Nama : Nadia Nur Oktaviani Sukma

Nim : 202010370311320

UTS Analisis Big Data – PCAP

Analisis Pola Koneksi TCP Menggunakan XGBoost

Dikumpulkan minggu depan di hari UTS BIG DATA (google classroom):

1. Find at least A REFERENCE INTERNATIONAL JOURNAL about network dataset or IoT modelling with machine learning/ deep learning

2. Create dataset (extraction based on your knowledge about PCAP - network dataset) —> ALL OF DATA AMOUNT, not recommendation using sample.

3. Create Summary Data and Exploratory Data Analysis and describe clearly.

4. Create machine learning model/ deep learning model from your dataset.

5. Describe clearly step by step in your modelling (preprocessing, what model that you should use, evaluation)

6. Give conclusion about model and recommendation.

7. Collect in PDF. This PDF must include:

   1. Explanation about extraction of dataset. Why and what data that you use from PCAP dataset.

   2. Your dataset sample (head or tail dataset)

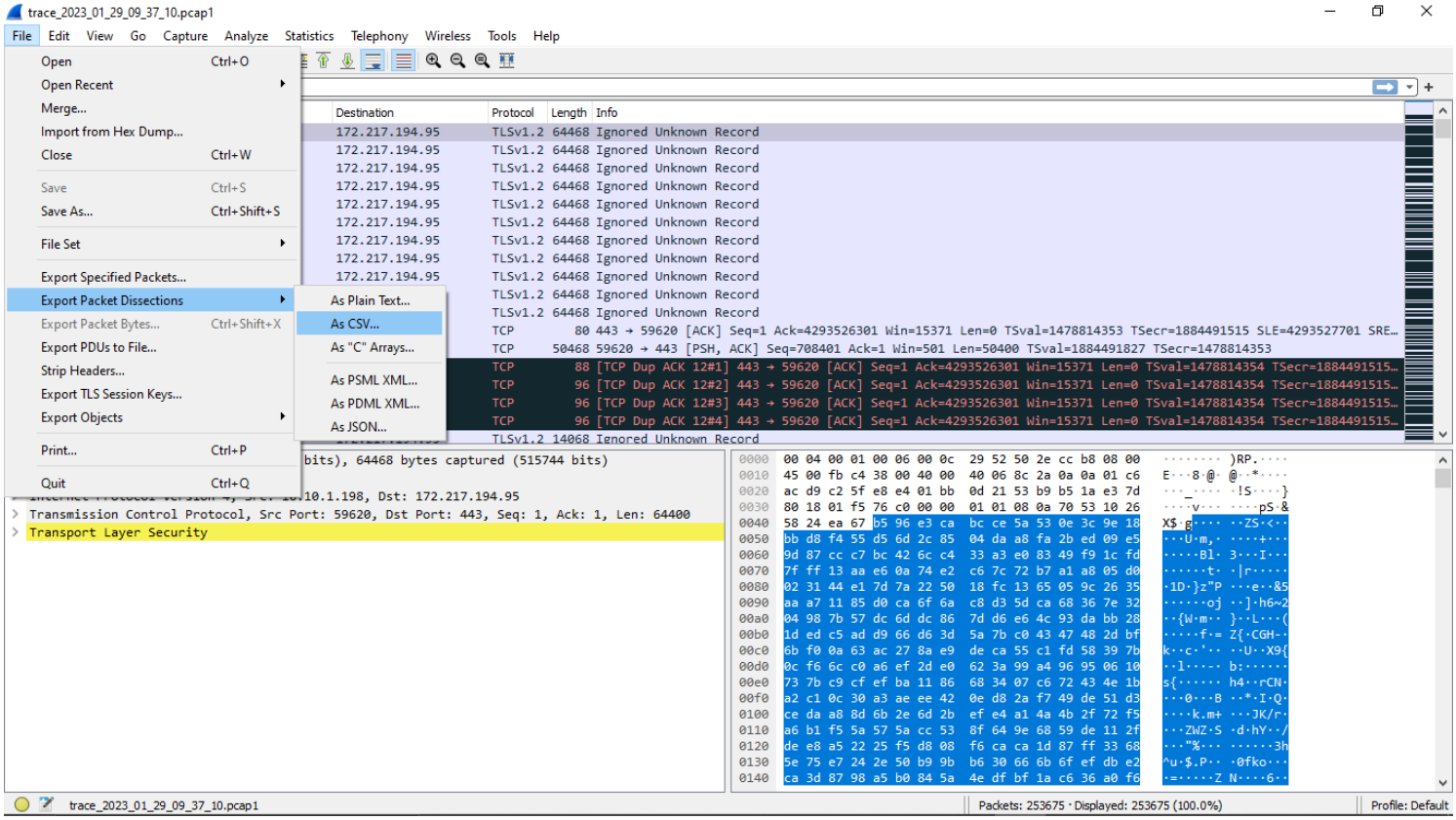
   3. Explanation about modelling (preprocessing until evaluation)

   4. Conclusion and recommendation

   5. Source code

   6. JOURNAL REFERENCE that you used (link journal and screenshot the title, author, and abstract)

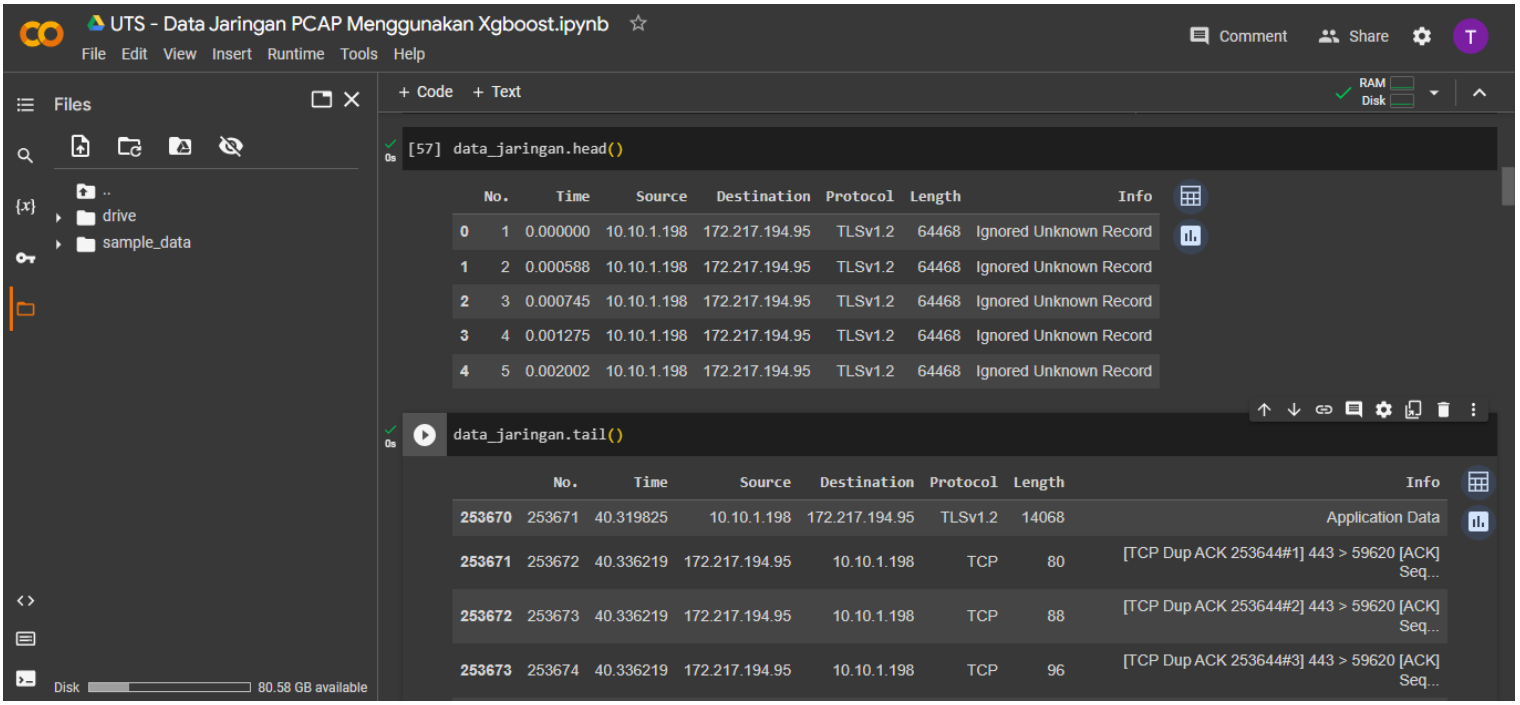
1. Penjelasan Ekstraksi Dataset



Disini saya melakukan ektraksi dataset dengan bantuan wireshark. Setelah memasukkan data PCAP ke Wireshark, saya melakukan ekspor dari analisis paket (packet dissections) yang dilakukan oleh Wireshark. Ekspor ini dilakukan dalam format CSV (Comma-Separated Values), yang merupakan format yang umum digunakan untuk menyimpan data dalam bentuk tabel terstruktur. Penulis menyatakan bahwa mereka telah memilih hasil analisis ini untuk dijadikan sebagai dataset yang akan disimpan dalam format CSV.

2. **Sampel dataset Anda (bagian awal atau akhir dataset)**

- 5 data teratas dan 5 data terbawah sebelum di pre-processing



UTS - Data Jaringan PCAP Menggunakan Xgboost.ipynb

File Edit View Insert Runtime Tools Help

Comment Share

Files

+ Code + Text

RAM Disk

drive

sample_data

Disk 80.58 GB available

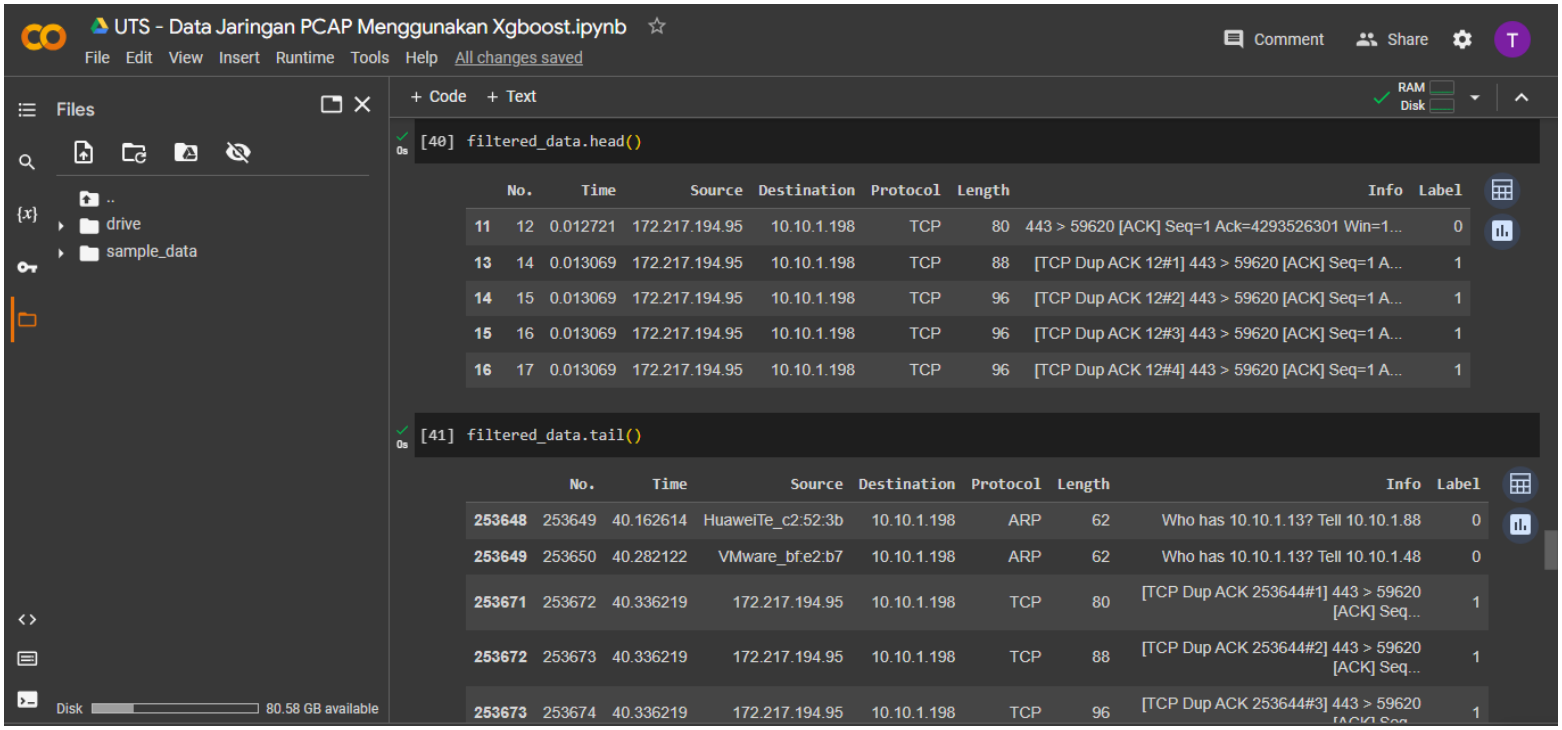
```
[57] data_jaringan.head()
```

	No.	Time	Source	Destination	Protocol	Length	Info
0	1	0.000000	10.10.1.198	172.217.194.95	TLSv1.2	64468	Ignored Unknown Record
1	2	0.000588	10.10.1.198	172.217.194.95	TLSv1.2	64468	Ignored Unknown Record
2	3	0.000745	10.10.1.198	172.217.194.95	TLSv1.2	64468	Ignored Unknown Record
3	4	0.001275	10.10.1.198	172.217.194.95	TLSv1.2	64468	Ignored Unknown Record
4	5	0.002002	10.10.1.198	172.217.194.95	TLSv1.2	64468	Ignored Unknown Record

```
data_jaringan.tail()
```

	No.	Time	Source	Destination	Protocol	Length	Info
253670	253671	40.319825	10.10.1.198	172.217.194.95	TLSv1.2	14068	Application Data
253671	253672	40.336219	172.217.194.95	10.10.1.198	TCP	80	[TCP Dup ACK 253644#1] 443 > 59620 [ACK] Seq...
253672	253673	40.336219	172.217.194.95	10.10.1.198	TCP	88	[TCP Dup ACK 253644#2] 443 > 59620 [ACK] Seq...
253673	253674	40.336219	172.217.194.95	10.10.1.198	TCP	96	[TCP Dup ACK 253644#3] 443 > 59620 [ACK] Seq...

- 5 data teratas dan 5 data terbawah sesudah di pre-processing



UTS - Data Jaringan PCAP Menggunakan Xgboost.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Files

+ Code + Text

RAM Disk

drive

sample_data

Disk 80.58 GB available

```
[40] filtered_data.head()
```

	No.	Time	Source	Destination	Protocol	Length	Info	Label
11	12	0.012721	172.217.194.95	10.10.1.198	TCP	80	443 > 59620 [ACK] Seq=1 Ack=4293526301 Win=1...	0
13	14	0.013069	172.217.194.95	10.10.1.198	TCP	88	[TCP Dup ACK 12#1] 443 > 59620 [ACK] Seq=1 A...	1
14	15	0.013069	172.217.194.95	10.10.1.198	TCP	96	[TCP Dup ACK 12#2] 443 > 59620 [ACK] Seq=1 A...	1
15	16	0.013069	172.217.194.95	10.10.1.198	TCP	96	[TCP Dup ACK 12#3] 443 > 59620 [ACK] Seq=1 A...	1
16	17	0.013069	172.217.194.95	10.10.1.198	TCP	96	[TCP Dup ACK 12#4] 443 > 59620 [ACK] Seq=1 A...	1

```
[41] filtered_data.tail()
```

	No.	Time	Source	Destination	Protocol	Length	Info	Label
253648	253649	40.162614	HuaweiTe_c2:52:3b	10.10.1.198	ARP	62	Who has 10.10.1.13? Tell 10.10.1.88	0
253649	253650	40.282122	VMware_bf:e2:b7	10.10.1.198	ARP	62	Who has 10.10.1.13? Tell 10.10.1.48	0
253671	253672	40.336219	172.217.194.95	10.10.1.198	TCP	80	[TCP Dup ACK 253644#1] 443 > 59620 [ACK] Seq...	1
253672	253673	40.336219	172.217.194.95	10.10.1.198	TCP	88	[TCP Dup ACK 253644#2] 443 > 59620 [ACK] Seq...	1
253673	253674	40.336219	172.217.194.95	10.10.1.198	TCP	96	[TCP Dup ACK 253644#3] 443 > 59620	1

3. Penjelasan tentang pemodelan (preprocessing hingga evaluasi)

Dalam proses menganalisis Pola Koneksi TCP dengan Fokus pada TCP Dup ACK (Label 1) dan Non-Dup ACK (Label 0). Tujuan analisis ini adalah untuk menganalisis pola koneksi TCP antara Sumber (Source) dan Destinasi (Destination) dengan fokus pada koneksi yang memiliki Label 0 (tidak ada TCP Dup ACK) dan Label 1 (TCP Dup ACK). Memahami perbedaan karakteristik waktu (Time), panjang koneksi (Length), dan informasi terkait (Info) antara koneksi dengan Label 0 dan Label 1. Sebelum melakukan filtered dataset, saya sudah melakukan satu preprocessing untuk dataset seperti yang sudah dijelaskan di atas. Kemudian ada beberapa hal yang saya lakukan dari pre preprocessing hingga evaluasi dan menampilkan akurasi, diantaranya:

a. Mengelompokkan Data

Melalukan Group_data untuk melakukan pengelompokan atau grouping pada DataFrame filtered_data berdasarkan munculnya string "TCP Dup ACK" dalam kolom 'Info', kemudian Setiap grup melibatkan baris-baris yang memiliki "TCP Dup ACK" berturut-turut, Dimana tujuan potensial dari pengelompokan ini adalah memudahkan analisis atau statistik tambahan yang spesifik terhadap setiap grup koneksi yang mengandung "TCP Dup ACK".

b. Menambah Label 1 dan 0

Menambahkan Label 1 dan 0 pada DataFrame filtered_data dilakukan dengan tujuan memberikan label (nilai 0 atau 1) pada setiap baris dalam DataFrame tersebut. Penetapan nilai label ini bergantung pada apakah string 'TCP Dup ACK' terdapat dalam kolom 'Info' pada setiap baris data. Dengan adanya label tersebut, kita dapat melakukan analisis lebih lanjut atau pemodelan untuk memahami pola dan sifat khusus dari koneksi jaringan yang memiliki ciri tersebut.Dup ACK'.

c. Label Encoding

Label encoding adalah Proses encoding ini mengubah nilai-nilai dalam kolom-kolom tersebut dari bentuk teks menjadi bentuk angka sehingga dapat digunakan dalam pemodelan atau analisis data. Disini saya mengubah semua atribut yang bertipe data kategori menjadi tipe data numerik, jika tidak data yang akan kita olah tidak bisa dilakukan karena masih ada tipe data yang tidak numerik.

d. Membagi Dataset menjadi Fitur dan Target

Dalam proses membagi dataset menjadi fitur (features) dan target (label) adalah salah satu langkah penting dalam pemodelan data. Proses ini dilakukan agar Anda dapat melatih model untuk memahami hubungan antara fitur-fitur tertentu dan label atau target yang akan diprediksi oleh model. Dalam proses kali ini Target yang digunakan adalah 'Label' dan selain label merupakan features

e. Pemisahan Train dan Test

Pemisahan data menjadi train dan test digunakan untuk membagi dataset menjadi data pelatihan (training data) dan data uji (testing data) menggunakan fungsi train_test_split dari Scikit-Learn. Pemisahan ini penting untuk mengukur kinerja model dan menghindari overfitting. Data pelatihan digunakan untuk melatih model, sedangkan data uji digunakan untuk menguji kinerja model dan mengukur sejauh mana model dapat melakukan prediksi yang akurat pada data yang belum pernah dilihat.

f. Melatih Model XGBoost

Dalam analisis kali ini saya menggunakan metode XGBoost. Setelah membuat objek model dengan konfigurasi tersebut, yaitu melatih model menggunakan data pelatihan yang telah dibagi sebelumnya. Disini saya akan menggunakan metode.fit() untuk melakukan pelatihan. Dengan melatih model menggunakan data pelatihan, dapat memberikan model informasi yang diperlukan untuk memahami pola dalam data dan dapat digunakan untuk melakukan prediksi pada data uji atau data baru yang belum pernah dilihat.

Setelah itu melakukan prediksi menggunakan model klasifikasi yang telah Anda latih sebelumnya (dalam hal ini, model Random Forest) pada data uji. Setelah Anda memiliki hasil prediksi (y_pred), Anda dapat melanjutkan dengan berbagai metrik evaluasi untuk mengukur kinerja model, seperti akurasi, presisi, recall, F1-score, dan lainnya.

g. Evaluasi Model

Dalam evaluasi model menggunakan XGBoost untuk menganalisis pola koneksi TCP antara Sumber (Source) dan Destinasi (Destination), fokus diberikan pada koneksi yang memiliki Label 0 (tidak ada TCP Dup ACK) dan Label 1 (TCP Dup ACK). Tujuan evaluasi adalah memahami perbedaan karakteristik waktu (Time), panjang koneksi

(Length), dan informasi terkait (Info) antara koneksi dengan Label 0 dan Label 1. Hasil evaluasi yang diinginkan adalah mencapai akurasi sebesar 100% atau 1.00, menunjukkan bahwa model secara tepat mengidentifikasi dan memisahkan koneksi dengan Label 0 dan Label 1 dengan sempurna dan juga karna klasifikasinya 100 persen bener mengenai jaringan yg sampai ke alamat atau tidak.

4. **Kesimpulan dan Rekomendasi**

Proses analisis pola koneksi TCP dengan fokus pada TCP Dup ACK (Label 1) dan Non-Dup ACK (Label 0) dilakukan dengan langkah-langkah yang terinci. Ekstraksi dataset dilakukan menggunakan Wireshark, diikuti oleh preprocessing data, pengelompokan data, penambahan label 1 dan 0, label encoding, pemisahan dataset menjadi fitur dan target, serta pemisahan data pelatihan dan pengujian. Model XGBoost digunakan untuk melatih dan menguji data, dengan hasil evaluasi yang diinginkan adalah akurasi sebesar 100%, menunjukkan kemampuan model dalam mengidentifikasi dan memisahkan koneksi dengan Label 0 dan Label 1 secara sempurna.

5. **Kode Sumber**

https://colab.research.google.com/drive/1B2Jb4Dqkwry0kZRnV2JUl1x-GDr0ozZH?usp=sharing

6. **Reference Jurnal**

Link Reference:

https://jis-eurasipjournals.springeropen.com/articles/10.1186/s13635-023-00141-4

Hu *et al.*
EURASIP Journal on Information Security (2023) 2023:6
https://doi.org/10.1186/s13635-023-00141-4

EURASIP Journal on Information Security

RESEARCH Open Access

Network traffic classification model based on attention mechanism and spatiotemporal features

Feifei Hu[1], Situo Zhang[1], Xubin Lin[1], Liu Wu[1], Niandong Liao[2*] and Yanqi Song[2]

Abstract

Traffic classification is widely used in network security and network management. Early studies have mainly focused on mapping network traffic to different unencrypted applications, but little research has been done on network traffic classification of encrypted applications, especially the underlying traffic of encrypted applications. To address the above issues, this paper proposes a network encryption traffic classification model that combines attention mechanisms and spatiotemporal features. The model firstly uses the long short-term memory (LSTM) method to analyze continuous network flows and find the temporal correlation features between these network flows. Secondly, the convolutional neural network (CNN) method is used to extract the high-order spatial features of the network flow, and then, the squeeze and excitation (SE) module is used to weight and redistribute the high-order spatial features to obtain the key spatial features of the network flow. Finally, through the above three stages of training and learning, fast classification of network flows is achieved. The main advantages of this model are as follows: (1) the mapping relationship between network flow and label is automatically constructed by the model without manual intervention and decision by network features, (2) it has strong generalization ability and can quickly adapt to different network traffic datasets, and (3) it can handle encrypted applications and their underlying traffic with high accuracy. The experimental results show that the model can be applied to classify network traffic of encrypted and unencrypted applications at the same time, especially the classification accuracy of the underlying traffic of encrypted applications is improved. In most cases, the accuracy generally exceeds 90%.

Keywords Traffic classification, CNN, LSTM, Attention mechanism