TIFFANY FRENCH

THINKFUL FINAL CAPSTONE

PROJECT GOALS & SCOPE

- To analyze job postings for potentially biased language, which may be a cause of very gender-skewed jobs.
- Scrape job postings, analyze with supervised and unsupervised NLP techniques.
- This could be the basis for a "Turn-it-in" Style tool that could take text input, and provide analysis and suggestions for neutralizing the language.

Gender Gaps and AI Skills

Skills where women outnumber men

- Text analytics
- Speech Recognition
- Text Mining
- National Language Processing

Skills where men outnumber women

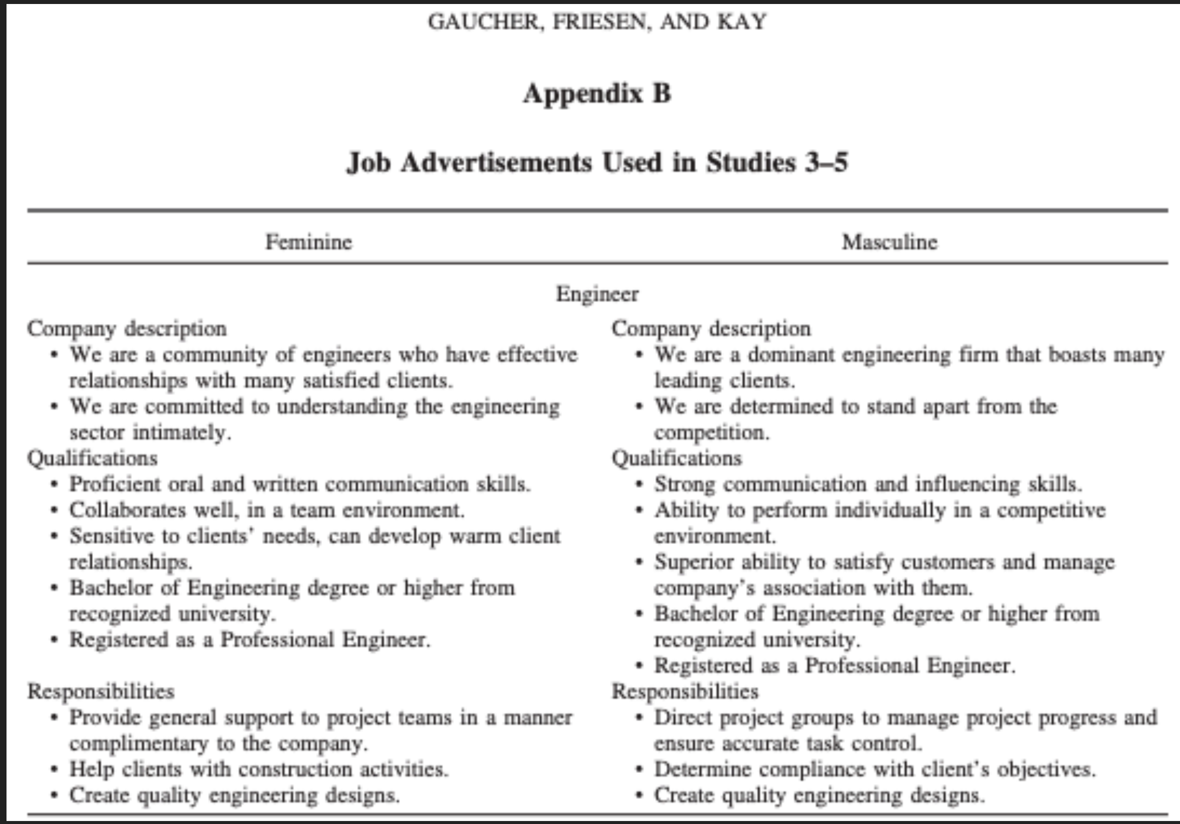
Deep Learning	66%
Apache Spark	74%
Artifiical Neural Networks	66%
Machine Learning	85%
Computer Vision	67%
Pattern Recognition	98%
Neural Networks	70%

Source: LinkedIn data featured in the Global Gender Gap Report 2018, World Economic Forum

EXAMPLES OF GENDERED LANGUAGE

- **Masculine:**
- Active
- Domina*
- Decisive
- Analy*
- Objective
- Self-reliant

- **Feminine:**
- Communal
- Connect*
- Cooperative
- Interdepend*
- Support*
- Together*



GAUCHER, FRIESEN, AND KAY

Appendix B

Job Advertisements Used in Studies 3–5

Feminine	Masculine
Engineer	
Company description • We are a community of engineers who have effective relationships with many satisfied clients. • We are committed to understanding the engineering sector intimately.	Company description • We are a dominant engineering firm that boasts many leading clients. • We are determined to stand apart from the competition.
Qualifications • Proficient oral and written communication skills. • Collaborates well, in a team environment. • Sensitive to clients' needs, can develop warm client relationships. • Bachelor of Engineering degree or higher from recognized university. • Registered as a Professional Engineer.	Qualifications • Strong communication and influencing skills. • Ability to perform individually in a competitive environment. • Superior ability to satisfy customers and manage company's association with them. • Bachelor of Engineering degree or higher from recognized university. • Registered as a Professional Engineer.
Responsibilities • Provide general support to project teams in a manner complimentary to the company. • Help clients with construction activities. • Create quality engineering designs.	Responsibilities • Direct project groups to manage project progress and ensure accurate task control. • Determine compliance with client's objectives. • Create quality engineering designs.

Gaucher, D., Friesen, J., & Kay, A. C. (2011, March 7). Evidence That Gendered Wording in Job Advertisements Exists and Sustains Gender Inequality. Journal of Personality and Social Psychology

DATASET

- Text analysis of job postings from indeed.com to assess for possible gender-biased language
- The job types are:
  - **Female**: Text Analytics, Text Mining, Speech Recognition, NLP,
  - **Male**: Machine Learning, Apache Spark, Pattern Recognition, Neural Networks
- Techniques used:
  - Beautiful Soup
    - I scraped over 7,800 job postings from indeed.com with an iterative scraper that worked through hundreds of pages of job postings.
  - Due to duplicates (I.e. an NLP/Machine Learning posting) the dataset was reduced to 4,300.
  - Additionally, I removed one of the job types (computer vision) to reduce the possibility of class imbalance. Female fields represented 34% of the dataset. The dataset ulitmately consisted of 3700 postings.

NOTEBOOKS AND CODE

```python
starts = list(range(700, 1000, 10))
requests = 0
start = time.time()

baseurl = 'https://www.indeed.com/'

nlp_jobs = []
for start in starts:
    my_urls = ('https://www.indeed.com/jobs?q=%22machine+learning%22&start=' + str(start),)
    my_url = my_urls[0]
    for my_url in my_urls:
        uClient = urlopen(my_url)
        html_input = uClient.read()
        uClient.close()
        soup = BeautifulSoup(html_input, "html.parser")
        cards = soup.findAll('div', {'class':'jobsearch-SerpJobCard'})
        it = iter(cards)
        next(it) # ads
        next(it) # ads
        #next(it)
        for curr in it:
            try:
                link = curr.find('h2').find('a', href=True)['href']
            except:
                pass
            with urlopen(baseurl + link) as uClient:
                list_url = uClient.read()
            listing = BeautifulSoup(list_url, 'html.parser')
            title = listing.find('h3',
                            {'class': 'icl-u-xs-mb--xs icl-u-xs-mt--none jobsearch-JobInfoHeader-t
itle'})
            if not title:
                    print('missing content @ ' + baseurl + link)
            body = listing.find('div',
                            {'class': 'jobsearch-JobComponent-description icl-u-xs-mt--md'}
                            )
            if not body:
                print('missing content @ ' + baseurl + link)
            requests += 1
            sleep(randint(5,7))
            end = time.time()
            #print("Done in", end, "seconds")
            print('Request: {}; Frequency: {} requests/s'.format(requests, requests/end))
            clear_output(wait = True)
            with db_session:
                Job(title=str(title),
                job_description=str(body),
                job_class='Machine Learning')
```

```
GET CONNECTION FROM THE LOCAL POOL
BEGIN IMMEDIATE TRANSACTION
INSERT INTO "Job" ("title", "job_description", "job_class") VALUES (?, ?, ?)
```

8 DIFFERENT JOB TITLES

BEAUTIFUL SOUP SCRAPER

21 lines (14 sloc) | 281 Bytes

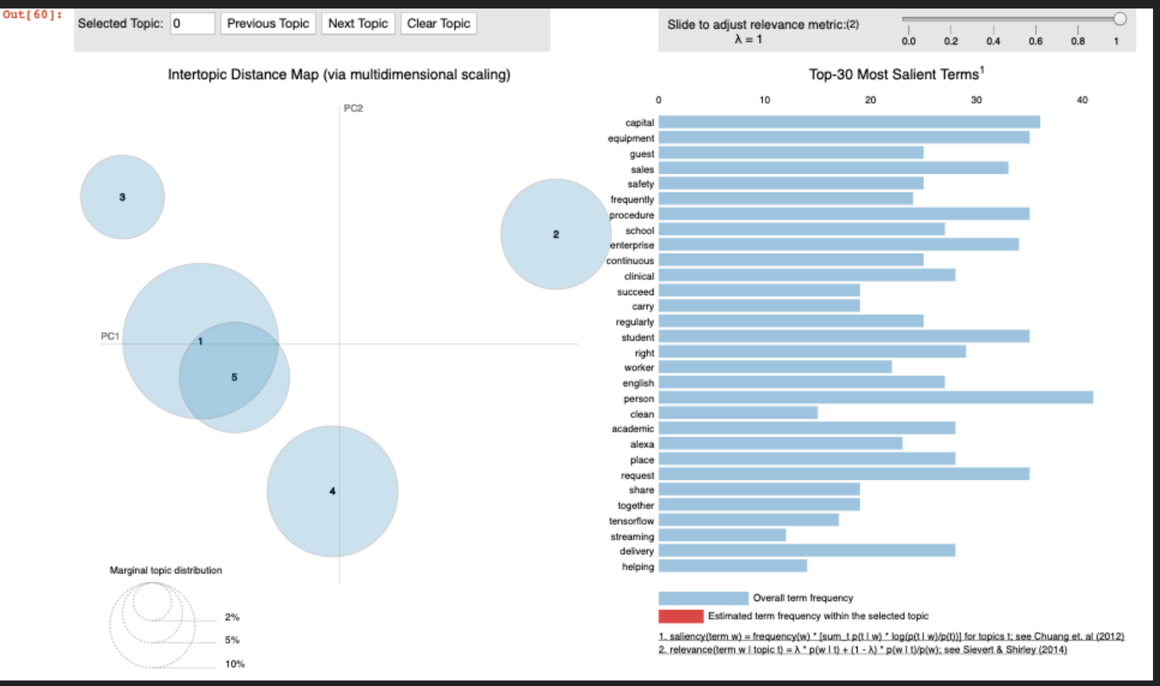
```python

# coding: utf-8

# In[1]:


from pony_orm_model import *
import csv

@db_session
def add_job(title, job_description):
    d = Job()
    d.title = title
    commit()
    d.job_description = job_description
    commit()
    d.job_class = job_class
    commit()

populate_database()
```

DATABASE MANAGEMENT

ORM AND SQLITE STORAGE

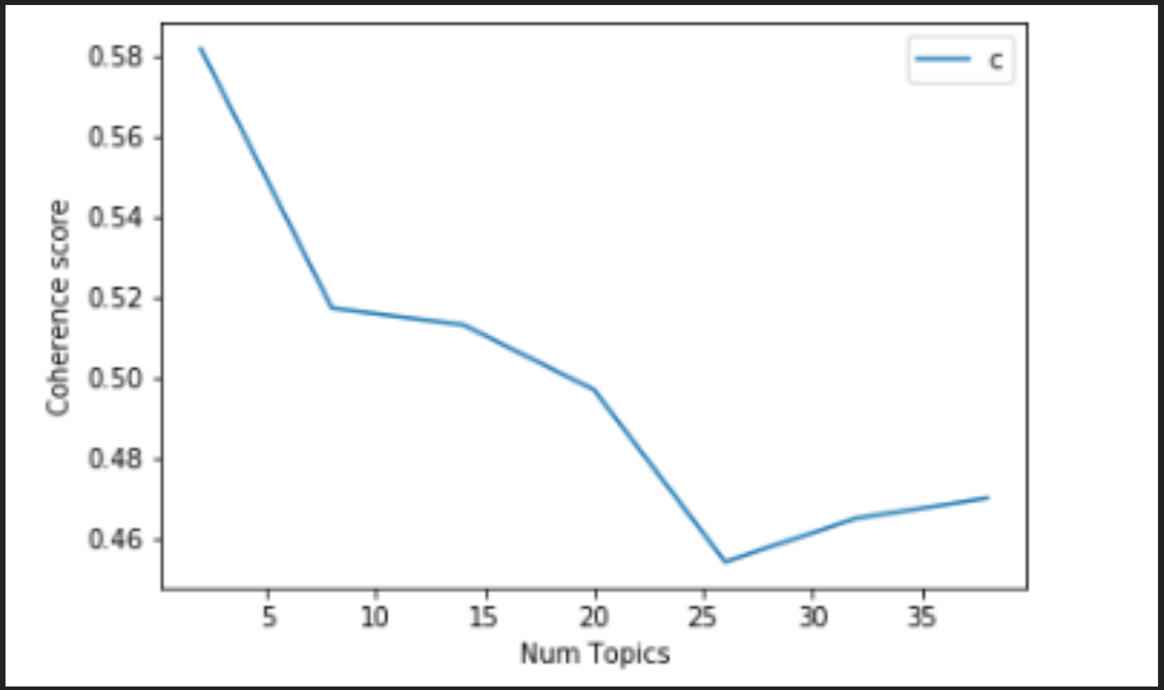
GENSIM AND PYLDAVIZ

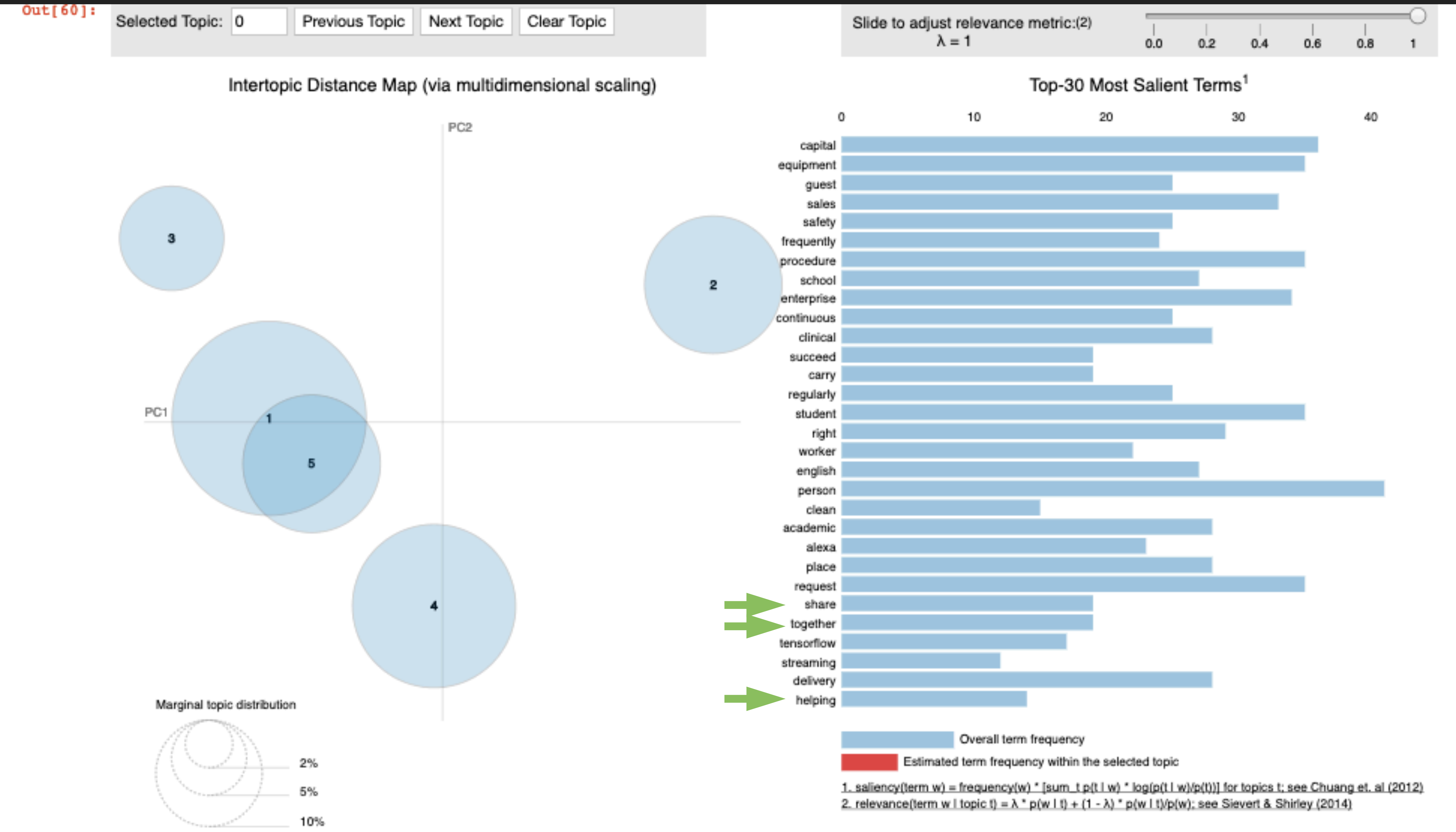
UNSUPERVISED APPROACH



EVALUATING COHERENCE

- After evaluating the coherence of the LDA, it would be unwise to go above about 10 topics since there is a plateau and drop-off at that point.




Out[60]:
Selected Topic: 0
Previous Topic
Next Topic
Clear Topic
Slide to adjust relevance metric:(2)
λ = 1
0.0
0.2
0.4
0.6
0.8
1
Intertopic Distance Map (via multidimensional scaling)
PC2
PC1
1
2
3
4
5
Marginal topic distribution
2%
5%
10%
Top-30 Most Salient Terms1
0
10
20
30
40
capital
equipment
guest
sales
safety
frequently
procedure
school
enterprise
continuous
clinical
succeed
carry
regularly
student
right
worker
english
person
clean
academic
alexa
place
request
share
together
tensorflow
streaming
delivery
helping
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

```
Cluster: 1
                                   job_description  MiniBatchLabels
job_class
Apache Spark                       19               19
Machine Learning                   183              183
Natural Language Processing        86               86
Neural Networks                    166              166
Pattern Recognition                45               45
Speech Recognition                 45               45
Text Analytics                     8                8
Text Mining                        4                4

Cluster: 2
                                   job_description  MiniBatchLabels
job_class
Apache Spark                       157              157
Machine Learning                   82               82
Natural Language Processing        103              103
Neural Networks                    188              188
Pattern Recognition                73               73
Speech Recognition                 8                8
Text Analytics                     75               75
Text Mining                        118              118

Cluster: 3
                                   job_description  MiniBatchLabels
job_class
Apache Spark                       3                3
Machine Learning                   48               48
Natural Language Processing        57               57
Neural Networks                    56               56
Pattern Recognition                24               24
Speech Recognition                 27               27
Text Analytics                     2                2
Text Mining                        7                7

Cluster: 4
                                   job_description  MiniBatchLabels
job_class
Apache Spark                       358              358
Machine Learning                   14               14
```

TF-IDF AND BOW

SUPERVISED APPROACH

BAG OF WORDS

- Models used:
  - K-Means: Unfortunately, did not perform well.
  - LSA with Bow: Much more helpful.
  - LSA with Bigrams: Even better.

TF-IDF

- Models used:
  - K-Means Mini-batch: Helpful in generating understanding “behind the scenes” batches 1 & 2 appear very balanced.
  - Gradient Boosting Classifier performed best here, and was able to match the job to its type (1 of the 8) with a score of 87.

The distance between the Deep Learning Engineer (66% male) and the NLP Position (mostly women).

In [82]:
```
scipy.spatial.distance.pdist([one_topics, two_topics])
```

Out[82]:
```
array([0.25583754])
```

The distance between the Deep Learning and Pattern Recogition (98% male).

In [83]:
```
scipy.spatial.distance.pdist([one_topics, three_topics])
```

Out[83]:
```
array([0.41192855])
```

The distance between NLP and Pattern Recognition.

In [84]:
```
scipy.spatial.distance.pdist([two_topics, three_topics])
```

Out[84]:
```
array([0.3271281])
```

This is very interesting. What we can see here is that a job description from a relatively balanced field has the greatest distance from the very-male dominated field. While the NLP and Pattern Recognition aren't as far apart as the Deep Learning and Pattern Recognition, I think it is still of note. What we see here is that a job description from a balanced field is different from one that is not balanced at all. This makes a great case that even job descriptions that might be skewed toward one gender or another are more similar than we realize as well.

In the end, I think this makes a great case for gender-balanced job descriptions that can attract the best and brightest from any gender (I'd, of course, like to broaden this to people who identify as gender non-binary).

I think something like this projection could help create a program that analyzes a job description, and gives feedback to the client about how balanced it might be. It would take web development, and extensive model training, but I think something like that could be valuable.

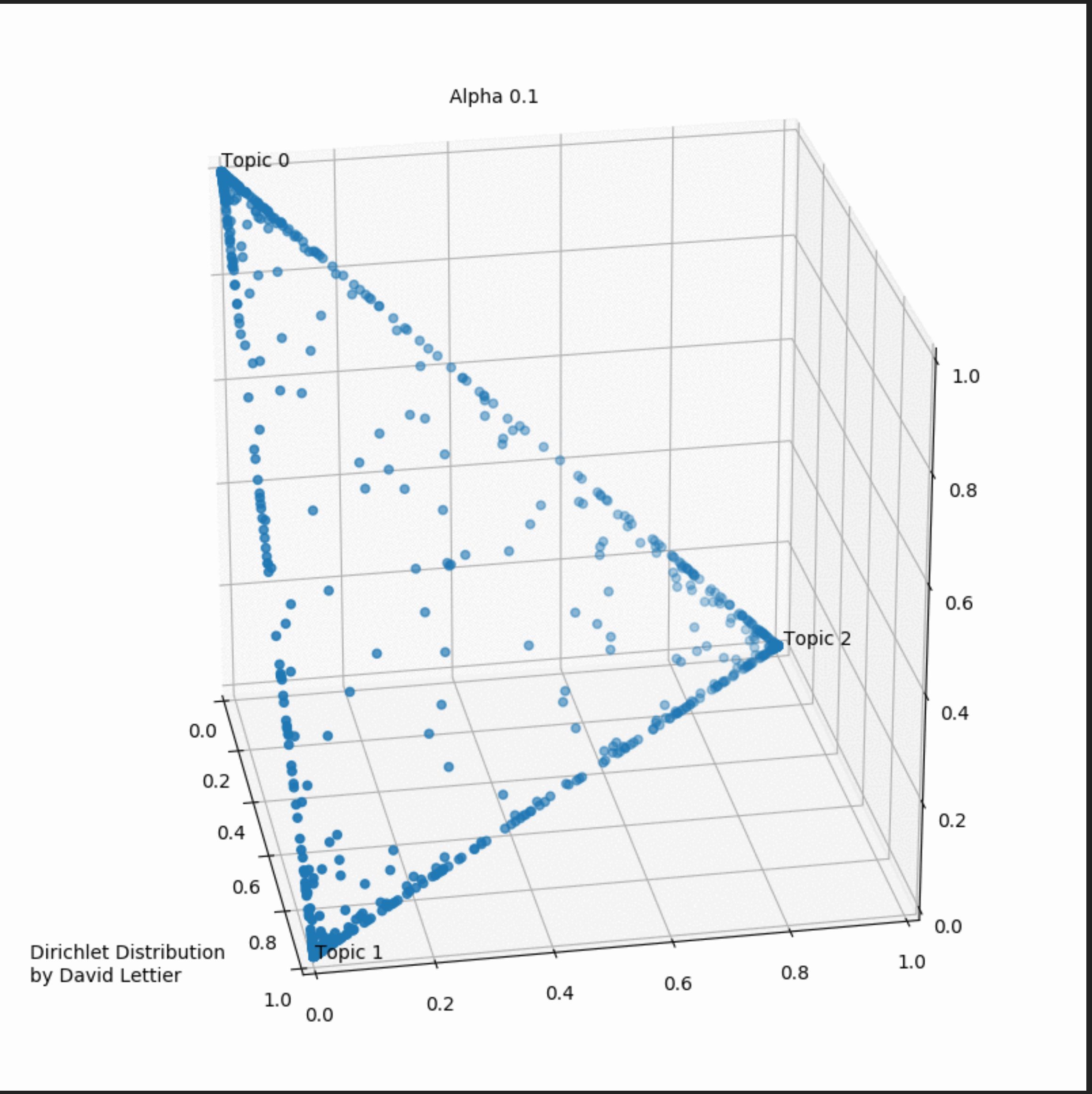
This projection model was created with extensive help from my mentor, Philip Robinson. So I'd like to give him credit here where it's due!

LDA PROJECTION

FURTHER UNSUPERVISED APPROACH

LDA PROJECTION

- With this, we project three new job descriptions into the LDA space, and measure the distance between the new postings, based on the understanding of the Latent Dirichlet Allocation.



https://medium.com/@lettier/how-does-lda-work-ill-explain-using-emoji-108abf40fa7d

OUTCOMES AND FURTHER RESEARCH

- In short, the project did not produce some of the definitive results I was looking for. However, I still think it had some valuable outcomes
  - LDA Projection
  - Modeling and classification
  - PyLDAviz
- A larger corpus could help promote understanding, so to improve the project, I would increase the corpus size and try some of the same approaches.

OUTCOMES AND FURTHER RESEARCH

- Something like this would be the ideal outcome from this project. However, I think just creating awareness with the project helps us not to skew a posting either way, but perhaps promote an equitable work environment that brings together all of the best talent available.