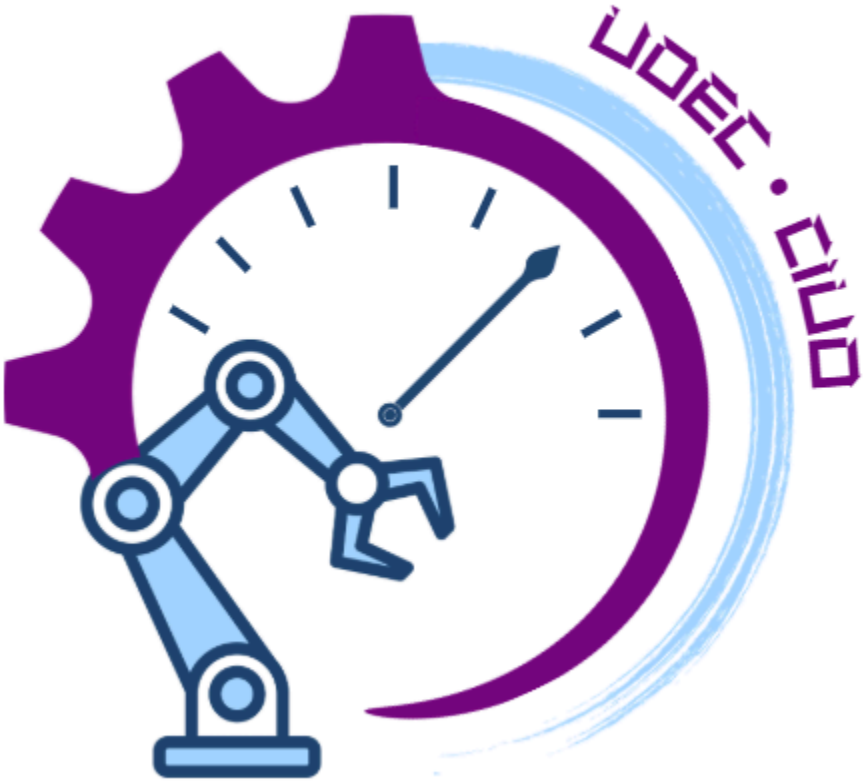


university of OTTAWA

ENGINEERING COMPETITION

Programming Competition 2026

uOttawa Engineering Students' Society

VP Internal

competition@uottawaess.ca

This document outlines requirements and expectations for competitors in the Programming category at uOEC 2026. Please read this document carefully and in full.

Overview

The goal of the programming category is to encourage engineering students to produce a piece of readable software. The teams will use their software development skills, their technical writing abilities, and their project management skills to design a solution to a posed problem. The judges will evaluate the software as a technical solution to the provided problem.

Scenario

As you embark on your time-travelling journey, you accidentally press a button that takes you back in time, to an unknown land. You have a translator, but you do not know the language to translate from in this unknown land. Going through each language at a time is time-consuming, so you should create a machine learning model that can predict which language you must translate from. Use the dataset provided and create a machine learning model that can identify the given language: Language Identification Dataset. How you choose to accomplish this is entirely up to your team. The dataset contains 22,000 text samples of 22 different languages. Each language in the dataset has 1000 different text samples, and each text sample has a unique identifier specifying the correct language. The model must correctly identify all the languages presented in the dataset to some degree of high accuracy. Teams are expected to download/access the dataset themselves through Kaggle with the link provided in the resources.

Objective

- Build a machine learning model that can predict the given language.
- A successful model should have a high degree of accuracy when predicting correct language.
- Provide a way to demonstrate the accuracy of the model such as:
    - Confusion matrix
    - Validation data
    - Accuracy in the sense of (Number of correct predictions divided by the total number of prediction)
    - Etc…

Rules and Constraints

Restrictions:

- You cannot use any external datasets, only the dataset provided by uOEC.
    - Only the dataset provided in the resources section of this booklet should be used to train the model.

- You cannot use any pretrained machine learning model. This applies to models trained for language identification. Standard open-source NLP and vectorization libraries are permitted (see Resources).
- The use of an artificial intelligence-backed API is forbidden
    - ChatGPT
    - Copilot
    - Gemini
    - Claude
    - AI-generated code
    - Or ANY AI-assisted coding

Constraints:

- The maximum number of individuals for a Programming team is four (4).

Durations

- Programming time: 3 hours and 30 minutes, from 8:30 am to 12:00 pm. Includes setting up the environment, downloading libraries, dataset preparation, testing, and presentation preparation. GitHub link and presentation due 12:00 pm. Lunch 12:00 to 1:00 pm.
- Presentations: from 1:00 pm, no audience. Per team: ten (10) minute presentation, five (5) minute question period.

Deliverables

- Presentation: The presentation must show us a live run of the machine learning model, as well as displaying the accuracy metrics of the machine learning model. Meaning the team should run the machine learning model in front of the judges to show them that it works as intended. Also consider displaying the accuracy of the model that you develop in percentage. You are also required to explain the process of training your model, such as how you divided up the work.
- Make sure that the code is available in a GitHub repository, and to have it be public so that it can be accessed by the judges at any time. Make sure to submit the link of the GitHub repository alongside the presentation.
- The GitHub link and presentation must be submitted by 12:00 pm.

Resources

- The internet
    - Competitors have internet access throughout the programming period.
- Any open source libraries
    - Pretrained libraries/models such as standard NLP/vectorization tools are allowed.
    - Competitors may install additional open-source libraries during the competition.
- Any programming language, so long as it runs on your computer.

- Kaggle data: https://www.kaggle.com/datasets/zarajamshaid/language-identification-datasst?utm_source
    - Only the dataset provided by uOEC can be used.
    - Kaggle is the location to where the dataset is provided.

Software Resources

- If a team decides to reuse pre-existing code, it must be clearly cited in the presentation. The team is required to clearly explain why and where the recycled code was used in their software.
    - This includes code copied from Stack Overflow, GitHub, or any documentation on the internet.
    - The citations must be in APA format, including the link to the authors work.
- The judges hold the right to ask team members who contributed to programming the machine learning model to describe what a particular section of the code does at any given point during the presentation.

Judging and Scoring

For your reference, the judging matrix is provided below.

Strategy/Algorithm	Accuracy of Model Deliverable Compliance with Expectations Innovation & Creativity	/ 40
Code	Readability Efficiency Structure Testing i.e. Model Runs & Works	/ 35
Presentation	Design Process and Justification Voice Articulation and Timing Visual Aids Response to Questions	/ 25
Penalties	Plagiarism Insufficient Citation Documents Received After Deadline AI restriction Violations Absent Team Member	-50 -50 -50 -50 -25
Total		/ 100

Refer to the uOEC 2026 rulebook for general rules. Where this package and the rulebook differ, this package takes precedence.

The Dataset:

Contents of the dataset:

- 22,000 Text samples
- 22 different languages
- 1,000 different text samples per language
- Unique identifiers per text sample clarifying and stating the language.
- Languages present:
    - English, Arabic, French, Hindi, Urdu, Portuguese, Persian, Pushto, Spanish, Korean, Tamil, Turkish, Estonian, Russian, Romanian, Chinese, Swedish, Latin, Indonesian, Dutch, Japanese, Thai.
- Teams are responsible for creating their own split ratio to validate the model.

https://www.kaggle.com/datasets/zarajamshaid/language-identification-datasst?utm_source