

Pemetaan Lokasi Kejadian dalam Sistem Deteksi Kejadian dengan Data Twitter Menggunakan Teori Graf

¹Fatra Nonggala Putra, ²Iza Arfiana Fauziah

Program Studi Ilmu Komputer, Universitas Nahdlatul Ulama Blitar,
Blitar, Indonesia

Email: ¹fatranp@unublitar.ac.id, ²izaarfiana@gmail.com

Tersedia Online di

<http://www.jurnal.unublitar.ac.id/index.php/briliant>

Sejarah Artikel

Diterima pada 30 April 2020
Disetujui pada 15 Mei 2020
Dipublikasikan pada 30 Mei 2020
Hal. 431-441

Kata Kunci:

Sistem deteksi kejadian; twitter;
FASM; graph

DOI:

<http://dx.doi.org/10.28926/briliant.v3i4.472>

Abstrak: Sistem deteksi kejadian dengan data twitter sebagai data masukannya telah banyak diteliti sebelumnya. Suatu informasi kejadian dapat dikategorikan sebagai kejadian penting adalah jika minimal terdapat informasi lokasi/tempat kejadian. Dalam sebuah *tweet*, seringkali terdapat lebih dari satu entitas lokasi kejadian karena kejadiannya memiliki dampak lebih dari satu titik lokasi. Hal ini berarti ada keterhubungan atau relasi antar lokasi kejadian sehingga diperlukan cara merubah relasi yang berupa teks ke dalam bentuk graf. Selain itu, banyak juga suatu informasi kejadian menulis entitas lokasi yang dengan ejaan yang berbeda atau terdapat kesalahan penulisan. Hal ini menyulitkan pemetaan lokasi terjadinya kejadian ke dalam maps karena entitas lokasi tidak terdapat di dalam *gazetteer*. Maka dari itu, diperlukan standarisasi entitas nama lokasi kejadian dengan *Fast Approximate String Matching* (FASM) dan

mengkonversi relasi antar lokasi ke dalam graf. Dalam penelitian ini menunjukkan bahwa merubah relasi antar lokasi ke dalam bentuk graf yang sebelumnya telah dilakukan standarisasi entitas nama lokasi sangat memudahkan sistem dalam pemetaan atau memvisualisasikan informasi lokasi kejadian ke dalam GoogleMaps.

PENDAHULUAN

Penduduk indonesia setiap tahunnya mengalami peningkatan jumlah populasi penduduk yang pada tahun 2019 jumlah pendudukan Indonesia diproyeksikan mencapai 271 jiwa. Dengan banyaknya penduduk di Indonesia, membuat kepadatan dan mobilitas masyarakatnya menjadi sangat tinggi terutama di kota-kota besar. Tingginya mobilitas menimbulkan dampak meningkatnya jumlah kecelakaan dan kemacetan di jalan raya. Disamping itu, kemacetan adalah hal yang tidak diinginkan oleh setiap orang. Setiap orang selalu berupaya secepat mungkin mencapai tempat kerja, tempat wisata, rumah, dan tempat lainnya yang menjadi tujuannya.

Beberapa cara telah dilakukan untuk memudahkan masyarakat menghindari kemacetan atau paling tidak memberikan informasi seperti kemacetan, kebakaran, dan banjir di daerah tertentu. Mulai dari cara konvensional dengan menempatkan sensor-sensor deteksi dan kamera hingga pemanfaatan data media sosial seperti Twitter dan Facebook (Gelernter & Balaji, 2013; Gu, Qian, & Chen, 2016; Li et al., 2017). Pemanfaatan media sosial sebagai sumber informasi dibandingkan dengan cara konvensional dengan sensor dan kamera dianggap

lebih murah dan cepat. Twitter merupakan salah satu *microblogging* populer untuk pertukaran informasi penggunanya. Twitter mampu menghasilkan 500 juta tweet per harinya dan setiap bulannya terdapat 330 juta pengguna aktif, maka pantaslah jika twitter menyandang *microblogging* terpopuler untuk saat ini (Gu et al., 2016). Pada saat ini, pemanfaatan Twitter salah satunya adalah sebagai media berbagi informasi kejadian penting oleh penggunanya yang mayoritas kejadian tersebut memang diketahui secara langsung oleh si pengguna. Ada 2 jenis pengguna dalam media sosial yaitu pengguna sebagai individu dan pengguna sebagai representasi perusahaan ataupun lembaga seperti akun BMKG, Dishub, Radio, dan lain-lain.

Sejumlah penelitian dengan memanfaatkan data twitter sebagai pendeteksi kejadian secara *realtime* telah dilakukan antara lain oleh Yiming Gu dkk dengan melakukan pengklasifikasian biner menggunakan semi-Naive Bayes terhadap data stream twitter. Pada Hasby dan Khodra melakukan ekstraksi lokasi kemacetan menggunakan named-entity-recognition (NER) dan relation extraction (RE) yang kemudian digunakan untuk mencari jalur perjalanan lain yang paling optimal (Hasby, Khodra, & Purwarianti, 2013). Penelitian lainnya, melakukan ekstraksi informasi data *stream* twitter menggunakan *NeuroNER* untuk mendeteksi entitas lokasi kejadian dalam teks tweet dan RCNN untuk mengklasifikasikan tweet sebagai tweet informatif atau tweet non-informatif (Putra & Fatichah, 2018).

Pada beberapa penelitian deteksi kejadian belum memperhitungkan kesalahan penulisan nama lokasi sehingga jika terjadi kesalahan penulisan lokasi, maka tidak akan bisa dipetakan ke dalam peta atau *maps*. Dalam sebuah teks tweet yang melaporkan suatu kejadian, kadangkala menyebut satu nama lokasi lebih dari satu kali namun berbeda dalam penulisannya atau terjadi kesalahan penulisan. Hal ini sangat disayangkan jika ada informasi kejadian penting tapi tidak dapat di visualisasikan ke dalam map dikarenakan ada kesalahan penulisan nama lokasi kejadian.

Dilain sisi, setiap informasi kejadian haruslah minimal mempunyai 1 (satu) entitas lokasi. Dengan memanfaatkan teori graf, suatu lokasi dapat dianggap sebagai sebuah node. Suatu kejadian bisa berdampak pada satu tempat tertentu atau beberapa tempat sekaligus yang berhubungan. Hubungan antar lokasi ini disebut sebagai relasi. Jadi, node satu dengan node lainnya akan dihubungkan dengan jenis relasi tertentu. Ada node sebagai titik awal, titik antara, dan node sebagai titik akhir.

Berdasarkan paparan permasalahan di atas, maka diperlukan standarisasi nama atau entitas lokasi kejadian berdasarkan *gazetteer* untuk memastikan entitas lokasi tersebut valid. Hal ini dilakukan untuk memudahkan dalam memetakan lokasi suatu kejadian ke dalam graf sebelum divisualisasikan ke dalam *Google Maps*.

METODE

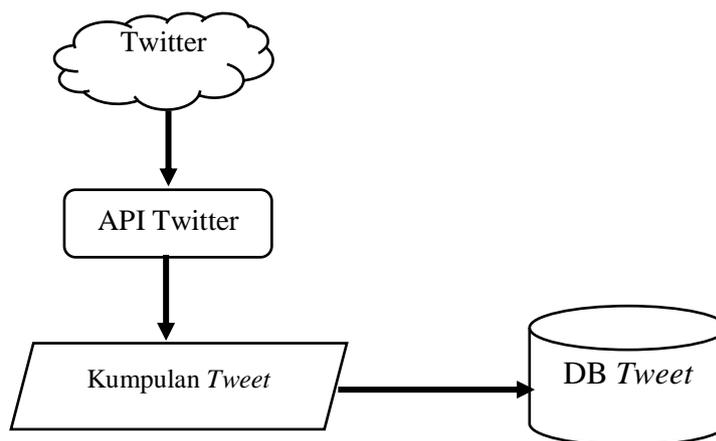
Bab ini memaparkan metode yang digunakan untuk menyelesaikan permasalahan pada visualisasi lokasi kejadian pada sistem deteksi kejadian dengan menggunakan teori graf. Tahapan ini dibagi menjadi 3 subbab yaitu; 1) persiapan data, 2) standarisasi entitas lokasi, dan 3) konversi lokasi ke dalam bentuk graf. Penjelasan setiap subbab sebagai berikut:

A. *Persiapan Dataset*

Pada penelitian ini ada dua jenis data yang dibutuhkan yaitu: 1) data *Tweet*, dan 2) data Gazetteer yang pada penelitian ini hanya fokus pada kota Surabaya. Setiap jenis data tersebut perlu diolah dahulu melalui tahapan praproses untuk memastikan data yang akan diolah merupakan data bagus yang dapat digunakan oleh sistem.

1) Data Tweet

Pengambilan data tweet dilakukan menggunakan API Twitter yang dengan bantuan *library* dari python yaitu *tweepy*. Target dari pengambilan data ini adalah berdasarkan ID Twitter langsung seperti akun Radio Suara Surabaya (@e100ss) dan berdasarkan *keyword* seperti ‘kebakaran’, ‘macet’, ‘kecelakaan’, dan ‘banjir’. Alur untuk mendapatkan data dari Twitter ditunjukkan pada Gambar 1.

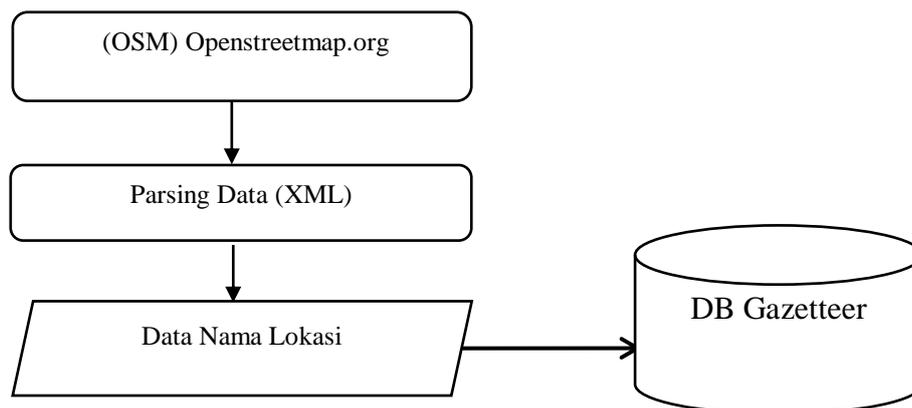


Gambar 1. Alur Pengambilan *tweet* dari Twitter.

2) Data Gazetteer

Selain data *tweet* dari pengguna Twitter yang memberikan informasi kejadian penting, juga dibutuhkan data gazetteer atau nama-nama tempat/lokasi di suatu lokasi. Dalam penelitian ini, peneliti memanfaatkan layanan peta *open source* yaitu OSM (*open street map*) untuk mendapatkan nama-nama tempat/lokasi yang berada di kota Surabaya. Alur pengambilan data gazetteer ditunjukkan pada Gambar 2.

Pengambilan data gazetteer menggunakan bantuan *libray* dari python yaitu *xmltree* karena struktur data dari OSM berupa XML. *Xmltree* memudahkan parsing struktur XML ke struktur teks sesuai kebutuhan. Data yang diperlukan dari OSM yaitu: 1) id lokasi, 2) kota, 3) alamat lokasi, 4) nama lokasi, 5) latitude dan longitude. Semua data yang berhasil diambil disimpan ke dalam database *mysql*. Fungsi dari DB *Gazetteer* ini adalah untuk standardisasi nama lokasi pada proses Standardisasi Entitas Lokasi.



Gambar 2. Alur Pengambilan Data *Gazetteer* dari OSM.

B. Standardisasi Entitas Lokasi

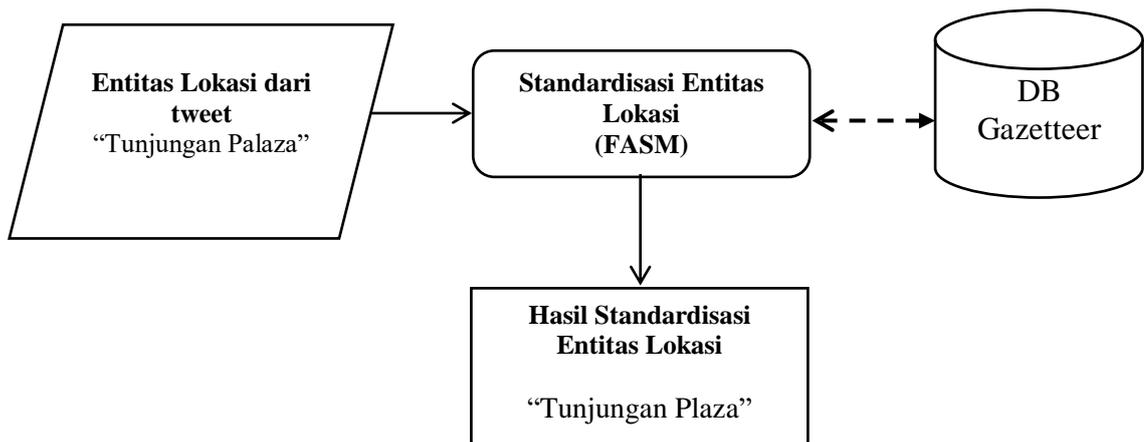
Pada tahap ini tidak akan dibahas bagaimana proses penentuan entitas lokasi dan klasifikasi tweet kejadian. Proses penentuan entitas lokasi dan klasifikasi dapat dilihat pada penelitian (Putra & Fatichah, 2018).

Standardisasi Entitas Lokasi adalah proses memperbaiki kesalahan penulisan nama lokasi yang pada penelitian ini menggunakan *Fast approximate string matching* (FASM) dengan yang merupakan modifikasi dari metode pencarian *fuzzy string matching* (FSM). Perbedaan dari keduanya adalah FASM memanfaatkan teknik *edit distance* Damerau-Levenshtein Distance, sedangkan FSM memanfaatkan teknik *edit distance* populer lainnya yaitu Levenshtein Distance untuk mencari *string* yang cocok dengan prediksi kemungkinan yang paling mendekati (bukan tepatnya atau pasti) (Garbe, 2015). FASM memiliki performa prediksi dan waktu komputasi yang lebih baik daripada *fuzzy string matching* standar. Contoh hasil pencarian perbaikan *string* menggunakan FASM ditunjukkan pada Tabel 1.

Tabel 1 Contoh Pencarian Perbaikan String dengan FASM (Garbe, 2015)

String	Best Correction	Edit Dist.	Max. Edit Dist.	ms per 1000 lookups
zacamodation	accommodation	4	4	727
acamodation	accommodation	3	3	180
acomodation	accommodation	2	2	33
hous	house	1	1	24
house	house	0	1	1

Proses standardisasi nama dilakukan menggunakan teknik FASM yang alur prosesnya ditunjukkan pada Gambar 3. Entitas nama lokasi yang berhasil dikenali oleh sistem akan dilakukan pencocokan *string* dari entitas tersebut menggunakan teknik FASM. Pencocokan atau *matching* memanfaatkan kamus database Gazetteer yang telah diperoleh sebelumnya.



Gambar 3. Diagram Alur Standardisasi Entitas Nama Lokasi.

C. Konversi Lokasi Kejadian ke Dalam Bentuk Graf

Graf merupakan kumpulan *node* (V) bersama yang dihubungkan dengan *edges* (E) untuk menghubungkan antar node (Hagberg, Schult, & Swart, 2018). Suatu graf mempunyai kumpulan node (V) yang terdiri dari $\{v_1, v_2, v_3, \dots, v_n\}$ dan edges (E) yang menghubungkan antar node dalam V , dimana $E = \{e_1(v_1, v_2), e_2(v_2, v_3), e_3(v_1, v_3), \dots, e_n\}$. Beberapa jenis graf antara lain: 1) *directed-graph*, 2) *undirected-graph*, dan 3) *multi-graph*.

Beberapa istilah khusus pada *directed graph* antara lain : 1) *Successor*, yaitu node x dihubungkan langsung ke node y (x, y) dan y merupakan *successor* dari x , 2) *Predecessor*, node x dihubungkan langsung ke node y (x, y) dan x adalah *predecessor* dari y . Contoh graf berarah dengan 1 *successor* dan 1 *predecessor* ditunjukkan pada Gambar 4.



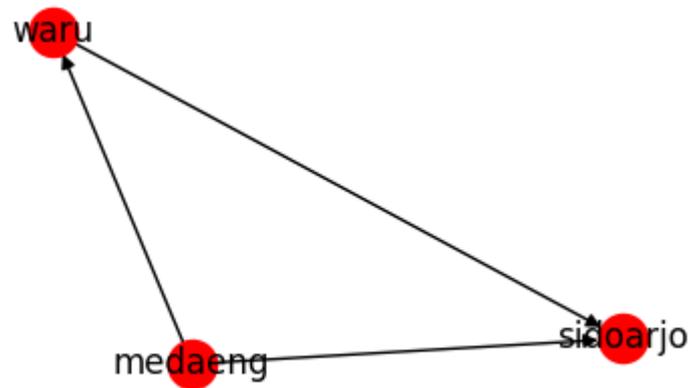
Gambar 4. Contoh *Directed Graph* dengan 1 *Successor* dan 1 *Predecessor*

Pada penelitian ini, setiap lokasi kejadian dinyatakan sebagai Node. Sedangkan edge merupakan relasi antar node yang didefinisikan menjadi 4 (empat) jenis relasi yaitu: 1) Highway-Position, 2) Street - Place, 3) StartingPoint - Destination, 4) Other. Setiap relasi tersebut memiliki perlakuan yang berbeda. Perlakuan setiap jenis relasi ditunjukkan pada Tabel 2.

Tabel 2 Perlakuan untuk masing-masing Relasi pada Graf

No	Relasi	Perlakuan
1	Highway-Position	Penggabungan Node antara Lokasi jalan Tol dengan Posisinya. Dan menghapus node posisi
2	Street - Place	Penggabungan Node antara

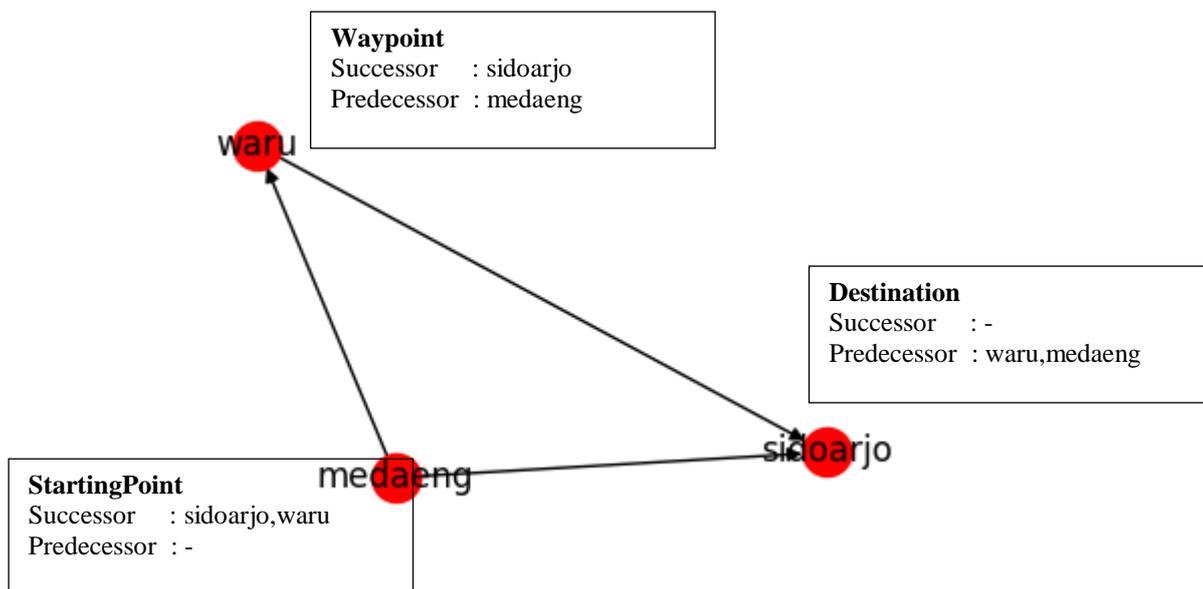
		Lokasi dengan Tempat. Dan menghapus node Tempat
3	StartingPoint - Destination	Node startingPoint dihubungkan dengan Edge ke node Destination
4	Other	Tidak ada Relasi



Gambar 5 Contoh Konversi Hasil Ekstraksi Relasi ke Bentuk Graf

Node dan edge yang sudah didapat akan dikonversi ke dalam bentuk graf yaitu directed-graph. Contoh hasil ekstraksi relasi jika diubah ke bentuk graf seperti pada Gambar 5. Hal ini dilakukan untuk mendapatkan node sebagai titik kejadian jika node tersebut berupa node tunggal, dan node sebagai titik awal bermula dari kejadian jika node tersebut memiliki hubungan dengan node yang lain. Jenis node yang berhubungan dengan node lainnya dibagi menjadi tiga jenis node yaitu: 1) node sebagai *startingpoint*, 2) node sebagai *destination*, 3) node sebagai *waypoint*.

Untuk menentukan siapa node yang menjadi startingpoint, destination, dan waypoint, digunakan konsep graf dengan menganalogikan *startingpoint* merupakan root dari sebuah graf, *destination* sebagai node akhir dari cabang graf, dan *waypoint* adalah node yang menghubungkan antara root dengan node akhir. Sebuah node dapat dijadikan sebagai root atau *startingpoint* jika node tersebut tidak memiliki *predecessor* dan sebaliknya memiliki *successor*. Kemudian untuk menentukan node sebagai *destination* adalah memilih node yang tidak memiliki *successor* namun memiliki *predecessor*. Sedangkan node yang mempunyai *predecessor* dan *successor* adalah merupakan *waypoint*. Node yang sudah dikenali dan diketahui jenis nodenya maka akan disimpan ke dalam database sebagai inputan pada tahap visualisasi data. Contoh hasil proses penentuan startingpoint, destination, dan waypoints ditunjukkan pada Gambar 6.



Gambar 6 Contoh Konversi Hasil Ekstraksi Relasi ke Bentuk Graf setelah melalui Proses Penentuan Startingpoint, Destination, dan Waypoint

HASIL

Uji coba dilakukan dengan menggunakan data tweet yang sudah tersimpan di database. Adapun batasan dalam uji coba ini adalah hanya menggunakan tweet yang berasal dari kota Surabaya. Pengujian dibagi menjadi dua yaitu uji kehandalan sistem dalam melakukan standarisasi entitas nama lokasi dan uji kehandalan sistem dalam konversi relasi antar lokasi kejadian ke dalam bentuk graf. Skenario pertama adalah menguji sistem menggunakan inputan tweet yang memiliki kesalahan penulisan entitas lokasi. Skenario kedua adalah menguji sistem dalam mengkonversi informasi ke dalam bentuk graf. Kedua Skenario tersebut dijabarkan sebagai berikut.

1. Ujicoba Standardisasi Entitas Lokasi

Pada ujicoba ini, data yang diproses adalah tweet informasi kejadian yang terdapat kesalahan penulisan entitas lokasi. Kesalahan penulisan nama tempat dalam sebuah tweet adalah hal yang wajar. Namun, dalam proses visualisasi nanti akan terjadi kendala geocoding oleh GoogleMaps karena nama lokasi tidak terdaftar atau dikenali sesuai standar penulisan nama tempat yang seharusnya. Beberapa tweet dengan kesalahan penulisan entitas nama lokasi ditunjukkan pada Tabel 3.

Tabel 3. Hasil Standardisasi Entitas Lokasi

Entitas Lokasi	Hasil Standardisasi
Tnjungan Plasa	Tunjungan Plaza
Surbaya	Surabaya
Krtajya	Kertajaya
Bnyu Urip	Banyu Urip

Dilain sisi, standarisasi entitas lokasi juga bisa justru mendapat hasil yang tidak sesuai. Hal ini bisa terjadi karena adanya penggunaan singkatan yang kurang tepat dalam penulisan nama lokasi. Selain itu, bisa juga dikarenakan penulisan nama lokasi sudah benar namun, karena tidak terdapat dalam kamus

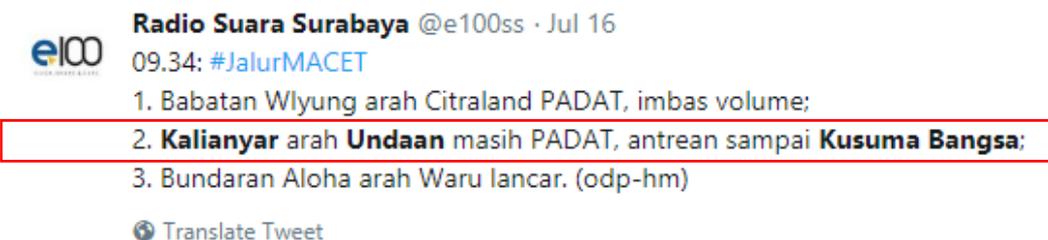
Gazetteer sehingga setelah melalui proses standardisasi nama, nama lokasi justru menjadi tidak sesuai dengan nama yang dimaksud dalam sebuah tweet. Hal tersebut dapat dilihat pada Tabel 4. Kesalahan ini terjadi karena pada penelitian ini hanya menggunakan *Gazetteer* wilayah kota Surabaya sehingga pencocokan nama akan dilakukan berdasarkan nilai *edit distance* terkecil.

Tabel 4 Contoh Kesalahan dalam Proses Standardisasi Nama *Gazetteer* dengan FASM

Teks Tweet	Hasil Standardisasi
lahan tebu ada api kebakaran di km 698 tol sumo dari jombang ke mojokerto	lahan tebu ada api kebakaran di km 698 tol sumo dari kembang ke mojokerto
arah probolinggo macet parah	arah pro lingga macet parah

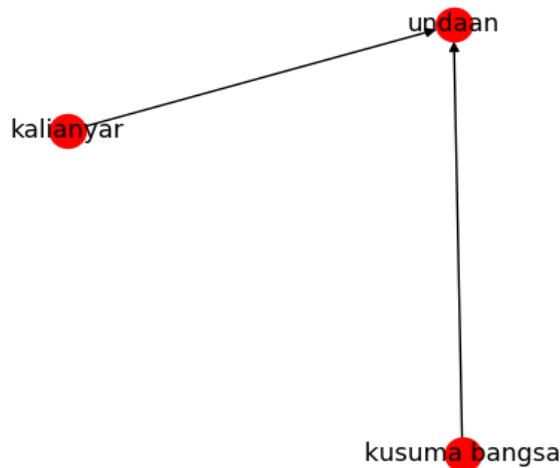
2. Ujicoba Konversi Relasi Antar Lokasi ke Bentuk Graf

Dalam satu tweet jika mempunyai lebih dari dua entitas maka akan memiliki lebih dari satu relasi, maka perlu didapatkan keterhubungan antar relasi yang terbentuk. keterhubungan antar relasi bisa didapatkan dengan mudah dengan mengubahnya ke bentuk directed-graf. Contoh tweet yang memiliki lebih dari satu relasi ditunjukkan pada Gambar 7. Pada gambar tersebut didapatkan hasil ekstraksi seperti pada Gambar 8. Penentuan startingpoint, destination, dan waypoints dari hasil ekstraksi dengan memanfaatkan graf seperti ditunjukkan pada Gambar 9.

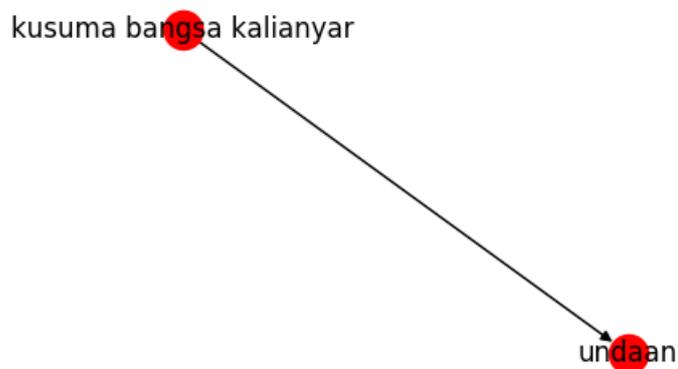


Gambar 7. Contoh Tweet Kejadian dengan Entitas Lokasi lebih dari Dua
`<e1>loc<e1> arah <e2>loc<e2> masih padat antrean sampai loc
 <e1>loc<e1> arah loc masih padat antrean sampai <e2>loc<e2>
 loc arah <e1>loc<e1> masih padat antrean sampai <e2>loc<e2>
 StartingPoint-Destination(e1,e2)
 [5. 3. 6.]`

Gambar 8. Hasil Merubah Teks Tweet ke dalam Struktur Jenis Relasi

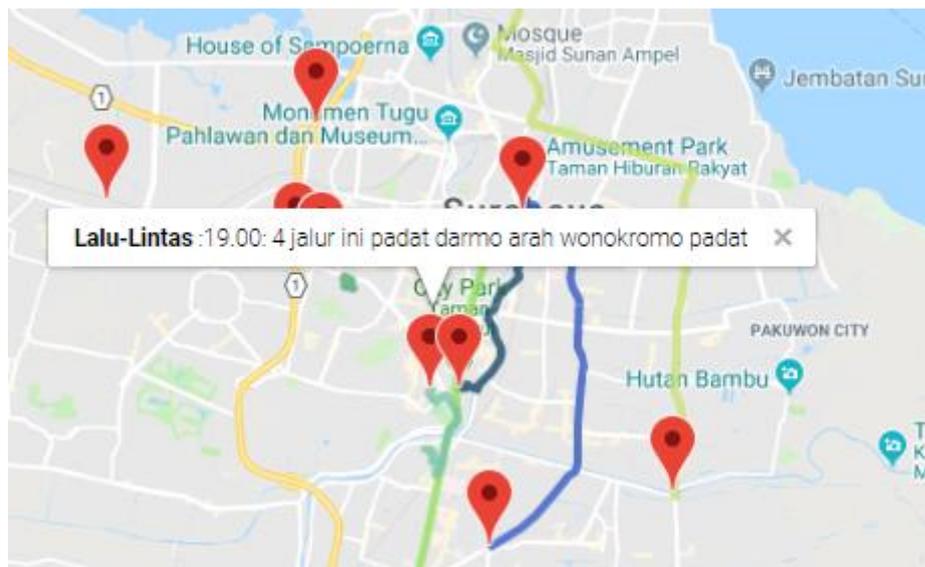


Gambar 9 Hasil Konversi Hasil Ekstraksi Relasi dalam Bentuk Graf



Gambar 10. Graf Hasil Penggabungan Node Kusuma Bangsa dan Kalianyar oleh Relasi Street-Place(e_1, e_2)

Dari Gambar 10 diterapkan perhitungan successor dan predecessor untuk menentukan node yang akan menjadi *startingpoint* dan *destination*. Berdasarkan hal tersebut dapat ditentukan bahwa kusuma bangsa kalianyar sebagai *startingpoint* dan undaaan sebagai *destination*. Hasil inilah yang digunakan sebagai inputan ke dalam GoogleMaps supaya informasi kejadian bisa dipetakan atau divisualisasikan secara tepat pada GoogleMaps. Hasil visualisasi informasi kejadian ditunjukkan pada Gambar 11.



Gambar 11. Hasil Visualisasi Informasi Kejadian dalam GoogleMaps

PEMBAHASAN

Berdasarkan hasil pengujian di atas yaitu uji standardisasi dan uji konversi lokasi kejadian dalam bentuk graf, dapat dilihat bahwa sistem dapat bekerja sesuai tujuannya dan mengatasi permasalahan yang telah disampaikan di awal. Pada proses standardisasi dapat terlihat kesalahan penulisan entitas lokasi dapat dibetulkan sesuai penulisan yang seharusnya di dalam Gazetteer. Sedangkan pada, penggunaan graf sebagai langkah memodelkan lokasi-lokasi kejadian berdasarkan relasinya menunjukkan hasil yang sangat memuaskan. Kerumitan dalam memahami relasi antar lokasi dapat dengan mudah ditampilkan dalam sebuah graf berarah.

KESIMPULAN

Pemetaan atau visualisasi lokasi kejadian pada Sistem Deteksi Kejadian dengan data Twitter sebagai inputannya dapat dengan mudah dilakukan dengan proses standardisasi entitas lokasi yang kemudian dilakukan konversi ke dalam bentuk graf berdasarkan relasi antar lokasi. Maka, jika terdapat beberapa lokasi yang saling terhubung maka akan terbentuk sebuah graf berarah atau *directed graph* yang terdapat lokasi sebagai *startingpoint*, *destination*, dan atau *waypoint* yang kemudian dijadikan sebagai inputan dalam proses visualisasi lokasi ke dalam *Google Maps*.

SARAN

Pada pengembangan selanjutnya untuk pemrosesan input data tweet bisa dilakukan dengan menerapkan model antrian dan pemrosesan paralel. Jadi, sistem akan memproses tweet berdasarkan antriannya. Hal ini guna mengantisipasi tidak diprosesnya data tweet oleh sistem dikarenakan banyaknya data tweet yang masuk secara bersamaan.

DAFTAR RUJUKAN

Garbe, W. (2015). Fast approximate string matching with large edit distances in Big Data. Retrieved March 25, 2018, from

<http://blog.faroo.com/2015/03/24/fast-approximate-string-matching-with-large-edit-distances/>

[Accessed: 25-Mar-2020]

- Gelernter, J., & Balaji, S. (2013). An algorithm for local geoparsing of microtext. *GeoInformatica*, 17(4), 635–667. <https://doi.org/10.1007/s10707-012-0173-8>
- Gu, Y., Qian, Z., & Chen, F. (2016). From Twitter to detector: Real-time traffic incident detection using social media data. *Transportation Research Part C: Emerging Technologies*, 67, 321–342. <https://doi.org/10.1016/j.trc.2016.02.011>
- Hagberg, A., Schult, D., & Swart, P. (2018). NetworkX Reference.
- Hasby, M., Khodra, M. L., & Purwarianti, A. (2013). Optimal Path Finding based on Traffic Information Extraction from Twitter. In *Prosiding International Conference on ICT for Smart Smart Society*.
- Li, J., He, Z., Plaza, J., Li, S., Chen, J., Wu, H., ... Liu, Y. (2017). Social Media: New Perspectives to Improve Remote Sensing for Emergency Response. *Proceedings of the IEEE*, 105(10), 1900–1912. <https://doi.org/10.1109/JPROC.2017.2684460>
- Putra, F., & Fatichah, C. (2018). Klasifikasi jenis kejadian menggunakan kombinasi NeuroNER dan Recurrent Convolutional Neural Network pada data Twitter, 4(2), 81–90.