Trayer Harvey
001341276
6/16/22

C951 Task 3:
MACHINE LEARNING PROJECT PROPOSAL

SCENARIO
In this assignment, you will assume the role of a recently hired machine learning engineer in an organization that has been asked to use available data to identify an organizational need that could be solved by machine learning. The organization then needs to outline a plan for designing and executing a machine learning model to solve this organizational need. You will explore available data sets and machine learning algorithms supporting the construction of a model that the organization would use, following the plan that you put together. You will outline the problem, the solution, and the project plan, as well as the framework for evaluating the success of the model and the project.

A. Create a proposal for a machine learning project

This proposal outlines our proposed machine learning solution to the issue of Disaster Fatigue and Outrage Promotion in our daily media consumption. Namely, we aim to use a machine learning model to process headlines and the reactions to them under categories, filtering out those deemed negative or anger-inducing, ideally leading to reduced anxiety among users.

A1. Describe an organizational need that your project proposes to solve.

As more users consume more content year after year, companies get more desperate for our views, and their algorithms no longer have the users well being at heart, instead prioritizing “clicks”. To remedy this, users should have a heads up on the kind of content they’re about to receive, and choose whether or not they want to be subjected to it. There is a need for a product to filter the incoming content stream, and give the user more control over what they’re fed by the algorithm. Sentiment-detecting neural networks could prove to be a solution here, as they could analyze the content and label anything that could promote anxiety in the user.

A2. Describe the context and background for your project.

More people than ever are using social media. And as these companies grow and thrive, capitalism dictates that they grow faster, and they begin implementing any possible tools they can think of to increase clicks and time spent on their platform. To that end, most of the loudest, most re-tweeted content ends up being outrage-promoting, or overtly sad, or needlessly dramatic. We humans, however, are physiologically affected by our emotions, and a deluge of negative-yet-engaging posts can cause real-world harm in its users.

In order to help improve the user experience, there needs to be a detection system in place that can “understand” the feeling (or sentiment) behind content. A basic neural network trained on human-labeled data could potentially differentiate between negative and positive posts, while a more advanced system could recognize more nuanced emotions like angry, sad, happy, or gross.

A3. Review three outside works that explore machine learning solutions that apply to the need described in part A1.

1. A Semi-supervised Learning Approach For Tackling Twitter Spam Drift [1]
    - These researchers also used a machine learning technique to solve a social issue, in this case, Twitter spam. The issue they found was “spam tweet characteristics are changing over time, which affect the performance of the traditional ML algorithms”(p. 15). So they tested an SSL using a ‘Yet Another Two Stage Idea’ (YATSI) algorithm to classify data that was extracted from Twitter. In the end “the results showed that SSLA can reduce the effect of Twitter Spam Drift”(p. 16) which helps give credence to our SSL achieving similar results.
2. Sentiment Analysis with Deep Learning Models: A Comparative Study on a Decade of Sinhala Language Facebook Data [2]
    - This research paper involved a general goal of proving that Facebook reaction data is worth exploring, and ultimately they achieve a specific goal of being reasonably certain (84.58%) that Facebook reactions are suitable predictors of text sentiment in a post. In order to determine this, they used a Bidirectional LSTM model to analyze over 15000 Facebook news comments with millions of reactions, comparing the results, concluding “it is safe to claim that Facebook reactions are suitable to predict the sentiment of a text”(p.1)
3. Deep learning for detecting inappropriate content in text [3]
    - Our challenge for this project involves using machine learning to protect the end user, and this paper is a study on just that. Specifically, they found that bad messages were “degrading the effectiveness of user experiences”(p.1), and sought a deep learning solution. They applied a “Convolutional Bi-Directional LSTM (C-BiLSTM)"(p.1) architecture for identifying inappropriate queries, and found that that a machine-learning solution “significantly outperforms both pattern-based and other hand-crafted feature-based baselines.”(p.13)

a. Describe how each reviewed work from part A3 relates to the development of your project.

1. The project involved using a Semi-Supervised Learning algorithm to analyze and label content spam with the end goal of helping the user experience, so it’s a helpful parallel as our solution uses an SSL as well. It also helps that our prototype will be based for operating on Twitter so having an SSL successfully moderate results on the same targeted platform is promising.
2. Facebook had a similar problem to our own, where their users desired labeled content. So in 2016 Facebook overhauled its ‘Like’ system in order to allow users to label their own content, with Reactions. Thus saving them from having to do any work at all, and creating a freely monetizable database! It’s such good data in fact, that this paper compared it to data labeled with leading machine learning algorithms, and found that reactions were able to beat the accuracy mark at predicting post sentiment. For this project we aim to use Facebook data as one of our training data sets for our model, and this paper validate that choice.

3. One of the questions that comes up often when explaining machine learning is ‘why bother with a machine at all, why not use human moderators/workers?’ and that’s a perfectly valid question. There’s no reason to use a machine learning model if it’s not more efficient than a human-based model in the first place. Thankfully, this paper discusses how machine learning solutions can outperform hand-crafted baselines when it comes to content moderation on social media. So our approach here is not unwarranted (double negative)

A4. Summarize the machine learning solution you plan to use to address the organizational need described in part A1.

In order to reduce the overall anxiety of user while they consume their daily content, we propose a system that will analyze what content the user is about to consume, categorize it into an emotion using a sentiment-analyzing neural network, then display to the user an option to view the content based on its feeling

A5. Describe the benefits of your proposed machine learning

Using Twitter as an example, most popular social media sites have millions of new pieces of content (in twitter’s case, tweets) added every day, so it remains infeasible that they could hire a workforce large enough to label the feeling of posts by hand. So instead of spending money hiring humans to do an impossible job, we spend the money to gather a data set, hiring an engineer to code a neural network, and train our own AI. The two main benefits of this solution are quantity: once programmed, the AI can run on any device with internet access, and speed: an AI can read text faster than a human can see, without even mentioning typing speed discrepancies.

B – Machine Learning Project Design

B1. Define the scope of the proposed machine learning project.

For the current proposed budget, the scope of the project involves creating an application that will act as the front end for the social media site “Twitter”, and categorize trending twitter posts into different emotional centers.

Planned features:

- A front-facing user interface allowing the user to select content
- A backed process to scan Twitter’s API for content
- A trained neural network ready to categorize content in real time
- A hiding feature for content the user doesn’t want seen

Out of Scope at Current Budget:

- Image/Video analysis
  - Training neural networks is expensive, and Twitter is primarily text-based, so for this prototype we’ll be focusing solely on text analysis
- Non-Twitter site content analysis
  - This application is designed to work with Twitter’s API only at the moment, but in the future can be expanded to analyze news sites as well as other social media platforms

B2. Explain the goals, objectives, and deliverables for the proposed project.

Goals:

Our application can be considered a success if it….

- Has a functional user interface to select and hide content
- Accurately and appropriately assigns feelings to content
- Can work in real time as the user browses and doesn’t hamper the experience

Objectives

- Improve upon the current Twitter browsing experience
- Function as a proof-of-concept to secure funding for further projects and research

Deliverables

- An application designed to help users who experience anxiety browsing Twitter

B3. Explain how you will apply a standard methodology (e.g., CRISP-DM, SEMMA) to the implementation of your proposed project.

Our chosen methodology is

SEMMA

- Sample
  - After a basic model has been constructed, large datasets will be fed to it for training. This data will be randomly sampled to identify variables or factors influencing the process, then sorted.
- Explore
  - Our data scientists will perform uni-variate (single factor) analysis, finding the relationships between individual factors and effects, and multivariate (multiple factor) analysis, finding the relationship between many data elements to identify gaps in the data
- Modify
  - The data is then analyzed and visualized, used as a guideline for how to change and ultimately improve the benchmarks of our model

- Model
  - Once processed and logged for tracking and archival purposes, the data can be modeled to evaluate how the efficiency and other goalposts have changed over time, and to make predictions about how they will change in the future
- Assess
  - The model is ultimately evaluated for how useful it is to the problem at hand. Data is compared to predictions made and benchmarks taken in the past, to determine it's overall level of performance. The cycle then repeats as long as funding is maintained and results prove fruitful.

B4. Provide a projected timeline for the proposed project, including the start and end dates for each task.

Project Start: Proposal is accepted

Week 1-2 : Form Team, assign roles
Week 3 : Create technical proof of concept, submit for approval end of week (EOW)
Week 4 : If approved, begin work on Application, begin gathering data sets to train on
Week 5-6 : Build and train neural network
Week 7 : Build user interface, back-end twitter interface, implement AI
Week 8 : Final testing with submitted deliverables at EOW

Sprint	Start	End	Tasks
1 - Preparation	07-04-2022	07-17-2022	-Form Team -Assign Roles
2 - Proof of Concept	07-18-2022	07-24-2022	-Create technical proof of concept
3 - Begin Full Application	07-25-2022	07-31-2022	-Begin programming front-end of application -Gather data sets from Yelp, Facebook, etc.
4 - Neural Network	08-01-2022	08-14-2022	-Build neural network -Train neural network
5 - Final Steps	08-15-2022	08-21-2022	-Finish UX -Add back-end Twitter API interface -Implement trained model into app algorithm
6 - Pre-Release Testing	08-22-2022	08-28-2022	-Final Testing and bug fixing -Submit deliverables for release 8/28

B5. List resources (e.g., hardware, software, work hours, third-party services) and all associated costs needed to implement the proposed solution.

	Resource	Description	Cost
Upfront	Servertime	Hosting Fees	$100
	Hardware	Computers (work from home, no purchase necessary	Free
			$100
Hourly	Software Engineers	1-2 programmers for the application	$200/hr
	Data Scientists	1-2 neural network programmers	$200/hr
			$128,000
	weeks	hours	
	8	320	**TOTAL COST**
			$128,100

B6. Describe the criteria that you will use to evaluate the success of the project once it is completed.

Objective	Success Criteria
Functional UI	All buttons and controls act as intended and are understood by the user intuitively
Neural network Efficiency	Our model should be able to accurately identify and label content in equal to or greater than 95% of cases
Application Speed	The application responds to user's actions in real time, and processing doesn't slow down the browsing device

C – Machine Learning Solution Design

C. Describe the proposed machine learning solution you will use to address the organizational need identified in part A1 by doing the following:

C1. Identify the hypothesis of the proposed project.

We hypothesize that our algorithm will satisfy the 95% accuracy required by our business case

C2. Identify the machine learning algorithm(s) (i.e., supervised, unsupervised, or reinforcement learning) you will implement in your proposed solution.

Given that the problem at hand is text classification, we determined the best option for our current resources is Semi-Supervised Learning

- a. Justify the selection of the algorithm in part C2. Include one advantage and one limitation of the selected machine learning method.
    - As the mid-point between Supervised and Unsupervised learning, Semi-Supervised Learning (SSL) aims to strike a balance between the weaknesses of each. SSL starts with a set of tagged training data that is used to form a model, which can then be used to tag similar, untagged data. This newly tagged data can then be fed back into the model, ideally leading to a larger data set and better model overtime. There are clear advantages to only needing to use a small data set to train a model for a large one, labeling data is expensive after all. And feeding output data back in after the model has categorized it is a great way to strength the biases of the system without needing gargantuan amounts of data (Amazon claims they need 40 times less data than they did after shifting their Alexa model to SSL[8]). But biases should only be reinforced if they're good, and feeding data back into a model can potentially lead to and overall worse model if the programmers aren't careful. The SSL model is unable to correct its own mistakes, the programmers must be wary of any high confidence-(but wrong) predictions that can corrupt the whole model.

C3. Describe the tools and environments that will be used to develop the proposed machine learning solution, including any third-party code.

Primary Language: Python

- Python has a nice balance between depth and ease of use and has been used in numerous deep learning projects, so it will be our language of choice for this algorithm

IDE: PyCharm

- As the industry standard for Python development, PyCharm will be more than sufficient for our purposes

Libraries:

SemiSupervised by PyPi

- As mentioned in C2, we will be using an SSL for our machine learning model. Specifically, we'll be using a Label Propogation Algorithm. On Python, the semisupervised library by PyPi provides support for a LabelPropogation classifier [4]

BeautifulSoup

- If our team is unable to procure datasets of our own or simply wants to gether data from the web to test, BS is used to scrape the internet for site information to convert into our data set [5]

TweePy

- Seeing as how our application ultimately has to work with Twitter, TweePy is the oldest and most highly reviewed Twitter API interactive Python Library [6]

C4. Explain the process you will use to measure the performance of your proposed machine learning solution.

We will focus on two main Performance Metrics when evaluating our algorithm:

Classification Accuracy

- When it comes to text analysis, classification accuracy is the big one, and can be determined by measuring the number of correct predictions and dividing that by the number of total predictions made.

Logarithmic Loss

- In addition to measuring success, we will also punish failures by way of Log Loss. This is ideal for multi-class classification (such as sentiment analysis), as the algorithm works by assigning a probability to each sample belonging to a class (or not) and measuring that against the actual label (), with the errors tabulated. Minimizing Log Loss gives greater accuracy as it's minimized (closer to 0)

Description of Data Set(s)

D. Describe the data for your proposed project by doing the following:

D1. Identify the source(s) of the data for your proposed project.

There are three categories of data for this project:

Labeled Training Data

- Certain academies already have open source labeled data repositories, imdb has open source data[7]. If we go the web scraping route, Yelp has reviews labeled with Stars to indicate sentiment, Facebook has posts with labeled reactions

Unlabeled Training Data

- This is possible but unrealistic, using scraped data and then manually labeling all entries with humans for use as training data

Unlabeled Data to test On

- Ultimately results will be pulled from Twitter's trending tags

D2. Describe the data collection method.

Certain data sets can simply be downloaded, while other data sets need to be assembled by scripts. The Python library BeautifulSoup provides tools to scrape the web, so python scripts can be written that will scrape our source sites such as Yelp, Facebook, etc. to create data sets for cleaning

a. Discuss one advantage and one limitation of the data collection method described in part D2.

Our method of data collection is limited by budget at the moment. When limited to capital free options, our choices become limited to investing our time instead. Namely using our programming time to write scripts before even starting the algorithm. As a positive, with a team of two programmers and two data scientists, we possess the skill sets to make a tool/bot to gather any piece of data from the free web, ideally giving us a wide variety of testing data to work with

D3. Explain how you will prepare your data for use by the machine learning algorithm(s) from part C2 for your proposed project, including data set formatting, missing data, outliers, dirty data, or mitigation of other data anomalies.

For data scraped from the web, we will be conducting a relatively straightforward sentiment analysis, so a large chunk of data can be stripped away via program. All punctuation, all common and non-emotional parts of speech ('a', 'the', 'or', etc.). Our labels need to be standardized across data sets, so if data from Facebook uses Reactions, data from Yelp that uses Stars needs to be programatically adjusted to match. To keep things simple, we should probably filter out emojis too, but I personally would like to keep them implemented if feasible (more study necessary).

D4. Describe behaviors that should be exercised when working with and communicating about sensitive data in your project.

Ideally all of our data comes from public, front-facing web-sites so sensitive data shouldn't be present in the first place. But in order to be safe, our data sets won't contain any names (user or otherwise) or locations. If any database in our processing needs to store names for duplication-prevention purposes, we'll hash the names so they're unreadable even to the researchers.

As for our sensitive data, we shouldn't have any **crazy** amount of data, so following the 3-2-1 rule should be fairly affordable: Keeping three copies of the data, on two different media, with one copy offsite. So we'll have a copy as an organization, we'll have a copy each personally on work hard drives, and we'll keep a backup in the cloud, syncing all three at the beginning and end of each work day.

Our workers are professionals, and should know not to send any sensitive information over email, and not to write their passwords down on a sticky note taped to their monitor. They should also know not to go decrypting user locations in the company database (unless it's for an organization-approved and supervised process like bug fixing)

E. Acknowledge sources, using in-text citations and references, for content that is quoted, paraphrased, or summarized.

1. Niddal Imam, Biju Isaac, & Jacob, S. M. (2019, June). *A Semi-supervised Learning Approach For Tackling Twitter Spam Drift*. (pp. 1-19). ResearchGate. Retrieved June 16, 2022, from https://www.researchgate.net/
2. Weeraprameshwara, G., Jayawickrama, V., de Silva, N., & Wijeratne, Y. (Jan 2022). (rep.). *Sentiment Analysis with Deep Learning Models: A Comparative Study on a Decade of Sinhala Language Facebook Data* (pp. 1–8). Arxiv. Retrieved June 16, 2022, from https://arxiv.org/abs/2201.03941
3. Yenala, H., Jhanwar, A., Chinnakotla, M. K., & Goyal, J. (2017). Deep learning for detecting inappropriate content in text. *International Journal of Data Science and Analytics*, *6*(4), 273–286. https://doi.org/10.1007/s41060-017-0088-4
4. *Semisupervised Library 0.0.28*. PyPI. (2021, January 7). Retrieved June 16, 2022, from https://pypi.org/project/semisupervised/
5. *Beautiful Soup Documentation*. Beautiful Soup 4.9.0 documentation. (n.d.). Retrieved June 16, 2022, from https://www.crummy.com/software/BeautifulSoup/bs4/doc/
6. Tweepy. (n.d.). *Tweepy: Twitter for python!* GitHub. Retrieved June 16, 2022, from https://github.com/tweepy/tweepy
7. Maas, A. (2011). Large Movie Review Dataset. Retrieved June 16, 2022, from https://ai.stanford.edu/~amaas/data/sentiment/.
8. Bezos, J. P. (1997). 2017 Amazon Shareholder Letter. https://www.sec.gov/Archives/edgar/data/1018724/000119312518121161/d456916dex991.htm