Web based Machine Learning for OpenEOcubes - WebMLOpenEO (Pflichtenheft)

Leonard Wittrock, Martin Kriger, Tim Ciroth, Maximilian Reiner, Michael Brüggemann

6. November 2023

Projekt:	Web based Machine Learning for OpenEOcubes - WebMLOpenEO
Auftraggeber:	Institut für Geoinformatik, Universität Münster
Auftragnehmer:	Eine Firma

Version	**Datum**	**Autor(en)**
1.0	06.11.2023	Leonard Wittrock, Martin Kriger, Tim Ciroth, Maximilian Reiner, Michael Brüggemann

Inhaltsverzeichnis

1 Unser Produkt

Um die Aufgabe einen webbasierten Service zur machine-learning gestützten Klassifikation von Sentinel-2 Satellitenbildern zu entwickeln zu erfüllen, stellen wir einen umfassenden Webservice zur Verfügung.

Dieser Webservice besteht aus einem Frontend-Server, einem Node.js-Backend und einem OpenEO-konformen R Backend. Der Frontend Server stellt der Nutzer:in eine gerenderte HTML Seite zur Verfügung, auf der die Klassifikationsaufgabe konfiguriert werden kann. Der Frontendserver stellt ebenfalls eine HTML mit der Dokumentation zu unserem Webservice zur Verfügung. Auf der Website unseres Webservices stehen der Nutzer:in ein Formular zur Interaktion, sowie eine Webkarte zur Verfügung. Mit diesen Elementen ist die Interaktion mit unserem Service möglich. Das Erscheinungsbild und die Ausgaben der Anwendung werden dabei von einer barriefreien Farbpalette unterstützt, was eine möglichst breite Zugänglichkeit ermöglicht. Durch das Verwenden von Vue.js ist außerdem eine umfangreiche Browserunterstützung gewährleistet (90%).

Im Node.js-Backend werden die Anfragen des Users gebündelt. Dort wird eine API bereitgestellt, die die Services unserer Anwendung beschreibt (API-self-documentation), sowie die Anfragen der Nutzer:innen verarbeitet, bzw. in neue Anfragen an andere Server umwandelt (Routing).

Die Anfragen zu einer Klassifikation werden dann an ein OpenEO-konformes Backend weitergeleitet, auf welchem die tatsächlichen Berechnungen stattfinden. Dieses nutzt die bereits bestehenden Strukturen von 'openEOcubes' und erweitert diese um geeignete Prozesse, um die machine-learning gestützte Klassifikation durchzuführen. So werden bereits bestehende Prozesse zur Verbarbeitung von Datacubes durch 'openEOcubes' genutzt und auch neue Prozesse im Bereich machine-learning ergänzt.

Weiterhin bietet unsere Anwendung die Möglichkeit einen Demoprozess zu starten. Dies ist ein geführter Prozess, indem die Nutzer:in eine mögliche Eingabe in unseren Webservice vorgestellt bekommt und diese Anfrage an den Server gesandt wird. Dort wird ein bestehender Datensatz angefragt, der dann für die Klassifikation genutzt wird. Während des Prozesses werden die Formularelemente für die Nutzer:in gesperrt, um sicherzustellen, dass der Demoprozess genau wie von uns vorbereitet durchlaufen wird. Der Prozess wird dabei von informativen Popups begleitet, die der Nutzer:in weitere Informationen zu dem Webservice bieten.

Der Webservice selbst wird auf DockerHub als Docker-Image bereitgestellt, welches direkt auf einer Amazon AWS/EC2 Instanz gehostet wird. So ist sichergestellt, dass der Service unabhängig vom lokalen System ausgeführt wird und der Webservice dadurch ebenfalls über die für die Klassifikationen nötige Rechenleistung verfügt. Dieses Docker Image besteht dabei aus 3 Services. Dies sind die oben beschriebenen Komponenten des Services (Frontend, Node.js-Backend, R Backend). Da jede Komponente nur eine einzelne Programmiersprache nutzt, kann auf komplexe Build-Prozesse mit Makefiles verzichtet werden. Unser Entwicklungsprozess besteht hier aus in den Testsuites 'Jest', 'vitest' und 'testthat' definierten Unit- und Integrations-Tests, sowie einem nach dem 4-Augenprinzip überwachten Pullrequest-Prozess in unserem Github-Repository.

Der gesamte Webservice wird unter einer OSI Lizenz veröffentlicht und steht als OpenSource-Software zur Verfügung. Der Kunde erhält über unser GitHub Repository während des Entwicklungsprozesses Zugriff auf den Quellcode, sowie alle dort verwalteten Projektmanagementdokumente (siehe Abschnitt 5).

2 Funktionale Anforderungen

2.1 Data Ingest

Nummer	Bezug	Beschreibung
/FA05/		Unsere Web-App wird als eine One-Page-Struktur implementiert, welche auf der linken Seite eine interaktive Kartenfunktion bereitstellt und auf der rechten Seite ein Formular besitzt, um die gewünschten Daten einzugeben und einzuladen.
/FA10/	support loading training data via direct file upload to the server, support at least file formats GeoJSON and GeoPackage	Im Formular wird ein Button bereitgestellt, der den File-Explorer öffnet, um Dateien für die Trainingsdaten hochzuladen. GeoJSON und GeoPackage werden unterstützt.
/FA15/	the server automatically discovers and processes (accesses, or downloads) the satellite imagery from other sources as required for model training and prediction based on the locations and extents of the area of interest and training data	Der Backend-Server basiert auf einem openEO-Backend namens 'openEOcubes', auf dem die Collection der Sentinel-2 Erdbeobachtungsdaten vorliegen.
/FA20/	the server automatically discovers and processes (accesses, or downloads) the satellite imagery from other sources as required for model training and prediction based on the locations and extents of the area of interest and training data, /F05/	Durch die gleichzeitige und gemeinsame Speicherung auf einer AWS-Instanz (us-west-2 region Oregon) wird der direkte Zugriff auf die Collection der Erdbeobachtungsdaten des Sentinel-2 Satelliten möglich.
/FA25/	example satellite data for demonstration purposes is being kept available on the server	Unsere Anwendung bietet die Möglichkeit über einen Button einen Demoprozess zu starten. Dies ist ein geführter Prozess, indem die Nutzer:in eine mögliche Eingabe in unseren Webservice vorgestellt bekommt und diese Anfrage dann an den Server gesandt wird.

2.2 Processes

Nummer	Bezug	Beschreibung
/FA30/	implement at least one standard machine learning algorithm for LULC clas- satellite image data	Wir implementieren den 'Random Forest' Algorithmus. Random Forest zeichnet sich durch eine Hohe Performanz in LULC Klassifikationsproblemen aus und ist daher gut für diese Anwendung geeginet
/FA35/	allow the user to specify the target area by means of polygon data (e.g.by file upload, or on-screen digitizing).	Der Nutzer wird die Möglichkeit haben in einer Karte, welche auf openLayers basiert, mithilfe eines Zeichenwerkzeuges, einen Kartenausschnitt einzuzeichnen.
/FA40/	allow the user to specify the target area by means of polygon data (e.g. by file upload, or on-screen digitizing)	Optional kann eine GeoJSON-Datei, welche den gewünschten Bereich als beliebiges Polygon definiert, über einen Button, welcher den File-Explorer öffnet, hochgeladen werden.
/FA45/	allow the user to specify the target area (rectangular) and date	Neben der Möglichkeit einen Kartenausschnitt zu bestimmen, ist es möglich im Formular über eine Datumsauswahl, den Zeitpunkt für die gewünschten Satellitendaten, festzulegen.
/FA55/	allow the user to specify a lower resolution to compute predic- tions on than the native resolution of the satellite imagery used	Wir stellen ein Input Feld zur Verfügung, in der die Resolution des Satellitenbildes eingestellt werden kann. Im Backend wird dies durch 'openEOcubes' in Verbindung mit 'gdalcubes' realisiert.
/FA60/	in addition to the classified map, create and return the map with class probabilities of all classes of that for which the probability is maximum.	Nach der Prozessierung wird dem Nutzer, über die Interaktive Karte, eine Möglichkeit gegeben die verschiedenen Kartentypen (predicted map, map with class probabilities) als Layerwechsel anzeigen zu lassen.
/FA65/	improve openEOcubes and submit one or more pull requests to its GitHub repo	Die Erweiterung wird ein neu definierter openEO-Prozess sein, der das Berechnen einer LULC-Klassifikation mit dem Algorithmus 'Random Forest' ermöglicht und dabei auf dem Backend laufen wird. Zusätzlich dazu werden die Teilprozesse einer ML-Klassifikation als openEO-konforme Prozesse implementiert.

2.3 Visualisation

Nummer	Bezug	Beschreibung
/FA70/	visualize the training areas/locations (if applicable) and area of interest on an interactive map (leaflet, openlayers or similar)	Die Web-App verfügt über eine Kartenansicht, die je nach ausgewähltem Schritt entweder die 'Area of Interest' oder die Training-Areas anzeigt.
/FA75/	visualize the resulting prediction (classification) on an interactive web map	Nach Abschluss des Prozesses zeigt die Karte das Ergebnis in Form eines ein- und ausblendbaren Klassifizierungs-Layers an. Dieser Layer verwendet eine barrierefreie Farblegende, die auf Grundlage der vom Backend gelieferten Daten berechnet wird. Die Farblegende ermöglicht den Benutzern eine klare und zugängliche Darstellung der Klassifizierungsinformationen auf der Karte, wodurch sie die Ergebnisse leicht verstehen können.
/FA80/	visualize the resulting quality of prediction (e.g. probibility of the most likely class) on an interactive web map	Am Ende des Prozesses hat der Nutzer die Möglichkeit, eine Karte mit den Klassenwahrscheinlichkeiten in der Web-App anzuzeigen.

2.4 Data Download

Nummer	Bezug	Beschreibung
/FA85/	enable the download of the prediction map in a standard raster data format	Unsere Web-App bietet einen Download-Button an, der es den Benutzern ermöglicht, die Rasterdaten im .TIFF-Format herunterzuladen.
/FA90/	enable download of the trained model (if applicable)	Es wird eine Möglichkeit geben, das trainierte Model in einem für das Model geeigneten Datenformat herunterzuladen.

2.5 Deployment

Nummer	Bezug	Beschreibung
/FA95/	Docker/docker-compose configurations are used for packaging and deployment	Das Projekt besteht aus drei Docker-Services: ein HTTP-Server mit Vue.js, ein Node.js-Server und ein R-Server mit openEOcubes. Diese Services sind für die Bereitstellung der Benutzeroberfläche und serverseitige Logik verantwortlich.
/FA100/	the developed system can be deployed to any host system without additional software requirements other than recent versions of Docker/docker-compose	Das Projekt kann mit Docker Compose gestartet werden, was die einfache Verwaltung und Bereitstellung der Anwendung ermöglicht.

3 Nichtfunktionale Anforderungen

Die meisten nichtfunktionalen Anforderungen werden bereits in dem Abschnitt 'Unser Produkt' dargestellt (siehe Abschnitt 1). Folglich wird hier weiterführend auf die Performance eingegangen.

3.1 Performance

Eine Änderung auf der Website erfolgt innerhalb einer Sekunde. Alle Inhalte werden asynchron geladen. Die interaktive Bedienung erfolgt innerhalb 0.1 Sekunden.

4 User Stories

- Als Nutzer:in möchte ich anhand einer interaktiven Benutzeroberfläche auf einer Karte den Kartenausschnitt wählen können, sodass ich den für mich interessanten Bereich untersuchen kann.
- Als Nutzer:in muss ich in der Lage sein das Datum zu bestimmen, Trainingsdaten über ein File hochladen zu können und einen Machine Learning Algorithmus mit Hyperparametern auswählen zu können, damit ich mein eigenes individuelles Modell trainieren kann.
- Als Stadt Münster möchten wir, dass das Hochladen von Trainingsdaten auf jeden Fall die Formate GeoJSON und GeoPackage umfasst, damit wir die meisten unserer Daten direkt nutzen können.
- Als Nutzer:in möchte ich eine Möglichkeit haben neue Trainingsdaten zu bestimmen, um den Algorithmus wichtige Informationen zu geben, sodass eine möglichst umfangreiche Vorhersage durch den 'Random Forest' Algorithmus stattfinden kann.
- Als Laie und unaffine Person im Umgang mit Erdbeobachtungsdaten möchte ich eine Hilfestellung besitzen, die mir zeigt, wie ich vorgehen muss, um ein Modell trainieren zu können.

- Als wissenschaftliche:r Mitarbeiter:in einer Universität möchte ich eine Möglichkeit haben, die resultierenden Vorhersage mittels einer interaktiven Webkarte zu begutachten und als Rasterdaten zu downloaden und das trainierte Modell herunterzuladen, sodass ich diese weiterführend in meine Arbeit einbinden kann.
- Als Arbeitsgruppe an der Universität Osnabrück im Bereich Remote Sensing wollen wir eine Möglichkeit besitzen, das Toolkit mittels Docker auf verschiedensten Endgeräten laufen zu lassen, damit auch im Außendienst, spontane Analysen getätigt werden können.
- Als farbenblinder Mensch möchte ich barrierefrei und ohne Einschränkungen die Anwendung nutzen können, damit bei der Zusammenarbeit mit meinen Mitmenschen, die gleichen Voraussetzungen für alle bestehen und keine Fehlinterpretationen untereinander entstehen.

5 Zugang

Der Quellcode sowie die Projektmanagement-Dokumente sind auf unserem GitHub-Repository unter folgendem Link zu finden:

- GitHub

6 Zeitplan

Unsere voraussichtliche Zeiteinteilung befindet sich als Gantt-Chart im Anhang.

Geosoft2

- **Ferien**
  - Urlaub
- **Angebotsphase**
  - Lastenheft
- **Planung**
  - Pflichtenheft
  - Abgabe Pflichtenheft
  - Überarbeitung Pflichtenheft
  - Abgabe Pflichtenheft
- **Implementierung**
  - Beginn Implementierung
  - **Frontend**
    - Implementierung
    - Implementierung 2
  - **ML**
    - Recherche
    - Implementierung
    - Implementierung 2
  - **Node**
    - Recherche
    - Implementierung
    - Implementierung 2
  - Status Report
  - Prelease vorbereiten
  - Pre-release
  - Status Report
  - Verknüpfung
  - Testen
  - Abgabe Endprodukt
- **Projektabschluss**
  - Projektbericht
  - Abschlusspräsentation