

Homework: Large Scale Generation of Falsified Scientific Literature and Detection

Due: Friday, April 3, 2020 12pm PT

1. Overview

Counteracting neural disinformation with Grover

Exploring the surprising effectiveness of a fake news generator for fake news detection



Rowan Zellers [Follow](#)
Jun 18, 2019 · 8 min read



By Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi



Artificial Intelligence has enormous potential to benefit society. However, the same technology can cause harm, particularly if used by malicious adversaries. One important threat is that of “Neural Fake News”: machine-written disinformation at scale. Our

Figure 1: Neural network generated falsified information.

In this second assignment, we will leverage your augmented Bik dataset and features and use it to generate believable, fake scientific literature in order to see if you can automatically classify whether or not the media was falsified like you attempted to do in the first assignment through feature augmentation and the five Vs. To do this, we will leverage two techniques for falsified content generation at scale – the [Grover Neural Disinformation Generation](#) to generate believable scientific text and the [Deep Convolutional Generative Adversarial Networks \(DCGAN\) technique](#) to generate believable scientific images from existing learned representations.

Your goal is to parse out and extract the text from the 200 scientific publications identified in the Bik dataset using Apache Tika, and use the text to create a new Grover model that can in turn generate new “fake” scientific, believable literature. You will also train your own DCGAN on the associated images with the literature, and then generate believable imagery to go along with your new papers and generate Image Captions to go along with those new paper figures. You will then generate 500 fake PDFs of scientific papers by automatically generating LaTeX papers (.tex files)

and then converting them to PDF. There are [plenty of sample posts on how to automatically generate LaTeX papers from Python](#), so search the web for a few of them you shouldn't have a problem there.

Face Generator - Generating Artificial Faces with Machine Learning 🧑

Creating Realistic Faces with DCGANs



Greg Surma [Follow](#)

Mar 4, 2019 · 5 min read



Figure 2: Using DCGAN to generate false faces – you will generate false scientific images.

You will also use another advanced extraction technique in order to generate information from your falsified multimedia (image/*) figures in your papers. As opposed to the first assignment where this was difficult to do automatically since we had not covered the particular lecture topics, in this assignment we will leverage two easy-to-use Tika Docker files to identify objects present in an image and to generate a textual (human-readable) caption for the image. Both of these Docker Files are available in Apache Tika and they leverage Machine Learning and Deep Learning extraction techniques in particular Google's Tensorflow technology and custom Deep Learning models built in the USC IRDS group. You can see some examples of the Image Captioning and Image Object identification in action below in Figure 2a-c showing 3 automatically generated labels (with only generic training). We will integrate this Tika capability and generate labels and text captions for your scientific papers generated with Grover and LaTeX and DCGAN.

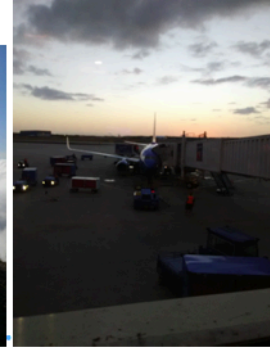


Figure 2: a) a light/orb shown in the daylight; b) an orb present against a mountain background; and c) an orb in a cloudy sky.

a plane flying in the sky over a field

a view of a mountain range with a mountain in the background

an airplane is parked on the tarmac at an airport

Machine Generated Labels for a).

b).

c)

The integration of these two datasets will allow you to apply knowledge gained from the Parsing/Extraction Lecture, the lectures on advanced extraction (including Deep Learning and Metadata), and also topics discussed including Large Scale Content Extraction. In particular, please consider techniques discussed in class to embark on this assignment.

2. Objective

The objective of the assignment is to take the next steps in applying content extraction and feature generation by seeing if you can generate a training set of falsified media. The training set ultimately will be used to see if you can automatically – based on the content – label something as likely falsified or not. In the first assignment you focused on the unintended consequences of Big Data and the associated features to see if you could you explain why the papers in the Bik dataset were manipulated. In this assignment, you will focus on using the content to explain and train a model for scientific paper falsification detection.

You will generate in this assignment a new TSV dataset which includes the following new columns:

1. (Grover) Falsified Media – Indicates whether or not Grover could detect that the media was falsified automatically based on your new dataset.
2. Falsified Media (Manually labeled) – Set this to True for all of the generated papers from this assignment.

And which will include the new rows from your falsified media generation of papers. To do this, you will need to generate the features present in the original Bik dataset for your new papers. My suggestion is to use services like [Fake Name Generator](#) and so on to generate author names, and to sample from the distributions of Institutions present in your first dataset to get the institutional information, and possibly from other features to generate number of students in the lab, etc. You can get creative and even generate these

features for your new 500 papers based on similarity of text, and other distance metrics like we talked about during the Clustering and Deduplication lectures.

Please generate a version 2 TSV file that includes this joined data.

The assignment specific tasks will be specified in the following section.

3. Tasks

1. Generate a copy of your TSV v1 dataset. Call it “v2” or something similar. You will add your new columns for Grover identified falsification after training your model and manual falsification as specified earlier. Additionally you will add the 500 new rows.
2. Write a Python or other script to download the text from all of the referenced papers in the Bik dataset. You can use Selenium, PhantomJS or whatever is needed. You should be able to access the papers institutionally through USC’s connection.
 - a. Store this dataset of PDFs you will use it later in the next steps
3. Run Tika Python (<http://github.com/chrismattmann/tika-python>) on the downloaded PDFs and extract out the text from the PDFs
4. Read the documentation on the Grover GitHub site
 - a. <https://github.com/rowanz/grover>
 - b. Read this paper: <http://papers.nips.cc/paper/9106-defending-against-neural-fake-news.pdf>
 - c. You will need to do a few things with Grover
 - i. Generate your own new Grover model based on the extracted text from the papers you downloaded in steps 2 and 3.
 - ii. Apply that Grover model to generate
 1. The text for 500 new falsified scientific papers
 - iii. Use Grover to test for falsification and to retroactively add that feature as a column to your original 200 papers from Bik
5. Generate your new fake scientific images to go along with the text for your 500 papers
 - a. Write a Python program to extract from the PDFs the associated images, and save them for each paper PDF in a new folder with title <paper name>-images
 - i. [You can find some hints on how to do that here in Python](#)
 - b. Use the generated images as input into the sample DCGAN Python notebook here <https://towardsdatascience.com/face-generator-generating-artificial-faces-with-machine-learning-9e8c3d6c1ead>
6. Generate captions for your new scientific images
 - a. Install Tika Dockers package for Image Captioning and Object Recognition
 - i. git clone <https://github.com/USCDataScience/tika-dockers.git>
 - ii. Read <https://cwiki.apache.org/confluence/display/TIKA/TikaAndVisionDL4J> and test out:

- iii. Read and test out: <https://github.com/apache/tika/pull/189>
- 7. Write a Python program and use sites like FakeNameGenerator and generate the Author Names, and affiliations sampled from your 200 Bik papers for your new falsified papers
- 8. Finally, bring it all together and generate 500 full fake papers
 - a. Check out the code here <https://tex.stackexchange.com/questions/270714/automatic-document-generation-based-on-a-database> and generate a Python program to generate LaTeX .tex files using
 - i. Your falsified images from DCGAN
 - ii. Your generated image captions and any labels from Tika Captioning
 - iii. Your Grover generated text for the papers
 - iv. Your Author Names and affiliations from step 7
 - b. Save the papers in a folder called falsified_media
- 9. Generate the new rows in your TSV v2 with your new papers and their associated features from all prior steps
- 10. **(EXTRA CREDIT)** Integrate Grover's falsification check and your model into Apache Tika
 - a. Look at the Tika Dockers that are used for Tika Image captioning
 - b. Call Grover as a service similar to the way TensorFlow is called
 - c. Submit a pull request to the tika-dockers repository with your Grover integration
- 11. **(EXTRA CREDIT)** Use DCGAN to generate faces for each of the false authors for your 500 new papers
 - a. You will need some sample faces you can find them on Google Images and so forth. Labeled Faces in the Wild is a decent choice, as are some of the datasets used in the DCGAN notebook.
 - b. Submit your new faces via a DropBox link.

4. Assignment Setup

4.1 Group Formation

You should keep the same group from your assignment one. There is no need to send any emails for this step.

5. Report

Write a short 4 page report describing your observations, i.e. what you noticed about the dataset as you completed the tasks. What did the falsified images look like? Were they believable? What about the Grover generated text? Would your associated ancillary features from assignment 1 have been able to discern what was false or not? Are your new papers detectable as false media?

Thinking more broadly, do you have enough information to answer the following:

1. How much do you think media falsification is solvable using ancillary metadata features, or using actual content based techniques? Is one better than the other?
2. What other types of datasets could have been used to generate the falsified papers? Pick at least 2 datasets from distinct MIME types.
3. How well did Grover do on detecting that the original Bik papers were modified? Why do you think it did well, or not?
4. What other sorts of “backstopping” would be required to generate a believable paper trail for these scientific literature?

Also include your thoughts about media falsification for text and images, and Image Captioning/Object identification – what was easy about using it? What wasn’t?

6. Submission Guidelines

This assignment is to be submitted *electronically, by 12pm PT* on the specified due date, via Gmail inf550spring2020@gmail.com. Use the subject line: INF 550: Mattmann: Spring 2020: EXTRACT Homework: Team XX. So if your team was team 15, you would submit an email to inf550spring2020@gmail.com with the subject “INF 550: Mattmann: Spring 2020: EXTRACT Homework: Team 15” (no quotes). **Please note only one submission per team.**

- All source code is expected to be commented, to compile, and to run. You should have at least the identified Python scripts that you used to generate the falsified media, download the scientific papers and generate associated features and text.
- Include your updated dataset TSV. We will provide a Dropbox location for you to upload to.
- Also prepare a readme.txt containing any notes you’d like to submit.
- If you used external libraries other than Tika Python and Grover and DCGAN, you should include those necessary files in your submission, and include in your readme.txt a detailed explanation of how to use these libraries when compiling and executing your program.
- Save your report as a PDF file (TEAM_XX_EXTRACT.pdf) and include it in your submission.
- Compress all of the above into a single zip archive and name it according to the following filename convention:
TEAM_XX_INF550_HW_EXTRACT.zip
 Use only standard zip format. Do **not** use other formats such as zipx, rar, ace, etc.
- If your homework submission exceeds the Gmail's 25MB limit, upload the zip file to Google drive and share it with inf550spring2020@gmail.com.

Important Note:

- Make sure that you have attached the file the when submitting. Failure to do so will be treated as non-submission.
- Successful submission will be indicated in the assignment’s submission history. We advise that you check to verify the timestamp, download and double check your zip file for good measure.
- Again, please note, only **one submission per team**. Designate someone to submit.

6.1 Late Assignment Policy

- -10% if submitted within the first 24 hours
- -15% for each additional 24 hours or part thereof