Final Year Project

Project Specification

Spoiler Alert: Deep Learning with Natural Language Processing to identify spoilers

By Yasmin Paksoy

October 2021

Table Of Contents

1 Work Overview

I plan to create a web browser extension for Twitter which will consist of two distinct components: the first, a deep learning algorithm tasked with natural language processing (NLP) that can identify which television show or movie a tweet is about. The second, the browser extension itself which will scan the users twitter feed and block them from seeing certain tweets related to that television show or movie.

This program will be aimed at helping users avoid spoilers for any television shows or movies they are yet to watch. Using natural language processing should lead to a much more advanced system as tweets tend to be very nuanced and no longer necessarily mention the exact TV show or movie they are referencing. While Twitter does allow users to "mute" certain words [1], the process cannot block via topic and can lead to users still seeing spoilers.

My main focus for this project will be developing the algorithm that will be able to identify spoilers. For this, I will develop a deep learning algorithm that uses natural language processing to classify the topic a tweet is about. I will be implementing this algorithm using Python and a range of libraries.

2 Technologies and Materials

The main technique I will be implementing for this project will be deep learning. This will require the use of a neural network with many layers of representations in order to achieve the highest level of accuracy possible.

As previously mentioned, I will be using Python for this project. This is the best option as all of the libraries I plan on making use of also use Python.

When coding, I plan on using Google Colab, which hosts Jupyter notebooks. Colab runs entirely on the cloud, meaning it can train large machine-learning and deep-learning models without needing access to a powerful machine, making it a better option than a local Jupyter notebook, which would have hardware limitations.

This project will not require the use of any physical materials other than a computer.

2.1 Libraries

There are various libraries that I will be using for this project, which include:

- Tweepy, a Python library that facilitates connection to Twitter's API.

  https://www.tweepy.org/

- TensorFlow, a machine learning library that facilitates training of deep neural networks.

  https://www.tensorflow.org/

- Natural Language Toolkit (NLTK), an open-source Python library for NLP.

  https://www.nltk.org/

3 Motivation Research

Twitter is an endless sea of information, and a user's feed is not only populated by the tweets of people they follow, but also the tweets those users have interacted with and other content the Twitter algorithm believes the user will enjoy, based on the most trending topics during that time. This leaves users feeds open to all content and makes it very difficult for them to manually avoid seeing content about certain topics.

In addition, Twitter is the number one social media platform where people go to share their opinions on topics such as television shows and politics. While there are other social media platforms that have generally more users, such as Instagram [2], Twitter is favoured by users to discuss the day's biggest events and news.

Both of these aspects combined makes Twitter a notorious place to see spoilers for television shows and movies. Many users avoid using Twitter if they have not yet seen a movie or television show [3]. Most recently, The Netflix television show "Squid Game" went viral, leading to many watchers steering clear of Twitter, knowing they were bound to come across spoilers, ruining the experience of watching the show.

As mentioned previously, Twitter only allows users to mute certain words, meaning users have to manually add every word that can relate to a subject in order to block it. This is not only tedious but does not work very successfully since users can miss certain words, resulting in them still see tweets relating to the topic, and tweets including muted words that are not actually related to that topic also get blocked.

These difficulties are the inspiration and motivation for this project; as an avid Twitter user myself, and someone who despises spoilers of my favourite shows, avoiding Twitter on days where new episodes are released has become the, frustrating, only option. This is a problem felt by many, as expressed in many recent online articles [4].

Previously, an AI called SpoilerNet was developed by researchers at the University of San Diego, who used book reviews with the "Spoiler Alert" from Goodreads [5], a community centred around books, to train a neural network. When testing using spoilers for television shows, they found their algorithm had 80% accuracy [6]. I would like for this project to be more focused on detecting television show spoilers to improve on their 80% accuracy.

4 Existing Knowledge

I have experience coding in Python, from one of my previous modules, Foundations of Programming, and from a module I am current studying, Artificial Intelligence.

Additionally, I studied Algorithms & Data Structures last year, which will assist me in the creation of the algorithm. The current modules I am studying, Neural Networks and Artificial Intelligence, will provide necessary knowledge in regards to using deep learning techniques to develop an NLP algorithm.

I will also be studying Data Mining next term, which will support this project by providing necessary knowledge in regards to data analysis.

5 New Knowledge

In order to complete this project, I will have to familiarise myself with the libraries that I have not worked with before, such as NLTK and Tweepy. Both of these libraries have extensive documentation that will be essential in order to implement them into my algorithm successfully. There are also many tutorials online, written by programmers, that show implementations of these libraries, along with educational articles shared by companies such as GeeksforGeeks [7], which will be beneficial.

I will also have to improve my knowledge of Python and TensorFlow to be able to develop the complex algorithm required here. This will mainly be achieved by completing my Artificial Intelligence module and referring to books such as Deep Learning with Python by Francois Chollet.

6 Timeline and Milestones

Date	Milestone	Tasks	Deliverables
11/10/21 - 22/10/21	Initial Research	Finalise project idea. Finalise minimum viable product. Decide which technologies to use.	Project Specification
23/10/21 - 29/10/21	Further Research	Research deep learning algorithms. Familiarize myself with Tweepy. Decide on a dataset.	
30/10/21 - 19/11/21	Prototyping	Develop prototype of algorithm.	Prototype
20/11/21 - 19/12/21	Implementation	Learn how to use NLTK. Start developing algorithm.	
20/12/21 - 2/01/22	Christmas break		
03/12/21 - 09/01/22	Report	Evaluate work so far. Decide on next steps. Write interim report.	Interim Report
10/01/22 - 10/02/22	Implementation	Finish working on algorithm. Work on testing during development. Make any changes resulting from tests.	
11/02/22 - 18/02/22	Design Presentation	Decide on aspects to focus on. Summarise processes as talking points. Make presentation slides.	Presentation Slides
19/02/22 - 25/02/22	Reading Week	Use time to catch up on other modules.	
26/02/22 - 13/03/22	Testing	Complete all testing.	
14/03/22 - 01/04/22	Report	Confirm report sections. Start write up and get feedback.	
02/04/22 - 02/05/22	Final	Complete any final fixes. Finish report write up and review. Extra time included as contingency planning.	Final Report & Code

6.1 Minimum Viable Project

While being able to develop and apply such an algorithm to a web browser extension would be ideal, it would not be possible to achieve this to a high enough standard in the given time frame. Because of this, I will be focusing on developing this algorithm to a high standard is instead. My MVP should be able to pass testing processes with accuracy above 80%.

6.2 GitHub Repository

Link: https://github.com/ysmnpksy/Final-Project

References

[1] Anon. How to use advanced muting options. Twitter. [online] Available at: https://help.twitter.com/en/using-twitter/advanced-twitter-mute-options [Accessed 18 October 2021].

[2] Statista. 2021. Most used social media 2021. Statista. [online] Available at: https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users [Accessed 20 October 2021].

[3] Alexander, J. 2021. How Trying to Avoid Spoilers in 2021 Became Futile. IGN Middle East. [online] Available at: https://me.ign.com/en/movies/186513/feature/how-trying-to-avoid-spoilers-in-2021-became-futile [Accessed 20 October 2021].

[4] Rose, S. 2021. A tangled web: have super-fans ruined the new Spider-Man movie? The Guardian. [online] Available at: https://www.theguardian.com/film/2021/feb/15/have-super-fans-ruined-new-spiderman-movie [Accessed 18 October 2021].

[5] Riccio, G. 2019. SpoilerNet, the anti spoiler AI: it signals them to you and avoids them. Futuroprossimo.it. [online] Available at: https://en.futuroprossimo.it/2019/07/spoilernet-anti-spoiler [Accessed 18 October 2021].

[6] Harin. 2019. Don't Want To See Spoilers? This AI Tool Will Help You. Mobygeek.com. [online] Available at: https://mobygeek.com/features/dont-want-to-see-spoilers-this-ai-tool-will-help-you-6951 [Accessed 18 October 2021].

[7] GeeksforGeeks. 2018. Extraction of Tweets using Tweepy - GeeksforGeeks. [online] Available at: https://www.geeksforgeeks.org/extraction-of-tweets-using-tweepy [Accessed 18 October 2021].