OGC Testbed-16

Machine Learning Training Data ER

Publication Date: 2021-01-13

Approval Date: 2020-12-14

Submission Date: 2020-11-19

Reference number of this document: OGC 20-018

Reference URL for this document: http://www.opengis.net/doc/PER/t16-D016

Category: OGC Public Engineering Report

Editor: Guy Schumann

Title: OGC Testbed-16: Machine Learning Training Data ER

OGC Public Engineering Report

COPYRIGHT

WARNING

LICENSE AGREEMENT

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD. THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications.

This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the

Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable.

Table of Contents

Chapter 1. Subject

The OGC Testbed-16 Machine Learning (ML) Training Data Engineering Report (ER) describes training data used for developing a Wildfire Response application. Within the context of the application, this ER discusses the challenges and makes a set of recommendations. The two scenarios for the wildfire use case include fuel load estimation and water body identification. The ML training data described in this ER are based on these two scenarios. Suggestions are also made for future work on a model for ML training dataset metadata, which is intended to provide vital information on the data and therefore facilitate the uptake of training data by the ML community. Additionally, this ER summarizes the discussions and issues about ML training data among the Testbed-16 ML thread participants and draws conclusions and recommendations for future work on the subject. Finally, this ER also links to current Analysis Ready Data (ARD) principles and efforts, in particular in the Earth Observation (EO) community.

Chapter 2. Executive Summary

2.1. General Purpose of the ML thread and this Engineering Report

The OGC Testbed-16 Machine Learning task focused on understanding the potential of existing and emerging OGC standards for supporting Machine Learning (ML) applications in the context of wildland fire safety and response. In this context, the integration of ML models into standards-based data infrastructures, the handling of ML training data, and the integrated visualization of ML data with other source data was explored. Emphasis was on the integration of data from the Canadian Geospatial Data Infrastructure CGDI [https://www.nrcan.gc.ca/science-data/science-research/earth-sciences/geomatics/canadas-spatial-data-infrastructure/10783], the handling of externally provided training data, and the provisioning of results to end-users without specialized software.

2.2. Requirements of the ML training data set components in particular

The Testbed-16 ML participants explored how to leverage ML technologies for dynamic wildland fire response. An objective was to also provide insight into how OGC standards can support wildland fire response activities in a dynamic context. Any identified limitations of existing OGC standards can be used to plan improvements to these frameworks.

Though this task uses a wildland fire scenario, the emphasis is not on the quality of the modeled results but on the integration of externally provided source and training data, the deployment of the ML model on remote clouds through a standardized interface, and the visualization of model output.

In summary, Testbed-16 addressed the following three challenges:

- Discovery and reusability of training data sets
- Integration of ML models and training data into standards-based data infrastructures
- Cost-effective visualization and data exploration technologies based on Map Markup Language (MapML)

This Machine Learning Training Data ER focuses explicitly on the first point. For ML models and their integration, readers are referred to the OGC Machine Learning Engineering Report [OGC 20-015].

2.3. Training Datasets

The Earth Observation (EO) user and developer community currently can access unprecedented capabilities. To combine these capabilities with the major advances in Artificial Intelligence (AI) in general and Machine Learning (ML) in particular, the community needs to close the gap between ML on one side and Earth observation data on the other. In this context, two aspects need to be addressed.

- The extremely limited discoverability and availability of training and test datasets.
- The interoperability challenges to enable ML systems to work with available data sources and live data feeds coming from a variety of systems and APIs

In this context, training datasets are pairs of examples of labelled data (independent variable) and the corresponding EO data (dependent variables). Together, these two types are used to train an ML model that is then used to make predictions of the target variable based on previously unseen EO data. Test data is a set of observations used to evaluate the performance of the model using some performance metric. In addition to the training and test data, a third set of observations, called a validation or hold-out set, is sometimes required. The validation set is used to tune variables called hyper parameters, which control how the model learns. In this ER, the set of training data, test data, and validation data together are referred to simply as training datasets.

To address the general lack of training data discoverability, accessibility, and reusability, Testbed-16 participants developed solutions that describe how training data sets can be generated, structured, described, made available, and curated.

2.4. (Research) Questions that this ER tries to address

As indicated in the Testbed-16 Call for Participation (CFP), this ER tries to address the following:

- Where do trained datasets go and how can they be re-used?
- How can we ensure the authenticity of trained datasets?
- Is it necessary to have analysis ready data (ARD) for ML? Can ML help ARD development?
- What metamodel structure should be used for ML TDS?
- What is the value of datacubes for ML?

2.5. About the Canadian Wildland Fire Information System (CWFIS)

The Canadian Wildland Fire Information System (CWFIS) [https://cwfis.cfs.nrcan.gc.ca/home] creates daily fire weather and fire behavior maps year-round and hot spot maps throughout the forest fire season, generally between May and September. The CWFIS monitors fire danger conditions and fire occurrence across Canada. Daily weather conditions are collected from across Canada and used to produce fire weather and fire behavior maps. In addition, satellites are used to detect fires, and reported fire locations are collected from fire management agencies.

The CWFIS comprises various systems. The Canadian Forest Fire Danger Rating System (CFFDRS) [https://cwfis.cfs.nrcan.gc.ca/background/summary/fdr] is a national system for rating the risk of forest fires in Canada . Forest fire danger is a general term used to express a variety of factors in the fire environment, such as ease of ignition and difficulty of control. Fire danger rating systems produce qualitative and/or numeric indices of fire potential, which are used as guides in a wide variety of fire management activities.

The CFFDRS has been under development since 1968. Currently, two subsystems – the Canadian Forest Fire Weather Index (FWI) System [https://cwfis.cfs.nrcan.gc.ca/background/summary/fwi] and the

Canadian Forest Fire Behavior Prediction (FBP) System [https://cwfis.cfs.nrcan.gc.ca/background/summary/fbp] – are being used extensively in Canada and internationally.

These are all available as training datasets for Testbed 16.

2.6. What does this ER mean for the EDM Working Group and the OGC

The purpose of the Emergency & Disaster Management (EDM) DWG is to promote and support the establishment of requirements and best practices for web service interfaces, models and schemas to enable the discovery, access, sharing, analysis, visualization and processing of information to the forecasting, prevention, response to and recovery from emergency and disaster situations. The mission lies in improving interoperability of geospatial products and other information consumables that can be shared across these communities. Two main objectives of work described in this ER are:

- Identify interoperability standards gaps and opportunities to support improved EDM information sharing, collaboration and decision making.
- Propose or encourage initiation of Interoperability Program studies, experiments, pilot initiatives, testbed threads or demonstrations to address technical, institutional and policy related interoperability challenges, and identify and engage the interest of potential sponsors for these activities.

The former OGC Law Enforcement And Public Safety (LEAPS) DWG promoted and supported the establishment of local, national, regional and international requirements and best practices for web service interfaces, data models and schemas for enabling the discovery, access, sharing, analysis, visualization and processing of information. This geospatial and temporal information is used comprehensively to address crime, terrorist activities and public safety incidents in an operationally effective way.

Given that the objectives and general purpose were very similar and overlapping for many applications, especially during disaster situations, these two groups were combined into the EDM/LEAPS DWG.

The objectives of this DWG are synergistic with the requirement and deliverables in the Testbed-16 Machine Modeling (ML) Thread. To facilitate access and formats of data for ML model training, validation and testing for the purpose of better managing emergencies such as wildland fires.

This situation is very common in the emergency response and disaster management sector where more ML and ML models are being developed and used. Hence the importance of this ER to the EDM/LEAPS DWG.

2.7. Document contributor contact points

All questions regarding this document should be directed to the editor or the contributors:

Contacts

Name	Organization	Role
Guy Schumann	RSS Hydro	Editor
Albert Kettner	Contractor to RSS Hydro	Contributor
Ignacio Correas	Skymantics	Contributor

2.8. Foreword

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

Chapter 3. References

The following normative documents are referenced in this document.

- OGC: OGC 06-121r9, OGC® Web Services Common Standard (2010) [https://portal.opengeospatial.org/files/?artifact_id=38867&version=2]
- OGC: OGC 06-042, OpenGIS Web Map Service (WMS) Implementation Specification version 1.3.0 (2006) [http://portal.opengeospatial.org/files/?artifact_id=14416]
- OGC: OGC 07-063r1, Web Map Services - Application Profile for EO Products (0.3.3) (2009) [http://portal.opengeospatial.org/files/?artifact_id=30912]
- OGC: OGC 12-111r1, OGC Best Practice for using Web Map Services (WMS) with Time-Dependent or Elevation-Dependent Data (1.0) (2014) [https://portal.opengeospatial.org/files/?artifact_id=56394]
- OGC: OGC 19-008r4, OGC GeoTIFF Standard version 1.1 (2019) [http://docs.opengeospatial.org/is/19-008r4/19-008r4.html]

Chapter 4. Terms and definitions

For the purposes of this report, the definitions specified in Clause 4 of the OWS Common Implementation Standard OGC 06-121r9 [https://portal.opengeospatial.org/files/?artifact_id=38867&version=2] shall apply. In addition, the following terms and definitions apply.

4.1. Abbreviated terms

CWFIS Canadian Wildland Fire Information System

EDM Emergency and Disaster Management

FWI Fire Weather Index

ML Machine Learning

MSE Mean Squared Error

NFIS National Forest Information System

NRCan Natural Resources Canada

SLD Styled Layer Descriptor

SSIM Structural Similarity Index

WMS Web Map Service

Chapter 5. Overview

Chapter 1 introduces the subject matter of this Testbed 16 OGC Engineering Report.

Chapter 2 provides an Executive Summary for the Testbed-16 ML Training Data activity.

Chapter 3 provides a reference list of normative documents.

Chapter 4 gives a list of the abbreviated terms and the symbols necessary for understanding this document.

Chapter 5 lists the content of each chapter (this chapter).

Chapters 6 to 9 contain the main technical details and recommendations of this ER. This section provides a high-level outline of the use case scenarios, followed by an in-depth description of the work performed and the challenges encountered, raising issues and discussing possible solutions.

Chapter 10 summarizes recommendations and suggests top-priority items for future work.

Annex A includes a history table of changes made to this document.

Chapter 6. Use Case Scenarios

6.1. Fuel Load Estimation (D136)

Problem statement: Explore interoperability challenges of training data for the wildfire use case specifically for fuel load estimation. In addition, solutions need to be developed that allow the wildland fire training data, test data, and validation data be structured, described, generated, discovered, accessed, and curated within data infrastructures.

6.2. Water Body Identification (D135)

Problem statement: Investigate how existing standards related to water resources, in conjunction with ML, can be used to locate potential water sources for wildland fire event response.

Chapter 7. Fuel Load Estimation: Data and Technical Details

7.1. Data analysis tools

7.1.1. Structural Similarity Index (SSIM)

The first challenge was data analysis. Specifically, how to compare two different variables, represented by their respective maps, to estimate their correlation. This is a basic step in feature selection. The process is to select those features that contribute most to the prediction variable or output in a Machine Learning algorithm. This process helps finding relevant features, leaving irrelevant features out of the training data set.

One of the most common methodologies used in data analysis is Mean Squared Error (MSE). This error estimate measures the average of the squares of the errors between two variables. However, as this error estimate only considers local values and ignores shapes of isolines, which is one of the main characteristics in maps, this method is unsuited for map comparison. For example, two maps showing very similar isolines but different absolute values will be considered unrelated by MSE methodology.

In order to overcome this challenge, the participants researched new methodologies in the field of image processing. They determined that Structural Similarity Index (SSIM) [https://en.wikipedia.org/wiki/Structural_similarity] was considered suitable for the task. The SSIM index is a method for measuring the similarity between two images. The SSIM index can be viewed as a quality measure of one of the images being compared provided that the other image is regarded as of perfect quality. This is a method for predicting the perceived quality of digital television and cinematic pictures, as well as other kinds of digital images and videos. However, SSIM can be applied to find common patterns and contours between two images and estimate their degree of similarity.

As displayed in the following example, when applying SSIM to comparing two different images the value will be indicative of the similarity of the images. When applying SSIM to maps, the hypothesis is that SSIM will be a better indication than MSE-related metrics for variable correlation or for the accuracy of ML predictions.

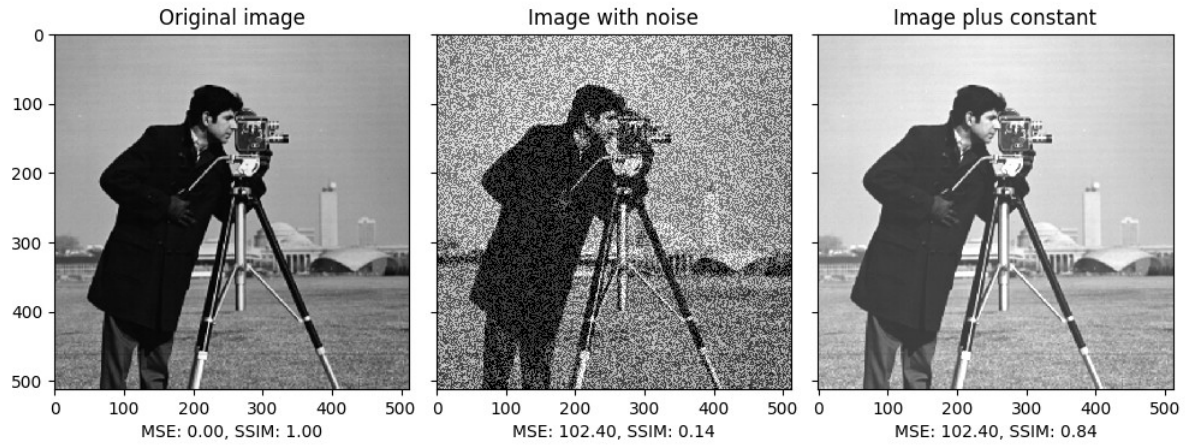


Figure 1. Example of MSE and SSI methodologies when comparing images. SSI clearly indicates high similarity when image lighting was adjusted and low similarity when an image was noisy. MSE provided the same metrics in both cases. (Source: scikit-image.org)

7.1.2. SSIM applied to Petawawa data

The SSIM method was tested with the Web Map Service (WMS) accessible Petawawa data. The Petawawa forest dataset provides a series of maps displaying the values of several measurements, such as volume, height or biomass. The dataset is for a relatively small and compact area but offers up to 26 different measurements with a spatial resolution of 30 meters and the quality of the data is assumed to be very high: that is, close to the real value distribution. Further, the maps come directly as gray scale that can be analyzed directly.

Both methods (MSE and SSIM) were run for all maps. SSIM is an index, topped at 1.0, which indicates identical maps. Values over 0.9 indicate a high similarity (90 percentile), and lower than 0.5 indicate very low similarity (10 percentile). In contrast, MSE is an absolute value, with 0 indicating identical images, values under 500 indicate very few differences (10 percentile), and over 5000 indicate substantial differences (90 percentile).

The interest in using SSIM becomes obvious when analyzing maps that show related variables, such as Lorey's height, the weighted mean height whereby individual trees are weighted in proportion to their basal area (RF_PRF_LOREYSHT), in relation to Co-dominant - dominant height (RF_PRF_CD_HT) and Top height, the average of the largest 100 trees per ha (RF_PRF_TOPHT). These variables all measure forest heights and should show a strong correlation.

When comparing Lorey's height in relation to Co-dominant - dominant height, both methods indicate high similarity, with SSIM value at 0.93 (very close to the maximum 1.0) and MSE at 161 (well below the 10th percentile value of 500).

NOTE	Please note in the following figures that each SSIM value and each MSE value represents a comparison of the two images (rather than the SSIM value being applied to one image and the MSE value being applied to the other).

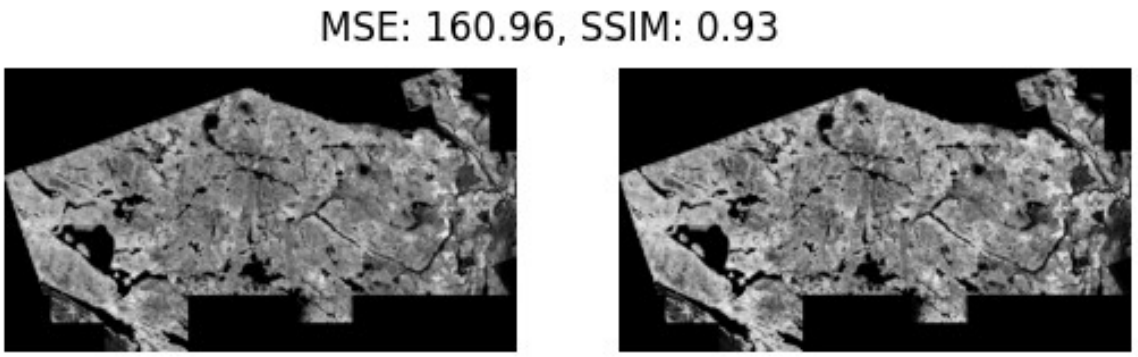


Figure 2. Lorey's height (SSIM) and Co-dominant - dominant height (MSE) both indicate strongly similar images

However, when comparing Lorey's height in relation to Top height, only SSIM, with a value of 0.92, indicates strong correlation. MSE gives a value of 1036, indicating that there might be some correlation. The reason for this misalignment is due to the fact that in spite of both maps showing similar shapes, the absolute values of the variables differ significantly.

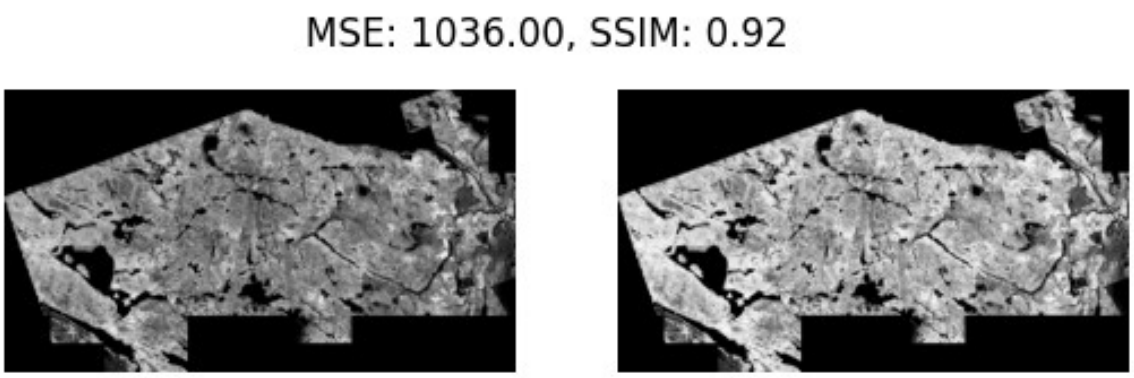


Figure 3. Lorey's height (SSIM) indicates strongly similar images, but Top height (MSE) indicates dissimilarity

Similar issues arise when comparing unrelated variables. For example, when comparing Top height (RF_PRF_TOPHT) in relation to Basal area in the Pole tree size class (10-24cm) (RF_PRF_BAPOLES), both methods indicate very low correlation or none at all, with values of 6746 for MSE and 0.39 for SSIM.

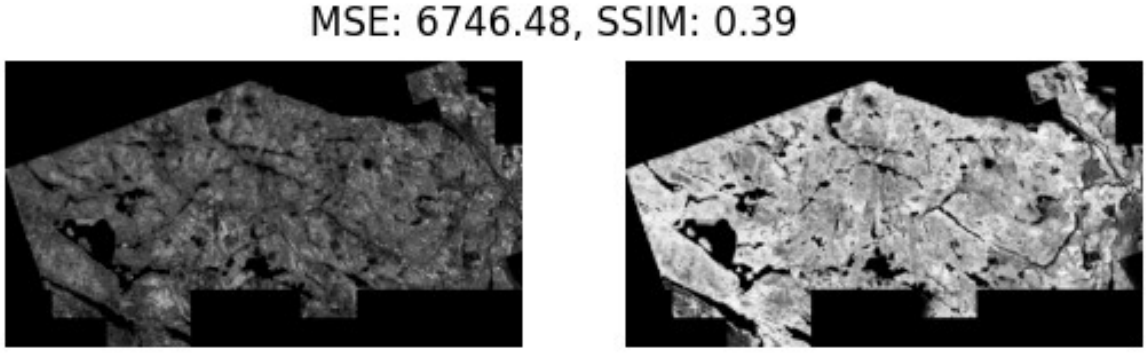


Figure 4. Both Top height (MSE) and Basal area (SSIM) indicate dissimilarity

However, when comparing Basal area in relation to Quadratic mean diameter for all trees >9.1cm

(RF_PRF_DBHQMERCH_MASKED) only SSIM results, with a value of 0.37, clearly indicate no correlation. The MSE value of 1666 which, although not pointing towards a clear correlation, does not reject the relationship either.

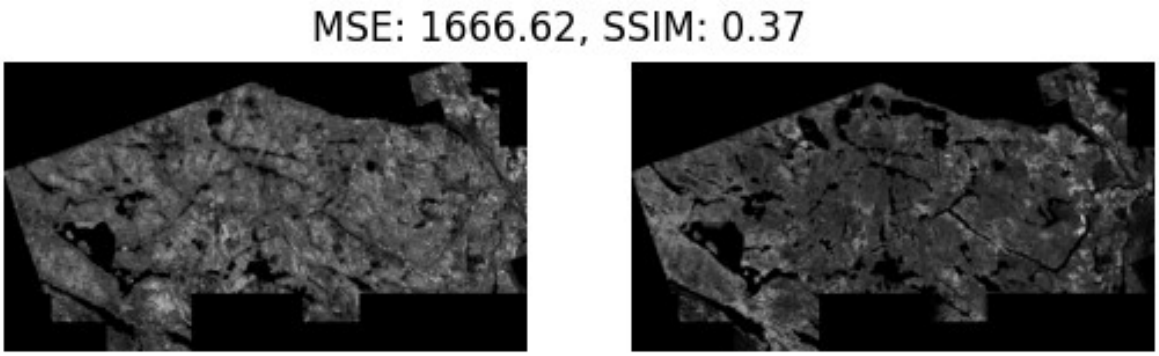


Figure 5. Basal area (SSIM) indicates dissimilarity while Quadratic mean diameter (MSE) is borderline

7.1.3. SSIM applied to Canadian Wildland Fire Information System (CWFIS)

All the previously described tests were carried out using the WMS accessible Petawawa dataset. This is high quality data with one continuous variable per layer and with maps served in grayscale. However, this dataset's geographic extent is very limited and conclusions therefore do not necessarily extrapolate to countrywide maps.

The CWFIS offers an alternative source of data to analyze the available data and showcase the Structural Similarity Index. The CWFIS maps are countrywide and many refer to discrete variables with their own color code, as ranges of the real-world variable. Moreover, many of these maps are related, such as the Fire Weather Index System or the Fire Behavior Prediction System, and offer a good testbed to analyze correlations.

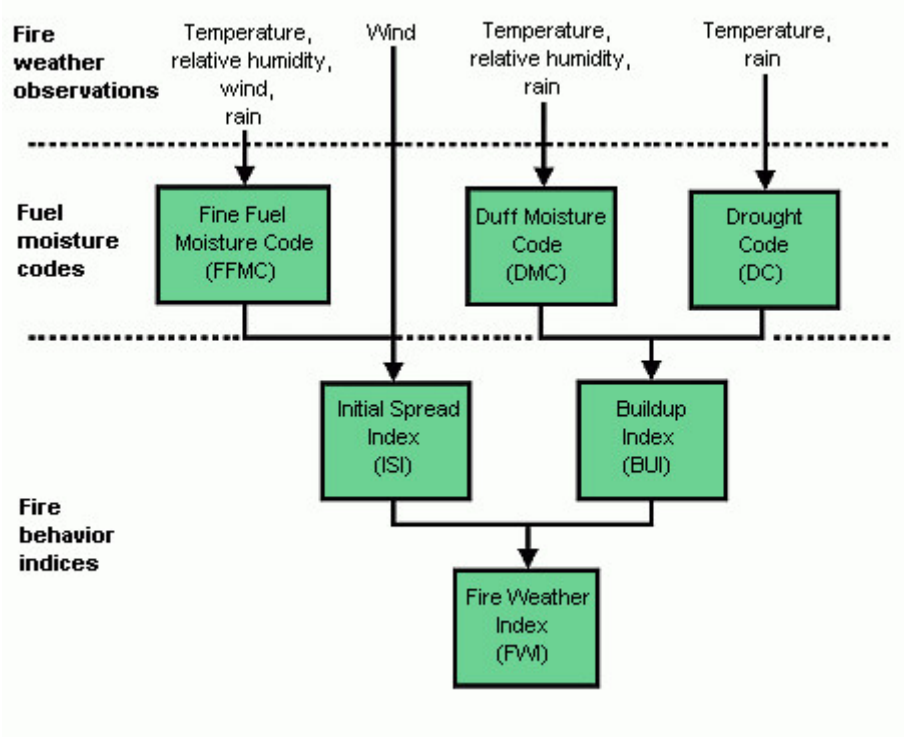


Figure 6. Fire Weather Index (FWI) structure and relations between variables/maps. (Source: NRCan)

The first challenge analyzing CWFIS maps is to convert them to a scale that is numerically significant and comparable by an algorithm. Maps in CWFIS have color codes that need to be transformed to a grayscale in which the higher volume according to the color code corresponds to either white or black and vice-versa. This challenge can be overcome as the CFWIS WMS offers a GetLegendGraphic option to associate a color to a value range, which then returns a "png" image with the map legend. This option is useful for visual inspection of the data. Alternatively, a WMS instance could offer a Styled Layer Descriptor (SLD) document. This is an XML file encoding of the legend information in a machine-readable format and accessible using a GetStyles operation. The SLD document can describe rules for rendering, and these rules can contain MinScaleDenominator and MaxScaleDenominator elements specifying the numerical values of the scale ranges. This is a valid method to communicate scale values in a machine-readable format using OGC standards. Both GetLegendGraphic and GetStyles operations are specified in an extension to the WMS standard. This extension is called OpenGIS SLD Specification [https://portal.opengeospatial.org/files/?artifact_id=1188], which is an optional extension.

Unfortunately, this extension does not seem to be fully implemented in NRCan's services. The GetLegendGraphic operation works properly and provides a descriptive "png" image that a user can visually interpret. However, when the GetStyles operation is used to retrieve an SLD document, the retrieved document is empty. This means that an ML algorithm does not have a numerical reference to interpret the colors and shades in a map and cannot make proper inferences and regressions. In order to overcome this limitation, the color value is considered the value of the variable.

In the following examples, the lowest value is set to black and the highest to white, while the remaining gray gradation is assigned evenly through the different ranges. The following maps show the gray-scaled maps for Wind Speed, Fine Fuel Moisture Code and Initial Spread Index.

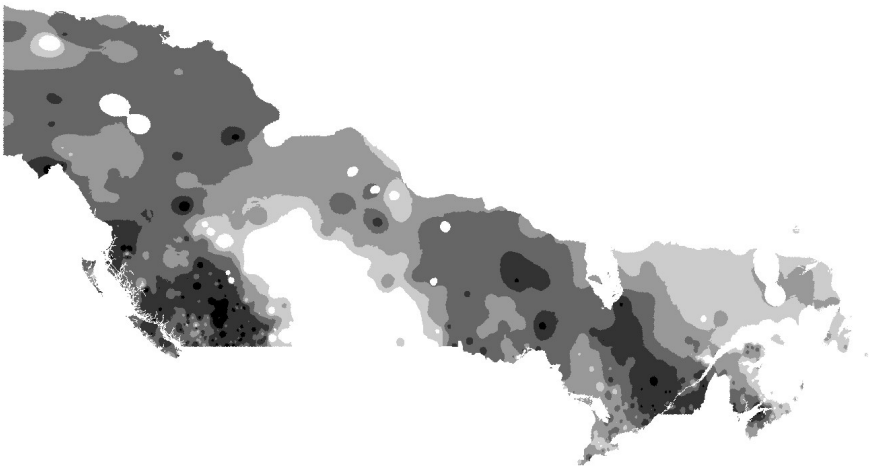


Figure 7. CWFIS Wind Speed.

Figure 8. CWFIS Fine Fuel Moisture Code.

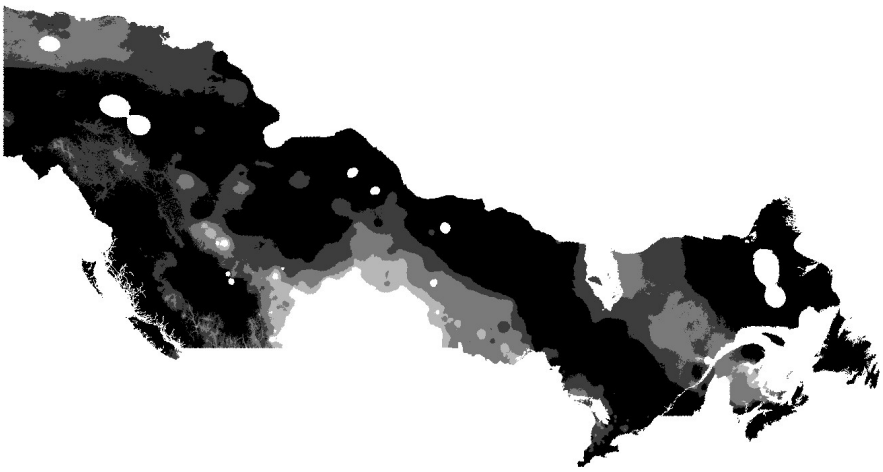


Figure 9. CWFIS Initial Spread Index.

According to the Fire Weather Index structure graph, the combination of Wind Speed (WS) and Fine Fuel Moisture Code (FFMC) maps generates the Initial Spread Index (ISI) map. This derived map when combined with the Buildup Index (BUI) generates the Fire Weather Index (FWI). Applying both MSE and SSIM methods indicates a strong correlation between FFMC and ISI (0.96 according to SSIM and 114 according to MSE) and between ISI and FWI (0.97 according to SSIM and 70 according to MSE). It is even possible to apply a simple regression and reverse-engineer the generation of ISI, as depicted in the following map, with a result pretty close to the original ISI map. Comparing this result to the original ISI map it scores relatively well (0.96 and 80 respectively).

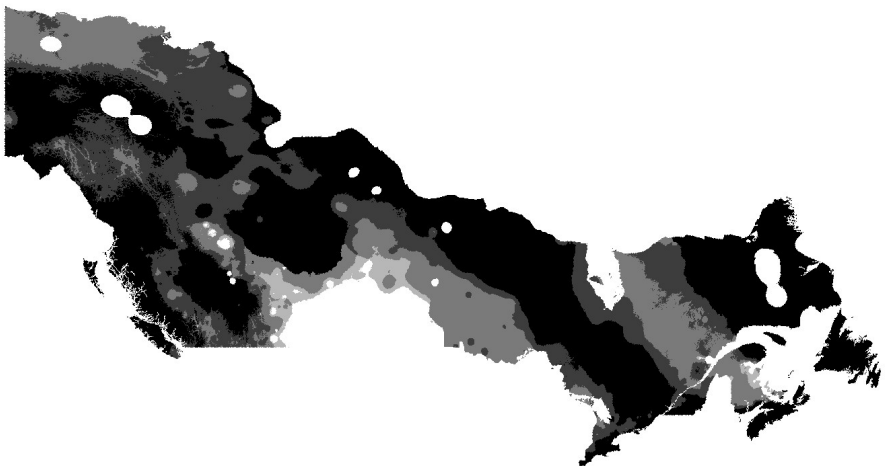


Figure 10. CWFIS reverse-engineered Initial Spread Index.

MSE and SSIM methods applied to all CWFIS maps draw consistent correlations, grouping maps by families according to their structural similarity. For example, Head Fire Intensity (HFI) is closely related to Rate of Spread (RoS) and Total Fuel Consumption (TFC) but they are also loosely related to FFMC, ISI or WS. This makes sense as HFI, RoS and TFC belong to the Canadian Forest Fire Behavior Prediction (FBP) System and FFMC, ISI and WS belong to the Canadian Forest Fire Weather Index (FWI) System).

7.1.4. Conclusions

- Feature selection requires reliable methods to compare variables and estimate correlations. SSIM, used in the field of image processing, has been tested to compare it with traditional methods such as MSE.
- MSE seems to perform well in simple, discrete-variable maps but is less accurate finding correlations in continuous-variable maps. SSIM is better at finding common shapes even with different absolute values, but SSIM is sensitive to noise.
- Although access to a graphical legend is possible for discrete-variable maps to visually interpret them, machines require access to machine-readable formats through WMS operations, such as accessing a descriptive SLD document through GetStyles operation. However, these are not always implemented and could not be tested in Testbed 16. Continuous-variable maps represent a similar challenge, as a machine needs to interpret the values of the different shades of grey but there is no available machine-readable format. The underlying issue seems to be related to the fact that the GetStyles operation is optional and is specified in an optional extension to WMS, weakly supported in the OGC web services interface standards.
- Discrete-variable maps are often just a simplification for humans, grouping values in ranges so that our eyes are able to find patterns in the isolines. However, machines are able to process more detailed maps, combine several sources and find more sophisticated patterns, making continuous-variable maps more interesting for machine learning applications.
- Existing OGC Web Services were successfully used to retrieve different maps and evaluate their level of correlation, with the aforementioned limitations. This evaluation was done programmatically and was performant, requiring a few seconds to complete for all the layers in the Canadian Wildland Fire Information System with each other. If required it should be feasible to execute automatically prior to training a Machine Learning algorithm.

7.1.5. Recommendations

- Continue research on image processing methods applied to map analysis, such as SSIM, that seem to provide more accurate results than other methods, such as MSE. These methods can eventually facilitate feature selection and automate part of the process of building new training datasets.
- Review the implementations of WMS services in NRCan and add an SLD document describing scale ranges, retrievable through GetStyles operation, to allow machine learning algorithms properly retrieve and interpret the data.
- Evaluate the possibility of creating new sources of data by turning the discrete-variable maps into continuous-variable ones, more suitable for machine learning applications.

7.2. Ground truth selection for fuel load estimation

Training a Machine Learning algorithm requires a ground truth dataset to serve as a reference. For the challenge of Fuel Load estimation, there are two available sources in the NRCan datasets: Petawawa (RF_PRF_BIOMASS_KG_DRY_MASKED layer) and the National Forest Information System (NFIS) [https://ca.nfis.org/index_eng.html] (tot_bio_r layer).

The Petawawa dataset is both accurate and of high resolution. However, the geographic area of coverage is small and too limited in tree species to provide a general training base. Therefore, extrapolating to countrywide area is not possible. Visualization of the Petawawa area shows smooth gradients and consistency as depicted in the image below.

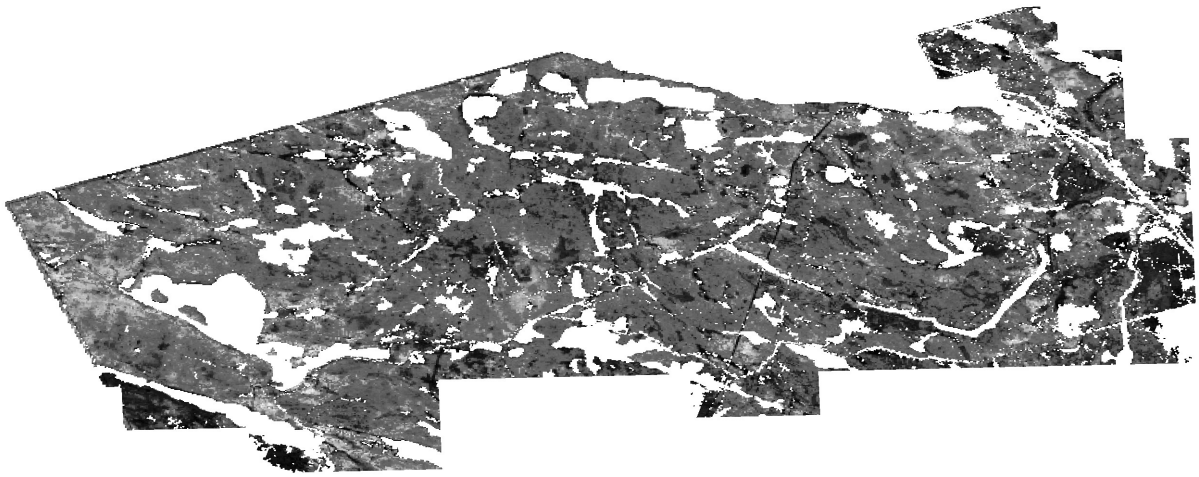


Figure 11. Petawawa Fuel Load

The NFIS [https://nfi.nfis.org/en/] dataset is very extensive, covering Canada with high resolution content and is the best candidate to use as the ground truth dataset for Fuel Load estimations. However, a close inspection shows a noisy map. This raises doubts about the accuracy of the data. The image below depicts the Petawawa area with NFIS Fuel Load data.

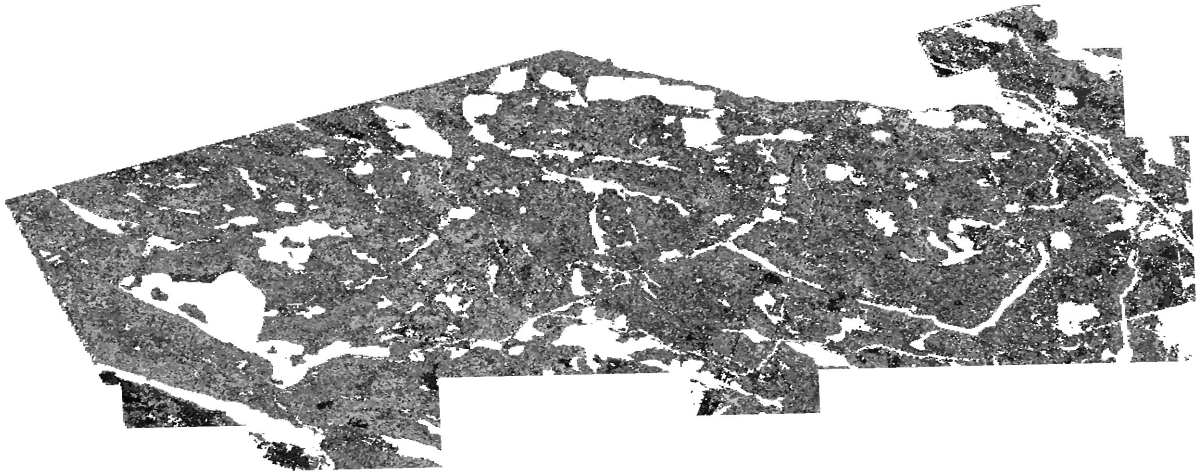


Figure 12. NFIS Fuel Load (Petawawa area)

Assuming that Petawawa data is highly accurate, and taking into consideration the results in SSIM applied to Petawawa data, comparing both maps should give values under 500 in MSE and values over 0.9 in SSIM. However, the actual computed values are out of these thresholds, with an MSE value of 1256 and a SSIM value as low as 0.59. The conclusion is that the values of NFIS Fuel Load dataset are not too accurate in absolute value and their variation with regards to the ground truth value is very high. As a consequence, there is no optimal dataset for Fuel Load ground truth, as the Petawawa dataset is too small and the NFIS dataset is pretty inaccurate.

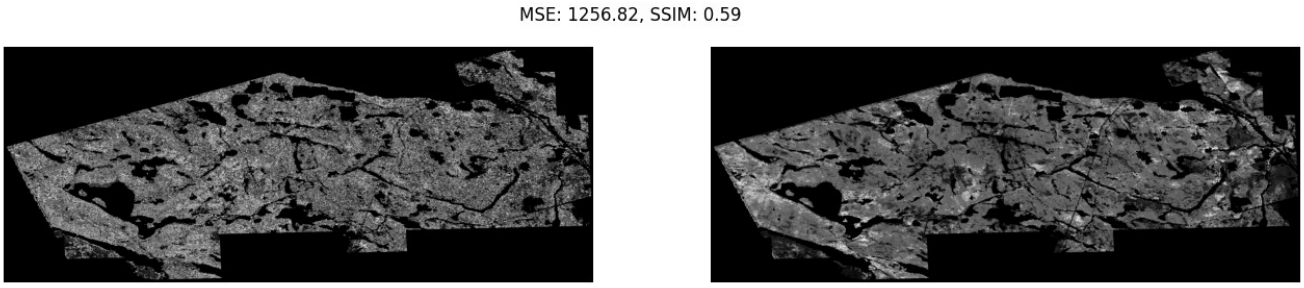


Figure 13. Comparison of NFIS and Petawawa datasets (in the Petawawa area)

For the purpose of this testbed, model accuracy is not important. That said, the goal is to test the process to automate the training of a ML algorithm. From this point of view, the NFIS Fuel Load dataset offers a valid base to work. The generation of the NFIS Fuel Load dataset is documented in the research paper Three decades of forest structural dynamics over Canada's forested ecosystems using Landsat time-series and lidar plots [https://www.sciencedirect.com/science/article/pii/S0034425718303572] (the "Three Decades Paper").

According to this paper, combining LIDAR-derived forest structure with time-series satellite imagery enables inferring countrywide annual forest structure estimates using a nearest neighbor imputation approach with a Random Forests-based distance metric with a relatively high level of accuracy. In particular, above ground biomass per hectare is calculated by summing the values of all trees within a plot and dividing that value by the area of the plot using species-specific equations. The calculated above ground biomass may be separated into various biomass components (e.g., stem, bark, branches, foliage). According to the model, the accuracy measured in terms of coefficient of determination (R^2) is 0.699, which seems reasonable for a first approach. The following image, extracted from the aforementioned research paper, shows the workflow for the modelling and mapping of the forest attributes.

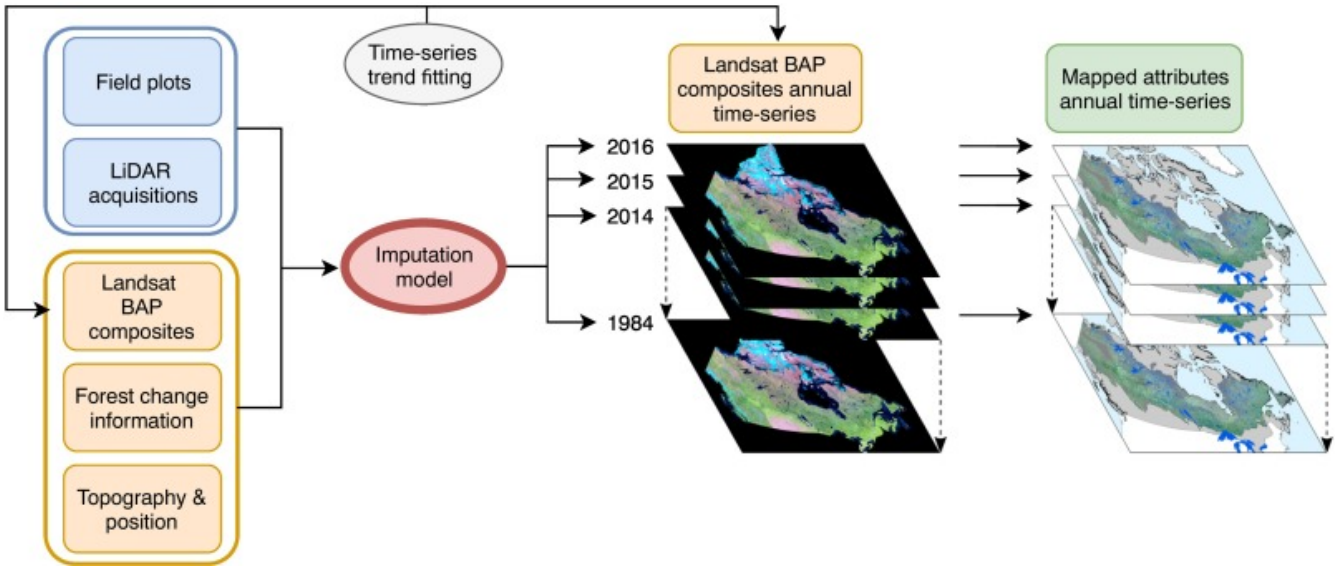


Figure 14. Workflow for the modelling and mapping of the forest attributes to generate the NFIS Fuel Load dataset (source: ScienceDirect)

7.2.1. Conclusions

- The *Three Decades Paper* provides a promising base to explore the generation of accurate Canadian annual forest structure estimates that can be used as a test to feed different Ml models. However, as demonstrated when comparing with the more accurate Petawawa dataset, the accuracy of the inferred dataset is not provable. If NRCan is interested in inferring the fuel load countrywide at any time with a sufficient degree of accuracy, new datasets would be required to form the ground truth.
- Access to the research paper was critical in understanding the validity of the dataset. Similarly, the level of accuracy of the data was essential in understanding the validity of the ground truth data. This information needs to be easily available together with the dataset.
- With regard to current testbed results, training a ML algorithm using the NFIS Fuel Load dataset (generated in summer 2015) and satellite imagery from a similar date to infer current fuel load using current satellite imagery should be possible.
- Unfortunately, the exact model and algorithms to infer the fuel load described in the research paper were not available and could not be evaluated for improvement in Tested 16.

7.2.2. Recommendations

- In order to be able to train ML algorithms with a high degree of accuracy, having a solid ground truth dataset is crucial. This ground truth dataset does not need to cover the whole country, but it should be representative of different tree species and ecosystems and be accurate. So, instead of measuring the whole country at a high cost, a few small but varied forests could be measured in a very accurate manner providing the ground truth to train a ML algorithm in combination with satellite imagery. Once trained, the ML algorithm would be able to infer the measurements for the rest of the country using just the satellite imagery.
- Providing the accuracy information in the metadata of each dataset that is meant for training is highly recommended. Providing additional information to help understand the process by which the data was generated is also recommended.

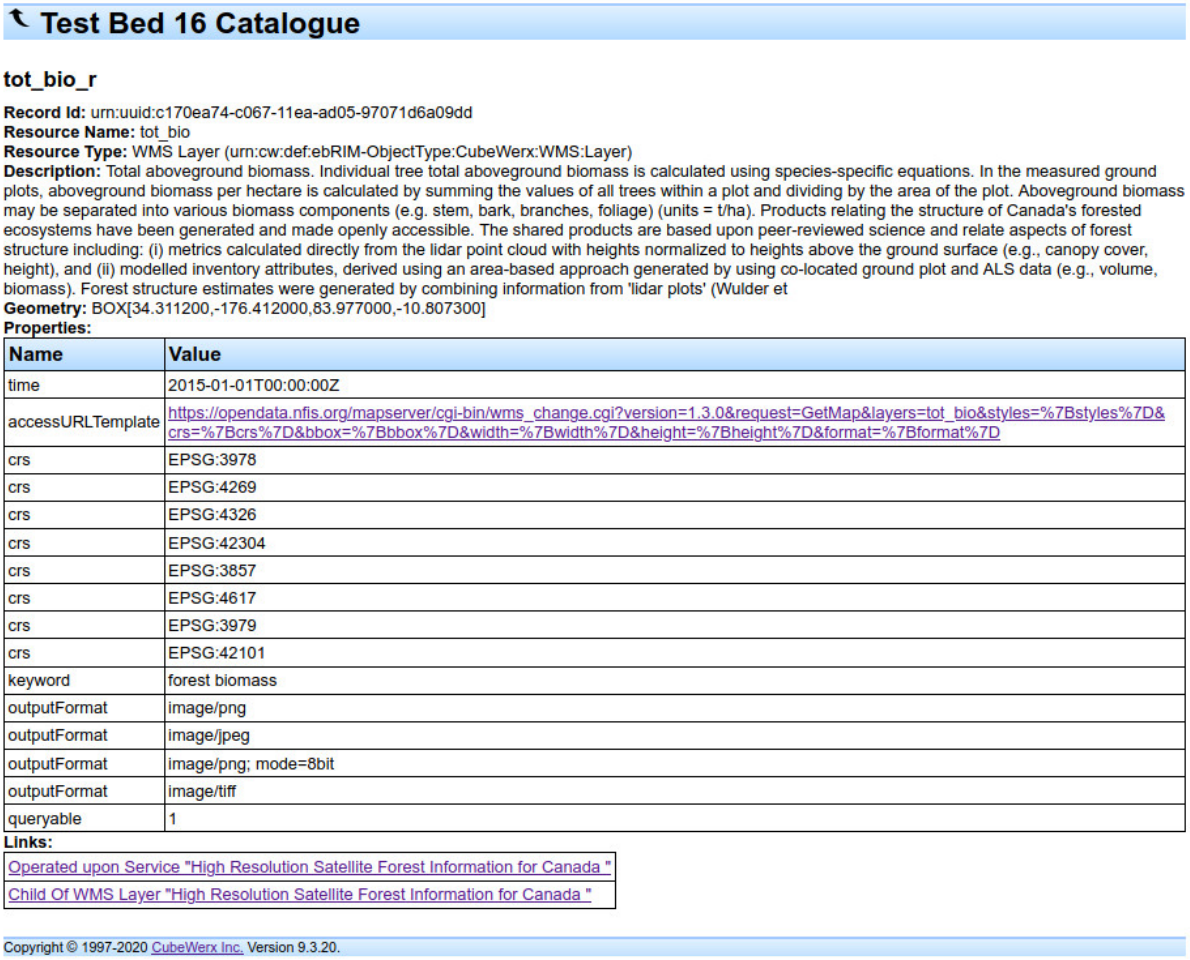
- Too often ML algorithms are improved in an iterative manner. Having a standard way to store and reuse a model (for example, using ONNX [https://onnx.ai/] or MXNet [https://mxnet.apache.org/] formats, evaluated in other sections of this Testbed16 task) could greatly accelerate and improve the development. In this case, having access to the model developed in the previously-mentioned research paper could have facilitated the evaluation.

7.3. Dataset discovery and training process

In order to implement the discovery and training process, Cubewerx deployed an experimental prototype of an OGC API - Records catalogue. The catalogue was populated by harvesting several WMS services offered by NRCAN, namely CWFIS and NFIS, as well as SENTINEL-1 [https://sentinel.esa.int/web/sentinel/missions/sentinel-1] imagery data. The landing page or entry point for the catalogue can be found at this URL [https://eratosthenes.pvretano.com/cubewerx/cubeserv/default/ogcapi/catalogues/collections/tb16cat].

The Cubewerx server supports both JSON and HTML output. The catalogue supports a variety of query parameters such as "q" (Space-separated list of query terms), DateTime (a time period to search), bbox (bounding box), lat/lon/radius (proximity search) and filter (Common Query Language filter).

Additionally, the experimental catalogue provides information on the OGC Web Service that serves the data, as well as a URL template to retrieve the data, description, bounding box, output formats and other useful information to query data from the original service. All this information is sufficient for a data analyst to find and discover useful datasets and to implement the extraction of data. The following image depicts a screenshot of the information provided for layer tot_bio_r in NFIS.



Test Bed 16 Catalogue

tot_bio_r

Record Id: urn:uuid:c170ea74-c067-11ea-ad05-97071d6a09dd
Resource Name: tot_bio
Resource Type: WMS Layer (urn:cw:def:ebRIM-ObjectType:CubeWerx:WMS:Layer)
Description: Total aboveground biomass. Individual tree total aboveground biomass is calculated using species-specific equations. In the measured ground plots, aboveground biomass per hectare is calculated by summing the values of all trees within a plot and dividing by the area of the plot. Aboveground biomass may be separated into various biomass components (e.g. stem, bark, branches, foliage) (units = t/ha). Products relating the structure of Canada's forested ecosystems have been generated and made openly accessible. The shared products are based upon peer-reviewed science and relate aspects of forest structure including: (i) metrics calculated directly from the lidar point cloud with heights normalized to heights above the ground surface (e.g., canopy cover, height), and (ii) modelled inventory attributes, derived using an area-based approach generated by using co-located ground plot and ALS data (e.g., volume, biomass). Forest structure estimates were generated by combining information from 'lidar plots' (Wulder et
Geometry: BOX[34.311200,-176.412000,83.977000,-10.807300]
Properties:

Name	Value
time	2015-01-01T00:00:00Z
accessURLTemplate	https://opendata.nfis.org/mapserver/cgi-bin/wms_change.cgi?version=1.3.0&request=GetMap&layers=tot_bio&styles=%7Bstyles%7D&crs=%7Bcrs%7D&bbox=%7Bbbox%7D&width=%7Bwidth%7D&height=%7Bheight%7D&format=%7Bformat%7D
crs	EPSG:3978
crs	EPSG:4269
crs	EPSG:4326
crs	EPSG:42304
crs	EPSG:3857
crs	EPSG:4617
crs	EPSG:3979
crs	EPSG:42101
keyword	forest biomass
outputFormat	image/png
outputFormat	image/jpeg
outputFormat	image/png; mode=8bit
outputFormat	image/tiff
queryable	1

Links:

Operated upon Service "High Resolution Satellite Forest Information for Canada "
Child Of WMS Layer "High Resolution Satellite Forest Information for Canada "

Figure 15. Screenshot of TB16 OGC API Records

The dataset discovery and training process designed and tested during this testbed is summarized in the following sequence diagram:

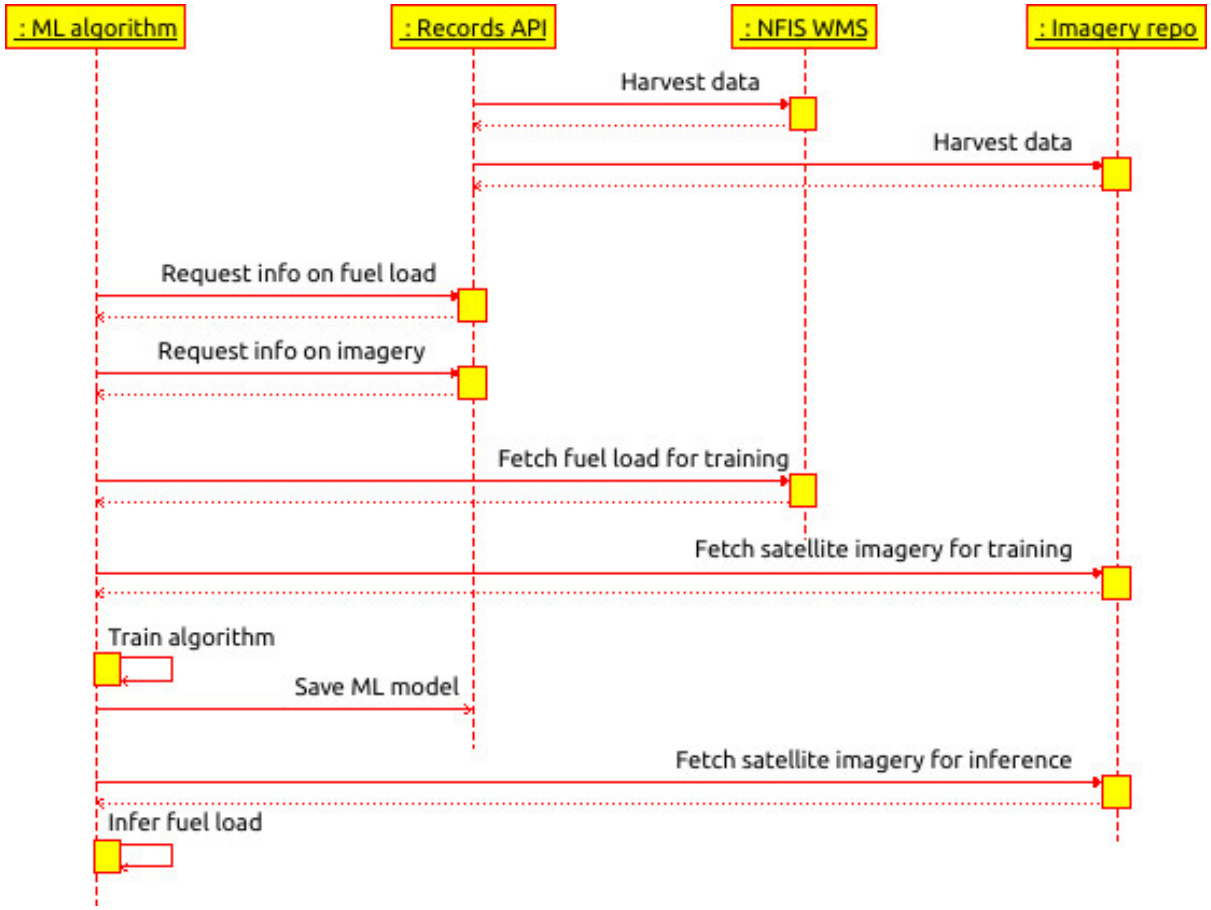


Figure 16. Dataset discovery and training process

The process is divided in five stages:

1. Catalogue gets populated by harvesting data from available services. A data analyst can then browse the catalogue and discover the useful datasets for the application and implement the data retrieval processes in the ML algorithm.
2. ML algorithm requests info from catalogue, including layers and descriptions, data type, styles and value ranges, DateTime, Coordinate Reference System (CRS), output format and URL template.
3. ML algorithm retrieves data for training using the information retrieved in the previous step.
4. ML algorithm gets trained and saves model in catalogue for future re-usage, using one of the standard formats (such as ONNX or MXNet).
5. ML algorithm can now infer new data.

7.3.1. Issues and limitations

- An issue was found when accessing CWFIS WMS via python script and OWSLib [https://geopython.github.io/OWSLib/] (a Python package for client programming with OGC web service interface standards) as the service did not implement the title attribute in the style definitions. This issue was quickly fixed by NRCan's staff.
- A critical issue was found when retrieving the value ranges from the WMSs. NRCan servers offer a legendUrl for some layers, which allow understanding the value ranges in a graphical way. The following image shows an example of a legend graphic. However, in order to do this automatically - that is programmatically - the catalogue tries to get a non-graphic representation

of the legend in the form of a SLD [https://www.ogc.org/standards/sld], an XML schema specified by OGC for describing the appearance of map layers. This descriptor is fetched via a GetStyles operation, which is supported by NRCan servers. However, when the catalogue gets the style, an empty document is returned.

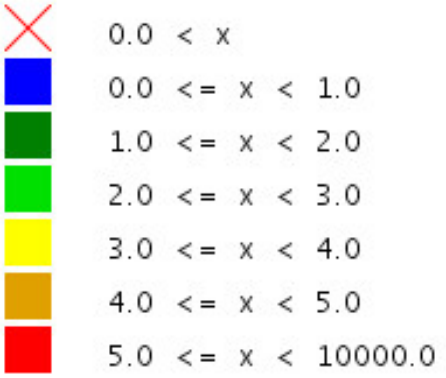


Figure 17. Example of legend (Surface Fuel Consumption)

- The NFIS Fuel Load layer does not provide a Style, so even manually interpreting the different shades and associating them with a number is not possible. There is not even information to understand whether lighter means a higher or a lower value. Eventually, the goal for the ML algorithm is to infer what color shades should correspond to each pixel in the map. Obviously, however, any ML model will be severely limited if it is not possible to numerically interpret the values in a training dataset.
- As date and time information will indicate which other datasets to fetch for training, the value for DateTime is critical in the ground truth dataset. The information in the NFIS Fuel Load layer indicated that the DateTime corresponded to January 1st, 2015 and the ML algorithm should then fetch satellite imagery of a similar date (winter imagery). However, the description on how the layer data was generated clearly specified that the proper date was around July 31st, 2015 which would correspond to Summer imagery. This is very different from the original conclusion.

7.3.2. Conclusions

The prototype OGC API - Records implementation was considered a useful API to browse and discover datasets, as well as provide all the required information to implement data retrieval by a ML algorithm.

Additionally, for a dataset to be useful as a source to train a ML algorithm the dataset needs the following attributes:

1. An extensive description that allows the user/application to fully understand the data, including the data provenance or the way the data can be interpreted.
2. The accuracy of the data in both human- and machine-readable format.
3. Values assigned to colors or color shades in a machine-readable format.
4. Accurate DateTime field for the time when the dataset was created.

NRCan's existing OGC Web Services can be useful for ML applications as long as they comply with the above requirements.

7.3.3. Recommendations

- When the OGC Membership formally approves the OGC API – Records, consider adopting that API as a catalogue for NRCan's web services. The API is compatible with NRCan's existing OGC Web Services and its versatile query capabilities can make it a useful complement to the Geospatial Web Services Harvester [https://www.nrcan.gc.ca/science-data/science-research/earth-sciences/geomatics/canadas-spatial-data-infrastruct/geospatial-web-services/19359].
- Review the implementations of existing WMS services and add an SLD through GetStyles operation to allow Machine Learning algorithms to properly retrieve and interpret the data.
- Make sure that DateTimes are properly and accurately set in datasets. Errors in datasets could lead to training machine learning algorithms with the wrong satellite imagery and thus inaccurate inferences.
- Consider the adoption of new OGC APIs to ease the coding of values in SLDs. The following shows a sample SLD for a raster layer from the GTOPO30 layer on CubeWerx's test server that illustrates how the values of a layer could be implemented in a machine-readable format.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!-- Generated by CubeWerx Suite 9.3.17 on Wed, 02 Sep 2020 11:20:31 -0400 -->
```

```
<sld:StyledLayerDescriptor version="1.1.0" xmlns="http://www.opengis.net/se"
 xmlns:sld="http://www.opengis.net/sld" xmlns:se="http://www.opengis.net/se"
 xmlns:exp="http://www.opengis.net/ogc"
 xmlns:xlink="http://www.w3.org/1999/xlink"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.opengis.net/sld
http://schemas.cubewerx.com/schemas/sld/1.1.0-cw/StyledLayerDescriptor.xsd">
```

```
<sld:NamedLayer>
  <Name>gtopo30</Name>
  <Description>
    <Title>GTOPO30</Title>
  </Description>
  <sld:UserStyle>
    <Name>default</Name>
    <Description>
      <Title>Default</Title>
    </Description>
    <sld:IsDefault>true</sld:IsDefault>
    <FeatureTypeStyle>
      <FeatureTypeName>gtopo30</FeatureTypeName>
      <Rule>
        <RasterSymbolizer>
          <ColorMap>
```

```
            <exp:Interpolate method="color">
              <exp:LookupValue>
                <exp:PropertyName>Rasterdata</exp:PropertyName>
              </exp:LookupValue>
              <exp:InterpolationPoint>
                <exp:Data>-500</exp:Data>
                <exp:Value>#00FF00</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>-1</exp:Data>
                <exp:Value>#64F014</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>0</exp:Data>
                <exp:Value>#7DEB32</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>30</exp:Data>
                <exp:Value>#78C818</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>105</exp:Data>
                <exp:Value>#38840C</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>300</exp:Data>
                <exp:Value>#2C4B04</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>400</exp:Data>
                <exp:Value>#FFFF00</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>700</exp:Data>
                <exp:Value>#DCDC00</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>1200</exp:Data>
                <exp:Value>#B47800</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>1400</exp:Data>
                <exp:Value>#C85000</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>1600</exp:Data>
                <exp:Value>#BE4100</exp:Value>
              </exp:InterpolationPoint>
              <exp:InterpolationPoint>
                <exp:Data>2000</exp:Data>
                <exp:Value>#963000</exp:Value>
```

```
            </exp:InterpolationPoint>
            <exp:InterpolationPoint>
              <exp:Data>3000</exp:Data>
              <exp:Value>#3C0200</exp:Value>
            </exp:InterpolationPoint>
            <exp:InterpolationPoint>
              <exp:Data>5000</exp:Data>
              <exp:Value>#F0F0F0</exp:Value>
            </exp:InterpolationPoint>
            <exp:InterpolationPoint>
              <exp:Data>13000</exp:Data>
              <exp:Value>#FFFFFF</exp:Value>
            </exp:InterpolationPoint>
          </exp:Interpolate>
        </ColorMap>
        <ShadedRelief/>
      </RasterSymbolizer>
    </Rule>
  </FeatureTypeStyle>
 </sld:UserStyle>
</sld:NamedLayer>
```

```
</sld:StyledLayerDescriptor>
```

Chapter 8. Water Body Identification: Data and Technical Details

8.1. Training dataset

8.1.1. Label data

Training data, also known as 'initial data' or 'sample data' are needed in ML to build mathematical models that can then be automatically detect certain features in the data. High quality training data is essential to produce reliable mathematical models. However, establishing these initial data sets can be incredibly daunting. One of the tasks in OGC Testbed 16 was to determine how well OGC standards can support ML in the process of building algorithms to automatically detect waterbodies in Sentinel-1 scenes.

For training datasets two earlier established waterbody databases were used:

A) WFS3 - Quebec_Rivers_and_Lake_TB15_ML_D104 [https://opendata.nfis.org/geoserver/OGC_TB15_ML_D104/wfs3/collections?f=text%2Fhtml].

B) WMS - Lakes, Rivers and Glaciers in Canada - CanVec Series - Hydrographic Features [https://open.canada.ca/data/en/dataset/9d96e8c9-22fe-4ad2-b5e8-94a6991b744b].

The Quebec Rivers and Lakes database (database A) created as part of the OGC Testbed15 ML activity was not regarded as it did not contain a sufficient number of labeled waterbodies. The labeled hydrological features, one of the 8 themes of the feature classes, included in the CanVec Series (dataset B) are watercourses, water linear flow segments, hydrographic obstacles (falls, rapids, etc.), waterbodies (lakes, watercourses, etc.), permanent snow and ice features, water wells and springs. This NRCan vector product is considered the best available geospatial data source covering Canada. The Hydrographic features theme is considered quality vector geospatial data (current, accurate, and consistent) of Canadian hydrographic phenomena. CanVec aims to offer a geometric description and a set of basic attributes on hydrographic features that comply with international geomatics standards, seamlessly across Canada. According to the metadata provided via the CanVec WMS endpoint, the hydrological features are from:

"the best available data sources covering Canadian territory, offers quality topographical information in vector format, and complies with international geomatics standards. CanVec is a multi-source product coming mainly from the National Topographic Data Base (NTDB), the Mapping the North process conducted by the Canada Center for Mapping and Earth Observation (CCMEO), the Atlas of Canada data, the GeoBase initiative, and the data update using satellite imagery coverage (e.g. Landsat 7, Spot, Radarsat, etc.)"

Although the CanVec hydrological features is a very comprehensive, state-of-the-art hydrological product, using these data as label data for training datasets might be less ideal for several reasons: 1) Merged product from different sources, 2) No date range of which imageries are used to produce this labeled dataset are provided, and 3) Metadata on uncertainty or data accuracy is missing. To be more specific:

1. Data from different satellites used to produce a merged labeled dataset makes it impossible to pair the one satellite scene with the identified waterbody feature. In the end, these pairs are what is required to have in the training dataset.
2. The metadata does not indicate the period for which imagery were analyzed. For dynamic features (like flooding or wildfires), exact date and time of each image that is used to create the labeled data is needed. Waterbody feature data is less dynamic and therefore the time-date component is maybe less impactful, but still these waterbodies could vary in area over time.
3. Using remote sensing to delineate waterbodies, there is a degree of uncertainty associated with the technique, especially for grid cells that reflect the boundary between water and land. Furthermore, vegetation can make this boundary fuzzier. This uncertainty should be captured in the metadata as ML can take this into account during the training phase of the model and as such the ML model would become more robust.

Water bodies might be less dynamic over time but Figure 18 illustrates how delineation of water bodies can differ over time or by using different datasets while extracting the same water body. Making additional metadata available through the WMS endpoint that includes what raw data sources were used over which period would most likely always improve the accuracy of the ML model.

Technical details

Both labeled and training datasets were provided as non-overlapping 512 x 512 map tiles. The ML algorithm applied uses raster data only, so data tiles for the vector data (*.mvt or *.geojson) were not needed and therefore not provided through an OGC service. Also, for the waterbody vector data, the styling was removed so the data used does not contain contours anymore. Furthermore, no data values were set to display transparent. The dataset was provided in the same projection and tiling scheme and is currently available through the following URL: https://eratosthenes.pvretano.com/cubewerx/cubeserv/default/ogcapi/mysql_tb16

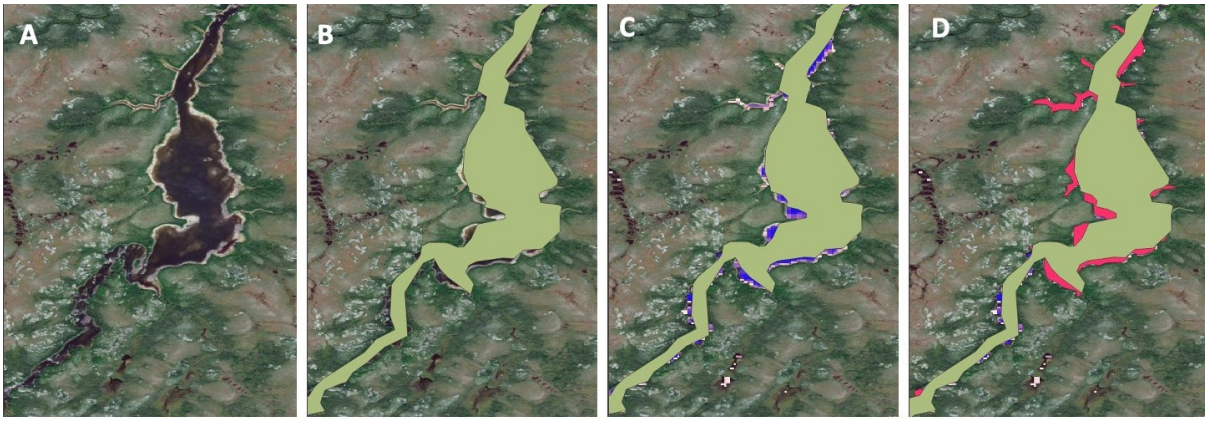


Figure 18. (A) Google Satellite Tile Map Services of a lake in Ontario Canada; (B) Google Satellite including the outline of a waterbody as made available through CanVec hydrological features dataset; © Google Satellite, CanVec waterbody outline, and the water body provided as occurrence grid derived from LandSat 1984-2019 data (Pekel et al., 2016) where blue indicates high occurrence over the years and purple showing less often occurrence of water; and (D) Google Satellite, CanVec water body outline, water occurrence and in red the waterbody as outlined by the HydroLakes database (Messager et al., 2016).

8.1.2. Conclusions

Derived datasets are authoritative but might be less suitable to use as labels in the ML chain when no adequate metadata is included. This often occurs when labeled data is initially created for another purpose than to train ML models. In general, for labeled data make sure to enrich label datasets with metadata when using OGC WMS standards. Metadata should include: a) Source of raw data, time period of raw data used to generate the label data, information on data accuracy, and b) precisely which raw data files were used (including a link to the data source).

8.1.3. Recommendations

- Identify key label datasets that might be very useful to train ML models with and investigate if it is still possible and worth the effort to include metadata.
- Update the WMS endpoint(s) for the vector waterbody data by including metadata to make the content more suitable as a ML training dataset.

8.1.4. Full References

- Messager, M.L., Lehner, B., Grill, G., Nedeva, I., Schmitt, O. (2016): Estimating the volume and age of water stored in global lakes using a geo-statistical approach. Nature Communications: 13603. doi: 10.1038/ncomms13603 [http://cnaes.ca/publication/messager-ml-lehner-b-grill-g-nedeva-i-and-schmitt-o-2016-estimating-the-volume-and-age-of-water-stored-in-global-lakes-using-a-geo-statistical-approach-nature-communications-7-13603-1/]
- Pekel, J-.F., Cottam, A., Gorelick, N., and Belward, A.S., 2016. High-resolution mapping of global surface water and its long-term changes. Nature 540, 418-422, doi:10.1038/nature20584 [https://www.nature.com/articles/nature20584].

8.2. Training images

8.2.1. RADARSAT-1

Initially RADARSAT-1 (1995 – 2013) Synthetic Aperture Radar (SAR) was selected to use as a training dataset. However, a geo-reference issue resulting in that a systematic offset was detected after applying the default pre-processing steps used in Sentinel Application Platform (SNAP) [https://step.esa.int/main/toolboxes/snap/]. These steps were:

- Import into SNAP (radarsat1 importer),
- Speckle Filtering (lee filter), and
- Terrain Correction (Ellipsoid Correction → Geogrid Location).

Following this procedure for RADARSAT-1 imagery, the resulting raster had an offset. This offset was caused by the poor orbit state vectors for RADARSAT-1. The orbit state vectors are updated every 8 minutes but the image acquisition takes less than 20 seconds. To estimate the satellite positions every 20 seconds by interpolation of the vectors is not accurate enough as this leads to incorrect pixel locations in terrain correction. This is a known issue with the RADARSAT-1 product. With RADARSAT-2 the orbit update period is 2 seconds.

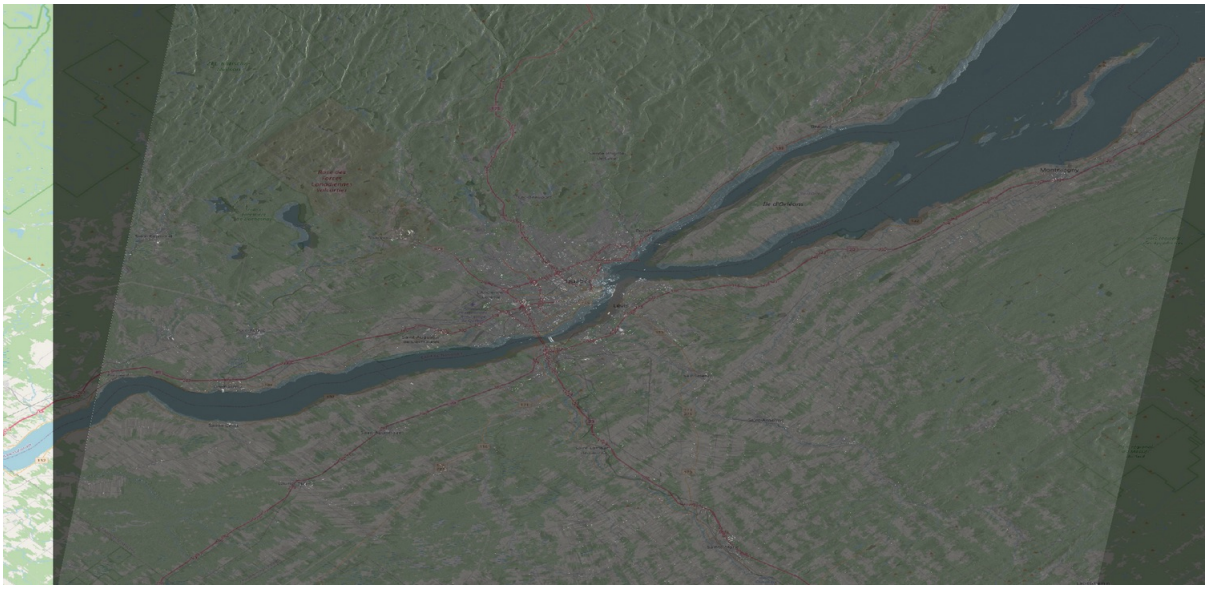


Figure 19. Offset detected when processing RADARSAT-1 imagery using the ESA SNAP toolbox standard processing steps.

8.2.2. Sentinel-1

Having identified the problem with pre-processing RADARSAT-1, the OGC Testbed16 participants decided to use Sentinel-1 (S1) SAR data for the remaining part of this project. Sentinel-1 data is considerably more difficult to display than other satellite data in near real time (which are no trivial task either). Each pixel in an image needs to be converted to meaningful physical quantities (the backscatter coefficient) and geopositioned individually. There is no global mapping which can take care of this. Orthorectification is even more complicated as it requires the use of a digital elevation model (DEM), and again, adjusted for each pixel individually. Many service providers do give the option of downloading orthorectified S1 imagery as end users might want to use a different DEM for their specific needs. As such S1 data is only limited available as GeoTIFF. Note that the Sentinel Hub streamed imagery from Amazon Web Services (AWS) is not (yet) orthorectified either but AWS will support this soon.

Although Sentinel-1 data is not new to Sentinel Hub, the fast AWS architecture and cloud optimized GeoTIFFs allow for faster response times compared to the current service running on a different platform with original, non-cloud-optimized files.

In general, serving Sentinel-1 data via a WCS endpoint can be challenging as the data is marked as 16-bit grayscale. As such the data will generate 8-bit JPEG grayscale overviews which will be almost entirely black for Data retrieval, and a contrast-stretching RasterSymbolizer will need to be applied to make the data visible for Map access or else it will look entirely black.

Level 0 products of the side viewing sensor of Sentinel-1 SAR records variability of the surface height of the earth. So a correction through an orthorectfication process needs to take place by using a digital elevation model before being able to host this data as a WMS or WCS endpoint. Therefore Sentinel-1 data was pre-processed by NRCan. The SNAP pre-process steps applied to Sentinel-1 SLC data are:

1. Calibration (Output sigma nought);
2. Apply a Terrain Observation with Progressive Scans SAR (TOPSAR) Deburst (What this will do is

seamlessly join all bursts in a swath into a single image);

3. Apply Multilook: 6x2 (This will improve the phase fidelity, creating Ground Range square pixels and reduces the file size. In essence, multi-looking performs a spatial average of a number of neighboring pixels);
4. Speckle Filter: Boxcar 5x5 (reduce speckle noise);
5. Terrain Correction: For higher latitudes, the Shuttle Radar Topography Mission (SRTM) [https://www2.jpl.nasa.gov/srtm/] would not work, so use the Canadian Digital Elevation Model (CDEM) and output GeoTIFF with pixel spacing 50m.

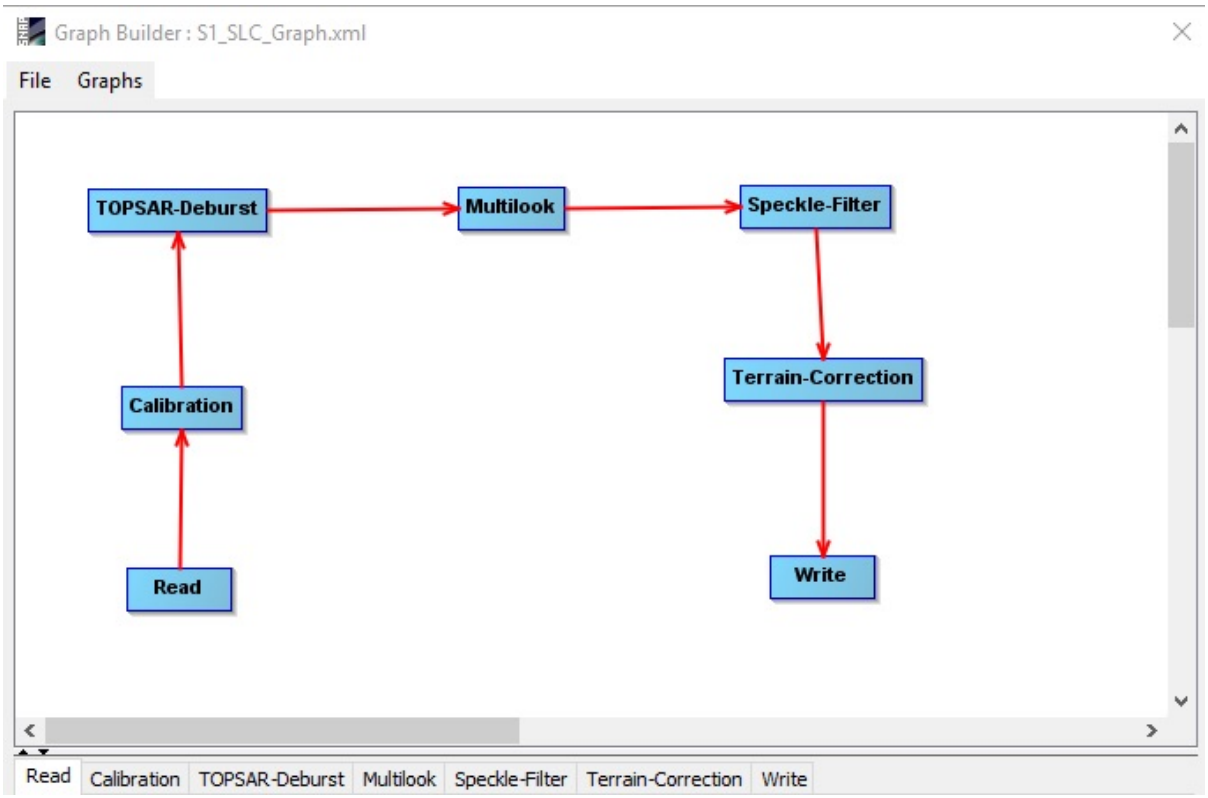


Figure 20. SNAP toolbox Graph Builder schematic overview of the pre-process steps.

On the output channels, VH and VV were assigned to channel 1 and channel 2 respectively. Then, for the RGB GeoTIFF displays the following was assigned to the RGB channels:

R: VH (Channel 1)

G: VV (Channel 2)

B: VV - VH (Channel 2 - Channel 1)

In total 89 sceneries from Sentinel-1 SLC data were processed, covering a large part of Canadian territories.

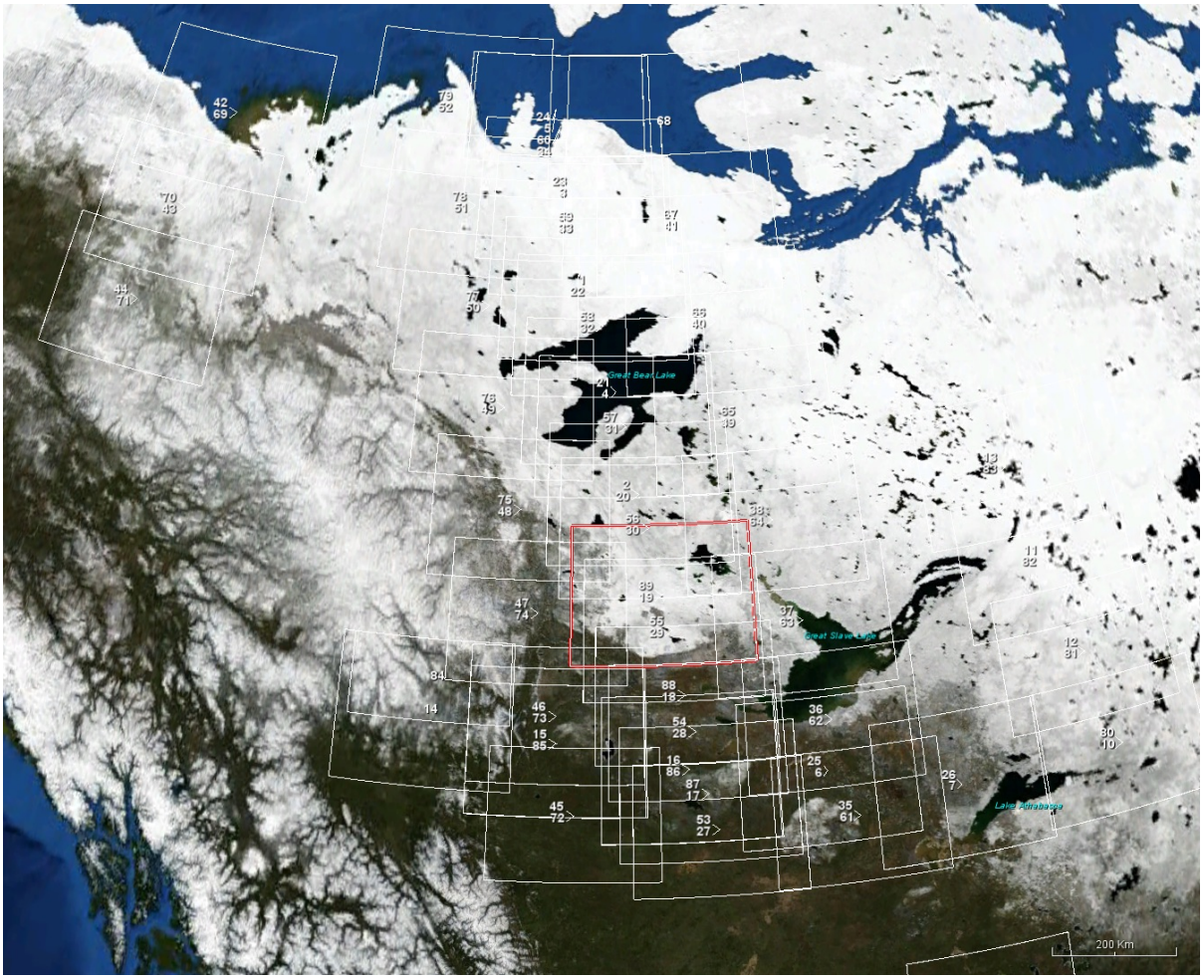


Figure 21. Sentinel images that were processed using SNAP.

Once processed the properties for the GeoTIFF were:

$ tiffinfo S1B_IW_SLC__1SDV_20170620T143323_20170620T143353_006135_00AC6F.tif

```
TIFFReadDirectory: Warning, Unknown field with tag 65000 (0xfde8) encountered.
```

```
TIFFReadDirectory: Warning, Sum of Photometric type-related color channels and
ExtraSamples doesn't match SamplesPerPixel. Defining non-color channels as
ExtraSamples.
```

```
Image Width: 19642 Image Length: 8167
```

```
Bits/Sample: 32
```

```
Sample Format: IEEE floating point
```

```
Photometric Interpretation: min-is-black
```

```
Samples/Pixel: 2
```

```
Rows/Strip: 8167
```

```
Planar Configuration: separate image planes
```

```
Tag 65000: <?xml version="1.0" encoding="ISO-8859-1"?>^M
```

```
<Dimap_Document
name="S1B_IW_SLC__1SDV_20170620T143323_20170620T143353_006135_00AC6F.dim"
```

The file information shows that for a non-standard tag 65000 a big XML document in some random format was included. This does not pose any problems except that the document cannot be parsed and the data-capture time is available inside that document. However capture time is also in the filename, although that is not parsed either.

Following the SNAP procedure, a technical flaw in creating a GeoTIFF is that the TIFF specifies a "Samples/Pixel: 2" tag but it doesn't include an "Extra Samples: 1<unspecified>" tag. The TIFF spec requires the "Extra Samples" tag in this place, but it isn't a problem since our code and the TIFF library assume the information of this tag.

A practical flaw that does have a big impact is the "Rows/Strip: 8167" element. To understand this, the developer has to understand the bitmap structure of TIFFs. There are three bitmap schemes TIFF supports, single strip images, stripped images and tiled images. For the sake of simplicity tiled images are not discussed here. For single stripped images, all of the bitmap is stored in one large block. A general recommendation is that no one strip should take more than 8 kilobytes uncompressed, which, with black-and-white images limits you to 65,536 pixels in a single strip. For stripped images, horizontal blocks of the image are stored together, and more than one strip is joined vertically to make the entire bitmap. However, if you allocate more strips than needed, then automatically the file size increases significantly, and that occurs in the pre-processing steps. Given the pixel size of the image there should be something like 3 strips. As such, the TIFF library reads scanline-organized images in Strips, which should be around 256 KB in size. However, for these preprocessed images, the Strip size is 8167 scanlines, which equals ~642 MB. This is a significant waste of memory, but worse than that, the TIFF-lib Scanline interface is used to read the image, which re-reads the entire 642 MB of data for each scanline.

Reading this image organized as "Rows/Strip: 8167" takes 7531 seconds (2.1 hours). Reading it organized as "Samples/Pixel: 2" takes 2.3 seconds.

Either files need to be reprocessed instead of using the original source data or the way the program

reads Scanline TIFFs has to be changed. Once done, performance should be much faster but still wastes 642 MB of memory. This is quite a lot of machine resource where the RAM is divided between many processors.

When reprocessing, the file properties are:

$ cwimage reprocessed.tif

```
type=FLOAT, nChannels=2, width=19642, height=8167, nPixels=160416214
```

```
samples: nBits=32, minValid=-inf, maxValid=inf, phys=FLOAT, physSize=4
```

```
channel 1: minSample=0, maxSample=0.907213032245636
```

```
channel 2: minSample=0, maxSample=5.18951034545898
```

This is different from the integer version of the images. Assumed in the above example is that the first value is the Amplitude and the second is the Phase angle in radians. These values will need to be adjusted to visualize them.

8.2.3. Conclusion

Providing Sentinel 1 data as an OGC service endpoint can be a cumbersome and not a straightforward process as data is provided in a format that is not supported by any of the OGC standards.

8.2.4. Recommendations

As a default, Sentinel-1 SAR data is not provided as GeoTIFF. To setup an OGC WMS or WCS endpoint or something similar, the data needs to be transformed into the GeoTIFF format. To accomplish this, processing steps are needed to generate a GeoTIFF, including orthorectification of the data. The SAR data provider should consider providing pre-processed data for ML workflows preferable for scenes that are used to create labeled data such that the same digital elevation model is used.

Readers are also invited to review the discussions in the D015 Machine Learning ER [https://portal.ogc.org/files/?artifact_id=95716] surrounding the use of OGC APIs in this task.

Chapter 9. Key Elements for Metadata Content for ML Training Data

At a high level, a machine learning meta-model consists of an objective, a learning algorithm, an optimizer, and dataset metadata.

Since this ER focused on ML Training Data, this section focuses on the dataset metadata. These metadata should at least contain statistical information about the data set, such as source, size, dimension, license, update status and other elements as well as of course features.

Creating and generating metadata for ML or research data and datasets in the ML training data "lifecycle" preserves the data in the long run and will also facilitate the use of ML training data for non-experts.

In addition to the recommendations on metadata for ML training datasets outlined hereafter, the reader is referred to https://portal.ogc.org/files/?artifact_id=75261 . This document captures research the CDB SWG has conducted on metadata standards and common mandatory elements across standards.

A set of rules and recommendations are easily found and can be defined as follows:

1. Consider what information is needed for the data to be read and interpreted in the future.
2. Understand requirements for data documentation and metadata. Several instructive examples can be found under the Funder Requirements [https://dmptool.org/public_templates] section of the Data Management Plan Tool (DMPTool) [https://dmptool.org/].
3. Consult available metadata standards for your domain of interest. Refer to Common Metadata Standards and Domain Specific Metadata Standards for details.
4. Describe data and datasets created in the research lifecycle, and use software programs and tools to assist in data documentation. Assign or capture administrative, descriptive, technical, structural and preservation metadata for the data. Some potential information to document:

```
Descriptive metadata
    Name of creator of data set
    Name of author of document
    Title of document
    File name
    Location of file
    Size of file
Structural metadata
    File relationships (e.g. child, parent)
Technical metadata
    Format (e.g. text, SPSS, Stata, Excel, tiff, mpeg, 3D, Java, FITS, CIF)
    Compression or encoding algorithms
    Encryption and decryption keys
    Software (including release number) used to create or update the data
    Hardware on which the data were created
    Operating systems in which the data were created
    Application software in which the data were created
Administrative metadata
    Information about data creation (e.g. date)
    Information about subsequent updates, transformation, versioning, summarization
    Descriptions of migration and replication
    Information about other events that have affected the files
Preservation metadata
    File format (e.g. .txt, .pdf, .doc, .rtf, .xls, .xml, .spv, .jpg, .fits)
    Significant properties
    Technical environment
    Fixity information
```

5. Adopt a thesaurus in your field or compile a data dictionary for your dataset.
6. Obtain persistent identifiers (e.g. doi) for datasets, if possible, to ensure data can be found in the future.

(Source: DMPTool: https://dmptool.org/. Digital Curation: A How-To-Do-It Manual; Digital Curation Centre: http://www.dcc.ac.uk/)

Chapter 10. Recommendations & Future Work

The Testbed participants made the following recommendations and suggested areas for future work.

10.1. Recommendations made with respect to the Fuel Load Estimation task

- Continue research on image processing methods applied to map analysis.
- Review the implementations of WMS services and add an SLD capability.
- Evaluate the possibility of creating new sources of data by turning the discrete-variable maps into continuous-variable ones, more suitable for machine learning applications.
- Make available a solid ground truth dataset.
- Provide accuracy information in the metadata of each training dataset.
- Establish a standard way to store and reuse a model.
- Consider adopting OGC API - Records in the future as a catalogue for NRCan's web services.
- Make sure that datetimes are properly and accurately set in datasets.
- Consider the adoption of new OGC APIs to ease the coding of values in SLDs.

10.2. Recommendations made with respect to the Water Identification task

- Identify key label datasets.
- Update the WMS for the vector waterbody data by including metadata to make it more suitable as ML training dataset.
- Consider providing pre-processed data for ML workflows preferable for EO scenes that are used to create labeled data.

10.3. Future work

For future work suggestions, the team has decided to distill the many recommendations listed in the previous subsections and provide the following as a top priority.

(i) From the many recommendations listed in the previous subsections, there is a real need to work out a best practice for a generalizable metadata model (framework) for ML training datasets. The Key Elements for Metadata Content section contains several items that could form a basis of this best practice in the future.

(ii) Furthermore, EO datasets should be rendered AI-ready as described, for example. by the aireo.net reference [https://www.aireo.net/] to this topic. Also in this context, efforts towards Analyses

Ready Data (ARD) such as those proposed by CEOS [http://ceos.org/ard/] will likely become vital for future ML applications.

(iii) Solid and reliable ground truth datasets should be developed, including accuracy levels of the ML training data.

Appendix A: Revision History

Table 1. Revision History

Date	Editor	Release	Primary clauses modified	Descriptions
May, 2020	G. Schumann	.1	all	IER version
June-September, 2020	G. Schumann; A. Kettner; I. Correas	.1	all	version ready for OGC review
October, 2020	G. Schumann; A. Kettner; I. Correas	1.0	various	implementation of comments from C. Reed
November 3, 2020	G. Schumann; A. Kettner; I. Correas	1.0	various	implementation of comments from G. Hobona
January 6, 2021	S. Serich	1.1	all	top-to-bottom feedback from NRCan sponsor