**Division SIGMA - Instruction and Rules**

\*\*\* Please read the following descriptions carefully \*\*\*

**Competition Instructions**

Division Sigma follows a problem-based hackathon model. A competition problem will be released at 0:00 am EST on August 22nd. Each participating team should submit a solution to the problem before the competition deadline at 11:59 pm EST on August 23rd. The general category of the problem focuses on the application of AI, specifically the **supervised learning of text data**. Therefore participants should submit an AI model as their solution.

On August 22 at 0:00 am EST, we will provide participants with a base (template) code that serves as a simple solution to the competition problem. This template code provides simple solutions using some conventional Machine Learning models from the Sci-Kit Learn library. To create a unique solution, each participant/team can either:

* Customize one of the sample Machine Learning models provided
* Use another AI model and customize it (encouraged)

Note that participants are **NOT Constrained** to the Sci-Kit Learn library, teams can use any AI model, including both not limited to neural networks, deep learning, etc.

\* Submissions of the template code with **NO CUSTOMIZATION** will **NOT** be allowed!

**Requirements & Prerequisites**

Division SIGMA is designed for the more **experienced programmers** who wish to learn about Machine Learning. There are some **prerequisites** for this division:

* Introductory Algebra
* Fluency in English
* Fluency in at least one programming language (preferably Python)
  + Basic data structures, loops, if statements and functions
  + Some knowledge of libraries (as known as packages, modules)

Note that not all of the concepts mentioned above will be used during the competition, but it is a good benchmark to check if you are qualified for the difficulty of this division.

We require all participants in Division SIGMA to use **Python**, as it is one of the most popular languages for Machine Learning. If you are proficient in another language, make sure to learn about Python’s basic programming syntax using the resources below. We believe the transition into using Python will not be too challenging, since all concepts and logics are the same, but with different syntax.

**Rules**

**Definition of Terms:**

* Ignition Hacks: The Ignition Hacks Competition that starts on August 22nd, 2020, EST, and ends on August 23rd, 2020, EST.
* Division Sigma: The division of Ignition Hacks that targets experienced programmers and focuses on Artificial Intelligence, especially machine learning.

**BEFORE** the date of the competition:

* If any participant by chance discovers or predicts competition information not yet released, which includes but is not restricted to Division Sigma’s competition problem, please do not discuss with others.
* Any malicious attempt or action to attack the Ignition Hack website or the Ignition Hacks media platform is prohibited. This includes spams, inappropriate language usages, retrieval of confidential information, etc.

**DURING** the competition:

* Plagiarism of an existing solution on the internet is prohibited. Any team caught copying code from the internet will be disqualified.
* As Division SIGMA focuses on helping beginners to AI, we recommended participants to use simple supervised machine learning models to solve the problem. However, other more complicated techniques such as **neural networks and deep learning are allowed**.
* Participants CAN join or switch teams during the competition. Team information should be finalized at submission.
* Collaboration with other teams is prohibited. Similarities in code structure will be carefully examined. Therefore if collaboration is suspected, teams will risk disqualification.

**SUBMISSION** and post-competition:

* Submit the entire project before the deadline, which includes source code files and a CSV file containing the predicted labels **(in the order that the features are given)** for the judgment data
* Submissions of the template code with no customization will not be allowed.
* All projects submitted must be created during the timeframe of the hackathon
* Submission is closed on August 23rd at 11:59 pm EST.
* You can discuss information about the hackathon after the submission deadline.
* Results will be released approximately a week after the competition ends. Scored will be determined based on the **accuracy, creativity, and commenting/explanation** of your solution. The rubrics will be released on the date of the competition.
* More information about submissions will be released on the competition date.

**Guides**

**BEFORE** the date of the competition:

* Sign up on Google Form and Devpost.
* Read the rules and requirements and familiarize yourself with the hackathon schedule and the submission process.
* Register for a GitHub account.
* It is recommended to download the GitHub Desktop version.
* **Watch recommended resources** provided to learn about GitHub, Google Colab, Python, and Machine Learning.

**DURING** the competition:

* Download the training and judgment dataset and sample files provided on the competition day.
* Research about the methods of **checking the accuracy of a model using only the training data**. There are plenty of pre-built functions that accomplish this goal. **HINT:** Split the training dataset into two groups. Train your model using the first group, while testing the accuracy using the second group.
* **Tip**: Effective communication between team members and creativity is one of the keys to success.
* **Tip**: There are a plethora of models with a variety of hyperparameter tuning, so always try to incorporate some new ideas and be creative to develop the optimal model.
* **Tip**: Google Colab provides an opportunity to utilize a virtual GPU for training purposes instead of a regular CPU which can save a lot of time and resources! The details are listed in the official documentation.

**SUBMISSION** and post-competition:

* Create your repository on GitHub.
* Upload your solution to Git Repository.
* Submit your solution on Devpost by providing a link to your GitHub Repository.
* It is highly recommended to include a README in your GitHub repository with explanations of the files and code.
* **IMPORTANT**: It is mandatory to include a detailed description of your entire project (such as models/techniques implemented and your approach towards the problem) with your Devpost submission.