Using CA-ANN Model for Southwestern Coastal Bangladesh.” Environmental Monitoring and Assessment 189 (11).

Rußwurm, M and Körner, M. 2017. “Multi-Temporal Land Cover Classification with Long Short-Term Memory Neural Networks.” International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives 42 (1W1): 551–58.

———. 2018. “Multi-Temporal Land Cover Classification with Sequential Recurrent Encoders.” ISPRS International Journal of Geo-Information 7 (4): 129.

Samardžić-Petrović, M, Kovačević, M, Bajat, B, and Dragićević, S. 2017. “Machine Learning Techniques for Modelling Short Term Land-Use Change.” ISPRS International Journal of Geo-Information 6 (12): 387.

Sauter, T, Weitzenkamp, B, and Schneider, C. 2010. “Spatio-Temporal Prediction of Snow Cover in the Black Forest Mountain Range Using Remote Sensing and a Recurrent Neural Network.” International Journal of Climatology 30 (15): 2330–41.

Srivastava, N, Hinton, G, Krizhevsky, A, and Salakhutdinov, R. 2014. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting.” Journal of Machine Learning Research. Vol. 15.

Sutskever, I, Hinton, G, Krizhevsky, A, and Salakhutdinov, RR. 2014. “Dropout : A Simple Way to Prevent Neural Networks from Overfitting.” Journal of Machine Learning Research. Vol. 15.

Turner, BL, Lambin, EF, and Reenberg, A. 2007. “The Emergence of Land Change Science for Global Environmental Change and Sustainability.” In Proceedings of the National Academy of Sciences. Vol. 104. National Academy of Sciences.

Understanding LSTM Networks. (2016). Retrieved July 17, 2019, from http://colah.github.io/posts/2015-08-Understanding-LSTMs/

van Rossum, G. 2016. Python 3.6 Language Reference. United Kingdom: Samurai Media Limited.

van Vliet, J, Bregt, AK, and Hagen-Zanker, A. 2011. “Revisiting Kappa to Account for Change in the Accuracy Assessment of Land-Use Change Models.” Ecological Modelling 222 (8): 1367–75.

Zadbagher, E, Becek, K, and Berberoglu, S. 2018. “Modeling Land Use/Land Cover Change Using Remote Sensing and Geographic Information Systems: Case Study of the Seyhan Basin, Turkey.” Environmental Monitoring and Assessment 190 (8).

Zhu, XX, Tuia, D, Mou, L, Xia, GS, Zhang, L, Xu, F, and Fraundorfer, F. 2017. “Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources.” IEEE Geoscience and Remote Sensing Magazine 5 (4): 8–36.

Chapter 3.

Analyzing the Effects of Temporal Resolution and Classification Error Propagation for Modeling Land Cover Change with Long Short-Term Memory Networks

3.1. Abstract

Land Cover Change (LCC) is typically characterized by infrequent changes over space and time. Data-driven methods including Deep Learning (DL) approaches have proven effective in many domains for predictive and classification tasks. Applied to real-world geospatial data, sequential DL methods such as Long Short-Term Memory (LSTM) have yielded promising results in GIScience and remote sensing studies. However, the nature of geospatial datasets selected for use with these methods has been shown impactful to method performance. With LCC processes propagating relatively slow changes over time and errors resulting from classification procedures influencing model performance, it is unknown whether such datasets are compatible with the LSTM method. As such, the main objective of this study is to explore the capacity of LSTM to forecast patterns that have emerged from LCC dynamics given varying temporal resolutions, persistent land cover classes, and integration of classification confidence layers. Stacked LSTM modeling approaches are applied to geospatial datasets focused on the province of British Columbia, Canada. The 17-year MODIS land cover data selected was reclassified to four major LC classes. The evaluation considers this dataset at four different temporal resolutions to demonstrate the significance of geospatial data characteristics on LSTM method performance. Results indicate that LSTM can be utilized for forecasting LCC patterns when there are few limitations on temporal intervals of the datasets provided. Furthermore, this study demonstrates heightened performance measures when fewer classes undergo changes. Including classification confidence data as ancillary input also demonstrated potential for improving scenarios where the number of timesteps or temporal resolution is limited.

3.2. Introduction

Land Cover Changes (LCCs) are typically slow changes occurring across Earth's surface over long periods of time (Geist et al. 2006). The aggregated effects of changes have global implications, contributing to local, regional, and global climate changes, loss of biodiversity and, ultimately, disturbing the capacity of systems to sustain people (Lambin et al. 2001; Findell et al. 2017). Therefore, analyzing and representing LCC processes are important tasks in various disciplines such as geography (Meyer and Turner 1996), hydrology (Yang, Long, and Bai 2019), and climatology (Chase et al. 2000).

Previous studies have introduced and assessed methodologies for forecasting LCC. Such models have included cellular, agent-based, data-driven, and hybrid modeling approaches (Ren et al. 2019). In particular, data-driven methods including Machine Learning (ML) approaches have been increasingly considered for LCC forecasting (Patil et al. 2017). The goal of data-driven methodologies is for models to "learn" patterns existing in datasets while reducing the manual operations required to utilize the method (Bishop 2006). Applied to LC datasets, the aim is to use these automated, statistical methods to identify and analyze spatial patterns that have resulted from underlying processes of LCC over time. ML methods employed in LC simulations and assessments have previously included Neural Networks (NNs) (Maithani 2014), Decision Trees (DTs), and Support Vector Machines (SVMs) (Otukei and Blaschke 2010).

While ML approaches have demonstrated promising results for forecasting and detecting LCC, challenges include the infrequence of LC changes and obtaining appropriate labeled training datasets (Karpatne et al. 2016). Recent advances in a subfield of ML called Deep Learning (DL) have demonstrated the capacity of these increasingly complex models. Such models exhibit aptness for learning more complicated relationships existing in training datasets while simultaneously achieving substantial improvements in predictive performance measures (Karpatne et al. 2016). DL approaches such as Recurrent Neural Networks (RNNs) are best suited for sequential or timeseries data (Hochreiter and Schmidhuber 1997). An improved RNN architecture called Long Short-Term Memory has garnered increased attention for geospatial applications, with demonstrated aptitude to forecast LCC (Jia et al. 2017) and to classify LC (Sun, Di, and Fang 2018) by leveraging patterns obtained from timeseries data. However, the

effectiveness of these methods to forecast or classify change is impacted by the characteristics of the training dataset selected.

As observed in Chapter 2, the geospatial data characteristics selected for use with LSTM have important implications on the effectiveness of the method. Previous work employed a stacked LSTM modeling approach to assess the capacity of the method to forecast LCC as temporal resolution and LC class cardinality increased. With the sparsity of changed cells existing in typical LC datasets, the focus of the assessment was to evaluate how well the method could forecast transitioned cells. Using synthetic datasets, the work demonstrated that an increased number of timesteps, fewer LC classes, and finer temporal resolutions yielded the most capable models for LCC forecasting. However, the behaviour of this method is yet to be assessed in a scenario utilizing real-world geospatial datasets. Thus, this study aims to evaluate the effectiveness of data-driven approaches such as LSTM for real-world geospatial applications where the amount of change is typically small or occurring at slow rates over long periods of time. Additionally, it is aimed to demonstrate the importance and implications of the characteristics of geospatial datasets selected for use with this method, namely temporal resolution. Such an assessment of the method's sensitivity to temporal resolution has not yet been conducted in the presence of classification errors that may exist in real-world datasets.

It is expected that trends obtained using the real-world dataset will follow those observed in Chapter 2, where coarser temporal resolutions will have negative repercussions on method performance. Likewise, it is expected that erroneous values due to classification procedures will impact the results. To further observe the effects of potentially inaccurate values, a hypothetical scenario is adapted from the original LC dataset to exhibit persistence in a single class. The classification confidence layer associated with the original LC data product will be incorporated to a secondary modeling approach applied to scenarios involving real-world and hypothetical scenarios. By including this ancillary data layer, the aim is to reduce epistemic imperfection and determine if improvements can be obtained by indicating potential at each cell to contain erroneous values (Boulila, Ayadi, and Farah 2017). Thus, the three main objectives of this study are (1) to evaluate the sensitivity of the method to varying temporal resolutions, (2) to assess the implications of persistent LC classes in a hypothetical scenario, and (3) to examine potential improvements by providing classification confidence layers as model input in both real-world and hypothetical scenarios.

3.3. Theoretical Background

RNNs were introduced as a variant of traditional NN modeling approaches, devised for sequential data (Hochreiter and Schmidhuber 1997). To do this, a recurrent connection was added to traditional neurons, allowing information from previous elements in timeseries data to be propagated when weight updates occur for subsequent elements observed by the network. The recurrent connection allowed information to be propagated through the entire timeseries input, with weights being updated with respect to the entire timeseries input. However, a major problem arose when these networks were utilized to learn dependencies across long input sequences. For instance, information pertaining to critical features from early in a sequence could not be connected to data elements occurring later in the sequence. This phenomenon ensues from the Vanishing (or exploding) gradient problem (Hochreiter 1998). This implies that network weights tend toward either very small (vanishing) or very large (exploding) values, negating the ability of the network to learn important information as weight updates become infinitesimal or massive.

3.3.1. Long Short-Term Memory

Long Short-Term Memory (LSTM) is an improved variation of the traditional RNN architecture (Hochreiter and Schmidhuber 1997). With internal memory cells and gating functions controlling the propagation of information through a unit, LSTMs have proven capable in mitigating the effects of the Vanishing Gradient Problem that was detrimental to earlier RNN implementations (Ball, Anderson, and Chan 2017). The *input gate* (i_t), *forget gate* (f_t), *output gate* (o_t), and *input modulation gate* (g_t) control the propagation of new information in, within, or out of the unit to be considered with the next element in the input sequence (Figure 3.1). While other LSTM variants exist, it has been determined in large-scale studies that the standard LSTM architecture performance remains most effective (Greff et al. 2017). The following equations have been obtained from Donahue et al. (2015).

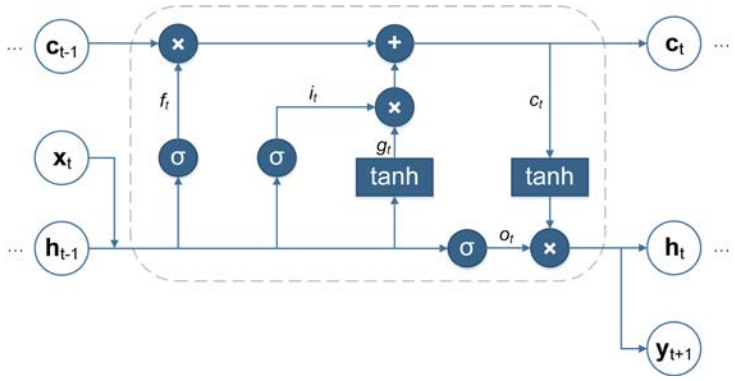


Figure 3.1. An LSTM unit (Figure adapted from Rußwurm and Körner (2017) and "Understanding LSTM Networks" (2016)).

When an input sequence is provided, (x_0, x_1,…, x_{t-1}, x_t), a hidden state, h_{t-1}, is either initialized at the start of the sequence or propagated from a computation considering a previous input sequence element. The amount of data from the next element in the input sequence, x_t, and how much of the hidden state from the previous timestep, h_{t-1}, should be committed to the internal memory cell c_t is determined by the *input gate* (i_t). How much information propagated through this gate is specified as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \qquad (1)$$

where the sigmoid function (σ) limits resulting values to the range of (0,1). W_{xi} and W_{hi} are learnable weight matrix parameters and b_i is a learnable bias parameter (Kratzert et al. 2018).

Next, a critical component of the LSTM unit is called a *forget gate* (f_t) (Ball, Anderson, and Chan 2017). By using gating functions to "forget" information, this component allows the network to ignore non-critical information during training procedures (Gers 1999; Kratzert et al. 2018). The vector resulting from the forget gate is denoted as follows:

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \qquad (2)$$

where the sigmoid function (σ) limits resulting values to the range of (0,1). W_{xf} and W_{hf} are learnable weight matrix parameters and b_f is a learnable bias parameter.

The next component of the LSTM cell is called the *output gate* (o_t), which mitigates the degree to which the value or state stored in the cell's internal memory, c_t, should be

propagated to the new hidden state, h_t, to be computed. The behaviour of the *output gate* is represented as follows:

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \qquad (3)$$

where W_{xo} and W_{ho} are learnable weight matrix parameters and b_0 is a learnable bias parameter.

Utilizing a hyperbolic tangent function, the *input modulation gate* (g_t) scales input values x_t and h_t before all or part of the resulting value is committed to the cell's internal memory in (5). The output of the *input modulation gate* is computed as:

$$g_t = \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \qquad (4)$$

where W_{xc} and W_{hc} are learnable weight matrix parameters and b_c is a learnable bias parameter.

The cell's internal memory is then updated, using the outputs of the *forget gate* (f_t), the previous internal memory state (c_{t-1}), *input gate* (i_t), and *input modulation gate* (g_t):

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \qquad (5)$$

where element-wise multiplication is symbolized by $\odot$.

Finally, the new hidden state is computed as follows:

$$h_t = o_t \odot \tanh(c_t) \qquad (6)$$

where h_t becomes the next h_{t-1}, and the next element in the input sequence is considered.

3.3.2. LSTM for Geospatial Applications

LSTM approaches have been previously utilized for simulating patterns resulting from dynamic geospatial systems (Lyu, Lu, and Mou 2016; Kong et al. 2018; Zhang et al. 2017; Liu et al. 2018). LSTM has been leveraged to learn LC changes in transfer learning applications (Lyu, Lu, and Mou 2016). It proved effective in learning changes in binary and multi-class LC datasets created for three variable-sized study areas focused on different

cities in China. LSTM has also been employed to reveal natural disturbances such as fires or floods, or human disturbances such as deforestation and urban growth, using satellite image timeseries datasets from a moderate-resolution imaging spectroradiometer (MODIS) data product (Kong et al. 2018). LSTM has also been effectively utilized in short- and long-term forecasts of sea surface temperature (Zhang et al. 2017). Using geographic datasets pertaining to weather, pollution, and influenza spread, LSTM has also been successfully applied in prediction of influenza propagation in the state of Georgia in the United States (Liu et al. 2018).

Though prior research has made evident that LSTM is useful for geospatial applications, the implications of varying temporal resolution as explored in Chapter 2 have not yet been applied to real-world datasets. The goal of this study is to evaluate the effectiveness of LSTM networks for LCC forecasting in real-world and hypothetical scenarios. Considering variable geospatial properties such as temporal resolution and the number of data layers obtainable to train LSTM models, the aim is to quantify the repercussions of changing temporal resolution in a real-world application and a hypothetical scenario derived from the real-world dataset. In contrast with the study applied to synthetic data developed in Chapter 2, real-world LC datasets feature inevitable classification errors, further impacting method performance. In addition to changing temporal resolution, this study intends to demonstrate the impacts of persistent LC classes and to determine if improved performance can be obtained using available classification confidence data.

3.4. Methods

3.4.1. Study Area and Datasets

The “MODIS Terra+Aqua Combined Land Cover product” was first obtained, featuring global annual land cover data from 2001 to 2017 (Friedl, M., Sulla-Menashe 2015). This dataset features 13 scientific data layers, including land cover, surface hydrology, and classification confidence layers. This dataset features 17 LC classes using the “MCD12Q1 International Geosphere-Biosphere Programme (IGBP)” classification system (Sulla-Menashe and Friedl 2018). It features 500-meter spatial resolution with data layers available annually from 2001 to 2017. Additional details regarding this dataset are shown in Table 3.1. The number of timesteps, temporal resolution, and numerous

geospatial data layers available in the selected MODIS dataset motivated the selection, despite coarse spatial resolution inhibiting studies at smaller spatial scales. Thus, the data is further processed to consider the only province of British Columbia.

Table 3.1. Overview of original MODIS land cover dataset characteristics.

Name	Details
Data Product	"MODIS Terra+Aqua Combined Land Cover product" global land cover dataset, featuring land use layers, surface hydrology, and classification confidence layers (13 scientific data layers)
Dataset Source URL	https://lpdaac.usgs.gov/products/mcd12q1v006
Coordinate System	Sinusoidal
Spatial Resolution	500m
Temporal Resolution	Yearly
Spatial Extent of Original Data	Global Coverage
Temporal Extent	2001-01-01 to 2017-12-31 (17 timesteps)
Land Cover Layer	"Land Cover Type 1: Annual International Geosphere-Biosphere Programme (IGBP) classification" (Friedl, M., Sulla-Menashe 2015)
Land Cover Confidence Layer	"LCCS1 land cover layer confidence" (Friedl, M., Sulla-Menashe 2015), with assessments recorded as percentages for each cell
Number of Land Cover Classes	17 LC classes using the "MCD12Q1 International Geosphere-Biosphere Programme (IGBP)" classification system
Data Format	HDF-EOS
Data Acquisition Tools	LP DAAC2Disk – An executable required to directly download MODIS data from the Land Processes Distributed Active Archive Center (https://lpdaac.usgs.gov/data_access/daac2disk)

First, using the Geospatial Data Abstraction Library (v2.2.4), the desired 17-class LC data layer and respective classification confidence layer were obtained and combined to create a multidimensional mosaic dataset for each timestep (GDAL/OGR contributors 2019). The raster mosaics were then extracted and re-projected to the NAD 1983 BC Environment Albers projected coordinate system with the provincial boundary of British Columbia, Canada obtained from the 2016 Canadian Census Boundary Files ("Boundary Files, 2016 Census" 2016). These operations were conducted using the data management tools available in Esri's ArcGIS Pro (v2.4.0) and applied to both the LC and classification confidence layers (Esri 2017a). When using the "Project Raster" tool available in ArcGIS

Pro (Esri 2017b), the “Nearest Neighbor” resampling technique was used as “it is suitable for discrete data, such as land cover” (Esri 2017b). The nearest neighbor resampling technique is utilized for the discrete and continuous datasets to maintain the related confidence layer data with the respective LC class for each cell (Esri 2016). Different resampling techniques including “bilinear interpolation” and “cubic convolution” are only relevant for continuous datasets (Esri 2016).

To integrate findings from Chapter 2 pertaining to favourable geospatial dataset characteristics, classes are aggregated to form four LC classes to improve model performance. The 17 available LC classes have been aggregated to forest, anthropogenic areas, non-forest areas, and water as presented in Table 3.2. Details of the original LC classes are shown in Table 3.3. This reclassification procedure was conducted in Esri’s ArcGIS Pro (Esri 2017a). The new aggregated class labels were named as per Voight et al. (2019). Datasets were subsequently resampled to 1 km spatial resolution to reduce the computation time required (Figure 3.2). Confidence data layers have been processed to this spatial resolution as well, featuring continuous percentage values indicating classification confidence (Figure 3.3).

Table 3.2. Detailed LC class composition following reclassification procedures to produce a four-class LC dataset.

Aggregate Class Name	Forest	Anthropogenic Areas	Non-Forest	Water
Original Class Names from the “MCD12Q1 International Geosphere-Biosphere Programme (IGBP)” classification system (Sulla-Menashe and Friedl 2018)	• Evergreen Needleleaf Forests • Evergreen Broadleaf Forests • Deciduous Needleleaf Forests • Deciduous Broadleaf Forests • Mixed Forests	• Croplands • Urban and Built-up Lands • Cropland/Natural Vegetation Mosaics	• Closed Shrublands • Open Shrublands • Woody Savannas • Savannas • Grasslands • Permanent Wetlands • Permanent Snow and Ice • Barren	• Water Bodies

Table 3.3. "MCD12Q1 International Geosphere-Biosphere Programme (IGBP) legend and class descriptions" from Sulla-Menashe and Friedl (2018).

Name	Value	Description
Evergreen Needleleaf Forests	1	Dominated by evergreen conifer trees (canopy >2m). Tree cover >60%.
Evergreen Broadleaf Forests	2	Dominated by evergreen broadleaf and palmate trees (canopy >2m). Tree cover >60%.
Deciduous Needleleaf Forests	3	Dominated by deciduous needleleaf (larch) trees (canopy >2m). Tree cover >60%.
Deciduous Broadleaf Forests	4	Dominated by deciduous broadleaf trees (canopy >2m). Tree cover >60%.
Mixed Forests	5	Dominated by neither deciduous nor evergreen (40-60% of each) tree type (canopy >2m). Tree cover >60%.
Closed Shrublands	6	Dominated by woody perennials (1-2m height) >60% cover.
Open Shrublands	7	Dominated by woody perennials (1-2m height) 10-60% cover.
Woody Savannas	8	Tree cover 30-60% (canopy >2m).
Savannas	9	Tree cover 10-30% (canopy >2m).
Grasslands	10	Dominated by herbaceous annuals (<2m).
Permanent Wetlands	11	Permanently inundated lands with 30-60% water cover and >10% vegetated cover.
Croplands	12	At least 60% of area is cultivated cropland.
Urban and Built-up Lands	13	At least 30% impervious surface area including building materials, asphalt, and vehicles.
Cropland/Natural Vegetation Mosaics	14	Mosaics of small-scale cultivation 40-60% with natural tree, shrub, or herbaceous vegetation.
Permanent Snow and Ice	15	At least 60% of area is covered by snow and ice for at least 10 months of the year.
Barren	16	At least 60% of area is non-vegetated barren (sand, rock, soil) areas with less than 10% vegetation.
Water Bodies	17	At least 60% of area is covered by permanent water bodies.
Unclassified	255	Has not received a map label because of missing inputs.

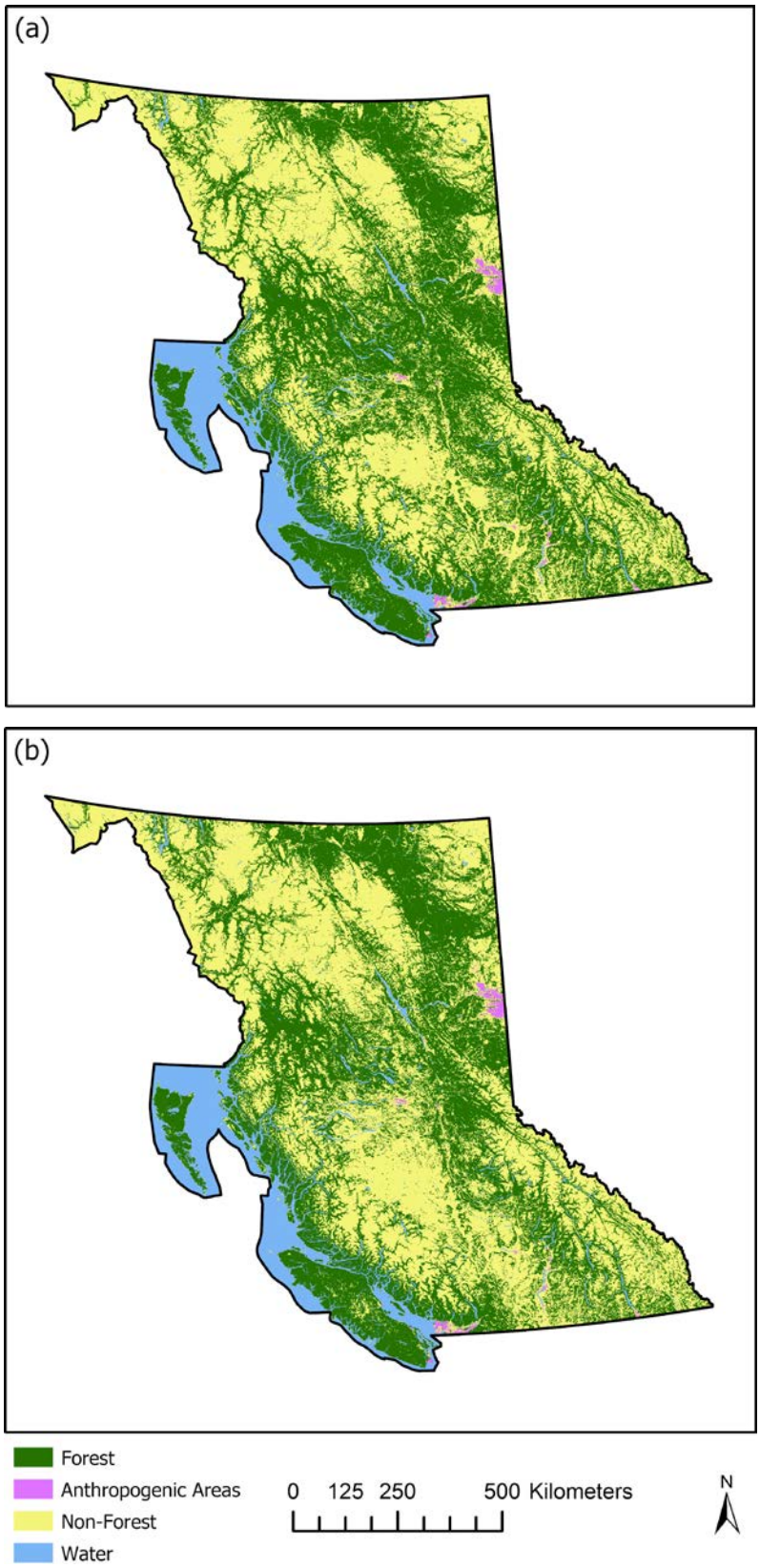


Figure 3.2. **Land cover maps of British Columbia generated for years: (a) 2001 and (b) 2017.**

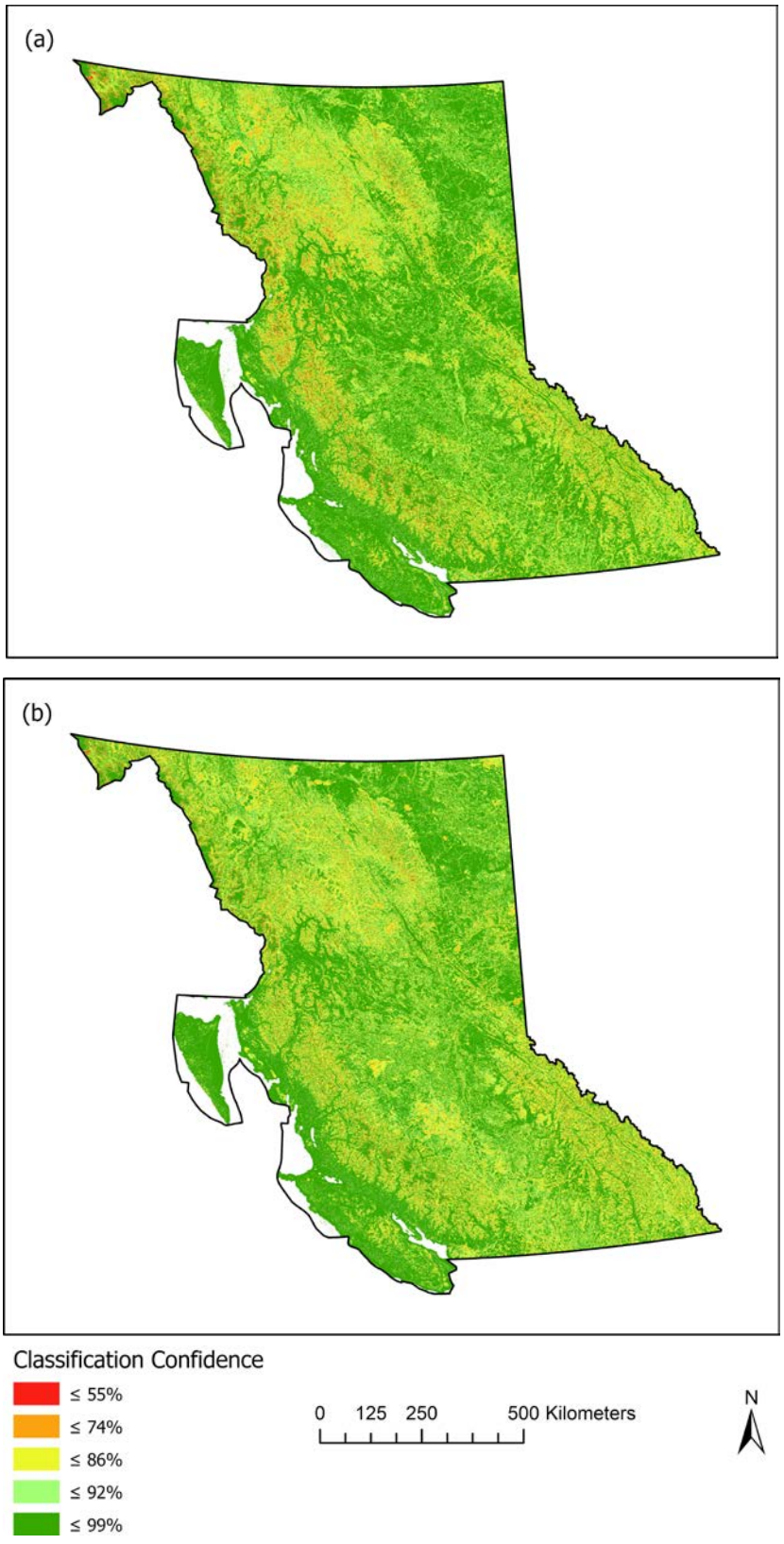


Figure 3.3. Examples of maps using classification confidence data for years: (a) 2001 and (b) 2017.

For the purpose of investigating the performance of the LSTM method on a persistent class, a hypothetical scenario has been created. The water class has been chosen to become a persistent class given the observed fluctuation of its boundaries through all data layers that potentially are due to classification errors. For example, the number of cells denoted as water abruptly changes from 90,963 to 94,290 between 2016 to 2017 (Figure 3.4). To make the water class persistent through time, all water cells present in the study area from 2002 to 2017 were converted to “No Data.” Next, the cells containing water in the study area in 2001 were overlaid to the same location from 2001 through 2017. Cells that were occupied by water in 2002 to 2017 that were not water cells in 2001 remain “No Data” and are thus excluded in the model evaluation procedure. The persistent water datasets for 2001 and 2017 have been shown in Figure 3.5. These procedures were enabled by reading data using the Geospatial Data Abstraction Library and modifying the data programmatically using the Python programming language (v3.6.5) (van Rossum 2016).

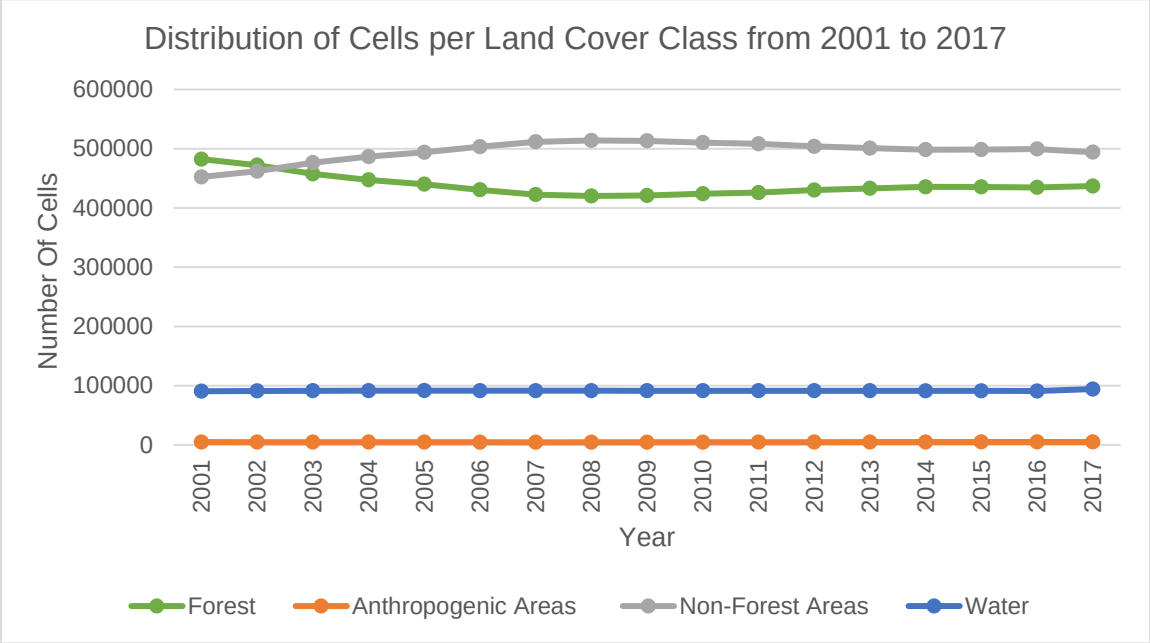


Figure 3.4. Class membership for each land cover class per year in the British Columbia study area.

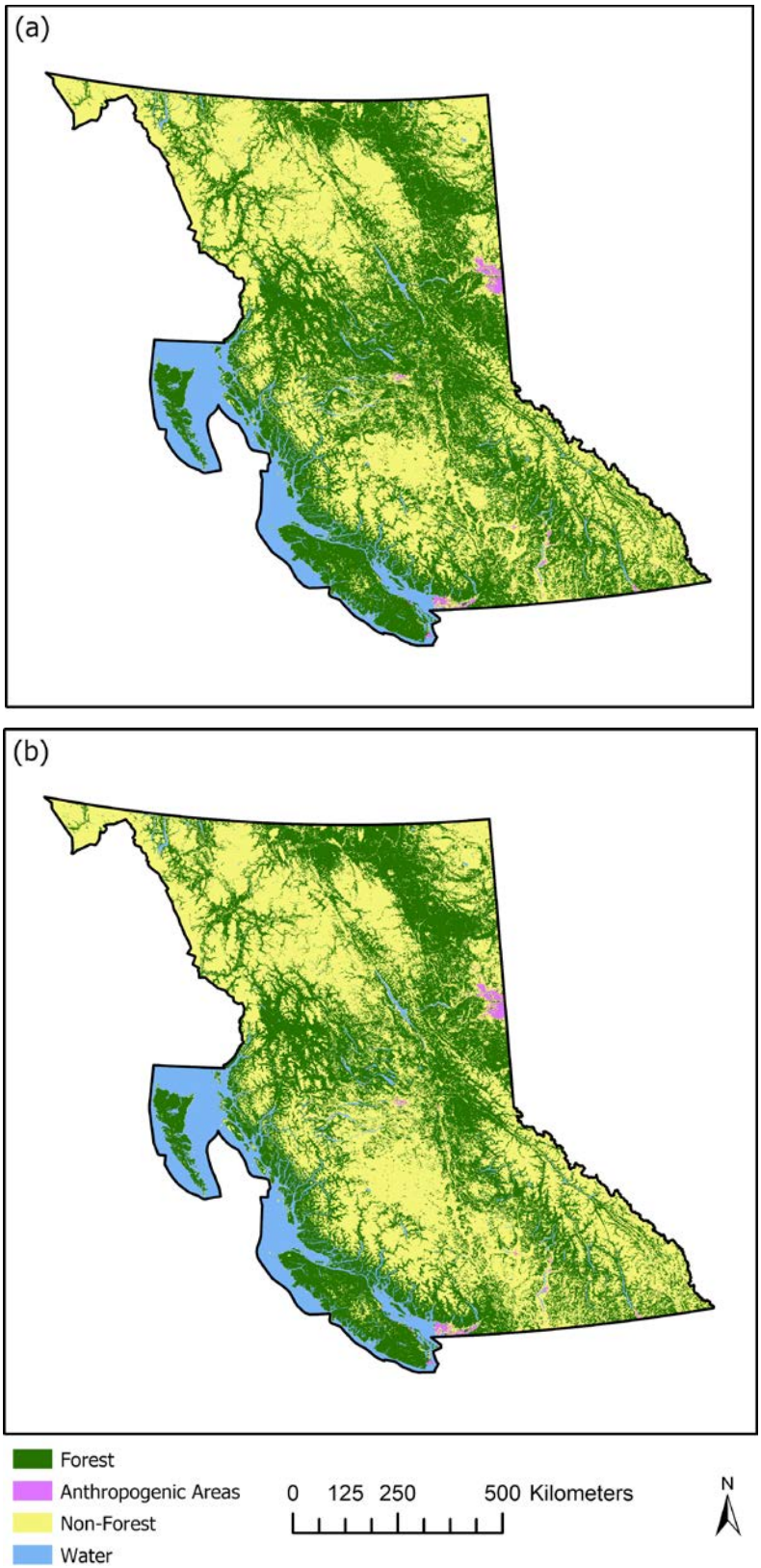


Figure 3.5. Land cover maps of British Columbia for the hypothetical scenario featuring a persistent water class for years: (a) 2001 and (b) 2017.

3.4.2. Training and Testing Procedures

Following the data preparation procedures, datasets must be further processed to create training and test sets for use with the LSTM method. Input sequences composing the training and testing datasets are created using a moving-window approach (Kong et al. 2018). Training sets are thus denoted as ($x_0, x_1, x_2, \ldots, x_{T-3}$), while the target LC class is denoted by (y_{T-2}). Input sequences in the test set are denoted as ($x_1, x_2, x_3, \ldots, x_{T-2}$), while the target LC class is denoted by (y_{T-1}). The training and test sets are dependent on the temporal resolution being utilized with the currently trained model. Table 3.4 presents the years used in training and testing procedures with respective temporal resolutions. The subsequent timestep to be forecasted, or the training targets (y_{T-2}) and testing targets (y_{T-1}), have been underlined (Table 3.4). Both training and testing datasets are one-hot encoded to represent LC classes at each timestep (Lyu, Lu, and Mou 2016; Rußwurm and Körner 2018). The LC classification confidence layer is processed in the same way, without the one-hot encoding procedure. The confidence layers are comprised of continuous percentages ranging from 0 to 100.

Table 3.4. Years used to compose training and test datasets considering the four temporal resolution scenarios.

Temporal Resolution (years)	Years used in model training (where the last entry in each input sequence is the training target)	Years used in model testing (where the last entry in each input sequence is the testing target)
1	2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016	2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017
2	2001, 2003, 2005, 2007, 2009, 2011, 2013, 2015	2003, 2005, 2007, 2009, 2011, 2013, 2015, 2017
4	2001, 2005, 2009, 2013	2005, 2009, 2013, 2017
8	2001, 2009	2009, 2017

The overall number of cells per class considering the real LC dataset has been shown in Figure 3.5. Although there is a multitude of ways of composing the training sets, due to the scarcity of cells undergoing change (Karpatne et al. 2016), it is necessary to apply an improved sampling strategy to form the training sets used in each scenario. This means that strategies such as random sampling or obtaining equal counts from each class are disadvantageous for phenomena such as LCC. Samardžić-Petrović et al. (2016) propose a balanced sampling strategy, showing the benefits of equal counts of persistent cells and changed cells used to compose the training set. Changed cells are denoted as such if they have undergone one or many changes between x_0 and y_{T-2}. However, due to the large discrepancy between changed and persistent cell counts, persistent cells cannot be simply randomly sampled. Therefore, all changed cells are added to the training data set, while persistent cells are sampled at random across the entire study area while maintaining the original distributions of classes found in the entire set of persistent cells (Samardžić-Petrović et al. 2016, 2017). This technique impacts the number of inputs available for training for each temporal resolution. That is, if changes occurred during timesteps unavailable in the creation of the training set for the temporal resolution being considered, the cell would be marked as persistent. This sampling procedure is also applied to the derived hypothetical scenario, ignoring the additional cells marked as "No Data." Due to the increased number of cells ignored in the hypothetical scenario, training and testing datasets are smaller in these scenarios.

Previous methods have utilized subsets of available datasets for testing, following suit with training dataset creation procedures (Samardžić-Petrović et al. 2016, 2017). However, in this study, the balanced sampling strategy has been employed for only the training dataset creation procedure. To create the test set, all cells (except for those containing "No Data") are considered in the evaluation of the entire forecasted map. That is, the number of cells forecasted correctly as changed or persistent are calculated for all cells available in the forecasted map, in both the real and hypothetical scenarios.

3.4.3. Model Specifications

The LSTM-based approach for forecasting LCC includes a stacked LSTM model (Figure 3.6). Figure 3.6 depicts the model considering training input sequences x_0 to x_{14}, considering one-year temporal resolution. The "Dense" layer refers to a fully-connected neural network layer used as the output layer. By stacking LSTM layers, these models

have been demonstrated to have improved capacity to capture increasingly complex relationships subsisting in input datasets (Pascanu et al. 2014; Hermans and Schrauwen 2013). In this study, three LSTM layers are used in a stacked modeling approach accepting categorical LC input sequence data. Each LSTM layer is composed of 128 neurons per layer. The input layer is compatible with one-hot encoded input sequences. Between each layer, various Dropout regularization factors were also tested before settling on a factor of 0.5 between each layer (Srivastava et al. 2014; Kong et al. 2018). The Dropout factor controls the probability of neurons being discarded during the training procedure. For instance, if a factor of 0.5 is applied between each layer, the probability that the information from a neuron is “dropped” becomes 50%. This simple tactic has proven effective in preventing overfitting and improving representations learned by RNNs (Pham et al. 2014).

To incorporate the classification confidence layer, this configuration is concatenated with an additional input layer and LSTM layer. This branched configuration is required to provide mixed input types to models, a prevalent approach in applications such as image captioning (Karpathy and Fei-Fei 2017; Wu et al. 2018). The model branch accepts the confidence layer as input. The output of this branch is concatenated with output resulting from the stacked LSTM used to consider the LC class sequences. This configuration is demonstrated in Figure 3.7. The application of the Dropout regularization terms remains the same in the model branch considering LC input sequences.

Models were developed using Python (v3.6.5) and the Keras API (v2.2.0) (Chollet 2015). The Keras API aids in simplifying DL model prototyping workflows while affording users the functionality of Google’s TensorFlow GPU implementation (v1.8.0) (Chollet 2015). TensorFlow is “an open-source machine learning framework” that provides users advanced functionality to fine-tune ML and DL models (Abadi et al. 2016). The optimization method used was the Adaptive Moment Estimation or “Adam” algorithm (Kingma and Lei Ba 2015) due to its demonstrated success and robustness to model hyperparameters. Categorical cross-entropy is utilized as the objective function to accommodate the multi-class data being used with this method (Rußwurm and Körner 2017).

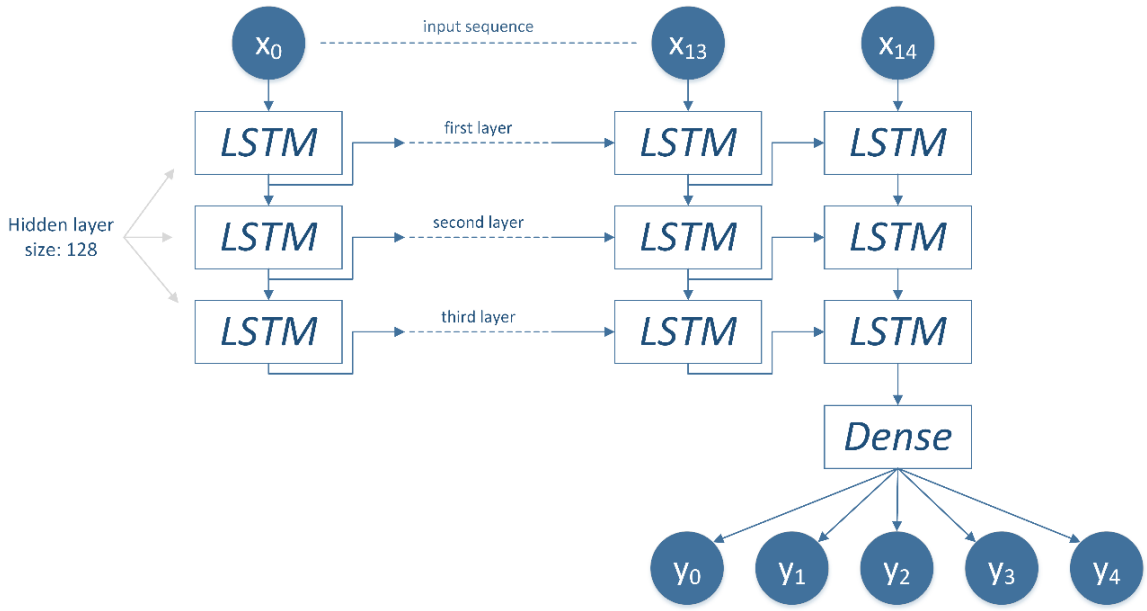


Figure 3.6. Stacked LSTM Model with a land cover input sequence example using 1-year temporal resolution.

The Softmax activation function (Bishop 2006) was employed in the output layer. This ensures a vector of probabilities corresponding to each class label for each cell is output from the model. This activation function is commonly used with multi-class classification and predictive models (Bishop 2006; Ienco et al. 2017). In this study, there are four LC classes. The fifth class (y_0 in Figures 3.6 and 3.7) denotes a "NoData" option which is unrepresented in the training and test sets. The maximal number of epochs models are enabled to train for is 1,000 epochs, with early-stopping callbacks utilized to prevent overfitting (Raskutti, Wainwright, and Yu 2011). Early stopping terminates model fitting when there have been no improvements to a model's objective function within a specified number of epochs. Models have been trained and tested using a NVIDIA GeForce GTX 1080 Ti GPU.

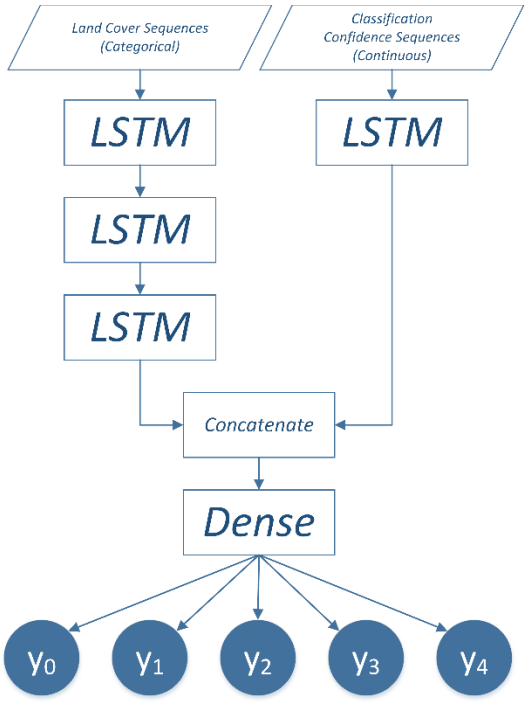


Figure 3.7. Stacked LSTM model configured for land cover and classification confidence input layers.

3.4.4. Experiment Overview

Using the original datasets that have undergone pre-processing procedures described in 3.3.1, two experiments are performed, including (1) training and evaluating the stacked LSTM model considering only the LC dataset (Figure 3.6) and (2) training and evaluating the mixed input stacked LSTM model considering both the LC dataset with the ancillary classification confidence layer (Figure 3.7). In the evaluation procedure, all cells apart from those assigned “No Data” are involved.

The hypothetical scenarios are used to assess whether performance may increase or decrease when a class remains persistent through all timesteps. In the hypothetical scenarios, the two experiments conducted using the persistent water dataset include (1) training and evaluating the stacked LSTM model considering the persistent water dataset (Figure 3.6) and (2) training and evaluating the mixed input stacked LSTM model considering the classification confidence layer (Figure 3.7). In the evaluations considering the persistent water dataset, a greater number of cells marked will be marked as “No Data” due to data creation procedures described in 3.3.1, implying more cells will be omitted during the evaluation procedure.

3.5. Results and Discussion

Constructing four models for each experiment, each model was trained with training sets featuring one-, two-, four-, and eight-year temporal resolutions, respectively. Training set composition has been shown in Table 3.5 for experiments involving the real-world datasets (Table 3.5a) and for the hypothetical scenario (Table 3.5b). The number of training samples for each temporal resolution option in Table 3.5a and 3.5b indicate that as the temporal resolution becomes coarser, the number of cells marked that would be observed changed becomes smaller with the lesser number of years available. Likewise, the number of persistent cells in the respective training sets becomes smaller to uphold the balanced sampling strategy posed in 3.4.2. Though the water class is persistent through all timesteps (Table 3.5b), the percentage of persistent water cells obtained for the respective training sets maintains the sampling strategy. The years composing the test set will vary for each modeling scenario due to changing temporal resolution. The timesteps involved in training and testing each model in all experiments is demonstrated in Table 3.4. To examine differences in forecasted outputs, metrics and maps are produced at the provincial extent. Additionally, a smaller spatial extent focused on the Central Okanagan region is selected for visual assessment (Figure 3.8). This data was extracted from the “Regional District Boundaries” file available for the province of British Columbia (Elections BC 2019).

To evaluate the method, cells are marked as changed if they have undergone a transition between 2001 and 2017. Likewise, the evaluation does not consider cells marked as “No Data.” This operation considers all timesteps and is then used to compare to the forecasted output generated from each of the models. Each model in every experiment is tested to forecast the LC geospatial data available for 2017 in both real-world and hypothetical scenarios, respectively. Evaluation metrics are considered per category by “cell-by-cell comparison,” indicating which cells were forecasted as persistent or changed (Ahmed et al. 2013).

Across all experiments, in utilizing the coarsest temporal resolution possible (eight-year temporal resolution), results obtained were poor and erratic (Sections 3.5.1 and 3.5.2). This scenario involves one input feature mapped to one output feature for training and testing. This behavior is consistent was observations made in Chapter 2, which showed poor performance with coarsest temporal resolutions at a slightly lesser degree.

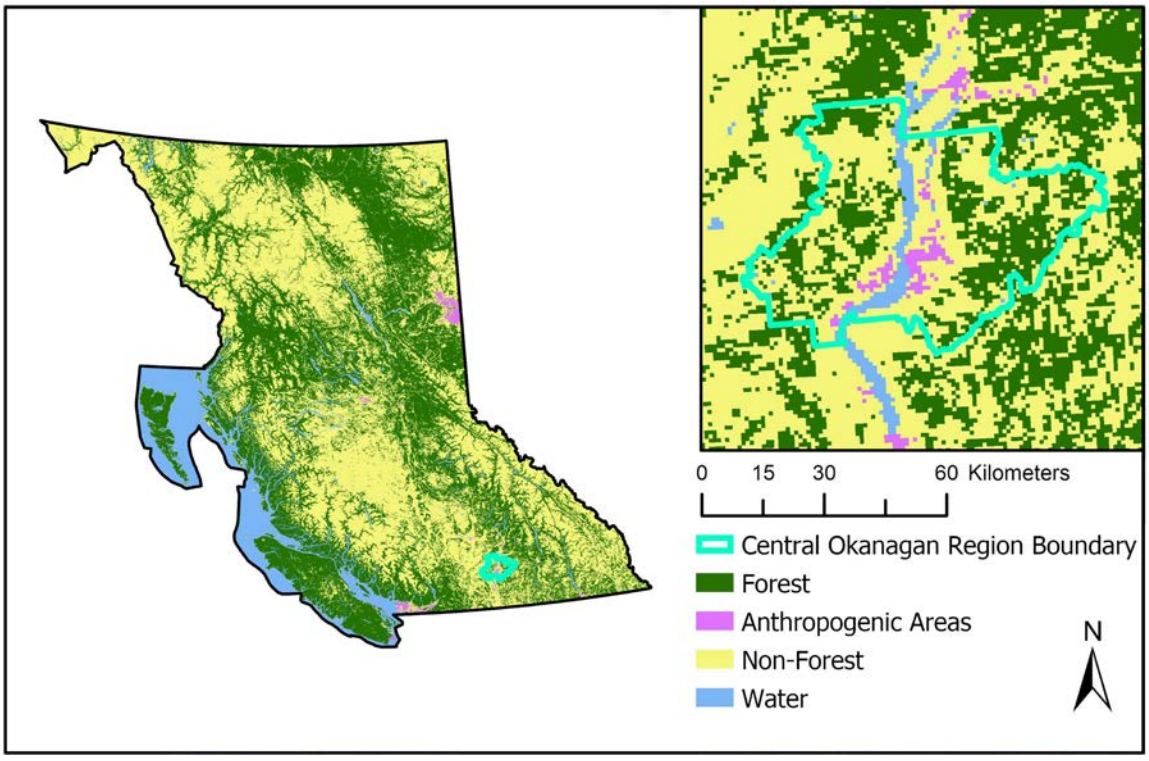


Figure 3.8. Maps created with actual land cover data for the Central Okanagan Region used for visual comparisons between forecasted outputs for 2017.

Table 3.5. Composition of each training set and for each temporal resolution. The number of cells belonging to each class has been shown, along with the original percentage of cells belonging to each class in the respective datasets for (a) the actual dataset and (b) the persistent water dataset.

(a)

Temporal Resolution	Number of Training Samples	Changed Cell Counts Number of Changed Cells *(% of changed cells)*				Persistent Cell Counts Number of Persistent Cells *(% of persistent cells)*			
		Forest	Anthro-pogenic Areas	Non-Forest Areas	Water	Forest	Anthro-pogenic Areas	Non-Forest Areas	Water
1	322278	65341 *(40.55%)*	1102 *(0.68%)*	94037 *(58.36%)*	659 *(0.41%)*	68493 *(42.51%)*	719 *(0.45%)*	75182 *(46.66%)*	16745 *(10.39%)*
2	307964	62129 *(40.35%)*	982 *(0.64%)*	89991 *(58.44%)*	880 *(0.57%)*	65618 *(42.61%)*	690 *(0.45%)*	71795 *(46.63%)*	15879 *(10.31%)*
4	274386	50080 *(36.50%)*	731 *(0.53%)*	85436 *(62.27%)*	946 *(0.69%)*	58835 *(42.89%)*	619 *(0.45%)*	63851 *(46.54%)*	13887 *(10.12%)*
8	204328	19457 *(19.04%)*	309 *(0.30%)*	81456 *(79.73%)*	942 *(0.92%)*	44204 *(43.27%)*	464 *(0.45%)*	47536 *(46.53%)*	9960 *(9.75%)*

(b)

Temporal Resolution	Number of Training Samples	Changed Cell Counts Number of Changed Cells *(% of changed cells)*				Persistent Cell Counts Number of Persistent Cells *(% of persistent cells)*			
		Forest	Anthro-pogenic Areas	Non-Forest Areas	Water	Forest	Anthro-pogenic Areas	Non-Forest Areas	Water
1	317754	64981 *(40.90%)*	1102 *(0.69%)*	92794 *(58.41%)*	N/A	67626 *(42.56%)*	711 *(0.45%)*	73931 *(46.53%)*	16610 *(10.45%)*
2	303742	61789 *(40.69%)*	982 *(0.65%)*	89100 *(58.67%)*		64814 *(42.68%)*	682 *(0.45%)*	70624 *(46.50%)*	15750 *(10.37%)*
4	270628	49746 *(36.76%)*	731 *(0.54%)*	84837 *(62.70%)*		58124 *(42.96%)*	613 *(0.45%)*	62804 *(46.41%)*	13772 *(10.18%)*
8	201628	19314 *(19.16%)*	309 *(0.31%)*	81191 *(80.54%)*		43707 *(43.35%)*	459 *(0.46%)*	46770 *(46.39%)*	9878 *(9.80%)*

3.5.1. Real-world Dataset Experiments

Using the real-world dataset, the models trained with one-year temporal resolution yielded highest counts of correctly simulated cells across the forest, anthropogenic areas, and non-forest area classes. This is exhibited by results including only the LC data layer (Figure 3.9), and the LC data layer with the confidence layer as ancillary data (Figure 3.10). This includes both changed and persistent cells belonging to the respective classes.

Similarly, the model trained with the finest temporal resolution forecasted the least number of changed cells incorrectly as persistent. The number of changed cells simulated as persistent increases with temporal resolution. It is also observed that the highest number of simulation errors exists in “changed cells simulated incorrectly as persistent” counts in both Tables 3.6 and 3.7.

The exemption to the aforementioned trends is the water class, exhibiting slightly increased counts of correctly simulated cells as temporal resolution becomes coarser. Overall, models performed poorly when forecasting water. Assumed to be mostly persistent, erroneous values may have factored in to poor performance for simulating this specific class. For instance, from 2016 to 2017, the number of cells occupied by water changes from 90,963 to 94,290 (Figure 3.5). This abrupt change occurring at the end of the available timeseries may be due to classification errors or discrepancies when annual data products were created.

Overall, the effects of increasing or decreasing temporal resolution were as expected. It was observed that method performance is impeded in scenarios involving coarser temporal resolutions. When forecasting land cover, the LSTM method forecasts a greater number of persistent cells than changed cells as temporal resolution becomes coarser. This is demonstrated in Table 3.6, where the number of persistent cells forecasted correctly increases as temporal resolution becomes coarser. The capacity of the models to forecast changed cells increases as temporal resolution becomes finer (Table 3.6). For instance, the model trained with the finest temporal resolution (one-year) produced the forecast with the most correctly transitioned cells for forest and anthropogenic areas. Across all scenarios, the models demonstrated their bias for forecasting persistent cells, despite the sampling regime instated. Therefore, it is observed that LSTM models are most effective when provided increased numbers of timesteps and finer temporal resolutions in this real-world application.

It was hypothesized that adding the classification confidence layer would mitigate the effects of erroneous values and thus enhance LCC forecasts when considering all temporal resolution options. However, results obtained using the mixed input models failed to significantly improve in scenarios involving finer temporal resolutions (Table 3.7). This is also evident in the map produced for the Central Okanagan region (Figure 3.11). By including the LC classification confidence layer, the capacity for the models to forecast

changes improved marginally as temporal resolution became coarser. For instance, Table 3.6 shows the model trained with eight-year temporal resolution data to forecast only two of the four possible classes, including the majority class (non-forest areas) and water cells. While still obtaining suboptimal results, models trained with both types of inputs forecasted some cells in forest and anthropogenic classes, albeit less than 1% in each class, respectively.

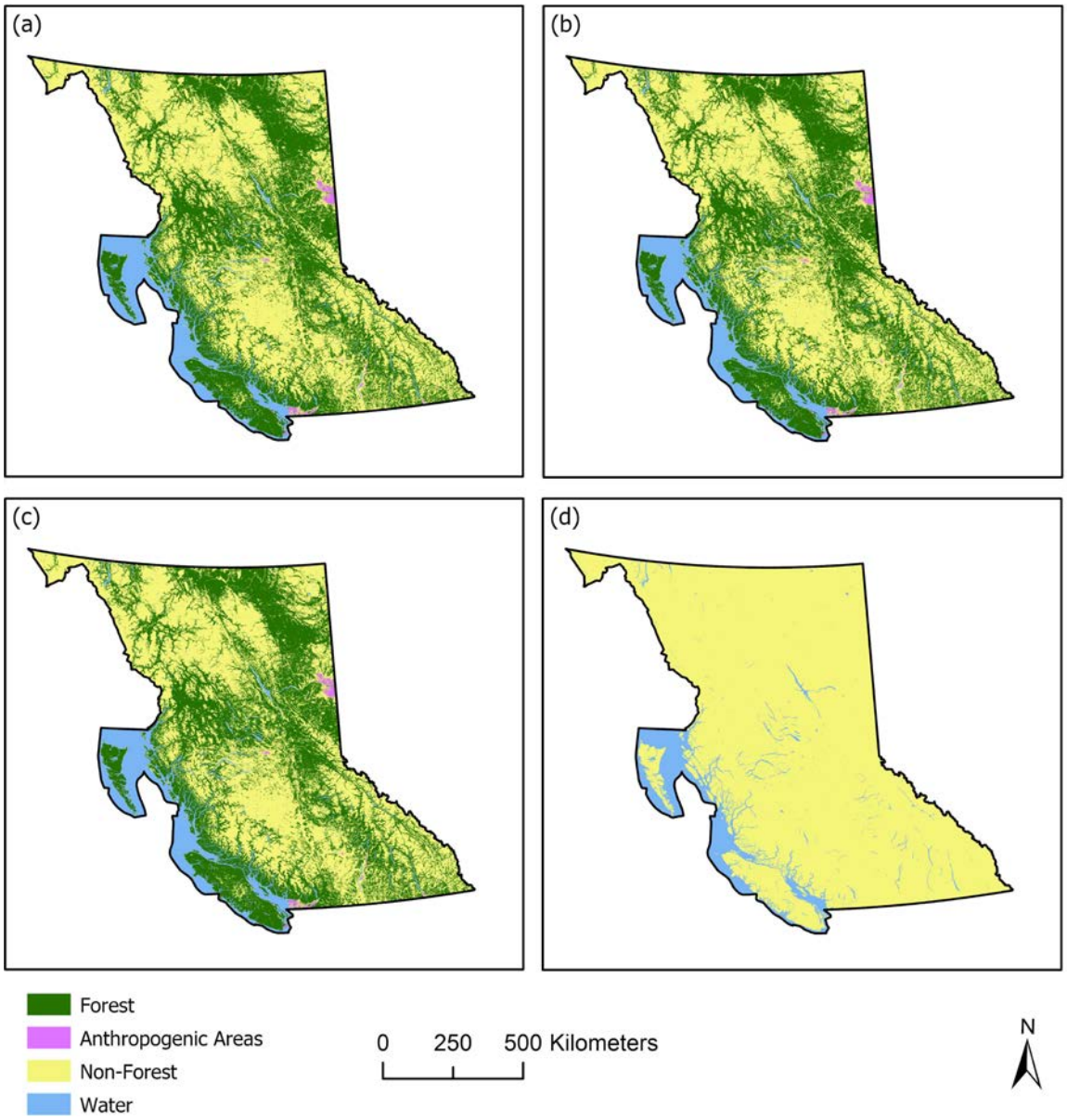


Figure 3.9. Land cover classes with actual data, and forecasted outputs obtained for year 2017 for model trained with (a) one-year, (b) two-year, (c) four-year, (d) eight-year temporal resolution datasets.

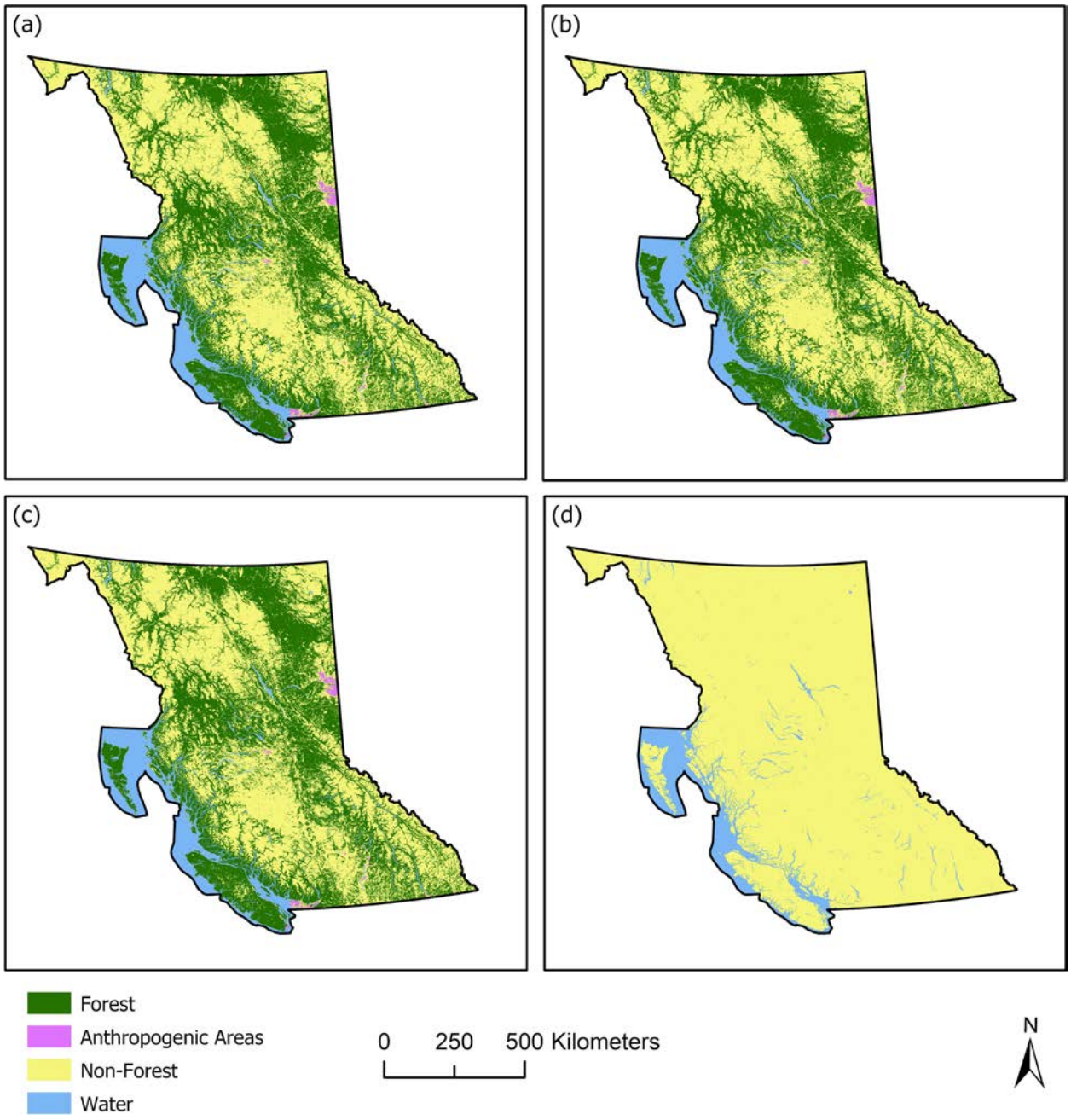


Figure 3.10. Land cover classes obtained for year 2017 with actual land cover and confidence data for model trained with (a) one-year, (b) two-year, (c) four-year, (d) eight-year temporal resolution datasets.

Table 3.6. Number of cells correctly and incorrectly simulated per class using the four temporal resolution options considering the real-world land cover dataset.

		Land Cover Class			
		Forest	Anthropogenic Areas	Non-Forest Areas	Water
Real # of Changed Cells from 2001 to 2017		40050	780	85481	3786
Real # of Persistent Cells from 2001 to 2017		396842	4145	408548	90504
Measure	Temporal Resolution (Years)				
Number of Cells Correctly Simulated as Changed	1	33037	593	77228	485
	2	32147	547	75191	550
	4	28192	412	70475	516
	8	0	0	85435	524
% of Cells Correctly Simulated as Changed	1	82.49%	76.03%	90.35%	12.81%
	2	80.27%	70.13%	87.96%	14.53%
	4	70.39%	52.82%	82.45%	13.63%
	8	0%	0%	99.95%	13.84%
Number of Cells Correctly Simulated as Persistent	1	388738	4093	404179	90342
	2	388437	4082	404161	90357
	4	384974	4047	403954	90357
	8	0	0	408235	90412
% of Cells Correctly persistent	1	97.96%	98.75%	98.93%	99.82%
	2	97.88%	98.48%	98.93%	99.84%
	4	97.01%	97.64%	98.88%	99.84%
	8	0%	0%	99.92%	99.90%
Number of Changed Cells Simulated as Wrong Change	1	138	3	181	5
	2	165	2	144	26
	4	193	2	119	44
	8	0	0	739	55
Changed Cells Simulated Incorrectly as Persistent	1	8643	139	9628	17
	2	10679	154	10452	40
	4	15370	200	14526	48
	8	0	0	43291	53
Persistent Cells Simulated Incorrectly as Changed	1	4158	155	8286	88
	2	4070	131	8562	239
	4	4291	102	12040	274
	8	0	0	401029	363

Table 3.7. Number of cells correctly and incorrectly forecasted per class using the four temporal resolution options with the actual land cover dataset combined with the land cover classification confidence layer.

		Land Cover Class			
		Forest	Anthropogenic Areas	Non-Forest Areas	Water
Real # of Changed Cells from 2001 to 2017		40050	780	85481	3786
Real # of Persistent Cells from 2001 to 2017		396842	4145	408548	90504
Measure	Temporal Resolution (Years)				
Number of Cells Correctly Simulated as Changed	1	33037	594	77228	502
	2	32145	542	75192	560
	4	28190	412	70478	553
	8	43	2	85387	524
% of Cells Correctly Simulated as Changed	1	82.49%	76.15%	90.35%	13.26%
	2	80.26%	69.49%	87.96%	14.79%
	4	70.39%	52.82%	82.45%	14.61%
	8	0.11%	0.26%	99.89%	13.84%
Number of Cells Correctly Simulated as Persistent	1	388731	4093	404172	90342
	2	388436	4069	404154	90356
	4	384819	4041	403860	90358
	8	1018	34	407562	90412
% of Cells Correctly persistent	1	97.96%	98.75%	98.93%	99.82%
	2	97.88%	98.17%	98.92%	99.84%
	4	96.97%	97.49%	98.85%	99.84%
	8	0.26%	0.82%	99.76%	99.90%
Number of Changed Cells Simulated as Wrong Change	1	138	3	176	5
	2	164	2	144	27
	4	193	2	120	49
	8	6	0	735	55
Changed Cells Simulated Incorrectly as Persistent	1	8643	139	9615	17
	2	10678	153	10450	40
	4	15362	200	14488	50
	8	45	2	43245	53
Persistent Cells Simulated Incorrectly as Changed	1	4158	155	8293	95
	2	4071	127	8577	249
	4	4295	102	12200	364
	8	671	2	399977	363

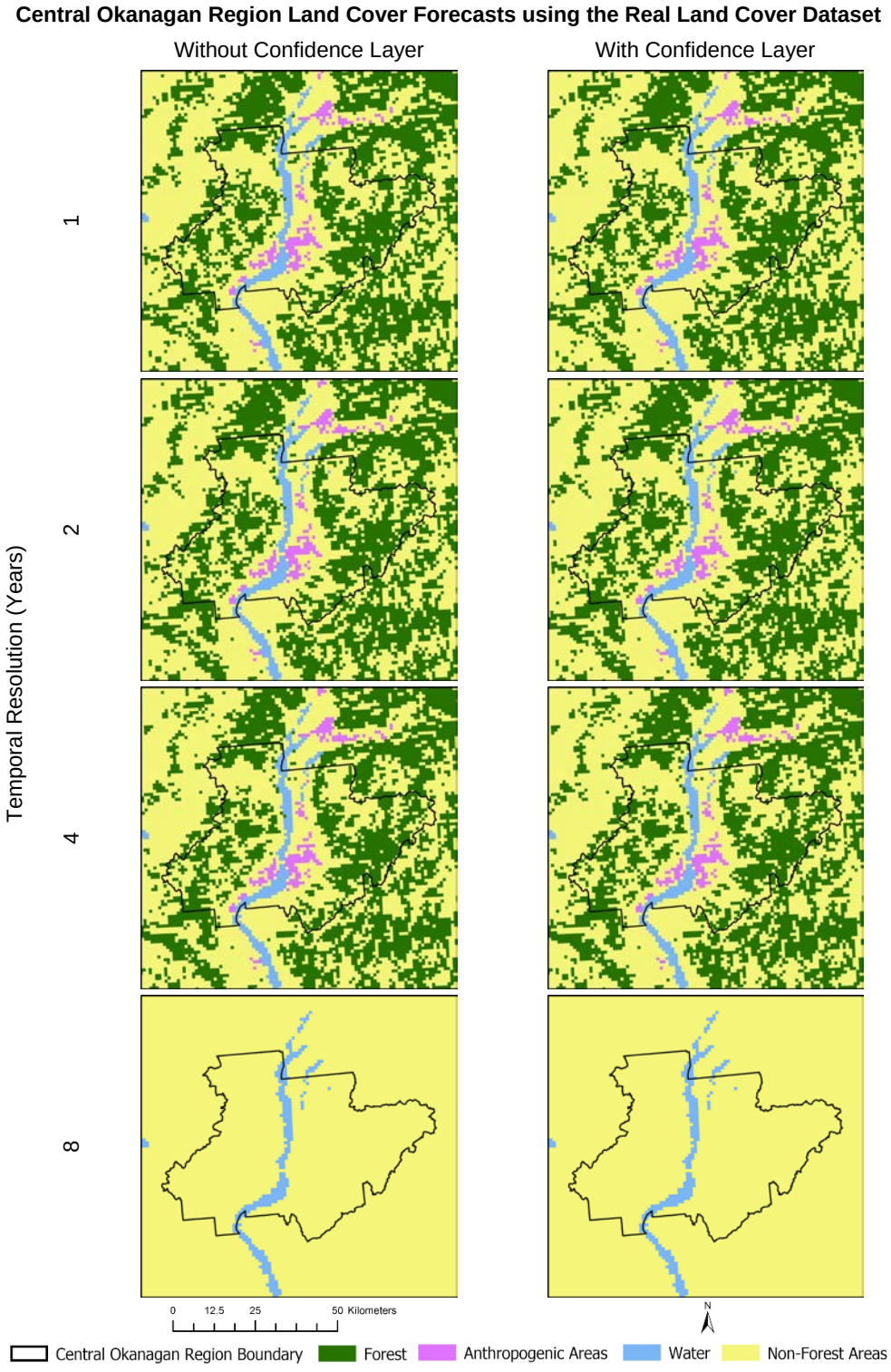


Figure 3.11. Comparison of land cover forecasts centred on the Central Okanagan region, British Columbia, using the real land cover dataset without and with classification confidence data.

3.5.2. Persistent Water Dataset Experiments

The persistent water dataset experiments consider the persistent water LC class developed from the real-world dataset, considering water cells from 2001 as persistent through all timesteps. The hypothetical scenario supports the observation that erroneous values pose significant detriments to forecasted results. Using this persistent water scenario, it was observed that study areas featuring LC classes that undergo slow or no changes greatly influence the method performance. Forecasted maps produced when considering the persistent water datasets have been shown in Figures 3.12 and 3.13.

Following suit with results obtained using the real-world datasets, the models trained with finer temporal resolutions obtained the most favourable results (Tables 3.8 and 3.9). In the case of the models considering the persistent water data layer with eight-year temporal resolution, it is observed that the model forecasted only non-forest areas and persistent water cells (Table 3.8). It is also observed that cells occupied by persistent water through all timesteps have been forecasted with no errors across all modeling scenarios considering both exclusively the persistent water land cover dataset as well as the mixed inputs.

Like results obtained when considering the real-world data, the addition of the classification confidence layer as input to the model increased model performance slightly when considering coarser temporal resolutions and fewer timesteps. It similarly had little to no effect on LCC forecasting performance in scenarios where finer temporal resolutions were considered. However, in the scenario considering eight-year temporal resolution, the addition of the classification confidence layer as model input contributed to the success in forecasting anthropogenic areas, with 92.6% of persistent anthropogenic areas forecasted correctly (Table 3.9). This can be viewed in Figure 3.14 at the smaller spatial extent.

As observed previously, the capacity of the method to forecast changes degrades as temporal resolution becomes coarser. An exemption to this trend is the number of non-forest areas correctly forecasted as changed (Tables 3.8 and 3.9). This is deemed a consequence of the bias toward the majority class, especially considering the “worst-case” scenario of eight-year temporal resolution.

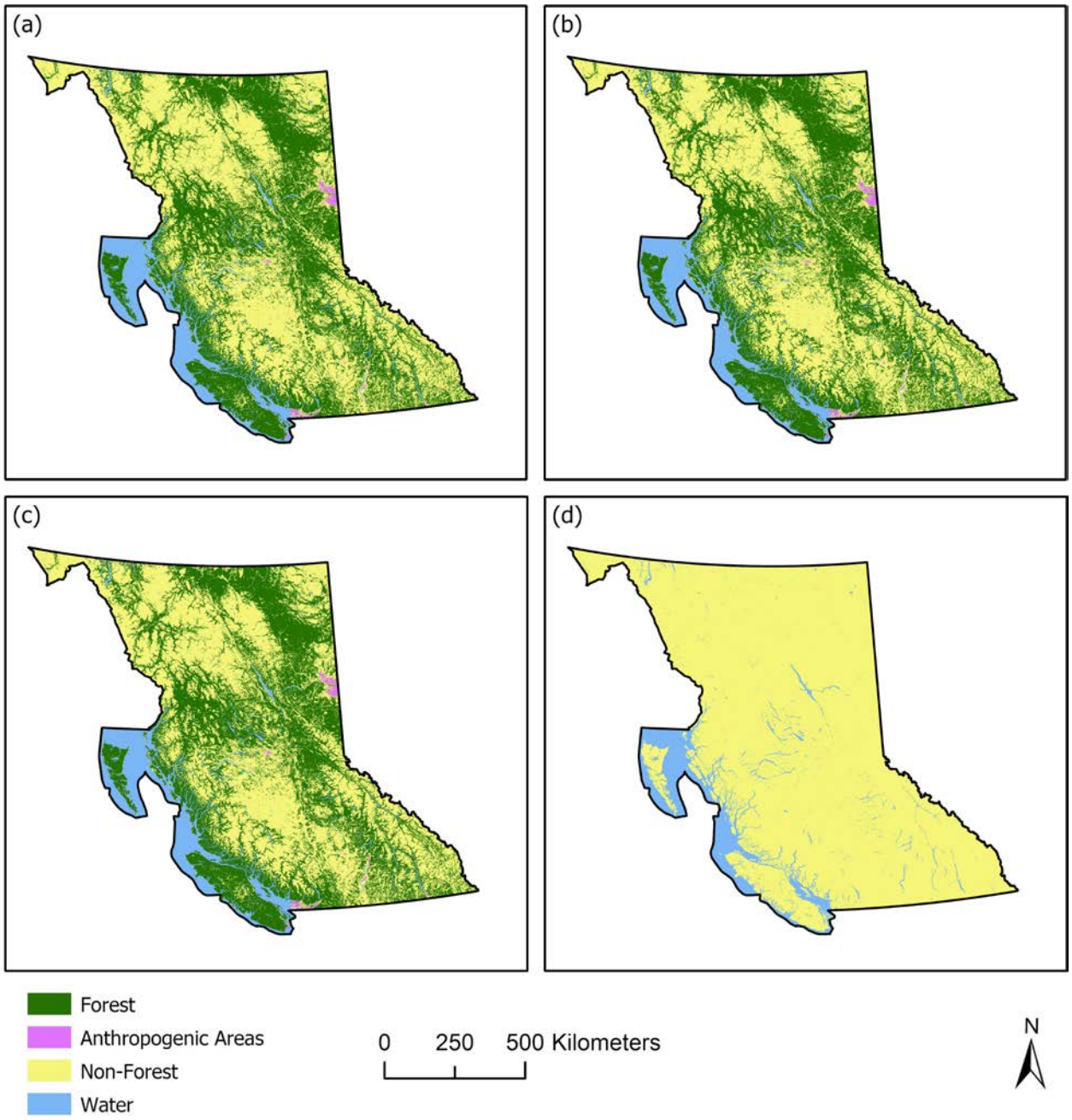


Figure 3.12. Forecasted land cover classes obtained for year 2017 in the hypothetical scenario for models trained with (a) one-year, (b) two-year, (c) four-year, (d) eight-year temporal resolution.

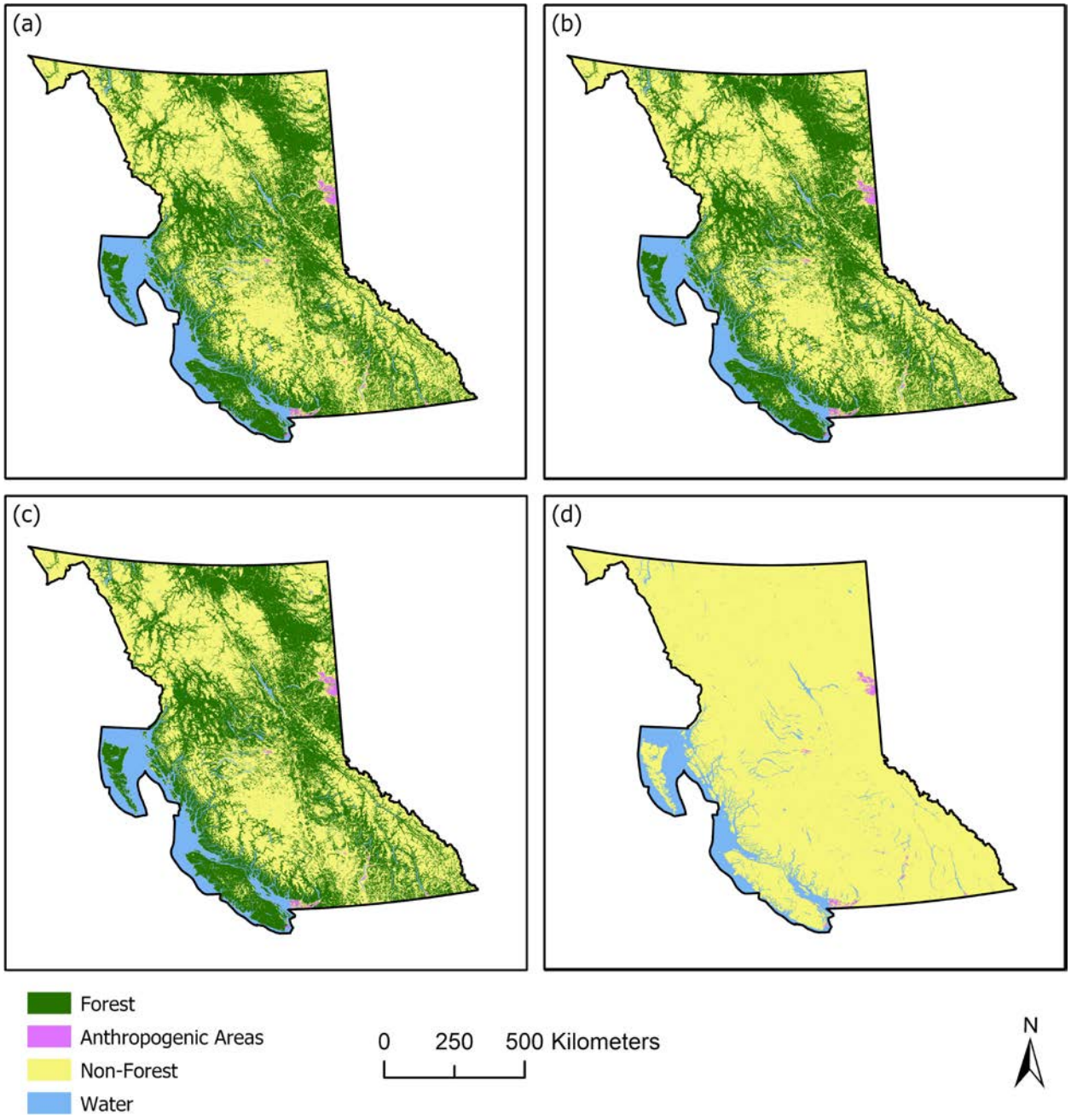


Figure 3.13. Forecasted land cover classes for year 2017 in the hypothetical scenario with classification confidence data for models trained with (a) one-year, (b) two-year, (c) four-year, (d) eight-year temporal resolution.

Table 3.8. Number of cells correctly and incorrectly forecasted per class using the four temporal resolution options considering the hypothetical scenario.

		Land Cover Class			
		Forest	Anthropogenic Areas	Non-Forest Areas	Water
Real # of Changed Cells from 2001 to 2017		39969	780	85379	0
Real # of Persistent Cells from 2001 to 2017		396781	4145	407873	90607
Measure	Temporal Resolution (Years)				
Number of Cells Correctly Simulated as Changed	1	32974	593	77145	0
	2	32104	542	75127	0
	4	28169	412	70418	0
	8	0	0	85379	0
% of Cells Correctly Simulated as Changed	1	82.50%	76.03%	90.36%	N/A
	2	80.32%	69.49%	87.99%	N/A
	4	70.48%	52.82%	82.48%	N/A
	8	0%	0%	100.00%	N/A
Number of Cells Correctly Simulated as Persistent	1	388684	4093	403597	90607
	2	388405	4081	403708	90607
	4	384961	4047	403520	90607
	8	0	0	407873	90607
% of Cells Correctly persistent	1	97.96%	98.75%	98.95%	100.00%
	2	97.89%	98.46%	98.98%	100.00%
	4	97.02%	97.64%	98.93%	100.00%
	8	0%	0%	100.00%	100.00%
Number of Changed Cells Simulated as Wrong Change	1	0	3	2	0
	2	0	2	2	0
	4	0	2	1	0
	8	0	0	10	0
Changed Cells Simulated Incorrectly as Persistent	1	8093	139	7179	0
	2	10098	153	8100	0
	4	14761	200	12165	0
	8	0	0	40739	0
Persistent Cells Simulated Incorrectly as Changed	1	4121	155	8149	0
	2	4038	127	8440	0
	4	4251	102	11918	0
	8	0	0	400926	0

Table 3.9. Number of cells correctly and incorrectly forecasted per class using the four temporal resolution options considering the hypothetical scenario with the land cover classification confidence layer.

		Land Cover Class			
		Forest	Anthropogenic Areas	Non-Forest Areas	Water
Real # of Changed Cells from 2001 to 2017		39969	780	85379	0
Real # of Persistent Cells from 2001 to 2017		396781	4145	407873	90607
Measure	Temporal Resolution (Years)				
Number of Cells Correctly Simulated as Changed	1	32974	593	77145	0
	2	32104	542	75128	0
	4	28167	413	70396	0
	8	41	198	85069	0
% of Cells Correctly Simulated as Changed	1	82.50%	76.03%	90.36%	N/A
	2	80.32%	69.49%	87.99%	N/A
	4	70.47%	52.95%	82.45%	N/A
	8	0.10%	25.38%	99.64%	N/A
Number of Cells Correctly Simulated as Persistent	1	388679	4090	403596	90607
	2	388405	4076	403708	90607
	4	384847	4048	403518	90607
	8	1018	3837	407103	90607
% of Cells Correctly persistent	1	97.96%	98.67%	98.95%	100.00%
	2	97.89%	98.34%	98.98%	100.00%
	4	96.99%	97.66%	98.93%	100.00%
	8	0.26%	92.57%	99.81%	100.00%
Number of Changed Cells Simulated as Wrong Change	1	0	3	2	0
	2	0	2	3	0
	4	0	2	1	0
	8	2	1	5	0
Changed Cells Simulated Incorrectly as Persistent	1	8093	139	7179	0
	2	10096	153	8100	0
	4	14756	227	12166	0
	8	40	267	40505	0
Persistent Cells Simulated Incorrectly as Changed	1	4122	155	8157	0
	2	4038	127	8445	0
	4	4251	104	12031	0
	8	671	99	396071	0

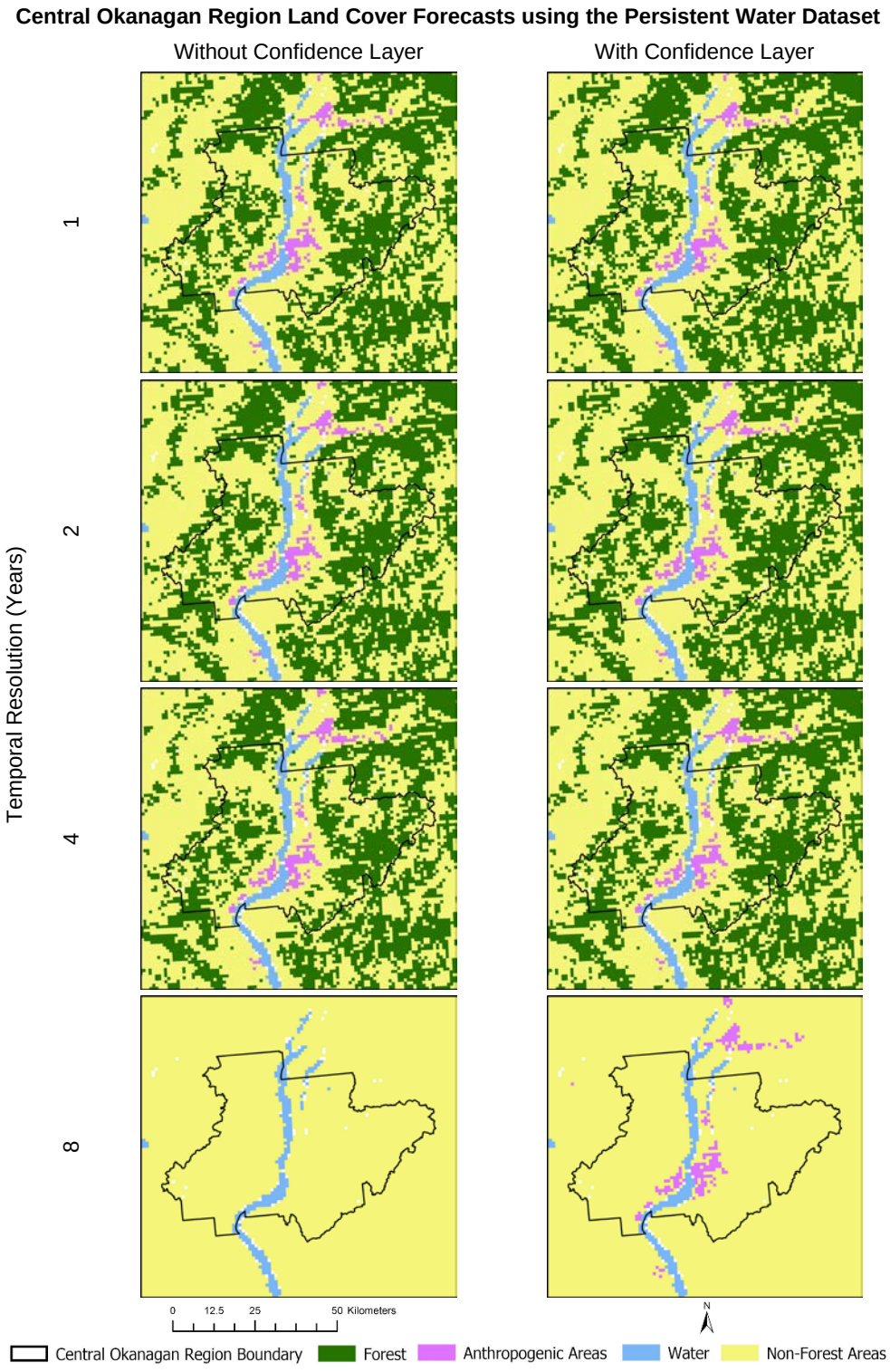


Figure 3.14. Comparison of land cover forecasts centred on the Central Okanagan region, British Columbia, in the hypothetical scenario without and with the classification confidence data.

3.6. Conclusion

The stacked LSTM modeling approach for forecasting LCC aims to detect patterns occurring across the temporal dimension to forecast forest, anthropogenic areas, non-forested non-anthropogenic areas, and water. Compared to previous applications of LSTM for prediction and classification tasks utilizing geospatial data, real-world applications thus far had not considered the implications and importance of the choice of geospatial input data characteristics on method performance. For slow-changing geospatial systems such as LCC, it is demonstrated that obtaining datasets that feature many timesteps and finer temporal resolutions enable more optimal models to be obtained. It is also indicative of potential issues arising when considering the LSTM method for use with geospatial datasets that are limited in the number of data layers or timesteps.

By comparing models trained with four temporal resolutions, it was observed that there is an overall bias for this method to forecast persistent cells, despite the balanced sampling regime used. This method also demonstrates a bias toward the majority class and persistent cells, especially as the number of timesteps decreased and temporal resolution became coarser in all experiments. Future work should consider improved sampling strategies to further address this issue. Similarly, a consequence of the sampling regime employed was a significant loss of potential training data samples. Maintaining and obtaining additional high-quality training samples available should continue to be a priority for further research involving LSTM.

While the classification confidence layer slightly increased the number of changed cells forecasted correctly in the real-world LC data scenarios, the greatest increase was seen in the hypothetical scenarios involving the persistent water class. This suggests the method's sensitivity to classification errors and suitability for situations where fewer classes undergo abrupt or rapid changes. Future work should consider reducing the cardinality of land cover datasets in order to use this method to its fullest potential given real-world geospatial datasets with limited layers available. Additional data layers should also be considered to increase method performance, especially if classes undergo more rapid changes.

Since this evaluation was conducted at a provincial scale, future works should also consider the implications of spatial extent and resolution on method performance. It is also

recommended that future work utilize additional evaluation metrics that consider not only location-based metrics, but spatial pattern metrics should be considered as well. While the integration of the available classification confidence layer available in the “MODIS Terra+Aqua Combined Land Cover Product” (Friedl, M., Sulla-Menashe 2015) was considered, further sensitivity analyses should be conducted to assess the effects of perturbations in input sequences and model response. Additional auxiliary or contextual data to provide more detail pertaining to spatial features to input sequences should also be explored. This could include deriving additional layers pertaining to local spatial autocorrelation. Though the classification confidence layer did not enhance LCC forecasting performance in situations involving fine temporal resolution data, this additional data layer may be advantageous in data-scarce scenarios where improving temporal resolution or increasing the number of timesteps is not an option. It should be assessed whether increasing the number of additional features may improve the forecasting performance of the method or increase robustness of the method to varying geospatial input data characteristics such as temporal resolution, LC class cardinality, and the number of timesteps available.

Given the lack of research endeavors exploring the effectiveness of LSTM for LCC forecasting, it was inconclusive as to what geospatial dataset characteristics were required to optimize the use of this modeling approach. By training and testing models using varying geospatial dataset characteristics, this work aimed to contribute to future LCC forecasting applications by providing recommendations and an assessment displaying under which circumstances the method is most effective. In this real-world application, it was determined that increasing the number of timesteps and obtaining data with finer temporal resolution enable the most optimal models to be developed for LCC forecasting. Likewise, the number of classes exhibiting change also demonstrated an impact on method performance. Lastly, integrating additional data layers such as classification confidence proved useful in mitigating the effects of coarser temporal resolution on the method’s capacity to simulate LCC.

3.7. References

Abadi, M, Agarwal, A, Barham, P, Brevdo, E, Chen, Z, Citro, C, Corrado, GS, et al. 2016. “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems.”

Ahmed, B, Ahmed, R, and Zhu, X. 2013. “Evaluation of Model Validation Techniques in Land Cover Dynamics.” ISPRS International Journal of Geo-Information 2 (3): 577–97.

Ball, JE, Anderson, DT, and Chan, CS. 2017. “A Comprehensive Survey of Deep Learning in Remote Sensing: Theories, Tools and Challenges for the Community.” Journal of Applied Remote Sensing 11 (04): 1.

Bishop, CM. 2006. Pattern Recognition and Machine Learning. Springer-Verlag New York.

Boulila, W, Ayadi, Z, and Farah, IR. 2017. “Sensitivity Analysis Approach to Model Epistemic and Aleatory Imperfection: Application to Land Cover Change Prediction Model.” Journal of Computational Science 23: 58–70.

“Boundary Files, 2016 Census.” 2016. Statistics Canada Catalogue no. 92-160-X.

Chase, TN, Pielke, RA, Kittel, TGF, Nemani, RR, and Running, SW. 2000. “Simulated Impacts of Historical Land Cover Changes on Global Climate in Northern Winter.” Climate Dynamics 16 (2–3): 93–105.

Chollet, F. 2015. “Keras: The Python Deep Learning Library.” Keras.Io. 2015.

Donahue, J, Hendricks, LA, Guadarrama, S, Rohrbach, M, Venugopalan, S, Darrell, T, and Saenko, K. 2015. “Long-Term Recurrent Convolutional Networks for Visual Recognition and Description.” In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 07-12-June:2625–34.

Elections BC. 2019. “Regional District Boundaries - Road Centreline Aligned - Datasets - Data Catalogue.” BC Geographic Warehouse. 2019.

Esri. 2017a. “ArcGIS Pro: 2.0.” Redlands, CA: Environmental Systems Research Institute.

Esri. 2017b. “Project Raster.” ArcGIS Pro Help. Redlands, CA: Environmental Systems Research Institute.

Esri. 2016. “Cell Size and Resampling in Analysis—ArcGIS Help | ArcGIS for Desktop.” Redlands, CA: Environmental Systems Research Institute.

Findell, KL, Berg, A, Gentine, P, Krasting, JP, Lintner, BR, Malyshev, S, Santanello, JA, and Shevliakova, E. 2017. “The Impact of Anthropogenic Land Use and Land Cover Change on Regional Climate Extremes.” Nature Communications 8 (1): 989.

Friedl, M., Sulla-Menashe, D. 2015. “MCD12Q1 MODIS/Terra+Aqua Land Cover Type Yearly L3 Global 500m SIN Grid V006 [Data Set].” NASA EOSDIS Land Processes DAAC. 2015.

GDAL/OGR contributors. 2019. “GDAL/OGR Geospatial Data Abstraction Software Library.” Open Source Geospatial Foundation. Open Source Geospatial Foundation.

Geist, H, McConnell, W, Lambin, EF, Moran, E, Alves, D, and Rudel, T. 2006. “Causes and Trajectories of Land-Use/Cover Change.” In Land-Use and Land-Cover Change, 41–70. Berlin, Heidelberg: Springer Berlin Heidelberg.

Gers, FA. 1999. “Learning to Forget: Continual Prediction with LSTM.” In 9th International Conference on Artificial Neural Networks: ICANN ’99, 1999:850–55. IEE.

Greff, K, Srivastava, RK, Koutnik, J, Steunebrink, BR, and Schmidhuber, J. 2017. “LSTM: A Search Space Odyssey.” IEEE Transactions on Neural Networks and Learning Systems 28 (10): 2222–32.

Hermans, M and Schrauwen, B. 2013. “Training and Analysing Deep Recurrent Neural Networks.” Advances in Neural Information Processing Systems 26 (NIPS 2013), 190–98.

Hochreiter, S. 1998. “The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions.” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 06 (02): 107–16.

Hochreiter, S and Schmidhuber, J. 1997. “Long Short-Term Memory.” Neural Computation 9 (8): 1735–80.

Ienco, Di, Gaetano, R, Dupaquier, C, and Maurel, P. 2017. “Land Cover Classification via Multitemporal Spatial Data by Deep Recurrent Neural Networks.” IEEE Geoscience and Remote Sensing Letters 14 (10): 1685–89.

Jia, X, Khandelwal, A, Nayak, G, Gerber, J, Carlson, K, West, P, and Kumar, V. 2017. “Incremental Dual-Memory LSTM in Land Cover Prediction.” In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD ’17, 867–76. New York, New York, USA: ACM Press.

Karpathy, A and Fei-Fei, L. 2017. “Deep Visual-Semantic Alignments for Generating Image Descriptions.” IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (4): 664–76.

Karpatne, A, Jiang, Z, Vatsavai, RR, Shekhar, S, and Kumar, V. 2016. “Monitoring Land-Cover Changes: A Machine-Learning Perspective.” IEEE Geoscience and Remote Sensing Magazine 4 (2): 8–21.

Kingma, DP and Lei Ba, J. 2015. “Adam: A Method for Stochastic Optimization.” In Proceedings of the 3rd International Conference on Learning Representations (ICLR). San Diego.

Kong, Y-LL, Huang, Q, Wang, C, Chen, JJ, Chen, JJ, and He, D. 2018. “Long Short-Term Memory Neural Networks for Online Disturbance Detection in Satellite Image Time Series.” Remote Sensing 10 (3): 452.

Kratzert, F, Klotz, D, Brenner, C, Schulz, K, and Herrnegger, M. 2018. “Rainfall-Runoff Modelling Using Long Short-Term Memory (LSTM) Networks.” Hydrology and Earth System Sciences 22 (11): 6005–22.

Lambin, EF, Turner, BL, Geist, HJ, Agbola, SB, Angelsen, A, Bruce, JW, Coomes, OT, et al. 2001. “The Causes of Land-Use and Land-Cover Change: Moving beyond the Myths.” Global Environmental Change. Vol. 11.

Liu, L, Han, M, Zhou, Y, and Wang, Y. 2018. “LSTM Recurrent Neural Networks for Influenza Trends Prediction.” In Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10847 LNBI:259–64. Springer, Cham.

Lyu, H, Lu, H, and Mou, L. 2016. “Learning a Transferable Change Rule from a Recurrent Neural Network for Land Cover Change Detection.” Remote Sensing 8 (6): 506.

Maithani, S. 2014. “Neural Networks-Based Simulation of Land Cover Scenarios in Doon Valley, India.” Geocarto International, July, 1–23.

Meyer, WB and Turner, BL. 1996. “Land-Use/Land-Cover Change: Challenges for Geographers.” GeoJournal 39 (3): 237–40.

Otukei, JR and Blaschke, T. 2010. “Land Cover Change Assessment Using Decision Trees, Support Vector Machines and Maximum Likelihood Classification Algorithms.” International Journal of Applied Earth Observation and Geoinformation 12 (SUPPL. 1): S27–31.

Pascanu, R, Gulcehre, C, Cho, K, and Bengio, Y. 2014. “How to Construct Deep Recurrent Neural Networks.” In Proceedings of the 2nd International Conference on Learning Representations (ICLR). Banff, Canada.

Patil, SD, Gu, Y, Dias, FSA, Stieglitz, M, and Turk, G. 2017. “Predicting the Spectral Information of Future Land Cover Using Machine Learning.” International Journal of Remote Sensing 38 (20): 5592–5607.

Pham, V, Bluche, T, Kermorvant, C, and Louradour, J. 2014. “Dropout Improves Recurrent Neural Networks for Handwriting Recognition.” In Proceedings of International Conference on Frontiers in Handwriting Recognition (ICFHR), 285–90. Crete, Greece.

Raskutti, G, Wainwright, MJ, and Yu, B. 2011. “Early Stopping for Non-Parametric Regression: An Optimal Data-Dependent Stopping Rule.” In 2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton), 1318–25. IEEE.

Ren, Y, Lü, Y, Comber, A, Fu, B, Harris, P, and Wu, L. 2019. “Spatially Explicit Simulation of Land Use/Land Cover Changes: Current Coverage and Future Prospects.” Earth-Science Reviews 190 (March): 398–415.

Rußwurm, M and Körner, M. 2017. “Multi-Temporal Land Cover Classification with Long Short-Term Memory Neural Networks.” International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives 42 (1W1): 551–58.

———. 2018. “Multi-Temporal Land Cover Classification with Sequential Recurrent Encoders.” ISPRS International Journal of Geo-Information 7 (4): 129.

Samardžić-Petrović, M, Dragićević, S, Kovačević, M, and Bajat, B. 2016. “Modeling Urban Land Use Changes Using Support Vector Machines.” Transactions in GIS 20 (5): 718–34.

Samardžić-Petrović, M, Kovačević, M, Bajat, B, and Dragićević, S. 2017. “Machine Learning Techniques for Modelling Short Term Land-Use Change.” ISPRS International Journal of Geo-Information 6 (12): 387.

Srivastava, N, Hinton, G, Krizhevsky, A, and Salakhutdinov, R. 2014. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting.” Journal of Machine Learning Research. Vol. 15.

Sulla-Menashe, D and Friedl, MA. 2018. “User Guide to Collection 6 MODIS Land Cover (MCD12Q1 and MCD12C1) Product,” 1–18.

Sun, Z, Di, L, and Fang, H. 2018. “Using Long Short-Term Memory Recurrent Neural Network in Land Cover Classification on Landsat and Cropland Data Layer Time Series.” International Journal of Remote Sensing.

Understanding LSTM Networks. (2016). Retrieved July 17, 2019, from http://colah.github.io/posts/2015-08-Understanding-LSTMs/

van Rossum, G. 2016. Python 3.6 Language Reference. United Kingdom: Samurai Media Limited.

Voight, C, Hernandez-Aguilar, K, Garcia, C, and Gutierrez, S. 2019. “Predictive Modeling of Future Forest Cover Change Patterns in Southern Belize.” Remote Sensing 11 (7): 823.

Wu, Q, Shen, C, Wang, P, Dick, A, and van den Hengel, A. 2018. “Image Captioning and Visual Question Answering Based on Attributes and External Knowledge.” IEEE Transactions on Pattern Analysis and Machine Intelligence 40 (6): 1367–81.

Yang, W, Long, D, and Bai, P. 2019. “Impacts of Future Land Cover and Climate Changes on Runoff in the Mostly Afforested River Basin in North China.” Journal of Hydrology 570 (March): 201–19.

Zhang, Q, Wang, H, Dong, J, Zhong, G, and Sun, X. 2017. “Prediction of Sea Surface Temperature Using Long Short-Term Memory.” IEEE Geoscience and Remote Sensing Letters 14 (10): 1745–49.

Chapter 4.

Conclusion

4.1. Synthesis of Research Results

With increasing interest in applying data-driven modeling approaches for geospatial applications, it becomes important to assess the capacity of these methods for capturing and forecasting complex geospatial systems. Land cover change (LCC) dynamics typically exhibit infrequent or slow changes over time, posing challenges for currently available Machine Learning (ML) methods (Geist et al. 2006; Karpatne et al. 2016). A subfield of ML, Deep Learning (DL), provides increasingly complex modeling approaches based on traditional Neural Networks (NNs). This subtype of data-driven methods has proven advantageous for predictive and classification tasks involving complex datasets, with performance measures improving as dataset size increases (Chi and Kim 2017).

With their heightened capacity for capturing intricate, non-linear relationships subsisting in datasets, few DL methods have been assessed in their ability to model LCC dynamics, where changes are typically infrequent. Likewise, it was unknown what characteristics of geospatial datasets were compatible or conducive for increasing the efficacy of the methods. Sequential DL methods such as Recurrent Neural Networks (RNNs) (specifically the Long Short-Term Memory variant) have been previously utilized in geospatial applications (Lyu, Lu, and Mou 2016; Chi and Kim 2017; Ienco et al. 2017; Zhang et al. 2017). However, geospatial data characteristics required to yield successful outcomes from these method applications were unspecified. For instance, the number of timesteps or temporal resolution required to produce improved results were undetermined.

This thesis research aims to assess not only the potential of sequential DL methods for forecasting LC changes, but to guide the selection of appropriate geospatial datasets for use with these methods. The effectiveness of sequential DL models in real-world scenarios where limitations exist in the temporal dimension of datasets is also assessed. This includes an evaluation of method response to limited timesteps or coarse temporal resolutions available. Likewise, it was also necessary to assess the effectiveness of LSTM for forecasting slow changes common in geospatial systems such as LCC.

To implement the proposed methodology, Python (v3.6.5) and the Keras API (v2.2.0) (Chollet 2015) were used. The TensorFlow GPU implementation (v1.8.0) was selected as the backend implementation, with the Keras API providing a high-level interface to utilize the functionality afforded by this open-source machine learning library (Abadi et al. 2016). Stacked LSTM models were developed for their known capacity to capture complex relationships existing in input datasets (Pascanu et al. 2014; Hermans and Schrauwen 2013). The Geospatial Data Abstraction Library (v2.2.4) (GDAL/OGR contributors 2019) and Esri's ArcGIS Pro (v2.4.0) (Esri 2017) were used to process input datasets and to process, evaluate, and visualize results. Model training, testing, and evaluations were conducted using NVIDIA Quadro 600 and NVIDIA GeForce GTX 1080 Ti GPUs.

First, favorable geospatial data characteristics were determined (Chapter 2). The primary objective of this study was to evaluate the repercussions of varying geospatial data properties on LSTM model performance. Localized LCC forecasts produced by the LSTM models were evaluated using a Sensitivity Analysis (SA) approach that considered varying (1) temporal resolution, (2) sequence length, (3) cardinality, and (4) rates of LCCs. It was determined that performance of LSTM models would be impacted by coarse temporal resolutions and limited numbers of timesteps. It was also observed that increasing the cardinality of the dataset yielding poorer performing models, despite model optimization techniques applied. These results indicated that this method benefits from fewer LC classes being present in the geospatial dataset being provided as input to the model. It was also observed that the method is biased toward persistent cells in all scenarios. For instance, it was observed that methods simulated persistent cells with near-perfect agreement between forecasted and actual maps (Figure 2.7). This occurred despite the variations in temporal resolution and the increase in number of LC classes being considered. Overall, results produced when using finer temporal resolution data, more correctly changed cells were captured in the forecasted land cover.

Utilizing findings from the preceding chapter, this research further explored the implications of geospatial data characteristics on method performance (Chapter 3). Methods were adapted and applied to LCC data obtained for the province of British Columbia, Canada. The purpose was to evaluate the potential of LSTM for LCC modeling in a real-world scenario where datasets may feature erroneous values resulting from prior classification procedures. By training the method using varying temporal resolutions,

results coincided with previous conclusions formed from using the localized, synthetic datasets. It was also observed that potential classification errors in the MODIS land cover dataset increased the number of errors, especially as temporal resolution became coarser (Tables 3.6 and 3.7). It was ascertained that more timesteps improved the model's capacity to forecast slow or rarer changes. The balanced training dataset creation procedures described in Samardžić-Petrović et al. (2017) were employed. This approach creates training datasets by randomly sampling changed and persistent cells for each land cover class according to their original distributions. This showed greater positive impacts on method performance than in the previous method focussed on the localized study area. Likewise, the model benefitted from greater breadth (number of units per layer) and use of optimization techniques such as Dropout that proved beneficial in Chapter 2 (Sutskever et al. 2014). Results showed that persistent cells were also forecasted with high accuracy measures for the one-, two-, and four-year temporal resolutions (Table 3.5). In the scenario using the training set characterized by four-year temporal resolution, the lowest percentage of cells forecasted accurately as persistent was still 97.01%. Cells simulated incorrectly as persistent was also the most frequent error type occurring in forecasts of the two majority classes (forest and non-forest areas). It was also observed that the worst-case scenario with the coarsest possible temporal resolution (where one input was mapped to one output) produced far more erratic, poor results in the tests utilizing the real-world dataset, despite the addition of the classification confidence layer and efforts to reconfigure training dataset creation procedures to improve or reduce biases in the training set.

This study further illustrates the importance of temporal resolution and opting for greater number of timesteps when choosing to utilize LSTM for forecasting LCC. Overall, the models trained with the coarsest temporal resolution, considering one timestep as input and one timestep as output, have shown to be ineffective in LCC modeling in such configurations. The addition of the classification confidence layer as input showed potential for enhancing scenarios in which the number of timesteps or temporal resolution is limited, albeit achieving poor simulation results across all experiments considering real-world and hypothetical scenarios.

By developing the persistent water class scenarios, evaluations demonstrated the sensitivity of the method to unchanging classes and to classification errors existing in the original dataset. With one class remaining static through time, changes occurring in other

classes were better captured at coarser temporal resolutions. Persistent cells were forecasted with no errors in all experiments, considering all temporal resolutions and both datasets. This concurs with results obtained in Chapter 2, indicating the method favours persistent cells and that decreased rates of change improves method performance.

By developing various modeling approaches, conducting a SA, and applying the method to real-world LC datasets, the method forecasted changed cells better when the finest temporal resolutions were used. It was also determined that geospatial dataset properties such as rates of change and LC class cardinality impact the performance of the LSTM models. For forecasting LCC when many timesteps are available, LSTM proves to be an accessible, capable method for forecasting patterns of change from these dynamic, geospatial systems.

4.2. Limitations and Future Directions

While the LSTM approaches have exemplified the potential of sequential DL models for LCC modeling, there exist limitations within the current implementations. In both chapters two and three, it was observed that the method was biased toward persistent cells. As temporal resolution became coarser, the method response became more erratic. Conversely, if the geospatial dataset featured increased depth in the temporal dimension, forecasts were drastically improved. The addition of ancillary data layers as demonstrated in chapter three pose promising directions for potentially improving LCC modeling performance in scenarios where the number of timesteps or temporal resolution is limited. Providing other derived data layers including layers pertaining to spatial autocorrelation may also be beneficial to improving model performance. To further improve the capacity of these methods to forecast LCC, there exist many avenues for future research endeavours. Following the development of solutions for each of the limitations described in following sections, expanded study areas should also be considered, including national and global scales.

First, additional SAs should be conducted to assess the implications of changing spatial resolution on LSTM performance, as data was resampled from 500m to 1km to expedite model training and testing procedures in Chapter 3. Likewise, the resampling technique of nearest neighbour impacts the quality of ancillary data layers. Error

accumulation occurs as a result of these resampling procedures and is expected to impact method performance in future work.

It is also recognized that explicitly spatio-temporal DL modeling approaches such as Convolutional LSTM exist (Shi et al. 2015). These approaches should be evaluated further for their potential to model LCC patterns in future work, especially for their capacity to forecast spatial patterns. However, these methods are affected by similar shortcomings as models presented in this thesis. It is important that future research should aim to address some of the inherent limitations of data-driven techniques for geospatial applications. As ML and DL methods are typically trained to fit a single set of parameters that minimize an error function with respect to a training set, problems arise for heterogeneous study areas. With the goal of "learning" a set of parameters that will achieve suitable performance when given new observations in classification or predictive applications (Bishop 2006), the procedure to train ML and DL methods involves utilizing all inputs with equal importance. That is, the influence of training samples that may or may not be adjacent or near to one another are used to fit a single set of internal network parameters that should generalize to all future inputs given. This implies that methods are typically not adapted for considering properties of geospatial datasets such as spatial heterogeneity and non-stationarity (Karpatne et al. 2016; Brunsdon, Fotheringham, and Charlton 1996). This concern continues to be exacerbated as study areas expand and diversify. Thus, future research endeavours should explore and integrate consideration of these properties in various aspects of future models constructed.

Next, sequential DL approaches are limited in their ability to model complex systems. Since LCC dynamics can be viewed as complex systems processes, their many interacting elements result in new, emergent behaviours and patterns that a data-driven model may not have observed in training procedures and thus not be able to forecast (Manson 2001; Batty and Torrens 2005). Data-driven approaches are pattern recognition methods which are incapable of capturing facets of complex systems such as emergence, evolution, feedback loops, bifurcation, and self-organization (Manson 2001). For instance, ML methods utilize historic trends to forecast changes and are therefore not inherently intended for simulating non-stationary patterns and feedback loops that characterize human and environmental interactions (National Research Council 2014). ML algorithms assume stationarity of patterns and are devised to extrapolate patterns, not the processes by which they were generated (National Research Council 2014). While LSTM is stated

to be able to capture path dependencies in timeseries data, it is also largely unknown or challenging to determine how well these “black-box” approaches are capturing these relationships through time (Vohra, Goel, and Sahoo 2015). Existing complex systems modeling approaches such as Cellular Automata have their own shortcomings, such as challenges involving manual encoding of transition rules (Yang, Li, and Shi 2008). Likewise, biases are introduced during the selection and definition of transition rules, which have significant effects on the emergent behaviour of the system (Roodposhti, Aryal, and Bryan 2019). Conversely, statistical learning algorithms are advantageous in scenarios where driving factors are unknown or theory is ill-defined (National Research Council 2014). Combining sequential DL methods such as LSTM with Cellular Automata to formulate novel modeling approaches may leverage beneficial aspects of both top-down and bottom-up modeling techniques.

The performance of LSTM versus other RNN modeling approaches should also be considered and compared in future work. For instance, the gated recurrent unit (GRU) (Chung et al. 2015), a simplification to the LSTM gated architecture, has been made increasingly available and accessible in popular high-level APIs such as Keras. While yielding similar results to LSTM modeling approaches (Greff et al. 2017), GRUs are stated to converge faster than LSTM due to fewer internal parameters. In future studies, it should be assessed if method performance similarly degrades across other RNN architectures such as GRUs as temporal resolution becomes coarser, the number of land cover classes increases, and the number of timesteps available for model training decreases. Hybrid Deep Neural Network approaches should also be implemented and assessed to capture different types of spatial features occurring over space and time. These include combined networks such as Convolutional LSTM (Shi et al. 2015), merged models used to accommodate different input data types (Wu et al. 2015), and merging fine-tuned DL models with traditional ML methods (Chen et al. 2014; Zhao and Du 2016).

Additional exploration of sampling schemes used to form better training sets should also be considered. Patterns resulting from processes amalgamating slow changes across Earth’s surface over time prove difficult to forecast without the use of balanced training set creation strategies. The sensitivity of sequential DL methods such as LSTM to training dataset composition has been demonstrated in this work. For instance, training dataset composition was heavily influenced by persistent cells and necessitated balanced sampling schemes to improve model performance, imposing limitations on the amount of

training data that could be used despite having full spatial and temporal extents available in all datasets used in this work. Knowing the propensity of the method to forecast persistence, future works should consider strategies that do not require so much data to be discarded when few changed cells or areas are available for training the method. Modifying the training dataset composition via improved sampling schemes or data augmentation procedures should be explored in future work to lessen the amount of unused data and improve model performance. Similarly, it may prove advantageous to utilize or develop new sampling strategies to account for spatial autocorrelation.

Finally, future studies should consider a wider array of validation approaches for data-driven models. While measuring locational agreement between a forecasted output and real map provides one means of assessment, the capacity of modeling approaches for forecasting spatial patterns should also be considered (van Vliet et al. 2016). Additionally, fuzzy methods for evaluation should also be employed (Ahmed et al. 2013). Errors resulting from procedures used to classify satellite imagery should not only considered, but the sensitivity of the method to erroneous values present in the timeseries inputs that constitute training datasets should be quantified in future work. Likewise, epistemic imperfections should be quantified and decreased by including additional data layers (Boulila, Ayadi, and Farah 2017).

4.3. Thesis Contributions

The proposed methodologies exemplify the potential of the LSTM method and sequential DL modeling approaches for forecasting LCC by determining patterns through time. The research aimed to determine favourable scenarios for sequential DL methods such as LSTM to forecast changes in systems where changes are infrequent. It is intended that this work informs future research endeavours by either assisting in method selection or by providing insight for geospatial data selection.

By conducting a formal analysis of the method's response to different geospatial data properties, researchers can be better informed of both the strengths and shortcomings of the method for modeling systems such as LCC. This thesis research provides improved, automated methodologies for forecasting LCC while formalizing the geospatial data requirements necessary for this method to be effective. These automated methodologies are developed and evaluated in hypothetical and real-world scenarios at

local and provincial scales in order to help form conclusions regarding the method's effectiveness for forecasting LCC. The SA and real-world application of the method highlight the importance of geospatial data selection if this method is to be utilized to its full potential.

This thesis research aims to contribute to the field of GIScience by elucidating the benefits and limitations of this data-driven modeling approach. The proposed research methodologies contribute to the expanding collection of geocomputational approaches available and tested for forecasting dynamic, geospatial systems such as LCC. These contributions can also be beneficial for other disciplines that consider LCC studies, such as land change science, forestry, resource management, and urban planning.

4.4. References

Abadi, M, Agarwal, A, Barham, P, Brevdo, E, Chen, Z, Citro, C, Corrado, GS, et al. 2016. "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems."

Ahmed, B, Ahmed, R, and Zhu, X. 2013. "Evaluation of Model Validation Techniques in Land Cover Dynamics." ISPRS International Journal of Geo-Information 2 (3): 577–97.

Batty, M and Torrens, PM. 2005. "Modelling and Prediction in a Complex World." Futures 37 (7): 745–66.

Bishop, CM. 2006. Pattern Recognition and Machine Learning. Springer-Verlag New York.

Boulila, W, Ayadi, Z, and Farah, IR. 2017. "Sensitivity Analysis Approach to Model Epistemic and Aleatory Imperfection: Application to Land Cover Change Prediction Model." Journal of Computational Science 23: 58–70.

Brunsdon, C, Fotheringham, AS, and Charlton, ME. 1996. "Geographically Weighted Regression: A Method for Exploring Spatial Nonstationarity." Geographical Analysis 28 (4): 281–98.

Chen, Y, Lin, Z, Zhao, X, Wang, G, and Gu, Y. 2014. "Deep Learning-Based Classification of Hyperspectral Data." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 7 (6): 2094–2107.

Chi, J and Kim, H. 2017. "Prediction of Arctic Sea Ice Concentration Using a Fully Data Driven Deep Neural Network." Remote Sensing 9 (12): 1305.

Chollet, F. 2015. “Keras: The Python Deep Learning Library.” Keras.Io. 2015.

Chung, J, Gulcehre, C, Cho, K, and Bengio, Y. 2015. “Gated Feedback Recurrent Neural Networks.” In ICML’15 Proceedings of the 32nd International Conference on International Conference on Machine Learning, 2067–75. Lille, France.

Esri. 2017. “ArcGIS Pro: 2.0.” Redlands, CA: Environmental Systems Research Institute.

GDAL/OGR contributors. 2019. “GDAL/OGR Geospatial Data Abstraction Software Library.” Open Source Geospatial Foundation. Open Source Geospatial Foundation.

Geist, H, McConnell, W, Lambin, EF, Moran, E, Alves, D, and Rudel, T. 2006. “Causes and Trajectories of Land-Use/Cover Change.” In Land-Use and Land-Cover Change, 41–70. Berlin, Heidelberg: Springer Berlin Heidelberg.

Greff, K, Srivastava, RK, Koutnik, J, Steunebrink, BR, and Schmidhuber, J. 2017. “LSTM: A Search Space Odyssey.” IEEE Transactions on Neural Networks and Learning Systems 28 (10): 2222–32.

Hermans, M and Schrauwen, B. 2013. “Training and Analysing Deep Recurrent Neural Networks.” Advances in Neural Information Processing Systems 26 (NIPS 2013), 190–98.

Ienco, Di, Gaetano, R, Dupaquier, C, and Maurel, P. 2017. “Land Cover Classification via Multitemporal Spatial Data by Deep Recurrent Neural Networks.” IEEE Geoscience and Remote Sensing Letters 14 (10): 1685–89.

Karpatne, A, Jiang, Z, Vatsavai, RR, Shekhar, S, and Kumar, V. 2016. “Monitoring Land-Cover Changes: A Machine-Learning Perspective.” IEEE Geoscience and Remote Sensing Magazine 4 (2): 8–21.

Lyu, H, Lu, H, and Mou, L. 2016. “Learning a Transferable Change Rule from a Recurrent Neural Network for Land Cover Change Detection.” Remote Sensing 8 (6): 506.

Manson, SM. 2001. “Simplifying Complexity: A Review of Complexity Theory.” Geoforum 32 (3): 405–14.

National Research Council. 2014. Advancing Land Change Modeling: Opportunities and Research Requirements. National Academies Press.

Pascanu, R, Gulcehre, C, Cho, K, and Bengio, Y. 2014. “How to Construct Deep Recurrent Neural Networks.” In Proceedings of the 2nd International Conference on Learning Representations (ICLR). Banff, Canada.

Roodposhti, MS, Aryal, J, and Bryan, BA. 2019. “A Novel Algorithm for Calculating Transition Potential in Cellular Automata Models of Land-Use/Cover Change.” Environmental Modelling & Software 112 (February): 70–81.

Samardžić-Petrović, M, Kovačević, M, Bajat, B, and Dragićević, S. 2017. “Machine Learning Techniques for Modelling Short Term Land-Use Change.” ISPRS International Journal of Geo-Information 6 (12): 387.

Shi, X, Chen, Z, Wang, H, Yeung, D-Y, Wong, W, and Woo, W. 2015. “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting.” Advances in Neural Information Processing Systems 28 (NIPS 2015), 802–10.

Sutskever, I, Hinton, G, Krizhevsky, A, and Salakhutdinov, RR. 2014. “Dropout : A Simple Way to Prevent Neural Networks from Overfitting.” Journal of Machine Learning Research. Vol. 15.

van Vliet, J, Bregt, AK, Brown, DG, Delden, H van, Heckbert, S, and Verburg, PH. 2016. “A Review of Current Calibration and Validation Practices in Land-Change Modeling.” Environmental Modelling and Software. Elsevier.

Vohra, R, Goel, K, and Sahoo, JK. 2015. “Modeling Temporal Dependencies in Data Using a DBN-LSTM.” In Proceedings of the 2015 IEEE International Conference on Data Science and Advanced Analytics, DSAA 2015, 1–4.

Wu, Z, Wang, X, Jiang, Y-G, Ye, H, and Xue, X. 2015. “Modeling Spatial-Temporal Clues in a Hybrid Deep Learning Framework for Video Classification.” In 23rd ACM International Conference on Multimedia, 461–70. Brisbane, Australia.

Yang, Q, Li, X, and Shi, X. 2008. “Cellular Automata for Simulating Land Use Changes Based on Support Vector Machines.” Computers & Geosciences 34 (6): 592–602.

Zhang, Q, Wang, H, Dong, J, Zhong, G, and Sun, X. 2017. “Prediction of Sea Surface Temperature Using Long Short-Term Memory.” IEEE Geoscience and Remote Sensing Letters 14 (10): 1745–49.

Zhao, W and Du, S. 2016. “Spectral-Spatial Feature Extraction for Hyperspectral Image Classification: A Dimension Reduction and Deep Learning Approach.” IEEE Transactions on Geoscience and Remote Sensing 54 (8): 4544–54.

Appendix.

Full Synthetic Dataset Overview

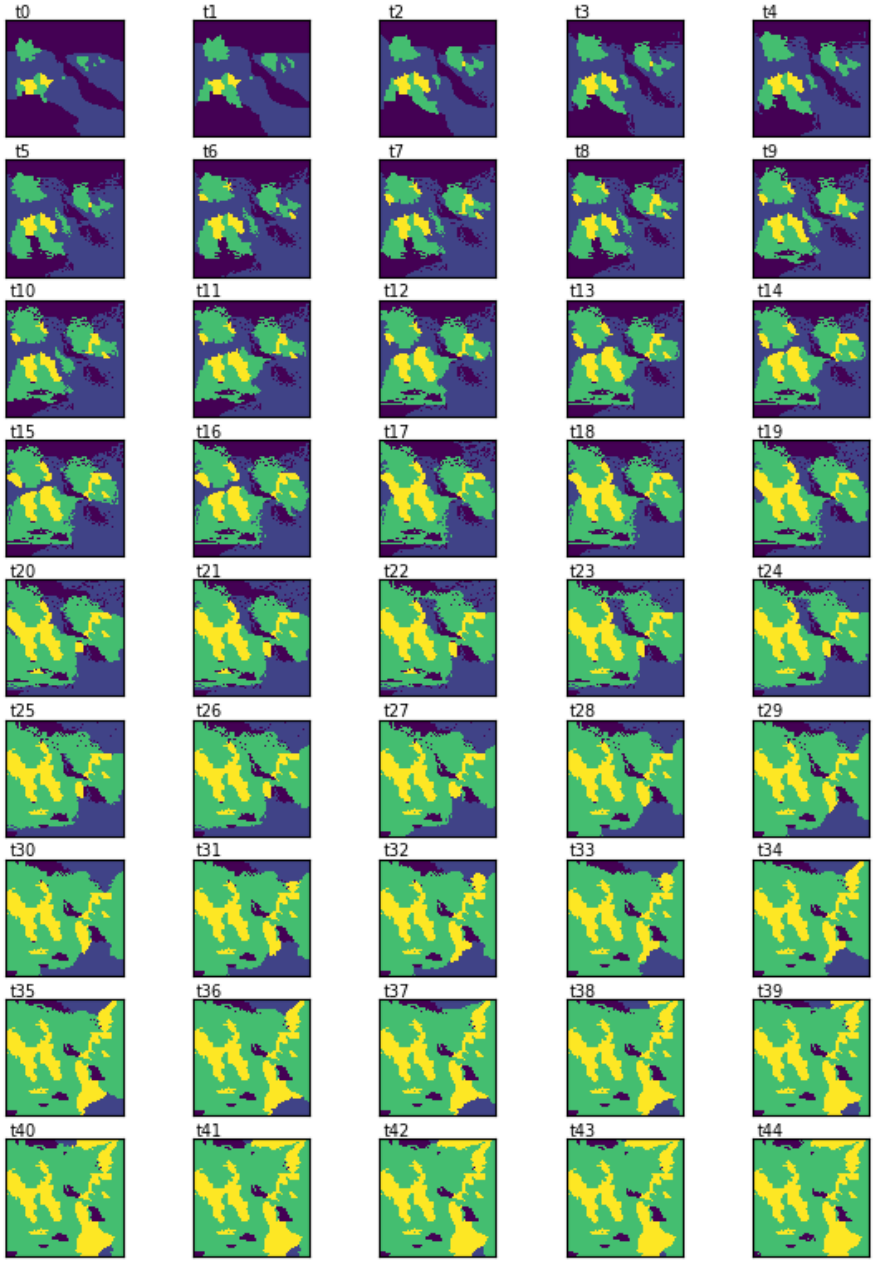


Figure A.1. Dataset featuring 4 Land Cover Classes

Figure A.2. Dataset featuring 8 Land Cover Classes

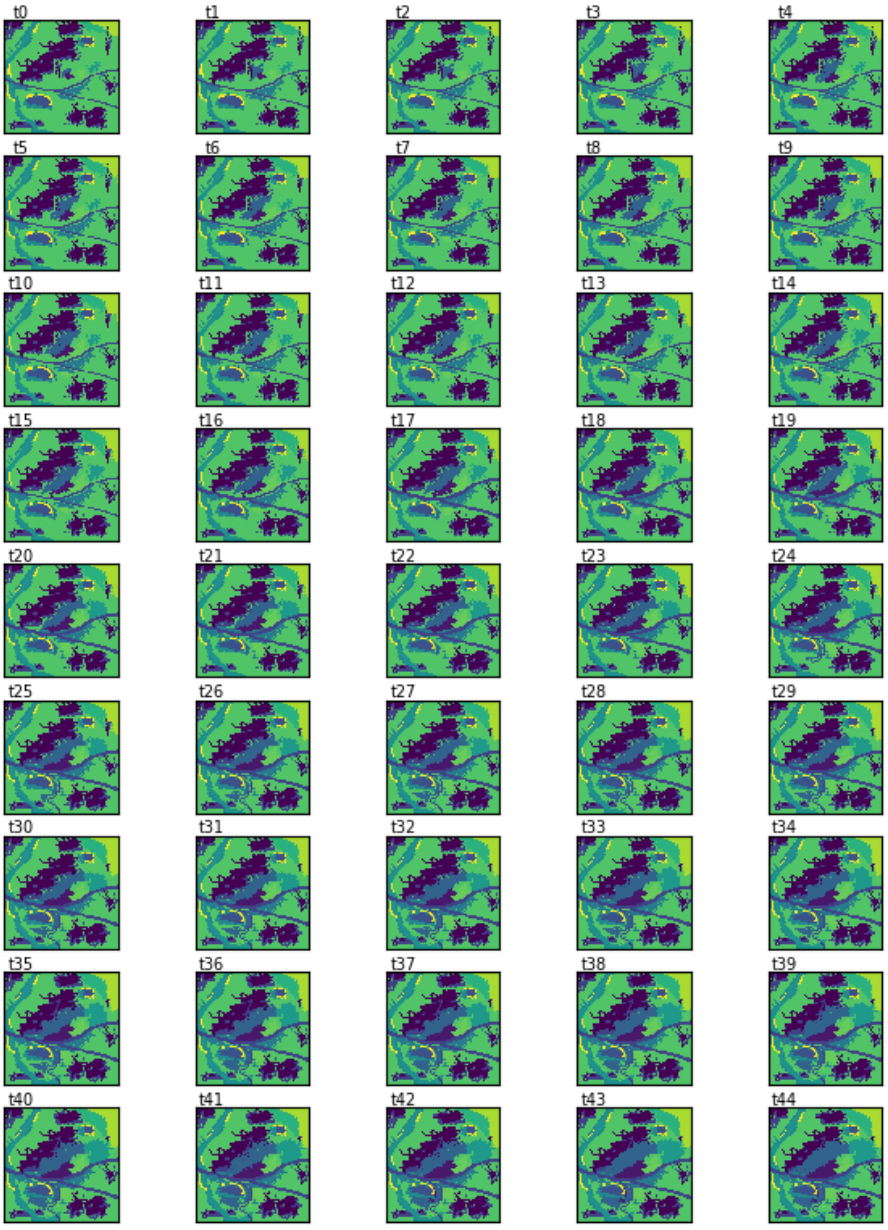


Figure A.3. Dataset featuring 16 Land Cover Classes