Machine Learning Engineer Nanodegree

Capstone Proposal

Ronaldo da Silva Alves Batista
07 de Março de 2017

Proposal

DonorsChoose.org Application Screening Kaggle's Competition launched on March 1st 2018

Background

Founded in 2000 by a high school teacher in the Bronx, DonorsChoose.org empowers public school teachers from across the country to request much-needed materials and experiences for their students. At any given time, there are thousands of classroom requests that can be brought to life with a gift of any amount.

DonorsChoose.org receives hundreds of thousands of project proposals each year for classroom projects in need of funding. Right now, a large number of volunteers is needed to manually screen each submission before it's approved to be posted on the DonorsChoose.org website.

I intend to apply Data Science skills to help NGO's connected to poverty alleviation that's why this project is of interest to me, we need a lot more than food to achieve a just and prosper society and education is an immensely important asset for this goal.

Problem Statement

Next year, DonorsChoose.org expects to receive close to 500,000 project proposals. As a result, there are three main problems they need to solve:

1. How to scale current manual processes and resources to screen 500,000 projects so that they can be posted as quickly and as efficiently as possible
2. How to increase the consistency of project vetting across different volunteers to improve the experience for teachers
3. How to focus volunteer time on the applications that need the most assistance

The goal is to predict whether or not a DonorsChoose.org project proposal submitted by a teacher will be approved, using the text of project descriptions as well as additional metadata about the project, teacher, and school. DonorsChoose.org can then use this information to identify projects most likely to need further review before approval.
With an algorithm to pre-screen applications, DonorsChoose.org can auto-approve some applications quickly so that volunteers can spend their time on more nuanced and detailed project vetting processes, including doing more to help teachers develop projects that qualify for specific funding opportunities.
The machine learning algorithm can help more teachers get funded more quickly, and with less cost to DonorsChoose.org, allowing them to channel even more funding directly to classrooms across the country.

Input and Data Sets

The competition's dataset contains information from teachers' project applications to DonorsChoose.org including **teacher attributes**, **school attributes**, and the project proposals including **application essays**. My objective is to predict whether or not a DonorsChoose.org project proposal submitted by a teacher will be approved.

File Descriptions

- **train.csv** - the training set
- **test.csv** - the test set
- **resources.csv** - resources requested by each proposal; joins with `test.csv` and `train.csv` on `id`

Data Fields

Data fields

test.csv and train.csv:

- id - unique id of the project application
- teacher_id - id of the teacher submitting the application
- teacher_prefix - title of the teacher's name (Ms., Mr., etc.)
- school_state - US state of the teacher's school
- project_submitted_datetime - application submission timestamp
- project_grade_category - school grade levels (PreK-2, 3-5, 6-8, and 9-12)
- project_subject_categories - category of the project (e.g., "Music & The Arts")
- project_subject_subcategories - sub-category of the project (e.g., "Visual Arts")
- project_title - title of the project
- project_essay_1 - first essay
- project_essay_2 - second essay
- project_essay_3 - third essay
- project_essay_4 - fourth essay
- project_resource_summary - summary of the resources needed for the project
- teacher_number_of_previously_posted_projects - number of previously posted applications by the submitting teacher
- project_is_approved - **target variable** - whether DonorsChoose proposal was accepted (0="rejected", 1="accepted"); `train.csv` only

`resources.csv`
Proposals also include resources requested. Each project may include multiple requested resources. Each row in resources.csv corresponds to a resource, so multiple rows may tie to the same project by id.

- id - unique id of the project application; joins with `test.csv` and `train.csv` on id
- description - description of the resource requested
- quantity - quantity of resource requested
- price - price of resource requested

A note on essay data

Prior to February 18th, 2010, for their DonorsChoose application, teachers had the option of writing either a free-form essay (split into `project_essay_1`, `project_essay_2`, `project_essay_3`, and `project_essay_4`) or writing free-form answers to the following four prompts:

- Introduce your classroom (`project_essay_1`)
- Describe the situation (`project_essay_2`)
- Describe the solution (`project_essay_3`)
- Empower your donors (`project_essay_4`)

Effective February 18th, 2010, the option to write a free-form essay was removed, and all teachers were required to respond to the following four prompts:

- Open with the challenge facing your students (`project_essay_1`)
- Tell us more about your students (`project_essay_2`)
- Inspire your potential donors with an overview of the resources you're requesting (`project_essay_3`)
- Close by sharing why your project is so important (`project_essay_4`)

Description of Solution

This is a binary classification problem with complex textual features. Given the mix between categorical variables in the form of the proposals meta-parameters and the actual content of the proposals (`project_essays`), I don't know yet which algorithm is appropriate, I'll try to apply some Ensemble methods, like XGBoost, and research how to deal with the NLP part of the challenge. Some heavy feature engineering is likely necessary. It's a recent open challenge on Kaggle and I'll perform this as a learning experience.

Benchmark

There is a Linear classifier benchmark provided with the Challenge which doesn't take into account the project essays at all, and for this reason is only slight better than chance: `Area under ROC Curve: 0.56`. A significant improvement over this benchmark is expected, my personal goal is `AUC > 0.7`. If the model performs as expected a bigger goal is to fine-tune the model and to be among the first half of competitors in the competition leaderboard.

Evaluation Metric

For each `id` in the test set, we'll predict a probability for the `project_is_approved` variable. The metric provided for the competition is area under ROC curve between the predicted probability and the observed target.

Project Design

Since the problem dataset, metric and goal are very well defined. The bulk of the workflow will be:

- Exploratory Data Analysis
  - Label Encoding
  - Create Category Aggregations
  - Cluster Categories

    - Explore relationships between the different fields in the data
    - Visualization
- Prepare the Data
- Possible algorithms to apply:
    - Ensemble Learning - XGBoost
    - Sparse Data and Embeddings
    - How to work with text data and vocabulary
    - Possibly some Neural Network
- Fine-tune the Algorithms
- Check the results compared to the benchmark