Fake Plastic Grass: Detecting Astroturfing by Plastic Waste Non-Profits

Daphne Walford
The University of British Columbia
December 14th, 2020

WRDS 150B Research and Writing in the Disciplines
712 / Mi-Young Kim

Abstract

In response to ecologically motivated pushback against plastic pollution, industry stakeholders use non-profit organisations (NPOs) to project an illusion of public support for their agenda. These NPOs spread misleading information to shift the onus of waste management away from manufacturers. Since these NPOs deceptively imitate grassroots movements, their activity is known as astroturfing. This study analysed the website copy of 9 grassroots and 5 astroturfing plastic pollution NPOs to compare thematic similarities and differences. Astroturfing organisations were found to use more prosocial language, refer less to concrete features of the outdoors, and discuss plastic pollution less explicitly than grassroots organisations. This provides a basis for future studies on the differences in messaging and mission between these organisations to help consumers make informed choices about the initiatives they support. Additionally, it provides guidance for grassroots organisations intending to effectively counteract the messaging of astroturfing organisations.

Key words: greenwashing, astroturfing, plastic, pollution

Introduction

Plastic waste is a growing problem (Prata et al., 2020). 90 countries have implemented restrictions on single-use plastics, which compose 46% of waste (Patrício Silva et al., 2020). However, industry lobbying has rolled back many of these regulations (Patrício Silva et al., 2020). “Grassroots” movements refer to collective action at the local level: large numbers of stakeholders who are embedded in the geographic and cultural contexts most impacted by the changes in question (Cho et al., 2011; Chowdhury et al., 2018). “Astroturfing” is a term coined by Lyon & Maxwell (2004) to describe campaigns and non-profit organisations (NPOs) that appear to be run by mobilised community members but are in fact orchestrated by industry interests and public relations firms. These advance industry interests while projecting grassroots credibility (Lyon & Maxwell, 2004). For example, Save the Plastic Bag Campaign (SPBC) is an NPO formed by a coalition of plastic bag manufacturers in response to plastic bag bans (Romer & Foley, 2012). SPBC has leveraged environmental law to relentlessly sue districts proposing similar bans and spread misinformation under the guise of correcting exaggerated claims of plastic bags’ negative ecological impact (Romer & Foley, 2012). Astroturfing therefore helps private interests maintain their market foothold in the face of social pressure for change. Although operated by corporate interests, many astroturfing organisations still fundraise and use volunteer resources, often shielding their intentions with deceptive language (Cho et al., 2011). Consumers attempting to positively engage in environmental action can inadvertently support the interests of the plastic industry if they participate in these activities without thoroughly investigating organisations beforehand (Cho et al., 2011).

Literature Review

The regression in anti-plastic regulations discussed above is closely linked to the COVID-19 pandemic (Prata et al., 2020). A letter from the Plastics Industry Association framed reusable bags as a safety risk, citing three studies whose generalisability to coronavirus was firmly refuted by Hale & Song (2020). Similarly, Prata et al. (2020) describe a lack of peer-reviewed studies demonstrating hygienic advantages of single-use plastic bags, and a tendency for the Canadian Plastics Industry Association to cite internal research in support of these claims. Concurrent to these policy transitions, changes in public opinion of these products have also been observed. The preliminary results of a Canadian survey found that support for a single-use plastic ban dropped from 90% in 2019 to 78% in 2020 (Kitz et al., 2020). It is thus apparent that attitudes surrounding single-use plastics are undergoing a tectonic shift, but by what mechanisms are organisations generating the social influence necessary to motivate this change?

Significant attention has been directed toward corporate “greenwashing,” token environmental efforts intended to improve public image rather than cause real change (Dixon, 2020). Laufer (2003) describes three main elements of greenwashing: confusion, fronting, and posturing. These deceptive strategies are directed both internally (within an organization or sector) and externally (toward stakeholders and potential regulators) (Laufer, 2003). The combination of astroturfing and greenwashing has been described as an “exemplary worst case” of misalignment between corporate social responsibility and corporate political activity (Schultz & Seele, 2020). Cho et al. (2011) conducted an experiment to assess the effectiveness of astroturfing in influencing public opinion. By disseminating misinformation on climate change via websites for false social movements and analysing its effectiveness in changing opinion, they demonstrated that astroturfing is effective in increasing uncertainty (Cho et al., 2011). Although this study thoroughly describes the rhetorical intentions and effects of climate astroturfing, it does not textually analyse the rhetoric itself for identifying details (Cho et al., 2011). This form of analysis, however, has been performed on mass communications astroturfing, such as “puppet” accounts on social media or secretly sponsored letters to news publications (Chen et al., 2017; Mahbub et al., 2019). One of the most common strategies for comparing word choice and themes between multiple documents is latent Dirichlet allocation (LDA), a form of topic modelling (Eickhoff & Wieneke, 2018). As retrieved by LDA, “topics” are probability distributions of all the words present within the corpus (a term referring to the totality of documents analysed), representing which words are most likely to occur together as part of an overarching theme (Eickhoff & Wieneke, 2018). This statistical method has been applied to sentiment analysis for detecting astroturfing in product reviews (Alallaq et al., 2018).

Previous research has therefore clearly demonstrated the powerful effects of astroturfing websites in deceiving the public and swaying opinion. It has also demonstrated practical approaches for detecting characteristic features of astroturfing content in the form of mass communications. However, the website rhetoric of astroturfing movements remains to be textually analysed and compared with that of grassroots movements. Specifically, it is not clear how web resources created for the general public differ between grassroots and astroturfing non-profit organisations (NPOs). This study aims to analyse website copy by grassroots and astroturfing plastic pollution NPOs by assessing readability and comparing themes in vocabulary.

Methodology

The first step to this analysis was website sourcing. The keywords “plastic pollution organisations” were entered into Google Canada. To focus on high-impact organisations, only the first five pages of search results were examined. In order to qualify, organisations had to feature plastic waste reduction as their primary target, and their websites had to be in English with an About, Mission, and/or Vision page. These pages were chosen since they communicate an organisation's strategic goals to a general audience. In addition, organisations had to declare their primary funding sources so that they could be labelled as grassroots (mostly individual donations) or astroturfing (mostly large corporations). From search results, 26 organisations met these criteria. Four were excluded because of limited English-language content and eight because of for-profit business models. A total of 14 organisations (9 grassroots, 5 astroturf) met eligibility criteria. These sources are listed in Table 1.

All text content on the About, Mission, and/or Vision pages of a given site was scraped into a file. Two Python scripts were written to process these files (see Appendices I and II). The first script runs files through three readability analyses (ARI, Flesch-Kincaid, and Smog) encoded within the *py-readability-metrics* library (DiMascio, 2018). These standardised formulae calculate the approximate grade level required for understanding a given text sample (DiMascio, 2018). The script then analyses the files using Python's Natural Language Toolkit (Bird et al., 2016). Because this study aims to identify themes from vocabulary, it uses a bag-of-words approach, where word order is unimportant: as discussed by McFarland et al. (2013), this is a common and effective strategy for identifying and comparing themes across documents.

Once text had been collected from these sources, the data were cleaned: each file was lemmatized, a process which simplifies words to their stems (i.e. “made” to “make”), punctuation was stripped, and “stop words” (such as prepositions) were removed (Bird et al., 2016). This ensured that only the most meaningful words underwent analysis, and that variations of one word were aggregated to reflect all usages (Bird et al., 2016). Basic metrics were then analysed (see Appendix I for the Python script written for this step). Internal comparisons between cleaned texts were performed within the two categories (grassroots and astroturf) to find words that appeared frequently across multiple texts, and the two resulting lists were compared to identify common elements. These results were compiled into a Venn diagram. The words were also qualitatively coded into subcategories to derive overarching themes.

The second Python script performed latent Dirichlet allocation. Two separate models were written, using the *gensim* and *scikit-learn* libraries respectively. The *scikit-learn* model (Appendix II) was selected due to lower overlap between topics. Once two topics had been extracted from the corpus, each document underwent individual analysis to determine which topic fit it best. The script was looped several thousand times, deriving slightly different topics each iteration, until at least 80% of grassroots organisations were assigned to one topic and at least 80% of astroturfing organisations were assigned to the other topic.

Table 1: Data sources (1-9 are grassroots, 10-14 are astroturfing)

	Name	***URL***
1	Plastic Pollution Coalition	https://www.plasticpollutioncoalition.org/
2	5 Gyres Institute	https://www.5gyres.org/
3	Surfrider Foundation	https://www.surfrider.org/
4	Greenpeace	https://www.greenpeace.org/international/campaign/toolkit-plastic-free-future/learn-about-plastic-pollution/
5	Bahamas Plastic Movement	https://www.bahamasplasticmovement.org/
6	Save Our Shores	https://saveourshores.org/
7	Californians Against Waste	https://www.cawrecycles.org/about-us
8	Bye Bye Plastic Bags	http://www.byebyeplasticbags.org/about/
9	Trash Hero	https://trashhero.org/how-we-are-funded/
10	Oceana	https://oceana.org/corporate_partnerships/corporate-partners
11	Ocean Conservancy	https://oceanconservancy.org/about/
12	Alliance to End Plastic Waste	https://endplasticwaste.org/about/
13	Ellen MacArthur Foundation	https://www.ellenmacarthurfoundation.org/
14	Lonely Whale	https://www.lonelywhale.org/about

Findings

Astroturfing material used more prosocial language and favoured abstract terms to describe strategies for addressing plastic pollution. Grassroots material used more concrete language to describe plastic pollution and strategies for addressing it.

As seen in Figure 1, the astroturfing material was slightly more readable by all three measures, although this was not statistically significant. Figure 2 displays the most common words in grassroots, astroturfing, and all documents; Table 2 organises these words into categories that were qualitatively selected after the lists were generated: “problem,” “environment,” “prosocial,” “abstract strategic,” and “concrete strategic.” The words shared between the two categories reflected the overarching focus of all organisations on plastic pollution in the marine environment.

Astroturfing material exhibited an overarching theme of prosocial language that emphasised collaboration and aid. However, specific strategies (like “policy”) were absent. In addition, the most common words did not include explicit negative references to plastic, such as “pollution” or “waste.” However, these were common in grassroots content. Grassroots content was also more likely to reference specific elements of the environment, such as “coastal,” and to discuss pollution as a problem with terms like “single-use” (hyphen removed) and “product.” These disparities reflect the pro-plastic astroturfing agenda, which is directly opposed to a negative view of plastic and to actionable strategies that might decrease overall plastic usage.

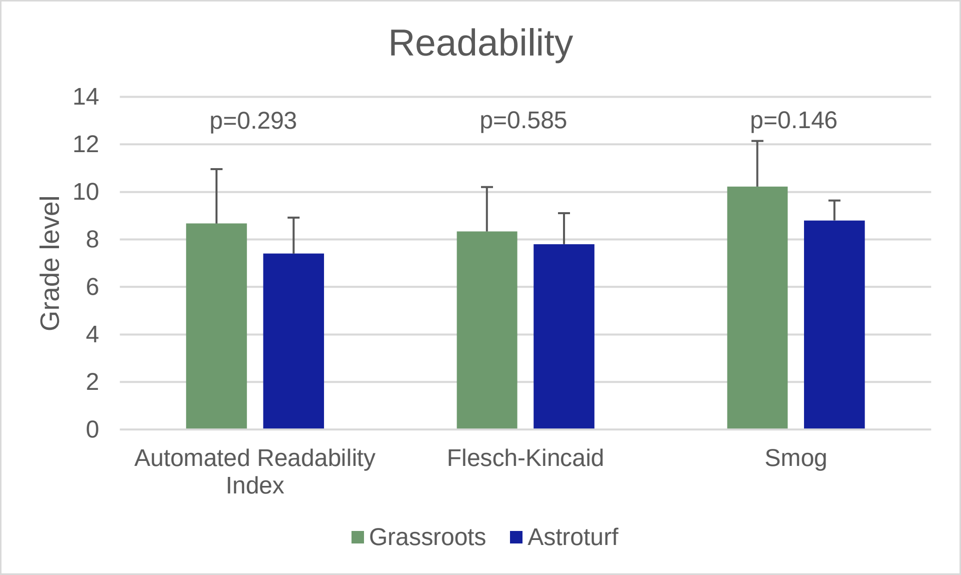

Figure 1: Average readability of each category

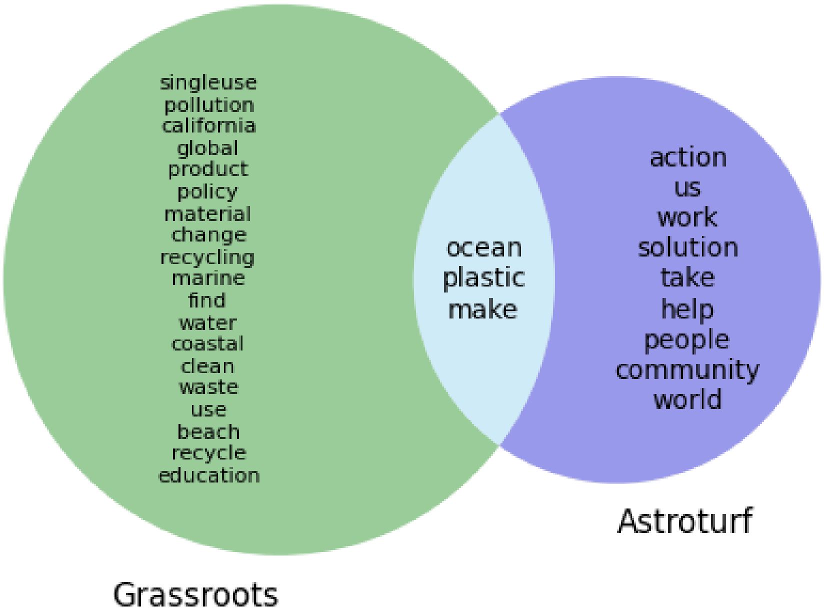

Figure 2: Venn diagram of most common words in each NPO category

Table 2: Words in Figure 2, above, separated into coded categories (green = grassroots, cyan = shared, mauve = astroturfing)

problem	***environment***	***prosocial***	***abstract strategic***	***concrete strategic***
plastic	ocean	us	make	policy
singleuse	marine	work	change	education
use	water	people	goal	recycling
pollution	coastal	community	find	recycle
product	beach	help	take	clean
material	california		action	
waste	world		solution	

Table 3: Topics derived via latent Dirichlet allocation, represented by the 10 most likely words to co-occur

	Word0	*Word1*	*Word2*	*Word3*	*Word4*	*Word5*	*Word6*	*Word7*	*Word8*	*Word9*
Topic0	ocean	plastic	trash	waste	solution	people	work	community	world	action
Topic1	plastic	pollution	california	water	waste	circular	economy	beach	use	ocean

Conclusion

Astroturfing organisations were more likely to employ prosocial and relational language that emphasised action and mobilization. This reflects effective co-opting of the community-oriented ethos associated with collective movements, as described by Barberá-Tomás et al. (2019). Astroturfing organisations were less likely to use actionable language and to explicitly describe pollution. Members of the public who are interested in engaging with environmental NPOs should read their About, Mission, and Vision pages critically, searching for explicit references to plastic as a polluting substance and for concrete goals.

This analysis is primarily limited by its small sample size and the decontextualized nature of bag-of-words analysis, which is not sensitive to meanings derived from word order, such as negation. Future research could analyse all text on these websites as well as resources created for the general public (such as PDFs) and content created by astroturfing coalitions that are not NPOs. In addition, optical character recognition (OCR) should be used to gather text from images and figures on these sites. (1905 words)

References

Alallaq, N., Al-khiza’ay, M., Dohan, M. I., & Han, X. (2018). Sentiment analysis to enhance detection of latent astroturfing groups in online social networks. *Communications in Computer and Information Science*. https://doi.org/10.1007/978-981-13-2907-4_7

Barberá-Tomás, D., Castelĺo, I., De Bakker, F. G. A., & Zietsma, C. (2019). Energizing through visuals: How social entrepreneurs use emotion-symbolic work for social change. *Academy of Management Journal*, *62*(6), 1789–1817. https://doi.org/10.5465/amj.2017.1488

Bird, S., Klein, E., & Loper, E. (2016). *NLTK Book*. O’Reilly.

Chen, T., Alallaq, N. H., Niu, W., Wang, Y., Bai, X., Liu, J., Xiang, Y., Wu, T., & Liu, J. (2017). A hidden astroturfing detection approach base on emotion analysis. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-319-63558-3_5

Cho, C. H., Martens, M. L., Kim, H., & Rodrigue, M. (2011). Astroturfing global warming: it isn’t always greener on the other side of the fence. *Journal of Business Ethics*, *104*(4), 571–587. https://doi.org/10.1007/s10551-011-0950-6

Chowdhury, R., Kourula, A., & Siltaoja, M. (2018). *Power of Paradox : Grassroots Organizations ’ Legitimacy Strategies Over Time*. 1–34. https://doi.org/10.1177/0007650318816954

DiMascio, C. (2018). *py-readability-metrics*. https://github.com/cdimascio/py-readability-metrics

Dixon, L. (2020). Autonowashing: The Greenwashing of Vehicle Automation. *Transportation*

Research Interdisciplinary Perspectives, *5*, 100113. https://doi.org/10.1016/j.trip.2020.100113

Eickhoff, M., & Wieneke, R. (2018). Understanding Topic Models in Context: A Mixed-Methods Approach to the Meaningful Analysis of Large Document Collections. *Proceedings of the 51st Hawaii International Conference on System Sciences*, 903–912. https://doi.org/10.24251/hicss.2018.113

Hale, R. C., & Song, B. (2020). Single-Use Plastics and COVID-19: Scientific Evidence and Environmental Regulations. *Environmental Science and Technology*, *54*, 7034–7036. https://doi.org/10.1021/acs.est.0c02269

Kitz, R., Charlebois, S., Walker, T., & Music, J. (2020). *Plastic food packaging: before and after COVID*. http://www.hospitalbarrosluco.cl/wp-content/uploads/2020/05/Cuenta-Publica-2019-HBLT.pdf

Laufer, W. S. (2003). Social Accountability and Corporate Greenwashing. *Journal of Business Ethics*, *43*(3), 253–261. https://doi.org/10.1023/A:1022962719299

Lyon, T. P., & Maxwell, J. W. (2004). Astroturf: Interest group lobbying and corporate strategy. *Journal of Economics and Management Strategy*, *13*(4), 561–597. https://doi.org/10.1111/j.1430-9134.2004.00023.x

Mahbub, S., Pardede, E., Kayes, A. S. M., & Rahayu, W. (2019). Controlling astroturfing on the internet: A survey on detection techniques and research challenges. *International Journal of Web and Grid Services*, *15*(2), 139–158. https://doi.org/10.1504/IJWGS.2019.099561

McFarland, D. A., Ramage, D., Chuang, J., Heer, J., Manning, C. D., & Jurafsky, D. (2013).

Differentiating language usage through topic models. *Poetics*, *41*(6), 607–625. https://doi.org/10.1016/j.poetic.2013.06.004

Prata, J. C., Silva, A. L. P., Walker, T. R., Duarte, A. C., & Rocha-Santos, T. (2020). COVID-19 pandemic repercussions on the use and management of plastics. *Environmental Science and Technology*, *54*(13), 7760–7765. https://doi.org/10.1021/acs.est.0c02178

Romer, J. R., & Foley, S. (2012). A Wolf in Sheep' s Clothing: The Plastics Industry' s "Public Interest" Role in Legislation and Litigation of Plastic Bag Laws in California. *Golden Gate University Environmental Law Journal*, *8*(2), 5–24. https://static1.squarespace.com/static/59bd5150e45a7caf6bee56f8/t/59bd52ab7e2a5fb4e246de6c/1514156744153/wolf-in-sheeps-clothing.pdf

Schultz, M. D., & Seele, P. (2020). Handbook of Business Legitimacy. *Handbook of Business Legitimacy*, 655–669. https://doi.org/10.1007/978-3-030-14622-1

Appendix I

```python
import string, collections, nltk, re, statistics, glob
from nltk.probability import FreqDist
from nltk import sent_tokenize, word_tokenize, pos_tag
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import RegexpTokenizer
from nltk.corpus import stopwords
from readability import Readability
import matplotlib
import matplotlib.pyplot as plt
import gensim
from gensim.utils import simple_preprocess
import numpy as np

#paragraph -> array of sentences with punctuation removed
def extract_sents(text):
    #list of sentences
    no_punct = []
    filtered_sent = []
    filtered_sents = []
    text = text.replace('\n','. ')
    sents = sent_tokenize(text)
    for sent in sents:
        no_punct = sent.translate(str.maketrans('','',string.punctuation))
        filtered_sents.append(no_punct)
    return filtered_sents

def extract_words(sents):
    filtered_words=[]
    stop_words=set(stopwords.words("english"))
    regex = re.compile('[^a-zA-Z]')
    lem = WordNetLemmatizer()
    for sent in sents:
        for word, tag in pos_tag(word_tokenize(sent)):
            sent.replace(word,word.strip())
            word = regex.sub('',word)
            if word:
                wntag = tag[0].lower()
                wntag = wntag if wntag in ['a', 'r', 'n', 'v'] else None
                lemma = lem.lemmatize(word, wntag).lower() if wntag else word.lower()
                if lemma not in stop_words:
                    filtered_words.append(lemma)
    return filtered_words

def extract_counts(sents):
    sentence = []
    n_words = 0
    for sent in sents:
        n_words += len(word_tokenize(sent))
    n_chars = sum(len(i) for i in sents)
    n_sents = len(sents)
    return n_chars, n_words, n_sents

def automatic_readability_index(n_chars, n_words, n_sents):
    return 4.71*(n_chars/n_words)+0.5*(n_words/n_sents)-21.43

#create frequency distribution of words
def freqDistWrds(abouts):
    freqList = []
    freqWrds = []
    for about in abouts:
        fdist = FreqDist(about)
        freqList = fdist.most_common(20)
        freqWrds.append(i[0] for i in freqList)
    counter = collections.Counter(x for xs in freqWrds for x in set(xs))
    return counter.most_common()

#append each text file to string list
list_of_files = glob.glob('./*.txt')
sentence_holder = []
preserved = []
abouts = []
aris = []
textAbouts = []

numbers = re.compile(r'(\d+)')
def numericalSort(value):
    parts = numbers.split(value)
    parts[1::2] = map(int, parts[1::2])
    return parts

for infile in sorted(glob.glob('*.txt'), key=numericalSort):
    print('filename: ',infile)
    f = open(infile,encoding="utf8")
    sentence_holder = extract_sents(f.read())
    tokens = word_tokenize(f.read())
    f.close()
    textAbouts.append(nltk.Text(tokens))
    preserved.append('. '.join(sentence_holder))
    abouts.append(extract_words(sentence_holder))
    n_chars, n_words, n_sents = extract_counts(sentence_holder)
    aris.append(automatic_readability_index(n_chars, n_words, n_sents))

grassroots = abouts[0:8]
gr_freq = []
gr_words = []
astroturf = abouts[9:]
at_freq = []
at_words= []

text_gr = textAbouts[0:8]
text_at = textAbouts[9:]

print('Grassroots: ')
for name, amount in freqDistWrds(grassroots):
    if amount > 1:
        print('\'%s\' is in %s %%' % (name, amount/len(grassroots)*100))
        gr_freq.extend([name,amount/len(grassroots)*100])
        gr_words.append(name)
print('Astroturf: ')
for name, amount in freqDistWrds(astroturf):
    if amount > 1:
        print('\'%s\' is in %s %%' % (name, amount/len(astroturf)*100))
        at_freq.extend([name,amount/len(astroturf)*100])
        at_words.append(name)

arisData = [sum(aris[0:8])/len(aris[0:8]),sum(aris[9:])/len(aris[9:])]

print('Flesch-Kincaid and SMOG')
ari = []
```

```
fk = []
smog = []
for text in preserved:
    r = Readability(text)
    ari_obj = r.ari()
    ari.append(ari_obj.grade_levels)
    fk_obj = r.flesch_kincaid()
    fk.append(fk_obj.grade_level)
    sm_obj = r.smog()
    smog.append(sm_obj.grade_level)

print('ARIs: ',ari)
print('FK ARIs: ',fk)
print('smog ARIs: ',smog)

from matplotlib_venn import venn2, venn2_circles
from matplotlib import pyplot as plt# set up the figure

gr = set(gr_words)
at = set(at_words)

v = venn2([gr,at], ('Grassroots','Astroturf'))

v.get_label_by_id('10').set_text('\n'.join(map(str,gr-at)))
v.get_label_by_id('01').set_text('\n'.join(map(str,at-gr)))
v.get_label_by_id('11').set_text('\n'.join(map(str,gr&at)))# add circle outlines

v.get_patch_by_id('10').set_color('g')
v.get_patch_by_id('10').set_edgecolor('none')
v.get_patch_by_id('10').set_alpha(0.4)

v.get_patch_by_id('01').set_color('mediumblue')
v.get_patch_by_id('01').set_edgecolor('none')
v.get_patch_by_id('01').set_alpha(0.4)

v.get_patch_by_id('11').set_color('skyblue')
v.get_patch_by_id('11').set_edgecolor('none')
v.get_patch_by_id('11').set_alpha(0.4)

label = v.get_label_by_id('10')
label.set_fontsize(8)

plt.axis('on')
plt.show()
```

Appendix II

```
import re, nltk, spacy, string
from sklearn.decomposition import LatentDirichletAllocation
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from pprint import pprint

import glob
import pandas as pd
import numpy as np

import pyLDAvis
import pyLDAvis.sklearn
import matplotlib.pyplot as plt

from plotly.offline import plot
import plotly.graph_objects as go
import plotly.express as px

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

all_files = glob.glob('./*.txt')
i = 0
df = pd.DataFrame(columns = ['text','index'])

numbers = re.compile(r'(\d+)')
def numericalSort(value):
    parts = numbers.split(value)
    parts[1::2] = map(int, parts[1::2])
    return parts

for infile in sorted(glob.glob('*.txt'), key=numericalSort):
    print('filename: ',infile)
    f = open(infile,encoding="utf8")
    df.loc[i] = [f.read(),i]
    f.close()
    i += 1

def clean_text(text):
    '''''Make text lowercase, remove text in square brackets, remove punctuation and remove words containing numbers.'''
    text = text.lower()
    text = re.sub(r'\[.*?\]', '', text)
    text = re.sub(r'[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub(r'\w*\d\w*', '', text)
    return text

df_clean = pd.DataFrame(df.text.apply(lambda x: clean_text(x)))

nlp = spacy.load('en')
def lemmatizer(text):
    sent = []
    doc = nlp(text)
    for word in doc:
        sent.append(word.lemma_)
    return " ".join(sent)

df_clean["text_lemmatize"] =  df_clean.apply(lambda x: lemmatizer(x['text']), axis=1)
df_clean['text_lemmatize_clean'] = df_clean['text_lemmatize'].str.replace('-PRON-', '')

```

```
vectorizer = CountVectorizer(analyzer='word',
                             min_df=3,
                             stop_words='english',
                             lowercase=True,
                             token_pattern='[a-zA-Z0-9]{3,}',
                             max_features=5000,
                            )

data_vectorized = vectorizer.fit_transform(df_clean['text_lemmatize_clean'])

i = 20
doc_topic_prior_set = 0.5 - 0.01*(i/2 + 1)
for x in range(i):
    doc_topic_prior_set += 0.01
    for i in range(1000):
        lda_model = LatentDirichletAllocation(n_components=2,
                                              max_iter=100,
                                              learning_method='online',
                                              learning_offset=50.,
                                              doc_topic_prior=doc_topic_prior_set,
                                              topic_word_prior=0.1)
                                              #,random_state=0)

        lda_output = lda_model.fit_transform(data_vectorized)


        def show_topics(vectorizer=vectorizer, lda_model=lda_model, n_words=10):
            keywords = np.array(vectorizer.get_feature_names())
            topic_keywords = []
            for topic_weights in lda_model.components_:
                top_keyword_locs = (-topic_weights).argsort()[:n_words]
                topic_keywords.append(keywords.take(top_keyword_locs))
            return topic_keywords

        topic_keywords = show_topics(vectorizer=vectorizer, lda_model=lda_model, n_words=10)

        df_topic_keywords = pd.DataFrame(topic_keywords)
        df_topic_keywords.columns = ['Word '+str(i) for i in range(df_topic_keywords.shape[1])]
        df_topic_keywords.index = ['Topic '+str(i) for i in range(df_topic_keywords.shape[0])]
        df_topic_keywords

        # column names
        topicnames = ["Topic" + str(i) for i in range(lda_model.n_components)]

        # index names
        docnames = ["Doc" + str(i) for i in range(len(df))]

        # Make the pandas dataframe
        df_document_topic = pd.DataFrame(np.round(lda_output, 2), columns=topicnames, index=docnames)

        # Get dominant topic for each document
        dominant_topic = np.argmax(df_document_topic.values, axis=1)
        df_document_topic['dominant_topic'] = dominant_topic

        df_list = list(df_document_topic['dominant_topic'])
        gr_list = df_list[0:9]
        at_list = df_list[9:14]

        gr_mean = sum(gr_list)/len(gr_list)
        at_mean = sum(at_list)/len(at_list)

        if abs(gr_mean-at_mean) > 0.7:
            print("GR List: ",gr_list)
            print("AT List: ",at_list)
            print('abs(%f - %f) = %f' % (gr_mean, at_mean, abs(gr_mean-at_mean)))

            print(df_topic_keywords.to_string())

            # Log Likelihood: Higher the better
            print("Log Likelihood: ", lda_model.score(data_vectorized))

            # Perplexity: Lower the better. Perplexity = exp(-1. * log-likelihood per word)
            print("Perplexity: ", lda_model.perplexity(data_vectorized))

            # See model parameters
            pprint(lda_model.get_params())

            print(df_document_topic)
```