

# Machine Learning & Analysis for Twitter Bot Detection

## Problem statement

There are an average of 6,000 tweets produced on Twitter per second. Twitter posts are mostly public and can be easily collected using Twitter's developer platform API. Also, frequent use of hashtags makes it more interesting to draw conclusions.

The role of so-called social media "bots" automated accounts capable of posting content or interacting with other users with no direct human involvement has been the subject of much scrutiny and attention in recent years [1]. These accounts can play a valuable part in the social media ecosystem by answering questions about a variety of topics in real time or providing automated updates about news stories or events. At the same time, they can also be used to attempt to alter perceptions of political discourse on social media, spread misinformation, or manipulate online rating and review systems [2]. As social media has attained an increasingly prominent position in the overall news and information environment, bots have been swept up in the broader debate over Americans changing news habits, the tenor of online discourse and the prevalence of fake news online [3].

In order to detect bots, classification models and/or natural language processing techniques such as topic modeling and sentiment analysis can be incorporated. This project will involve feature engineering and will provide a real-world data collection experience.

**Project goal:** In this project, the goal is to detect Twitter bots using tweets data from the Twitter developer API by utilizing machine learning techniques.

## Data Resources

You will collect your own data for this project. We provide a basic Python script, `tweepy_script.ipynb`, that utilizes the `tweepy` library [4] to access the Twitter API. The provided `tweets_sample.json` file demonstrates a sample of what you will collect and the attributes of a tweet made available by the Twitter API [5].

## High-level project goals

1. The first step is to create your own dataset. Mine the data for the project using the Twitter API and utilize feature engineering and pre-processing techniques to prepare the data for analysis.
2. Create several models to determine characteristics of different types of twitter users. Create at least one model that uses natural language processing techniques, such as topic modeling [6], and at least one model that uses a classification algorithm. You may decide to have models that use both. You should provide evidence of success at detecting bots when compared to human users or explain why it wasn't possible.

3. Perform a comparison of your models. This should include an error analysis and an evaluation of the predictive quality of your models.

## References

1. Stefan Wojcik, "Bots in the Twittersphere": <http://www.pewinternet.org/2018/04/09/bots-in-the-tweetsphere/>
2. Chris Baraniuk, "How Twitter Bots Help Fuel Political Feuds": <https://www.scientificamerican.com/article/how-twitter-bots-help-fuel-political-feuds/>
3. Chengcheng Shao et al., "The spread of low-credibility content by social bots": <https://arxiv.org/pdf/1707.07599>
4. The tweepy Python library: <http://www.tweepy.org>
5. Twitter's developer resources to learn about the API: <https://developer.twitter.com>
6. Asbjan Ottesen Steinskog et al., "Twitter Topic Modeling by Tweet Aggregation": <http://www.aclweb.org/anthology/N18-10210>