Lampiran 1. Preprocessing

```
# LOWER CASING ---------------------
df['full_text'] = df['full_text'].str.lower()

# CLEANING AND TOKENIZING ----------------------
import string
import re
import swifter

def remove_tweet_special(text):
    text = text.replace('\\t'," ").replace('\\n'," ").replace('\\u'," ").replace('\\',"")
    text = text.encode('ascii', 'replace').decode('ascii')
    text = ' '.join(re.sub("([@#][A-Za-z0-9]+)|(\w+:\/\/\S+)"," ", text).split())
    return text.replace("http://", " ").replace("https://", " ")


def remove_number(text):
    return  re.sub(r"\d+", "", text)

def remove_punctuation(text):
    return text.translate(str.maketrans("","",string.punctuation))

def remove_whitespace_LT(text):
    return text.strip()

def remove_whitespace_multiple(text):
    return re.sub('\s+',' ',text)

def remove_singl_char(text):
    return re.sub(r"\b[a-zA-Z]\b", "", text)

def tokenization(text):
    return re.split('\W+', text)

df['full_text'] = df['full_text'].swifter.apply(remove_tweet_special)
df['full_text'] = df['full_text'].swifter.apply(remove_number)
df['full_text'] = df['full_text'].swifter.apply(remove_punctuation)
df['full_text'] = df['full_text'].swifter.apply(remove_whitespace_LT)
df['full_text'] = df['full_text'].swifter.apply(remove_whitespace_multiple)
df['full_text'] = df['full_text'].swifter.apply(remove_singl_char)
df['tokens'] = df['full_text'].swifter.apply(tokenization)


# INDONESIAN STOPWORD REMOVAL ----------------------
import nltk
import pandas as pd

nltk.download('stopwords')
from nltk.corpus import stopwords

list_stopwords = stopwords.words('indonesian')
# additional_stopwords = pd.read_csv('stopwordbahasa.csv', dtype=str).values.tolist()
additional_stopwords = ['yg','gk','mw','gmw','ya','tdk','gtw','sm','bareng','krn','dg','ya','nya','bgmn','mnjd','mjd',
                        'dkk','kt','lha','lah','lh','ga','loh','tuh','blm','blom','blum','bnr','bener','benr','bner',
                        'si','kyk','kaya','doang','ny','nya','eh','klo','kalo','mo','kah','kh','gt','adlh','utk','untk',
                        'aja','dah','deh','jd','gak','pdhl','kali','kli','gn','gini','ngapain','tau','tp','dn','mulu']
list_stopwords.extend(additional_stopwords)
# convert list to dictionary
list_stopwords = set(list_stopwords)

def stopwords_removal(words):
    #remove stopword pada list token
    return [word for word in words if word not in list_stopwords]

df['sw_token'] = df['tokens'].swifter.apply(stopwords_removal)


# STEMMING ----------------------

# import Sastrawi package
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
import swifter


# create stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# stemmed
def stemmed_wrapper(term):
    return stemmer.stem(term)

term_dict = {}

for document in df['sw_token']:
    for term in document:
        if term not in term_dict:
            term_dict[term] = ' '

print(len(term_dict))
print("------------------------")

for term in term_dict:
    term_dict[term] = stemmed_wrapper(term)
    print(term,":" ,term_dict[term])

print(term_dict)
print("------------------------")


# apply stemmed term to dataframe
def get_stemmed_term(document):
    return [term_dict[term] for term in document]

df['swst_token'] = df['sw_token'].swifter.allow_dask_on_strings(enable=True).apply(get_stemmed_term)
```

Lampiran 2. User Extraction

```
# USER EXTRACTION

import regex as re

def returnMentions(text):
    # x = re.findall(r"@[a-zA-Z0-9.-_]+", text)
    # for i, s in enumerate(x):
    #     x[i] = re.sub('@','',s)
    # return x

    return re.findall(r"@[a-zA-Z0-9.-_]+", text)

def addTag(text):
    return '@'+text

df['mentions'] = df['full_text'].apply(returnMentions)
df['username'] = df['username'].apply(addTag)
```

Lampiran 3. Term Frequency

Lampiran 4. Wordcloud

```
import matplotlib.pyplot as plt
import regex as re
from wordcloud import WordCloud
%matplotlib inline

dfWordCloud = df['swst_token']

def delimitInit(text):
    return re.sub(r'[\'\[\],]+', '',text)

wordcloudMaterial = ''
for i, s in enumerate(dfWordCloud):
    wordcloudMaterial += delimitInit(s)

# print(wordcloudMaterial)

wordcloud = WordCloud(width=4282, height=2151).generate(str(wordcloudMaterial))

plt.figure(figsize=(55,55))
plt.imshow(wordcloud)
plt.axis("off")
plt.show()
```

Lampiran 5. Pembangunan Jaringan

```
# VECTORS
dfGraphCentrality = tempdf.drop(columns=['swst_token']).explode('mentions')
dfGraphCentrality['mentions'] = dfGraphCentrality['mentions'].apply(lambda y: np.nan if len(y) == 0 else y)
dfGraphCentrality = dfGraphCentrality.dropna()
dfGraphCentrality = dfGraphCentrality.rename(columns={'username': 'Source', 'mentions': 'Target'})
dfGraphCentrality.to_csv('tweets-data/classed/bts_csv_preprocessed.csv',encoding='utf-8',index=False)


dfNodes = dfGraphCentrality.drop(columns=['Target','id_str'])
dfNodes = dfNodes.rename(columns={'Source': 'Id'})
dfNodes['Label'] = dfNodes['Id']

dfNodes2 = dfGraphCentrality.drop(columns=['Source','id_str'])
dfNodes2 = dfNodes2.rename(columns={'Target': 'Id'})
dfNodes2['Label'] = dfNodes2['Id']

dfNodes = pd.concat([dfNodes, dfNodes2],ignore_index=True).drop_duplicates()
dfNodes['class'] = 'bts'
dfNodes.to_csv('tweets-data/classed/bts_csv_nodes2.csv',encoding='utf-8',index=False)
```

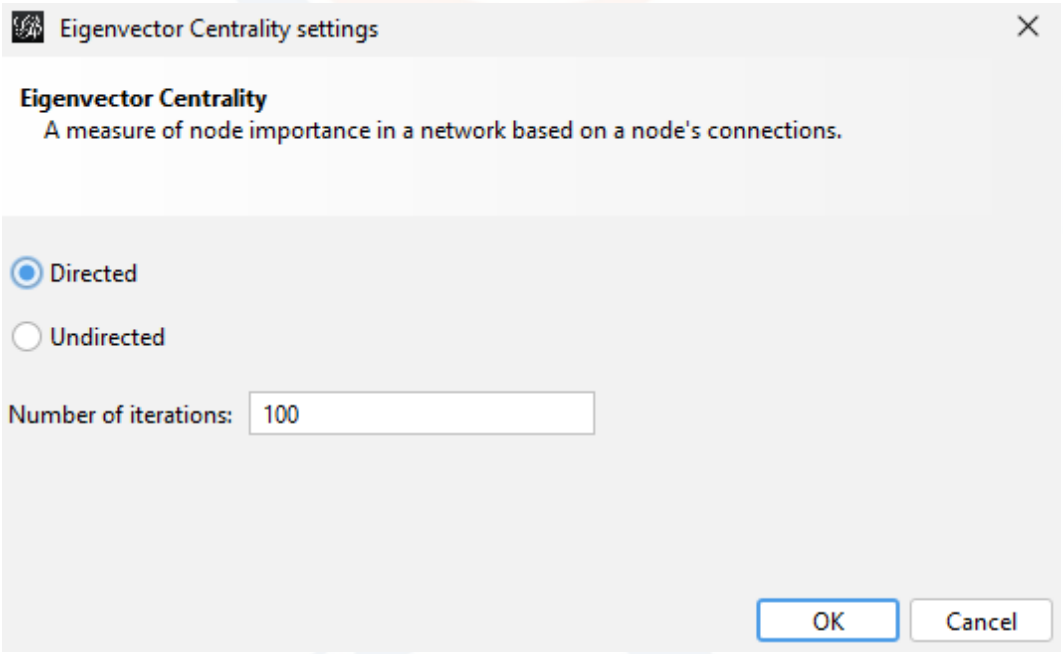
Lampiran 6. Algoritma Sentralitas

```
inDegCent = pd.DataFrame.from_dict(
    nx.in_degree_centrality(graphCentrality),
    orient='index',
    columns=['InDegree']
    )

outDegCent = pd.DataFrame.from_dict(
    nx.out_degree_centrality(graphCentrality),
    orient='index',
    columns=['OutDegree']
    )

closeCent = pd.DataFrame.from_dict(
    nx.closeness_centrality(graphCentrality),
    orient='index',
    columns=['Closeness']
    )

betweenCent = pd.DataFrame.from_dict(
    nx.betweenness_centrality(graphCentrality),
    orient='index',
    columns=['Betweenness']
    )
```



Lampiran 7. Daftar Bimbingan

Bimbingan

No	Dosen	Topik	Tanggal Bimbingan	Jenis Bimbingan	Catatan Perbaikan
1	5709 - MUNAWAR , S.TP, MM, Ph.D.	9 Oktober 2023 Bimbingan Revisi setelah Sidang Proposal	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
2	5709 - MUNAWAR , S.TP, MM, Ph.D.	16 Oktober 2023 Bimbingan perbaikan Bab 2	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
3	5709 - MUNAWAR , S.TP, MM, Ph.D.	23 Oktober 2023 Bimbingan Bab 4, visualisasi jaringan	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
4	5709 - MUNAWAR , S.TP, MM, Ph.D.	30 Oktober 2023 Bimbingan Bab 4, visualisasi jaringan, sentralitas	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
5	5709 - MUNAWAR , S.TP, MM, Ph.D.	20 November 2023 Bimbingan Bab 4, perubahan visualisasi jaringan, penambahan informasi akun	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
6	5709 - MUNAWAR , S.TP, MM, Ph.D.	3 Desember 2023 Bimbingan Bab 4, penambahan Interpretasi hasil	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
7	5709 - MUNAWAR , S.TP, MM, Ph.D.	7 Januari 2024 Bimbingan Bab 3 perbaikan tahapan penelitian dan Bab 4	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
8	5709 - MUNAWAR , S.TP, MM, Ph.D.	17 Januari 2024 Bimbingan dan perbaikan Bab 1 dan Bab 4 perbaikan interpretasi hasil	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
9	5709 - MUNAWAR , S.TP, MM, Ph.D.	22 Januari 2024 Bimbingan Bab 4 dan Bab 5	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	
10	5709 - MUNAWAR , S.TP, MM, Ph.D.	29 Januari 2024 Bimbingan persiapan sebelum sidang Akhir	27 Feb 2024	Skripsi/Tesis/BusinessPlan Proposal	

Lampiran 8. Form Pengajuan Sidang

UNIVERSITAS ESA UNGGUL
FAKULTAS ILMU KOMPUTER
Jl. Arjuna Utara No. 9, Tol Tomang, Kebon Jeruk, Jakarta Barat 11470

FORM PENGAJUAN SIDANG
MAGANG / SEMINAR PROPOSAL/ SKRIPSI/ TUGAS AKHIR

Nama	: Kristian Tirtawinata
NIM	: 20190801053
Program Studi	: Teknik Informatika / ~~Sistem Informasi~~ *
Judul	: Penerapan Sentralitas Graf Social Network Analysis Topik Kasus Korupsi BTS Kominfo Di Media Sosial Twitter
Periode	: Ganjil / ~~Genap~~* (Tahun Akademik 2023/2024)
Kategori	: ~~Sidang Magang~~ / ~~Seminar Proposal~~ / Sidang Skripsi *

**coret yang tidak perlu*

Jakarta, 29 Januari 2024

Menyetujui,

Pembimbing

(Munawar, S.TP, MMSI, Ph.D.)

Mengetahui,

Koordinator Tugas Akhir

(M. Bahrul Ulum, S.Kom, M.Kom)