TEXT ANALYSIS FOR SOCIAL SCIENTISTS AND LEADERS

Faculty: Michael Yeomans (m.yeomans@imperial.ac.uk)

Logistics: April 15-18; 12-3pm @ HBS (Hawes 201; except April 16 in Chao 340).

Office Hours: Immediately after each lecture, there will be an hour-long tutorial session where the coding activity will be walked through in detail. Additional office hours will be held by request.

INTRODUCTION

This course focuses on methods for turning words into numbers. That is, for quantitatively analysing text data, such as newspaper articles, political speeches, product reviews, job descriptions, social media posts, and conversation transcripts. The availability of such data is growing rapidly, and yet extracting valuable information from these corpora is a challenge. Accordingly, many machine learning methods have been developed for text, with even more to come. This course will introduce students to the modern text analysis toolkit, and discuss their application to problems in research.

The course will teach you to be a careful consumer of text analysis in research. The class will cover the strengths and weaknesses of different modelling approaches in the context of a variety of applications across many social science fields, including psychology, economics, policy, and business. While this proliferation of methods provides many opportunities for novel insights, the array of options can be daunting. Furthermore, social scientists are increasingly concerned about the validity, reproducibility, and transparency of their methods. These concerns compound for the text analysis toolkit, which provide near-infinite researcher degrees of freedom, which can have limited or situational validity, which can be difficult for outside researchers to implement themselves, and which can be inscrutable even to the researchers who designed them. Accordingly, this class will also teach a toolkit for evaluating text analysis methods, and their applicability to different research goals.

This course will also teach you to be an effective producer of text analysis research. Students will work on programming problems that implement different methods for quantifying text as part of activities that will require them to work with large-scale text datasets and accompanying meta-data. This will also necessarily reinforce skills for handling and manipulating large datasets computationally. There will be a special focus on creating reproducible, transparent analysis pipelines, so that students learn how to conduct research with technical and non-technical collaborators, and within a community of other researchers and software developers. Accordingly, some statistical and computational background beforehand will be necessary for most students to benefit from the material.

COURSE OBJECTIVES

- Represent text as data in a variety of ways to describe and analyze the content of vast corpora
- Use text to measure the characteristics, contexts intentions and outcomes of the source of that text
- Appreciate how the use of text in social science may differ from that in computer science
- Program text algorithms in R and apply them to example datasets (with support for Python)
- Compare algorithms for text to machine learning approaches for other forms of unstructured data

PREPARATION

Students should be prepared to work with code examples. This means having RStudio installed, along with the tidyverse and quanteda packages, before class. A GitHub account is also recommended. Additionally, there are two recommended articles to read before each class. There are also some optional readings - all will be discussed in the course, but the papers can also be read for more detail. There is no textbook for the course, however I highly recommend the following for more depth:

Jurafsky, D., & Martin, J. H. (2017). Speech and language processing. Vol. 3.
https://web.stanford.edu/~jurafsky/slp3/

Class 1: Humans & Word Counting

Recommended:

Hirschberg, J., & Manning, C. D. (2015). Advances in natural language processing. *Science*, *349*(6245), 261-266.

Poldrack, R. A., Huckins, G., & Varoquaux, G. (2020). Establishment of best practices for evidence for prediction: a review. *JAMA psychiatry*, *77*(5), 534-540.

Optional:

Gentzkow, M., Kelly, B., & Taddy, M. (2019). Text as data. *Journal of Economic Literature*, *57*(3), 535-574.

Lipton, Z. C. (2018). The Mythos of Model Interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue, 16(3),* 31-57

Kang, J. S., Kuznetsova, P., Luca, M., & Choi, Y. (2013, October). Where not to eat? Improving public policy by predicting hygiene inspections using online reviews. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 1443-1448).

Class 2: Interpretability & Categories

Recommended:

Jaidka, K., Giorgi, S., Schwartz, H. A., Kern, M. L., Ungar, L. H., & Eichstaedt, J. C. (2020). Estimating geographic subjective well-being from Twitter: A comparison of dictionary and data-driven language methods. *Proceedings of the National Academy of Sciences*, *117*(19), 10165-10171.

Yeomans, M. (2021). A concrete example of construct construction in natural language. Organizational Behavior and Human Decision Processes, 162, 81-94.

Optional:

Frankel, R., Jennings, J., & Lee, J. (2022). Disclosure sentiment: Machine learning vs. dictionary methods. *Management Science*, *68*(7), 5514-5532.

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77-84.

Grimmer, J., Roberts, M. E., & Stewart, B. M. (2021). Machine Learning for Social Science: An Agnostic Approach. *Annual Review of Political Science*, *24*, 395-419.

Class 3: Similarity - Embeddings & Similarity

Recommended:

Bhatia, S., Richie, R., & Zou, W. (2019). Distributed semantic representations for modeling human judgment. *Current Opinion in Behavioral Sciences*, *29*, 31-36.

Srivastava, S. B., Goldberg, A., Manian, V. G., & Potts, C. (2018). Enculturation trajectories: Language, cultural adaptation, and individual outcomes in organizations. *Management Science*, *64*(3), 1348-1364.

Optional:

Arora, S., Liang, Y., & Ma, T. (2017). A simple but tough-to-beat baseline for sentence embeddings. In *International conference on learning representations*.

Garg, N., Schiebinger, L., Jurafsky, D., & Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, *115*(16), E3635-E3644.

Bender, E. M., Gebru, T., McMillan-Major, A., & Mitchell, S. (2021, March). On the dangers of stochastic parrots: Can language models be too big?. In Proceedings of the 2021 ACM conference on fairness, accountability, and transparency (pp. 610-623).

Class 4: Sentence & Dialogue Structure

Recommended:

Voigt, R., Camp, N. P., Prabhakaran, V., Hamilton, W. L., Hetey, R. C., Griffiths, C. M., ... & Eberhardt, J. L. (2017). Language from police body camera footage shows racial disparities in officer respect. *Proceedings of the National Academy of Sciences*, *114*(25), 6521-6526.

Yeomans, M., Boland, F. K., Collins, H. K., Abi-Esber, N., & Brooks, A. W. (2023). A practical guide to conversation research: How to study what people say to each other. *Advances in Methods and Practices in Psychological Science*, *6*(4).

Optional:

Huang, K., Yeomans, M., Brooks, A. W., Minson, J., & Gino, F. (2017). It doesn't hurt to ask: Question-asking increases liking. *Journal of personality and social psychology*, *113*(3), 430.

Yeomans, M., Minson, J., Collins, H., Chen, F., & Gino, F. (2020). Conversational receptiveness: Improving engagement with opposing views. *Organizational Behavior and Human Decision Processes*, *160*, 131-148.

Stuhler, O. (2022). Who Does What to Whom? Making Text Parsers Work for Sociological Inquiry. *Sociological Methods & Research*, *51*(4), 1580-1633.

CODING ACTIVITIES

This class puts special emphasis on the practical application of the tools you will learn about. Most of the in-class lecture content will be paired with coding activities. You will spend an hour or two for each class completing an activity in a group. This should not be stressful! The activities are all about learning together, and if you make a sincere attempt to follow along, you will do well.

Please be prepared to code! Everyone should have RStudio installed, along with the tidyverse and quanteda R packages on their laptop before the first week. Every computer is different, and there is plenty of documentation online, so you must figure out how to do this on your own before the first class starts. If you cannot bring a laptop to class, please let me know immediately.

CODING LANGUAGE

In truth, any decent analyst will be able to at least modify code in several languages, even if they specialize in one language. However, any class like this can quickly descend into a debate over which programming language is most suitable. These debates are usually unproductive, and often stoked by the least informed. Accordingly, this will not be a discussion topic in this class.

This class will be primarily taught in R. I am an R expert, and frequently write production-ready code (e.g. R packages on CRAN). I can most fluidly teach you the basics of data manipulation, plotting, etc. in R. R is also a better language for this kind of work - the packages are specifically designed for exploring and analysing data. Furthermore, compared to Python, R packages are typically well-vetted, well-documented, and written by academic experts. If you are newer to programming then following my instruction in R is almost surely your best approach.

While Python is also popular, it is more chaotic, with difficult installations, little documentation or version control, and tailored to traditional computer science problems that have little relevance to the content of the class itself. There are also many different kinds of Python code editor software - though ironically, I've found that RStudio is the best available software for writing in Python.

However, I acknowledge most date science teams in industry rely on a mix of both languages, and all of the methods I will teach in class work well enough in both languages. More importantly, some of you are already proficient in Python. I will allow you to submit assignments in Python, if you prefer. But be warned: this is a risky strategy. I have had groups attempt this before and quickly get in over their heads! This is not a python class, it is an NLP class. So if you are not very familiar with the basics of python (including packages like pandas, scikit-learn, nltk, matplotlib) then you will learn much more about NLP if you are following along with R.

For each coding activity, I will briefly walk through R code in class before you work in your groups. We have also prepared a companion guide to the class in Python, mirroring every line of my assignment R code, though I will not teach this in the lecture proper. I will still give feedback on Python assignments. If there is persistent interest in using Python for this class, we will develop a more robust strategy.