Deep Spatio-Temporal Time Series Land Cover Classification

Bachelor's Thesis Exposé

Submitter:	Arik Ermshaus
Submission Date:	Thursday 29th March, 2018
Supervisor:	Dr. rer. nat. Patrick Schäfer
Supervisor:	Prof. Dr. Ulf Leser
Institution:	Department of Computer Science, Humboldt-Universität zu Berlin

Contents

1 Introduction

We consider the problem of classifying land cover classes in time series of satellite images. Nowadays, satellites like Landsat 8 provide high resolution images containing multi-spectral measurements per pixel. We can use this information to map a pixel to a pre-defined land cover class. In that manner we can e.g. discover the exact positions of urban areas, forests or grasslands. A time series in the context of land cover classification is an ordered sequence of pixels from satellite images that is associated with one land cover class. Given a dataset of land cover classification time series and their corresponding land cover classes, we can train a supervised classifier to learn representations from the temporal profiles of the time series in the dataset. We use such a classifier to predict the land cover class of a given pixel.

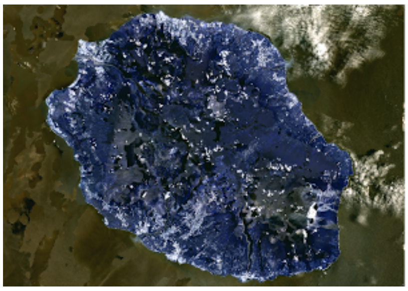

Figure 1: Reunion Island[1]

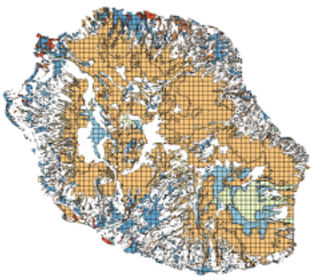

Figure 2: Reunion Island Land Cover Classes[1]

In this bachelor's thesis we analyse the winning solution submitted to the "Time Series Land Cover Classification Challenge" (TiSeLaC)[1], organized by the "European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases" (ECML PKDD 2017)[2]. The task of the challenge consists of predicting Reunion Island's land cover classes (Figure 1 & 2). The given dataset includes a time series of 23 annually taken Landsat 8 images from 2014. Each image comprises 2866x2633 pixels at 30m spatial resolution and contains seven surface reflectances and three computed indices per pixel.

The winning solution of the challenge achieves a weighted F1-score of 99.29%, using an artificial neuronal network (ANN) with three main components [MVB+17]. The first component learns every pixel's features among the time series and is inspired by the work in [ZLC+14]. The second component uses an image segmentation architecture as in [SLD17] to learn representations between the entire dataset. Lastly, the network's third component learns spatial information from the neighbourhood of each pixel. All components are then concatenated to predict the land cover class of each pixel.

The outstanding results of the ANN on the dataset motivate to understand how the three components work together. This thesis aims to investigate and deconstruct this approach.

2 Background and Related Work

Background An artificial neuronal network is an iterative, end-to-end learning system that predicts an output for a given input. A network comprises a collection of connected layers of nodes, called artificial neurons, that learn representations from the input data.

TiSeLaC's winning solution uses the following kinds of layers in its network. *1-dimensional convolutions* (conv) learn local features from the data. *Max pooling* (pool) regularizes and down-samples the learned features to generalize them and reduce their dimensionality. *Flattening* (flat) and *concatenating* (concat) change the shape of the data or rather connect layers. *Dense* (dense) and *dropout* (drop) memorize learned

[1]https://sites.google.com/site/dinoienco/tiselc
[2]http://ecmlpkdd2017.ijs.si/

representations and avoid over-fitting. The network's architecture arranges instances of these layers in the following way.

Independent Band Modeling (IBM) An Independent Band Model is an array of conv-pool-conv-pool-flat layers. Furthermore, a band consists of the 23 temporal measurements of a surface reflectance or computed index for a given pixel. An Independent Band Model learns the temporal connections in a band through 1-dimensional convolutional layers and regularizes the learned representations through max pooling. IBM consists of ten Independent Band Models, one for each surface reflectance or computed index. In addition, the outputs of the first and second convolution are respectively concatenated and connected to conv-flat layers to learn the correlations between bands. All outputs are then concatenated. (Figure 3) This component is inspired by the work in [ZLC+14].

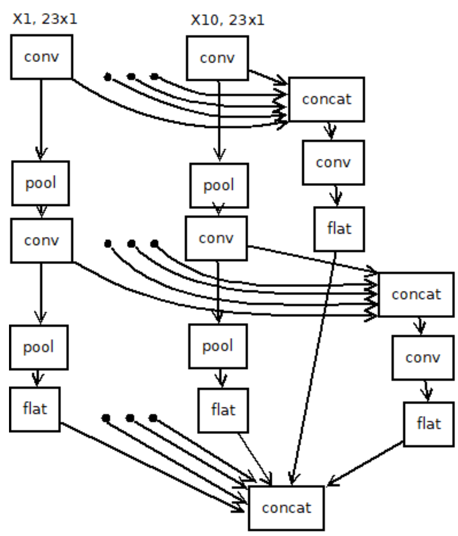

Figure 3: IBM

Fully-Convolutional Architecture (FCA) FCA consists of an array of conv-conv-flat layers that processes the 10 surface reflectances and computed indices for the 23 temporal measurements per pixel. This component captures every correlation between bands and identifies coherent units of land cover classes based on the work in [SLD17].

Spatial Dense Architecture (SDA) SDA consists of an array of dense-drop-dense-drop layers memorizing spatial correlations from a preprocessed version of the training data, called a "Spatial Feature Descriptor" (SFD). The SFD comprises a vector for each pixel, containing information from its spatial neighbourhood like land cover class frequencies and time progressions for the computed indices. This component uses the insights from [GMA15] and [PBB+09] that explore spatial classification on hyperspectral data.

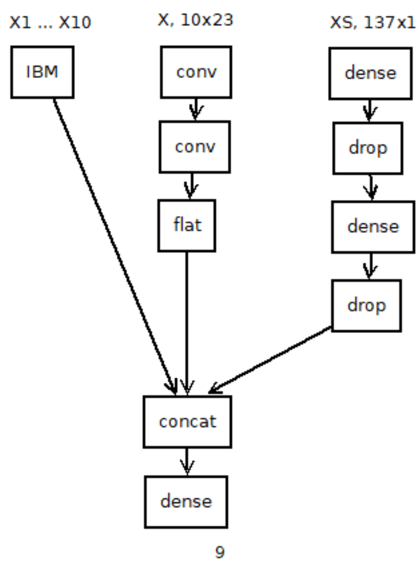

Figure 4: IBM, FCA & SDA

The outputs from IBM, the FCA & SDA are concatenated and forwarded to a dense layer that predicts a land cover class for a pixel. (Figure 4)

"End-to-end Learning of Deep Spatio-temporal Representations for Satellite Image Time Series Classification" explains the objective and properties of the challenge and describes the ANN's architecture in great detail. Furthermore, it evaluates the ANN on the given dataset and discusses its results [MVB+17]. We use this paper as a foundation to understand and implement the approach.

Related Work Land cover classification has recently advanced in accuracy through the use of ensemble classification methods. The Random Forest classifier is one of the most widely used ensemble classification algorithms. It utilizes a set of random decision trees and bootstrap aggregating (bagging). Bagging repeatedly fits the decision trees with random subsets of the training data in order to decrease the variance of the model and reduce its sensitivity to noise and over-fitting. "Classifying land cover from satellite images using time series analytics" [SPH+18] uses feature extraction and a univariate/multivariate bag-of-patterns classifier with a Random Forest to predict land cover classes. The approach reaches a 93.38% weighted F1-score on the TiSeLaC dataset.

Long short-term memory (LSTM) units enable artificial neuronal networks to remember data over arbitrary time intervals. An unit consists of a cell that remembers values and an input, output and forget gate that regulate the flow of values in the cell. [KMD+18] presents two neuronal networks comprising convolutional feature extraction combined with LSTMs for multivariate time series classification. Both networks outperform most of the state of the art classifiers.

In this thesis we experiment with Random Forests and the SFD to simplify TiSeLaC's winning solution. Also, we test an optimized version of IBM and the FCA on multivariate time series classification benchmark datasets. We use both papers as a start for further investigation.

3 Motivation and Objectives

The main motivation and objective of this bachelor's thesis is to investigate and deconstruct the winning solution to be able to understand and evaluate its components and to explore whether the used methods can be applied to different datasets. To achieve this, we use the following steps:

1 Introduce artificial neuronal networks, the underlying ideas and building blocks to gain an overview of the used methods. Analyse IBM, the FCA & SDA in detail to develop an understanding of their function and interaction. Implement the entire ANN in a modular architecture so that we can reproduce the results from the challenge and explore the effect of altering hyperparameters. Furthermore, analyse the impact on precision, recall, the F1-score and training & prediction time of the three aforementioned components.

2 Optimize and simplify the network if possible. Using the results of step 1, extract and change components of the network to optimize them or to make them available for broader use cases like domain-agnostic multivariate time series classification. For instance, analyse the influence of the Spatial Feature Descriptor and implement a SFD + Random Forest. Also, implement an optimized version of IBM and the FCA for multivariate time series classification datasets.

3 Use the original and optimized versions of TiSeLaC's winning solution and evaluate precision, recall, the F1-score, training & prediction time and model complexity on different time series land cover classification and multivariate time series benchmark datasets. Discuss the results and elaborate on the implications of the study in support of further scientific research in the field using the principals applied in IBM, the FCA & SDA.

The implementation, modification and application of the ANN reveals whether the applied methods are useful for time series classification and provides the computer science department a tool for further use and investigation.

4 Methods and Results

We reimplement the ANN in Python3 based on the author's code[3] and with TensorFlow[4] as the backend for the machine learning library. TensorFlow allows us to focus on the high-level design of the model and outsource the rest to existing and proven state of the art technologies. We divide the architecture of the implementation into the following parts:

1 Data Transformation, Scaling and Normalization. We transform the data to fit the input layers of the network and remove the mean and scale to unit variance to normalize the data. Furthermore, we create a binary representation of the class labels to fit the output layer of the network.

2 Spatial Feature Descriptor. We preprocess the Spatial Feature Descriptor and save it to reduce the future runtime of the software.

3 Network & NetworkBuilder. We define the network's architecture in a declarative structure using TensorFlow's internal Keras API.

4 Analysis. We measure the micro, macro and weighted F1-score of the training and test data, the training & prediction time and create a confusion matrix for further analysis.

The modular architecture separates data transformation, preprocessing, modelling and analysis and fits the use case of TiSeLaC's winning solution. This enables us to keep track of the complexity of the original and optimized models. Also, it creates a developing environment for this study that fosters the comparison and evaluation between the different versions. We use numpy[5], sklearn[6] and scipy[7] for data transformations and measurements. Also, we track the different versions of the software, the used data and the results on a private HU GitLab repository and publish the software on GitHub after we finish all experiments to make it accessible to the scientific community.

Furthermore, we use the TiSeLaC dataset, similar land cover classification and multivariate time series benchmark datasets from the UCI Machine Learning Repository[8] for the experiments.

[3]https://github.com/nicoladimauro/TiSeLaC-ECMLPKDD17
[4]https://www.tensorflow.org/
[5]http://www.numpy.org/
[6]http://scikit-learn.org/
[7]https://www.scipy.org/
[8]https://archive.ics.uci.edu/ml/

References

[PBB+09] Plaza, A., Benediktsson, J.A., Boardman, J.W., Brazile, J., Bruzzone, L., Camps- Valls, G., Chanussot, J., Fauvel, M., Gamba, P., Gualtieri, A., Marconcini, M., Tilton, J.C., Trianni, G. *Recent advances in techniques for hyperspectral image processing.* (2009), Remote Sensing of Environment 113. 110-122

[ZLC+14] Zheng, Y., Liu, Q., Chen, E., Ge, Y., Zhao, J.L. *Time series classification using multi-channels deep convolutional neural networks.* (2014), 15th International Conference on Web-Age Information Management. 298-310

[GMA15] Guccione, P., Mascolo, L., Appice, A. *Iterative hyperspectral image classification using spectral-spatial relational features.* (2015), IEEE Trans. Geoscience and Remote Sensing 53(7). 3615-3627

[MVB+17] Di Mauro, N., Vergari, A., Basile, T. M., Ventola, F. G., Esposito, F. *End-to-end Learning of Deep Spatio-temporal Representations for Satellite Image Time Series Classification* (2017)

[SLD17] Shelhamer, E., Long, J., Darrell, T. *Fully convolutional networks for semantic segmentation.* (2017), IEEE Trans. Pattern Anal. Mach. Intell. 39(4). 640-651

[KMD+18] Karim, F., Majumdar, S., Darabi, H., Harford, S. *Multivariate LSTM-FCNs for Time Series Classification* (2018)

[SPH+18] Schäfer, P., Pflugmacher, D., Hostert, P. and Leser, U. *Classifying land cover from satellite images using time series analytics.* (2018), 2nd Int. Workshop on Data Analytics Solutions for Real-Life Applications, Vienna, Austria