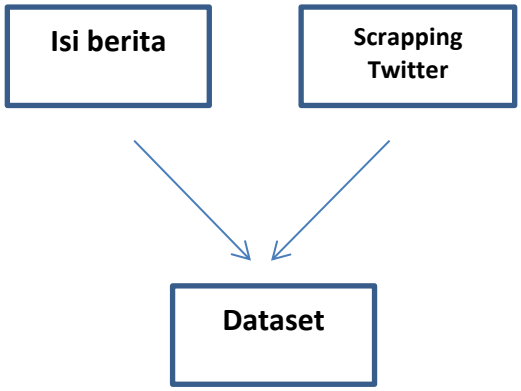
Report Tugas Akhir

1. Bussines Understanding

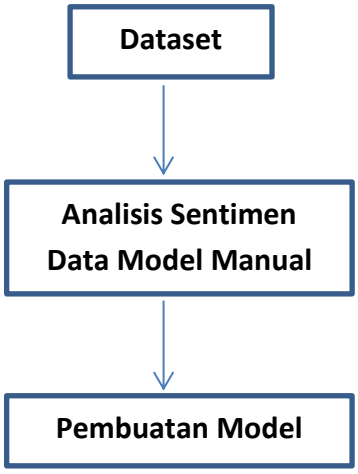
Menganalisis sentiment masyarakat terhadapat kejadian meletusnya Gunung Semeru dengan data twitter dan artikel berita.

2. Pengenalan Dataset

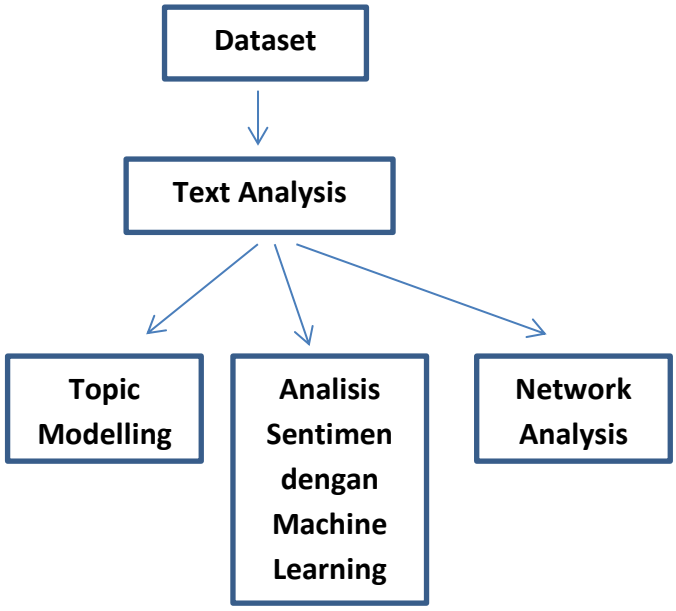
- Ada terdapat 2 Dataset yaitu dataset model dan dataset test.
- Dataset model digunakan untuk membuat model. Model dibuat dengan *machine learning* ***Naive Bayes***.
- Baik dataset model maupun dataset test di ambil dari dua sumber yaitu scraping data dari twitter dari tanggal 3 – 6 Desember 2021 dan isi artikel dari berita dengan keyword terkait.
- Untuk dataset model, keyword yang diambil mengenai **liburan**, **corona**, dan **pariwisata**. Dari ketiga keyword ini diharapkan bisa mendapatkan dataset mengenai liburan maupun pariwisata saat pandemi korona.
- Untuk dataset test, keyword yang dipilih tentang **semeru**,**bencana**, **jawa timur**. Dari ketiga keyword ini diharapkan bisa mendapatkan dataset mengenai bencana letusan Gunung Semeru di Jawa Timur.
- Untuk scraping data twitter, data retweet tetap diambil, karena penulis menganggap data retweet merupakan kegiatan user yang mem-forward dan setuju akan tweet tersebut.



2.1. Alur pada dataset model

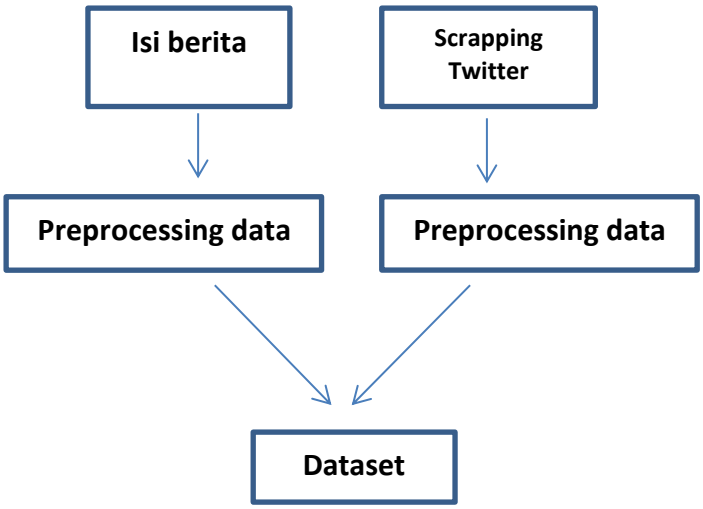


1.2. Alur pada dataset test



3. Preprocessing dataset

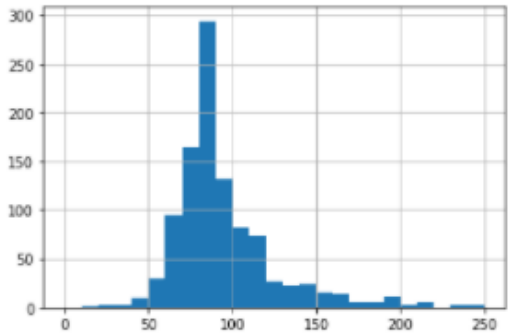
- Ada 4 teknik utama dalam melakukan text preprocessing, yaitu: case folding, tokenizing, stopwords, stemming.
- Case folding adalah teknik pembersihan dimana kita hanya menyisakan kata-kata dengan alfabet kecil, artinya semua angka, tanda baca, huruf besar, alamat situs, dsb diganti/dihilangkan.
- Tokenizing adalah teknik pemisahan sebuah kalimat menjadi kata per kata yang biasa disebut token.
- Filtering adalah menghilangkan kata-kata yang tidak terpakai atau banyak digunakan dari kalimat (stopword). Kata tersebut bisa kata sambung atau kata ganti orang.
- Stemming adalah teknik dimana kita mengembalikan semua kata ke dalam bentuk dasarnya.



4. Text Analysis

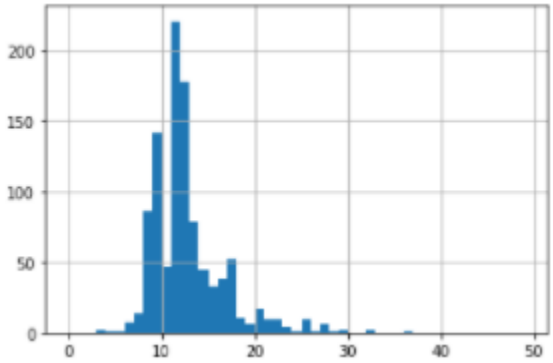
Terdapat 5 Analysis yang akan kita lakukan sekarang:

- Distribusi Frekuensi jumlah huruf pada suatu data



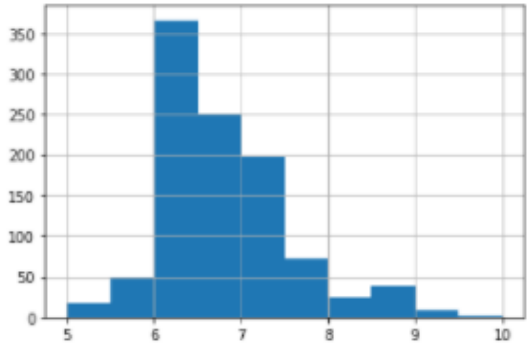
Dari sini kita bisa melihat, distribusi terbanyak berada pada 60 - 120 karakter per tweet/paragraph (berita). Dengan tweet/paragraph (berita) yang memiliki panjang diatas 150 karakter, setiap nilainya tidak lebih dari 50 tweet/paragraph (berita).

- Distribusi Frekuensi jumlah kata pada suatu data



Dari sini kita bisa melihat, distribusi terbanyak berada pada 11 - 15 kata per tweet/paragraph (berita). tweet/paragraph (berita) dengan jumlah kata di atas 20 bisa dibilang tidak terlalu banyak.

- Distribusi Frekuensi panjang kata rata-rata pada suatu data



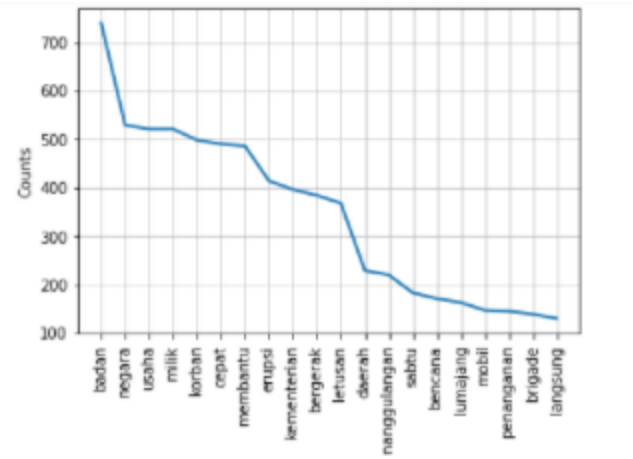
Dari sini kita bisa melihat, distribusi terbanyak berada pada 6 - 7 karakter per kata per tweet/paragraph (berita).Jumlah huruf pada kata yang umum ada pada bahasa indonesia. Jika kita gabungkan hasil analisa sebelumnya, maka kita tahu mayoritas tweet/paragraph (berita) berada pada 11 -15 kata dengan setiap kata berada di antara 6 - 7 karakter dan total 60 - 120 karakter. Bisa dibilang cukup banyak orang yang memberikan tweet dengan opini.

- Distribusi Frekuensi kata yang sering keluar

Hasilnya,

```
([('badan', 740),
  ('negara', 529),
  ('usaha', 521),
  ('milik', 521),
  ('korban', 498),
  ('cepat', 490),
  ('membantu', 486),
  ('erupsi', 414),
  ('kementerian', 396),
  ('bergerak', 384),
  ('letusan', 368),
  ('daerah', 229),
  ('penanggulangan', 220),
  ('sabtu', 183),
  ('bencana', 171),
  ('lumajang', 163),
  ('mobil', 147),
```

Jika kita visualisasikan,



Badan berada di urutan teratas, dikarenakan keyword yang kita cari sudah dihapus melalui stopword, jadi tidak heran jika keyword yang kita cari tidak ada di daftar frekuensi kata yang sering keluar. Negara, Usaha, Milik, Korban dan Cepat menjadi kata selanjutnya yang sering keluar. Dari sini kita bisa mengambil kesimpulan keselurahan tweet/paragraph (berita) adalah Badan Usaha Milik Negara yang Cepat Membantu Korban Erupsi.

- Distribusi bi-gram,

nlai n yang akan kita pakai adalah 2.

```
(badan, usaha)            521
(usaha, milik)            521
(milik, negara)           521
(kementerian, badan)      377
(bergerak, cepat)         373
(membantu, korban)        372
(korban, letusan)         363
(cepat, membantu)         351
(negara, bergerak)        347
(badan, penanggulangan)   206
```

Dengan melihat bi-gram tersebut kita mendapat gambaran yang lebih jelas, bahwa keseluruhan tweet/paragraph (berita) membahas tentang Badan Usaha Milik Negara yang Cepat Bergerak Membantu Korban

5. Topic Modelling

Pada tahap ini kita akan memanfaatkan teknik LDA (Latent Dirichlet Allocation) dan library Gensim. LDA akan menghitung koherensi dari suatu baris data terhadap topik-topik yang tidak diketahui, tapi diasumsikan ada. Saya membuat terlebih dahulu model awal dengan jumlah topik sebanyak 5. setelah itu kita jalankan untuk melatih model kita. Setelah selesai, kita tampilkan data kita.

```
[(0,
  '0.094*"lumajang" + 0.048*"bantuan" + 0.039*"personel" + 0.038*"erupsi" + '
  '0.026*"masyarakat" + 0.021*"bumnger" + 0.017*"logistik" + 0.017*"tidak" + '
  '0.016*"korban" + 0.015*"terdampak"'),
 (1,
  '0.066*"penanganan" + 0.065*"membantu" + 0.065*"erupsi" + 0.052*"tengah" + '
  '0.052*"jawa" + 0.046*"cepat" + 0.043*"respons" + 0.038*"saja" + '
  '0.030*"pemprov" + 0.029*"rismawidiono"'),
 (2,
  '0.059*"mobil" + 0.057*"brigade" + 0.042*"puan" + 0.040*"ketua" + '
  '0.037*"meletusnya" + 0.037*"prihatin" + 0.036*"dpr" + 0.035*"maharani" + '
  '0.029*"indonesia" + 0.028*"erupsi"'),
 (3,
  '0.043*"daerah" + 0.041*"penanggulangan" + 0.038*"badan" + 0.034*"bencana" + '
  '0.021*"desa" + 0.018*"terdampak" + 0.011*"pronojiwo" + 0.010*"wib" + '
  '0.010*"ganjar" + 0.010*"katanya"'),
 (4,
  '0.099*"badan" + 0.079*"negara" + 0.078*"milik" + 0.078*"usaha" + '
  '0.062*"bergerak" + 0.061*"korban" + 0.059*"cepat" + 0.058*"letusan" + '
  '0.057*"kementerian" + 0.056*"membantu"')]
```

Angka di awal tuple adalah indeks topiknya (0-4) karena kita memilih 5 topik. Angka di sebelah kiri kata adalah nilai bobot dari kata tersebut terhadap topik yang bersangkutan (misalnya badan bernilai 0.099 pada topik 4). Yang ditampilkan di atas 10 kata dengan nilai pembobotan terbesar.

Untuk interpretasi misalkan indeks 4 karena terdapat, kata badan, negara, milik, usaha, bergerak, korban, cepat bisa disimpulkan bahwa Topik pada indeks 4 menjelaskan tentang **Badan Usaha Milik Negara yang Bergerak Cepat Membantu Korban.**

Nilai Coherencenya sebesar 0.38688698631364604

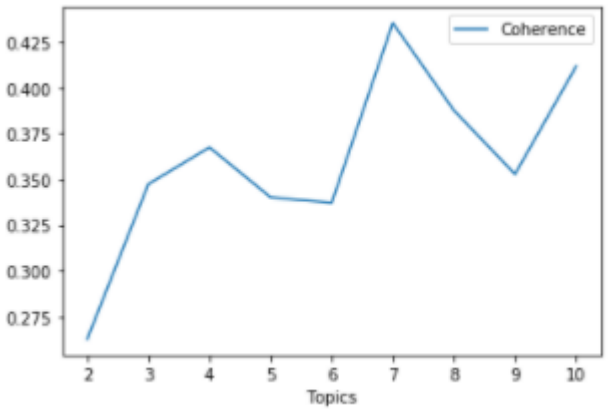
Kita kan membuat fungsi untuk melakukan hyper tuning parameter pada model kita untuk memperbaiki *coherence value* dan Setelah itu kita lakukan proses pencarian nilai koheren terhadap parameter yang dimiliki. Kemudian kita memvalidasi set. validasi set digunakan untuk memvalidasi data yang kita gunakan, pada kesempatan ini dibagi menjadi saat 75% data dengan 100% data. Tqdm adalah library yang digunakan untuk melihat persen proses berjalan.

hasilnya seperti ini,

	Validation_Set	Topics	Alpha	Beta	Coherence
0	75% Corpus	2	0.01	0.01	0.270377
1	75% Corpus	2	0.01	0.31	0.262670
2	75% Corpus	2	0.01	0.61	0.264501
3	75% Corpus	2	0.01	0.9099999999999999	0.264501
4	75% Corpus	2	0.01	symmetric	0.262670
...	...	...	...	...	...
535	100% Corpus	10	asymmetric	0.01	0.434893
536	100% Corpus	10	asymmetric	0.31	0.369701
537	100% Corpus	10	asymmetric	0.61	0.393574
538	100% Corpus	10	asymmetric	0.9099999999999999	0.372486
539	100% Corpus	10	asymmetric	symmetric	0.391308

Kita pilih salah satu angka dengan nilai yang sama, pada kesempatan ini saya memilih alpha = 0.01 & beta 0.3. Ini dilakukan untuk melihat nilai k (topik) terbaik

Lalu hasilnya kita plot



Semakin tinggi artinya semakin baik nilainya. Dengan berdasar pada hal tersebut maka kita akan memilih k = 3. Selanjutnya kita akan memilih alpha dan beta terbaik, caranya adalah dengan memilih k = 3 pada tabel, dan cari kombinasi alpha dan beta dengan *coherence value* terbaik.

	Validation_Set	Topics	Alpha	Beta	Coherence
305	100% Corpus	3	0.31	0.01	0.395042
315	100% Corpus	3	0.9099999999999999	0.01	0.385060
316	100% Corpus	3	0.9099999999999999	0.31	0.385060
319	100% Corpus	3	0.9099999999999999	symmetric	0.385060
317	100% Corpus	3	0.9099999999999999	0.61	0.385060

Terlihat dari tabel di atas, *coherence value* terbaik ada saat nilai alpha 0.31 dan beta 0.01.Setelah mendapatkan parameter terbaik, saatnya kita buat kembali model kita.

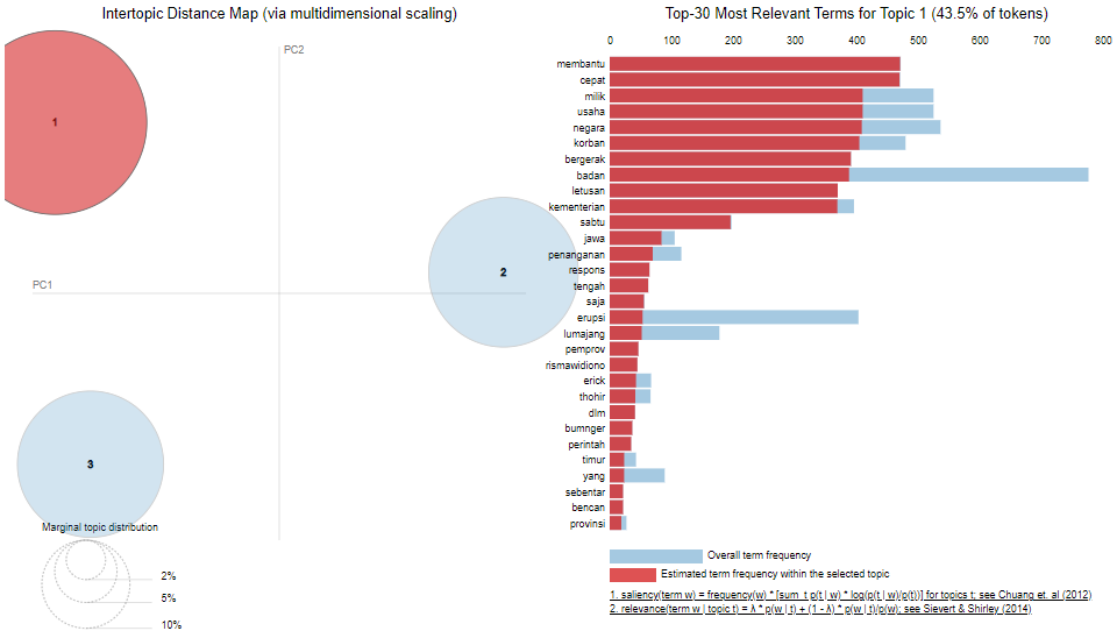
```
[(0,
  '0.112*"badan" + 0.081*"daerah" + 0.077*"penanggulangan" + 0.060*"bencana" + '
  '0.057*"erupsi" + 0.040*"berkoordinasi" + 0.036*"negara" + 0.033*"usaha" + '
  '0.033*"milik" + 0.033*"langsung"'),
 (1,
  '0.081*"membantu" + 0.081*"cepat" + 0.072*"milik" + 0.072*"usaha" + '
  '0.071*"negara" + 0.070*"badan" + 0.069*"korban" + 0.068*"bergerak" + '
  '0.064*"letusan" + 0.062*"kementerian"'),
 (2,
  '0.060*"mobil" + 0.057*"brigade" + 0.039*"puan" + 0.038*"ketua" + '
  '0.038*"erupsi" + 0.037*"prihatin" + 0.037*"dpr" + 0.037*"meletusnya" + '
  '0.031*"maharani" + 0.025*"indonesia"')]
```

Analisa terhadap setiap topik adalah:

1) Badan Penanggulangan Bencana Daerah berkoordinasi Langsung dengan Badan Usaha Milik Negara
2) Kementrian dan Badan Usaha Milik Negara Bergerak Cepat Membantu Korban
3) Mobil Brigade dan Ketua dpr Indonesia Puan Maharani Prihatin

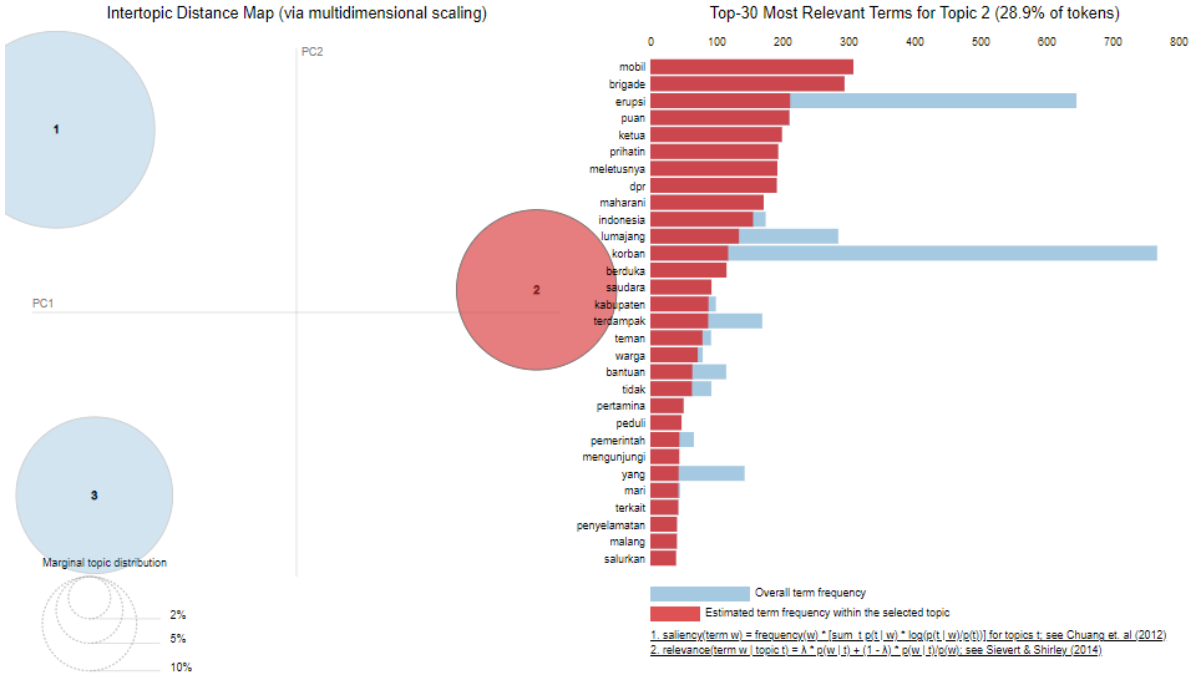
Mari kita visualisasikan datanya,

Untuk topic 1;



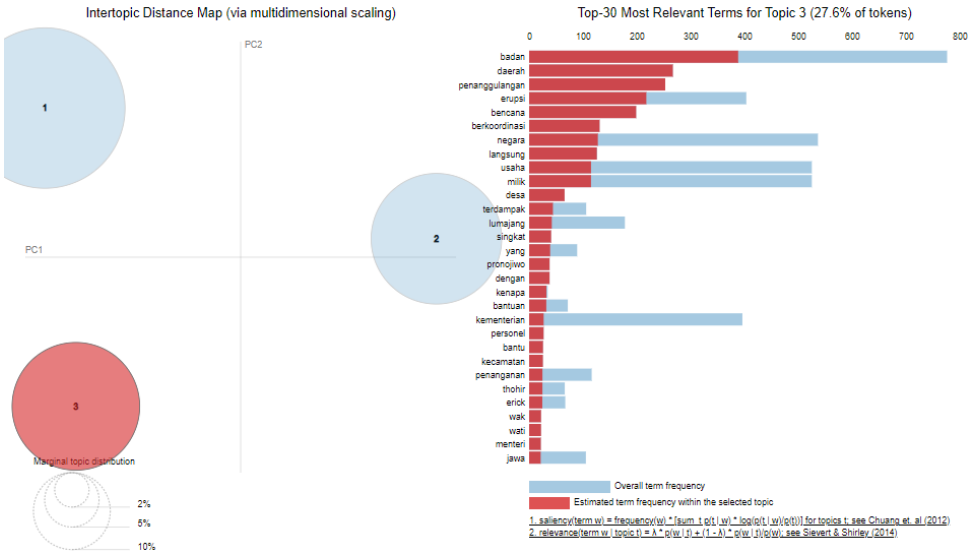
Bisa kita lihat bersama kata-kata yang sering keluar untuk topic 1 ini seperti membantu,dapat,milik,usaha dan lain lain

Untuk topic 2;



Bisa kita lihat bersama kata-kata yang sering keluar untuk topic 2 ini seperti mobil, brigade, erupsi, puan,ketua, dan lain lain

Untuk topic 3;



Bisa kita lihat bersama kata-kata yang sering keluar untuk topic 3 ini seperti badan, daerah, penanggulangan, erupsi, dan lain lain

6. Analisis Sentimen dengan Machine Learning

Pertama kita buat dulu model naïve bayes dengan membagi dataset model menjadi data train dan testing lalu fit kedalam model kita. Dan kita lakukan penghitungan confussion matrix, classification report, dan accuracy score. Lalu hasilnya,

```
[[ 32   7]
 [ 32 141]]
              precision    recall  f1-score   support

           0       0.50      0.82      0.62        39
           1       0.95      0.82      0.88       173

    accuracy                           0.82       212
   macro avg       0.73      0.82      0.75       212
weighted avg       0.87      0.82      0.83       212

nilai akurasinya adalah  0.8160377358490566
```

Nilai Akurasi, Recall, F1-Score diatas 80 dan precision ada 0.50 . Ini bearti sebenarnya model yang kita buat kinerjanya tidak terlalu baik untuk kasus ini. Support yang tidak terlalu jauh menandakan bahwa pembagian dataset kita cukup baik.

Sekarang mari kita gunakan untuk melakukan klasifikasi pada dataset test dan hasil klasifikasi sentiment positif dan negative sebagai berikut,

```
1    546
0    479
```

Masing-masing dari hasil klasifikasi tampak berimbang.

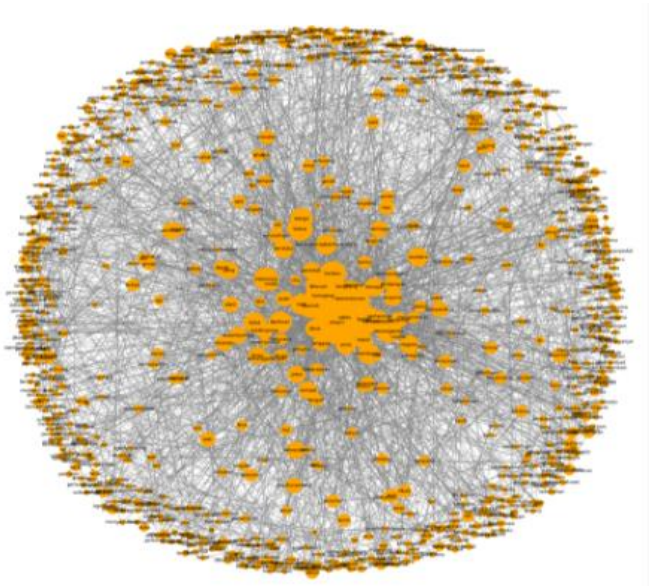
Lalu kita lihat kata-kata apa aja yang sering muncul dari kedua klasifikasi sentiment.

```
wordcloud_visualization(' '.join(isi_positif))
wordcloud_visualization(' '.join(isi_negatif))
```

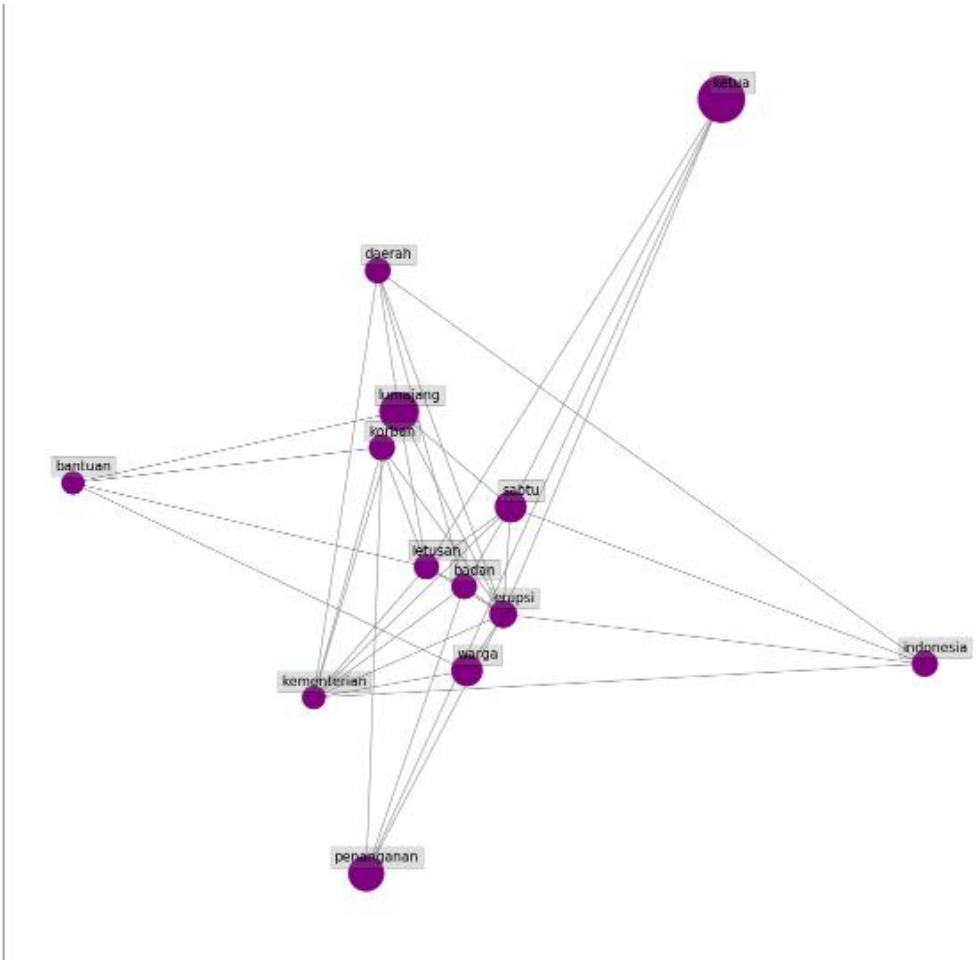
Baik sentiment positif maupun negatif keduanya sering muncul kata-kata seperti badan usaha,milik Negara ,dan lain-lain. Menariknya kata-kata seperti membantu korban sering keluar di sentiment positif dan nama seperti puan muncul di sentiment negatif.

7. Network Analysis

Kita ingin melihat relasi kata terhadap kata lainnya.



Setelah diproses dan di plot, ternyata hasilnya masih terlalu ramai, untuk lebih detailnya lagi kita filter berdasarkan nilai degree > 30. Degree adalah berapa banyak relasi yang terhubung kepada node yang jadi acuan.



Pada graph terakhir kita bisa melihat relasi antar kata yang bigram dan degree nya memiliki nilai yang tinggi. Ada kata seperti lumajang, bantuan, warga, dan lain-lain