Laporan Tugas Kelompok 1 - Graph Mining pada Keterhubungan Co-Starring Sinetron antar Aktor-Aktris Korea Selatan

Kelompok 2

Abdurrafi Arief - 1806205773
Naufal Hilmi Irfandi - 1806186673

Penambangan Data
Ganjil 2021/2022

Daftar Isi

Pendahuluan

Pada Tugas Kelompok 2 mata kuliah Penambangan Data, diberikan permasalahan untuk melakukan Graph Mining terhadap data kejadian ril. Permasalahan yang kelompok kami ambil yaitu Link Prediction terhadap hubungan Co-Starring antara Aktor-Aktris Korea Selatan. Output dari permasalahan ini merupakan edge-edge yang memiliki score terbesar untuk terjadinya keterhubungan di kedepannya. Kelompok kami menggunakan beberapa metode untuk menghitung score edge yang belum terbentuk.

Dataset yang Digunakan

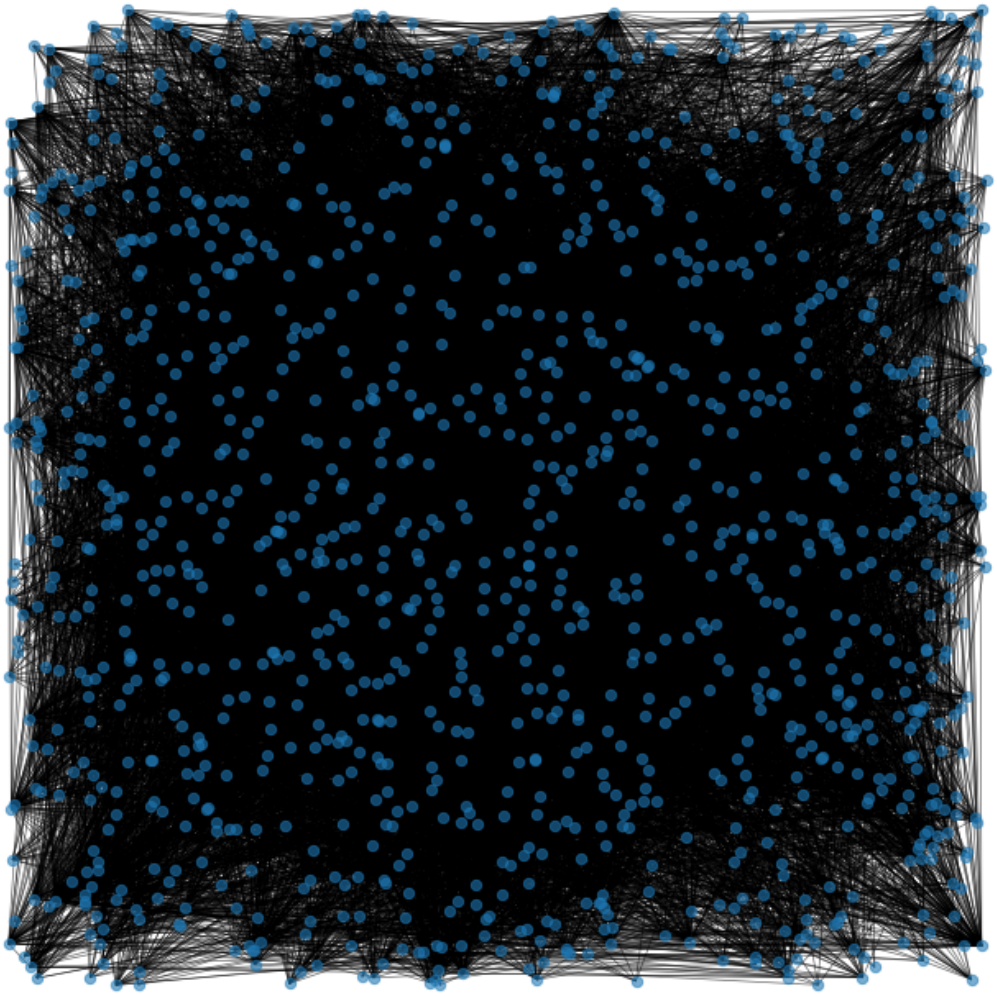
Dataset yang kami gunakan merupakan dataset yang kami ambil melalui sistem Wikidata dimana kami menggunakan query SPARQL untuk mendapatkan data sinetron Korea Selatan apa saja yang ada serta siapa saja Aktor-Aktris yang berperan pada drama tersebut. Berikut merupakan query SPARQL yang kami gunakan,

```
SELECT ?dramaLabel ?artistLabel
WHERE{
  ?a wdt:P31 wd:Q5398426;
     wdt:P495 wd:Q884;
     rdfs:label ?dramaLabel;
     wdt:P161 ?b .
  ?b rdfs:label ?artistLabel;
  MINUS{?a wdt:P136 wd:Q182415 .}
  FILTER(LANG(?dramaLabel) = "en")
  FILTER(LANG(?artistLabel) = "en")
}
```

Dari data yang kami dapatkan, kami berhasil mengambil 1317 Aktor dan Aktris yang memainkan sekitar 784 sinetron Korea Selatan. Pada query SPARQL diatas terdapat query FILTER, query ini bermaksud untuk melakukan filtrasi data dari Wikidata terhadap label yang mungkin muncul dikarenakan data yang di retrieve dari Wikidata memiliki label di berbagai bahasa. Kami juga memisahkan acara tv Korea Selatan yang bukan sinetron dengan query MINUS pada code SPARQL yang kami gunakan.

Data Preprocessing

Data yang sudah kami dapatkan, akan kami petakan ke bentuk adjacency list. Kami gunakan data adjacency list tadi untuk membentuk visualisasi Graph menggunakan library ‘networkx’ dimana library ini akan memberikan gambaran graf dan juga jumlah node serta edge dari graf. Berikut merupakan visualisasi graf yang kami buat,



Gambar 1 Visualisasi Graph

Setelah terbentuknya graf tersebut, didapatkan bahwa dari 1317 Aktor dan Aktris hanya 1230 Aktor dan Aktris yang memiliki keterhubungan dengan aktor dan aktris lain dimana 87 Aktor dan Aktrisnya tidak terdapat keterhubungan dengan aktor dan aktris lain. Graph tersebut juga memberikan informasi jumlah edge yang ada yaitu terdapat 13869 edge.

Dalam melakukan link prediction dibutuhkan list edge apa saja yang mungkin dapat dibentuk melalui data edge yang sudah terbentuk sebelumnya. Pencarian edge yang belum ada ini kami

lakukan dengan cara membuat adjacency matriks dari node-node yang ada lalu kami ambil hubungan node mana yang memiliki nilai 0. Dari proses ini kami mendapatkan node yang belum ada sebanyak 741966 edge yang belum ada pada graf.

Feature Engineering

Metode Implementasi

Implementasi yang kami lakukan untuk menentukan link prediction kami bagi menjadi dua implementasi. Implementasi pertama yaitu menggunakan teknik Non-Model Based seperti menghitung nilai similarity dengan Jaccard Similarity atau Common Neighbors lainnya. Implementasi kedua yaitu menggunakan Model Based seperti node2vec dalam menghitung link prediction pada graf yang kami miliki.

Non-Model Based

Implementasi Non-Model Based yang kami gunakan yaitu menggunakan dua tipe perhitungan yaitu Node Neighborhood-based Method dan Methods based on Paths Between Nodes. Untuk Node Neighborhood-based Method kami menggunakan empat perhitungan berbeda yaitu Common Neighbors, Jaccard Similarity, Adamic and Adar Measure, dan Preferential Attachment. Sedangkan untuk Methods based on Paths Between Nodes kelompok kami menggunakan perhitungan SimRank. Pada percobaan yang kami lakukan, kami hanya melakukan percobaan terhadap metode Node Neighborhood-based. Berikut merupakan pembahasan setiap metode pada perhitungan Node Neighborhood-based Method,

- Common Neighbors

  Perhitungan ini merupakan salah satu cara dalam menentukan nilai keterhubungan antara suatu node dengan node lain. Perhitungan ini melihat seberapa banyak tetangga dari suatu node yang beririsan dengan node yang ingin dihubungkan.

$$\sigma(x, y) = |N(x) \cap N(y)|$$

  Nilai dari jumlah intersect neighbors akan digunakan sebagai nilai penentu apakah suatu node x dengan node y dapat memiliki keterhubungan di kemudian hari.

Pada graf yang kami miliki, dari list of edge yang belum ada kami hitung menggunakan perhitungan ini dan mendapatkan hasil seperti berikut,

	Edge	Score
0	Um Hyo-sup - Kang Sin-il	20
1	Choi Woong - Im Se-mi	16
2	Yoo Yeon-seok - Choi Woong	16
3	Kim Soo-hyun - Chun Jung-myung	16
4	Um Hyo-sup - Jeong Man-sik	15
5	Kim Dong-gyun - Choi Woong	15
6	Chun Jung-myung - Kim Chang-wan	15
7	Kim Gap-su - Um Hyo-sup	14
8	Kim Hee-won - Kang Sin-il	14
9	Jin Kyeong - Park Yeong-gyu	14

Dari hasil diatas didapatkan informasi bahwa actor Um Hyo-sup menjadi Co-Starring dengan Kang Sin-il dan sebaliknya merupakan hubungan yang kemungkinan besar akan terhubung. Hal ini dipengaruhi dikarenakan aktor dan aktris yang intersect dari Um Hyo-sup dan Kang Sin-il memiliki nilai besar dibandingkan dengan kemungkinan edge lainnya.

- Jaccard Similarity

  Perhitungan ini memiliki cara yang hampir mirip dengan Common Neighbors namun terdapat nilai perata yang mana merupakan nilai dari gabungan neighbors antara dua node yang dianalisis. Nilai perata ini berfungsi untuk menegaskan kedekatan dan juga keterhubungan antar dua node. Teknik ini memberikan bobot untuk lebih menekankan bahwa bukan berarti jika dua node memiliki intersect yang banyak, node tersebut akan berteman dengan baik karena bisa saja teman dari masing-masing node sudah sangat banyak. Teknik ini juga melihat kedekatan yang lebih pasti karena akan melihat ruang neighbors yang lebih kecil namun intersectnya lebih besar. Perhitungan Jaccard Similarity dapat diperoleh dengan,

$$\sigma(x, y) = \frac{|N(x) \cap N(y)|}{|N(x) \cup N(y)|}$$

Nilai yang diperoleh akan menjadi tolak ukur pasangan node mana yang akan terbentuk edgenya. Perhitungan berikut kami implementasikan terhadap graf yang kami buat dan menghasilkan nilai berikut,

	Edge	Score
0	Nam Sang-mi - Baek Jin-hee	0.333333
1	Cha Hwa-yeon - Kim Yu-mi	0.333333
2	Baek Jin-hee - Jeon Mi-do	0.250000
3	Go Yoon - Seo Hyun-woo	0.250000
4	Kim Sang-joong - Jin Tae-hyun	0.250000
5	Jun Hyun-moo - Cho Kyuhyun	0.250000
6	Eric Mun - Jun Hyo-seong	0.250000
7	Won Bin - Kim Seong-su	0.250000
8	Kang Ho-dong - Jun Hyun-moo	0.250000
9	Kim Suk-hoon - Park Ha-na	0.250000

Dari hasil perhitungan Jaccard Similarity pada graf, didapatkan nilai Jaccard Similarity terbesar yaitu edge antara Nam Sang-mi dengan Baek Jin-hee dan edge antara Cha Hwa-yeon dengan Kim Yu-mi dengan nilai 0.33 untuk terjadinya pembentukan edge. Nilai 0.33 ini didapatkan dari jumlah intersect dari Cha Hwa-yeon dan Kim Yu-mi memiliki jumlah yang banyak namun variasi neighbors Cha Hwa-yeon dan Kim Yu-mi tidak berjumlah banyak.

- Adamic and Adar Measure

  Perhitungan yang dapat memprediksi link dari sebuah dua node dengan cara melihat penjumlahan dari inverse logarithmic tetangga yang merupakan intersection dari dua node yang di prediksi. Adamic and Adar Measure ini melihat bahwa antara node akan memiliki hubungan ketika node yang intersect tidak memiliki banyak teman kecuali dua node yang dianalisis. Berikut merupakan perhitungan Adamic and Adar Measure,

$$\sigma(x, y) = \sum_{z \in N(x) \cap N(y)} \frac{1}{\log |N(z)|}$$

Perhitungan ini akan diimplementasikan terhadap graf yang sudah dibuat dan menghasilkan data berikut,

	Edge	Score
0	Um Hyo-sup - Kang Sin-il	4.762552
1	Choi Woong - Im Se-mi	3.960515
2	Kim Soo-hyun - Chun Jung-myung	3.832310
3	Yoo Yeon-seok - Choi Woong	3.734029
4	Um Hyo-sup - Jeong Man-sik	3.633643
5	Chun Jung-myung - Kim Chang-wan	3.575241
6	Kim Dong-gyun - Choi Woong	3.513405
7	Chun Jung-myung - Moon Chae-won	3.509008
8	Moon Chae-won - Yoo In-na	3.482441
9	Daniel L - Moon Chae-won	3.470074

Sama seperti perhitungan Common Neighbors, hubungan antara Um Hyo-sup dengan Kang Sin-il memiliki nilai terbesar yaitu 4.762552. Hal ini dikarenakan terdapat banyak neighbors yang beririsan diantara Um Hyo-sup dengan Kang Sin-il sehingga jumlah nilai yang dijumlah juga meningkat.

- Preferential Attachment

  Perhitungan ini melihat seberapa padat dari neighbors yang dimiliki oleh dua node. Kepadatan ini akan menjadi bahan penentuan nilai dengan cara mengalikan jumlah neighbors node x dengan jumlah neighbors node y. Perkalian ini terjadi dikarenakan jika suatu node itu memiliki banyak teman, kemungkinan besar node tersebut akan memiliki hubungan juga dengan node lain yang juga memiliki banyak teman. Berikut merupakan perhitungan Preferential Attachment,

$$\sigma(x, y) = |N(x)| \cdot |N(y)|$$

Perhitungan ini akan diimplementasikan terhadap graf yang sudah dibuat dan menghasilkan data berikut,

	Edge	Score
0	Choi Woong - Chun Jung-myung	21978
1	Jung Dong-hwan - Chun Jung-myung	20394
2	Um Hyo-sup - Kim So-hyun	20160
3	Chun Jung-myung - Moon Chae-won	19800
4	Chun Jung-myung - Kwak Dong-yeon	19008
5	Kim Soo-hyun - Chun Jung-myung	18414
6	Jo Jung-suk - Chun Jung-myung	18414
7	Chun Jung-myung - Na Young-hee	18018
8	Um Hyo-sup - Kang Sin-il	17640
9	Jo Seong-ha - Chun Jung-myung	17226

Dari data hasil diatas terlihat bahwa Choi Woong dengan Chun Jung-myung memiliki nilai tertinggi dikarenakan jumlah tetangga dari Choi Woong dan Chun Jung-myung banyak yang membuat nilai pasangan ini menjadi nilai tertinggi.

Model Based

Untuk metode *model based* kami menggunakan metode Node2Vec dengan *logistic regression.* Jadi pertama node dari graf yang mau dipelajari diubah menjadi embeddings. Embeddings itu didapatkan dengan algoritma node2vec. Kemudian embeddings akan digunakan untuk melatih modelnya. Library yang digunakan untuk mengolah graf disini adalah StellarGraph.

Proses membuat modelnya sebagai berikut:

1. Pertama membagi grafnya menjadi *test graf* dan *test set*. Test graf adalah graf terreduksi yang didapatkan dari mengurangi edge test set dari graf. Test set berisi beberapa sampel positif edge (edge yang terhubung) dan negative edge (edge yang tidak terhubung) yang diambil dari graf.
2. Kemudian kami membuat *train graf* dari test graf dengan menggunakan fungsi *edgesplitter* dari stellargraf dan melakukan train/test split pada test graf. Untuk mendapatkan train graf, training set, dan set contoh link untuk seleksi model.

3. Setelah itu kami memanfaatkan library word2vec untuk menghitung embeddings dari setiap node. Pertama yang dilakukan adalah melakukan *random walks* pada graf untuk mendapatkan pasangan konteks. Pasangan konteks yang didapatkan digunakan untuk melatih model Word2Vec.
4. Untuk mendapatkan embeddings untuk edge positif dan negatif, diterapkan binary operator pada hasil embeddings node source dan target pada sampel edge. Untuk binary operator yang digunakan ada 4, yaitu:
    - operator hadamard: $u * v$
    - operator l1: $|u - v|$
    - operator l2: $(u - v)^2$
    - operator avg: $\frac{u+v}{2.0}$
5. Kemudian dengan embeddings yang didapatkan untuk edge positif dan negatif dari masing-masing operator, dilakukan training pada *logistic regression classifier* untuk memprediksi suatu nilai biner yang mengindikasi terdapat edge atau tidak. 1 menandakan ada edge. 0 menandakan tidak ada edge.
6. Setelah itu setiap classifier diuji coba dengan Train Graph kemudian dicari classifier terbaik.
7. Classifier terbaik kemudian digunakan pada test graf untuk mendapatkan skor evaluasi

Hasil evaluasi yang didapatkan dari langkah-langkah di atas sebagai berikut:

name	ROC AUC score
operator_hadamard	0.949446
operator_l1	0.967017
operator_l2	0.967840
operator_avg	0.699074

Berdasarkan percobaan diatas operator terbaik adalah operator l2 dengan skor ROC AUC 0.967840. Model classifier dengan operator l2 yang akan digunakan untuk memprediksi edge-edge tidak terhubung. Model yang kami buat, dirasa cukup baik karena memiliki skor ROC AUC yang relatif tinggi.

Analisis Hasil Percobaan

Setelah mendapatkan hasil setiap percobaan, dilakukan perbandingan hasil antara setiap metode non-model dengan metode model pada 10 nilai teratas. Didapatkan hasil sebagai berikut:

Perbandingan Jaccard Similarity dengan Node2Vec

	Edge	Score	Hasil Prediksi
0	Cha Hwa-yeon - Kim Yu-mi	0.333333	0
1	Baek Jin-hee - Nam Sang-mi	0.333333	1
2	Go Yoon - Seo Hyun-woo	0.250000	1
3	Yoo Se-yoon - Kang Ho-dong	0.250000	1
4	Jun Hyo-seong - Eric Mun	0.250000	1
5	Park Ha-na - Kim Suk-hoon	0.250000	1
6	Jun Hyun-moo - Song Min-ho	0.250000	1
7	Jun Hyun-moo - Kang Ho-dong	0.250000	1
8	Jun Hyun-moo - Cho Kyuhyun	0.250000	1
9	Kim Hee-ae - Jin Tae-hyun	0.250000	1
10	Yoo Se-yoon - Song Min-ho	0.250000	1

Pada tabel diatas terdapat 10 edge dengan nilai Jaccard Similarity tertinggi dan di kolom kanan adalah hasil prediksi edge oleh model kami.

Dari hasil perbandingan dapat dilihat apabila 9 dari 10 edge teratas yang didapatkan oleh Jaccard Similarity diprediksi akan memiliki edge oleh model yang kami buat. Jika berasumsi model yang kami buat adalah paling mendekati kebenaran bisa dibilang apabila metode Jaccard Similarity merupakan metode yang cukup baik untuk memprediksi edge yang terhubung.

Perbandingan Adamic and Adar Measure dengan Node2Vec

	Edge	Score	Hasil Prediksi
0	Um Hyo-sup - Kang Sin-il	4.762552	0
1	Choi Woong - Im Se-mi	3.960515	1
2	Kim Soo-hyun - Chun Jung-myung	3.832310	0
3	Choi Woong - Yoo Yeon-seok	3.734029	1
4	Jeong Man-sik - Um Hyo-sup	3.633643	0
5	Kim Chang-wan - Chun Jung-myung	3.575241	0
6	Choi Woong - Kim Dong-gyun	3.513405	0
7	Moon Chae-won - Chun Jung-myung	3.509008	0
8	Yoo In-na - Moon Chae-won	3.482441	0
9	Daniel L - Moon Chae-won	3.470074	0
10	Kim So-hyun - Kim Gap-su	3.391230	0

Pada tabel di atas terdapat 10 nilai tertinggi dari Adamic and Adar measure dengan hasil prediksi dari model kami di kolom paling kanan.

Dari hasilnya terlihat apabila 8 dari 10 data edge yang memiliki nilai tinggi, diprediksi tidak akan memiliki edge oleh model kami. Apabila berasumsi model kami lebih mendekati kebenaran, maka dapat dikatakan apabila Adamic and Adar measure kurang tepat untuk memprediksi hubungan artis dengan graf yang kita miliki.

Perbandingan Preferential Attachment dan Node2Vec

	Edge	Score	Hasil Prediksi
0	Choi Woong - Chun Jung-myung	21978	0
1	Jung Dong-hwan - Chun Jung-myung	20394	0
2	Kim So-hyun - Um Hyo-sup	20160	0
3	Moon Chae-won - Chun Jung-myung	19800	0
4	Kwak Dong-yeon - Chun Jung-myung	19008	0
5	Kim Soo-hyun - Chun Jung-myung	18414	0
6	Jo Jung-suk - Chun Jung-myung	18414	0
7	Na Young-hee - Chun Jung-myung	18018	0
8	Um Hyo-sup - Kang Sin-il	17640	0
9	Jo Seong-ha - Chun Jung-myung	17226	0
10	Yoo In-na - Chun Jung-myung	16830	0

Pada tabel diatas terdapat 10 edge dengan nilai tertinggi dari metode *Preferential Attachment*. Dengan hasil prediksi model kami di kolom paling kanan.

10 dari 10 data tersebut diprediksi oleh model kami tidak akan memiliki edge. Dengan asumsi model kami merupakan hasil yang mendekati paling tepat, maka dapat disimpulkan apabila metode *Preferential Attachment* kurang tepat untuk memprediksi hubungan artist pada graf kami.

Kesimpulan

Memprediksi apakah dua artis korea akan *co-star* atau tidak dapat dilakukan dengan berbagai cara. Kami memprediksinya menggunakan metode *similarity based* dan menggunakan model Node2Vec. Hasil model yang kami buat dengan Node2Vec dan *Logistic Regression Classifier* memiliki skor ROC AUC cukup baik, yaitu 0.967840. Bisa dikatakan

model yang kami buat cukup baik. Kemudian kami membandingkan hasil prediksi model dengan hasil metode similarity.

Dari analisa yang kami dapatkan edge dengan nilai tertinggi dari *Adamic and Adar Measure, preferential attachment* sebagian besar tidak diprediksi akan memiliki edge oleh model kami. Sementara edge dengan nilai tertinggi dari *Jaccard Similarity* sebagian besar diprediksi akan memiliki edge oleh model kami. Dengan asumsi, hasil prediksi model lebih tepat, maka dapat disimpulkan bahwa memprediksi apakah 2 artis korea akan menjadi *co-star* di masa depan, metode *Jaccard Similarity* lebih cocok dibandingkan dengan metode *Adamic and Adar Measure*, dan *Preferential Attachment*.

Terakhir, penulis menyarankan untuk hasil analisa yang lebih baik, dapat dilakukan percobaan dengan lebih dari 1 model untuk saling membandingkan model.

Referensi

1. Link prediction with Node2Vec — StellarGraph 1.2.1 documentation
2. Loading data into StellarGraph from Pandas — StellarGraph 1.2.1 documentation