مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (٩) العدد (٢) ٢٠٢٥
Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (9) No. (2) 2025

مراقبة حركة الأحمال في الشبكات المعرفة بالبرمجيات والتحكم بحركتها باستخدام تقنيات التعلم الآلي

أ.د. جمال خليفة**

د. مهند عيسى***

م. فيفاء ناصر مكائيل*

(تاريخ الإيداع ٢٠٢٤/١٠/١٥ . قُبِل للنشر في ٢٠٢٥/١/٢٦)

□ ملخّص □

تعد عملية مراقبة حركة الأحمال وتصنيفها من أبرز المجالات البحثية نتيجة النمو الهائل للتطبيقات والشبكات الحديثة. توفر هذه العملية فوائد عديدة تشمل تقليل ازدحام الشبكة، تحسين إدارتها، وتعزيز جودة الخدمة المقدمة. ومع التطور الكبير في الشبكات المعرفة بالبرمجيات Software Defined Network(SDN)التي تتميز بقدرتها على حل مشكلات الشبكات التقليدية من خلال تبسيط إدارة الشبكة، وإتاحة برمجتها، وتوفير رؤية شاملة لحركة البيانات، أصبحت الشبكات المعرفة بالبرمجيات منصة واعدة لتصنيف حركة الأحمال وتحسين مسارات التوجيه باستخدام تقنيات التعلم الآلي.

تم في هذا البحث بناء شبكة SDN لمراقبة حركة الأحمال وتصنيفها باستخدام تقنيات التعلم الآلي، حيث تم اعتماد نموذج الغابات العشوائية Random Forest (RF) كأحد نماذج التعلم الخاضع للإشراف لتصنيف حركة البيانات بناءً على التطبيقات المختلفة. تم تقسيم الأحمال إلى "كبيرة" و"صغيرة" بناءً على ميزات التدفقات المستخلصة. بعد التصنيف، يتم تحديد المسار الأمثل لكل حمل باستخدام وحدة التحكم RYU. أظهرت النتائج إمكانية تحقيق تصنيف دقيق وتحسين كفاءة التوجيه في الشبكات المعرفة بالبرمجيات، مما يعزز من أداء الشبكة واستجابتها للتغيرات الديناميكية في حركة البيانات.

الكلمات المفتاحية: الشبكات المعرفة بالبرمجيات (SDN)، المتحكم RYU ، التعلم الآلي، الغابة العشوائيةٌ(RF) ، التصنيف، أحمال كبيرة، أحمال صغيرة.

*أستاذ ، قسم هندسة الاتصالات والالكترونيات، كلية الهندسة الميكانيكية والكهربائية، جامعة اللاذقية .
** دكتور محاضر في قسم هندسة الاتصالات والالكترونيات، جامعة اللاذقية .
***طالبة دكتوراه ، قسم هندسة الاتصالات والالكترونيات، كلية الهندسة الميكانيكية والكهربائية، جامعة اللاذقية .

مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (٩) العدد (٢) ٢٠٢٥
Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (9) No. (2) 2025

Monitoring And Controlling Load Movements in Software-Defined Networks Using Machine Learning Techniques

E.Faifaa Naser Micaiel*
prof.Jamal khalifah**
D. Mohannad issa***

(Received 15/10/2024 . Accepted 26/1/2025)

□ ABSTRACT □

The monitoring and classification of traffic loads are among the most significant research areas due to the rapid growth of modern applications and networks. This process offers various benefits، including reducing network congestion، improving network management، and enhancing service quality. With the advancement of Software-Defined Networking (SDN)، which addresses the limitations of traditional networks by simplifying network management، enabling programmability، and providing comprehensive network visibility، SDN has become a promising platform for traffic classification and optimized routing using machine learning techniques.

In this study، an SDN environment was built to monitor and classify traffic loads using machine learning techniques. The **Random Forest** algorithm، a supervised learning model، was employed to classify network traffic based on application-level features. Traffic loads were categorized into "high" and "low" loads based on extracted flow features. Following classification، the RYU controller was utilized to determine the optimal path for each load. The results demonstrate the feasibility of accurate traffic classification and improved routing efficiency in SDN environments، contributing to enhanced network performance and adaptability to dynamic traffic changes.

Keywords: Software-Defined Networking, RYU Controller, Machine Learning، Classification, Random Forest, Elephant Flow, Mice Flow.

*Postgraduate Student (PhD student), Department of Communication and Electronics, Faculty of Mechanical and Electrical Engineering, Tishreen University, Lattakia,Syria. Email: faifaa.micaiel@gmail.com
** Professor .Doctor, Department of Communication and Electronics, Faculty of mechanical and Electrical Engineering, Tishreen University, Lattakia, Syria. Email: jam2kh58@hotmail.com
*** lecturer , Department of Communication and Electronics, Faculty of mechanical and Electrical Engineering, Tishreen University, Lattakia, Syria. Email mohannadissa@gmail.com

١ مقدمة (Introduction):

بسبب تضخم عدد الاتصالات وتزايد تقنيات الاتصال وتنوع الأجهزة في جميع أنحاء العالم، زادت حركة الأحمال وتنوعت في الشبكات بشكل كبير. بالإضافة إلى الاعتماد الكبير على أجهزة الاتصال ذات الميزات المتقدمة (مثل الحواسب والهواتف الذكية وأجهزة الكترونية ذكية، وحدات تحكم ألعاب الفيديو وأجهزة مراقبة.....) وتغيير بشكل كبير لأنماط تدفق حركة الأحمال وتنويع حركة البيانات في الشبكات. مما حوَل تصنيف حركة الأحمال في الشبكة إلى مشكلة حاسمة، وإن الحجم الكبير للحزم الواردة الذي يتميز بالسرعة العالية والتنوع الواسع، يضع تصنيف حركة الأحمال في الوقت الفعلي (Traffic Classification) ضمن أهم مجالات البحث لمعالجة مشاكل حركة البيانات. ويتم استخدام معلومات مختلفة مثل ترويسة الحزم في الشبكة كمعايير للتصنيف. تعمل هذه الطريقة بشكل جيد للغاية مع التطبيقات الشائعة، ولكنها لا تعمل بشكل جيد مع التطبيقات التي تستخدم التغييرات في أرقام المنافذ (Port Number) [1]. أحدث ظهور الشبكات المعرفة بالبرمجيات (SDN) نقلة نوعية في عالم الاتصالات حيث تعمل هذه الشبكات على فصل طبقة التحكم في الشبكة عن طبقة التطبيقات، مما يسمح لوحدة التحكم المركزية ببرمجة المستوى مباشرة، ويتم التعامل مع إدارة الشبكة من خلال المتحكم المركزي Controller تقع مسؤولية نقل الحزمة على عاتق مستوى البيانات الموجود على أجهزة التوجيه أو المحولات فقط المتوافقة مع بروتوكول(OFP) OpenFlow من بين جميع البروتوكولات التي يمكن استخدامها في بنية SDN، فإن بروتوكول OpenFlow هو الأكثر كفاءة، [2] يتم جمع البيانات المطلوبة لتصنيف حركة الأحمال SDN والتمييز بينها، تم في هذا البحث، التمييز بين الأحمال الكبيرة (Elephant Flow)و الأحمال الصغيرة (Mice Flow) ليتم اتخاذ قرار التوجيه الأمثل ضمن مسارات الشبكة. وقدمت شبكات (SDN) مرونة للشبكة وقابلية للبرمجة وكانت مجالا جيدا للتطبيق حيث أنه يمكن تخفيف تكاليف تشغيل الشبكة من خلال المحاكاة الافتراضية،[3] ويمكننا الاستفادة من تقنيات التعلم الآلي والذكاء الصنعي في هذا المجال.[4،5]

٢ الدراسات المرجعية (Reference Studies):

قام Raikar و آخرون بالدراسة [6] باقتراح بنية للشبكة المعرفة بالبرمجيات (SDN) وقاموا بتطبيق ثلاثة نماذج التعلم الخاضعة للإشراف مختلفة، وهي (SVM) Support Vector Machine، (NC) Nearest Centroid،

(NB) Naïve Bayes، على تصنيف حركة البيانات بناءً على التطبيقات الموجودة في منصة الشبكة المعرفة بالبرمجيات، و الدقة التي تم الحصول عليها لـ SVM هي ٩٢,٣%، و 96.79 لـ NC و ٩١,٠٢% لـ NB. بينما ركز Pradhan و آخرون بالدراسة [7] على زيادة قدرة وحدات تحكم SDN على اتخاذ القرار في شبكات الاستشعار تحت الماء باستخدام خوارزميات التعلم الآلي لتحقيق تصنيف حركة مرور الشبكة في الوقت الفعلي مع مراعاة جودة الخدمة. قدم Serag و آخرون في الدراسة [8] مسحًا شاملاً لتطبيق خوارزميات التعلم الآلي في مجالSDN ، مع التركيز بشكل خاص على تصنيف حركة الأحمال و ناقشوا الاختلافات بين طرق التصنيف التقليدية والقائمة على ML ، مع تسليط الضوء على المزايا التي توفرها تقنيات التعلم الآلي. كما ناقش Azab و آخرون في الدراسة [9] تطبيق خوارزميات التعلم الآلي في العديد من تقنيات التصنيف،

باستخدام الخصائص الإحصائية لتدفق حركة المرور على الشبكة، و استخدام التحليل في الوقت الفعلي للكشف عن أي أنشطة مشبوهة حيث أن التصنيف هو العنصر الأساسي لأنظمة اكتشاف التطفل على الشبكة (IDS) Intrusion Detection Systemss و تمت دراسة ارتباط التعلم العميق مع تقنيات التصنيف. واقترح Eissa و آخرون في الدراسة [10] إطار لضمان جودة الخدمة في شبكة SDN مع تصنيف باستخدام تقنيات التعلم الآلي .يشتمل الإطار على نظام تصنيف مكون من مرحلتين، المرحلة غير المتصلة بالإنترنت، حيث تم تدريب واختبار المصنف، والمرحلة عبر الإنترنت، حيث تتم المحاكاة و التعامل مع التدفقات واختبار سرعة المصنف على مجموعة بيانات "IP-network-traffic-flows-labeled-with-87-apps" ويحدد نوع حركة الأحمال بدقة ٩٩,٩٥% على مجموعة البيانات"ISCX-VPN-NONVPN" . بالإضافة إلى ذلك، ثبت أن سرعة المصنف تبلغ حوالي ٣٥٠٠ سجل/ثانية. وقام Shafiq و آخرون في الدراسة [11] بمقارنة الطرق التقليدية لتصنيف حركة المرور على الإنترنت مثل تقنية Port Based و Pay Load Based و Machine Learning Based. ثم تم اختبار تقنية التعلم الآلي (ML). باستخدام أربعة مصنفات للتعلم الآلي تدعم المتجهات ، يتم تطبيق شجرة القرار ومصنفات Bayes Net, Naïve Bays. واقترح الباحث Jang و آخرون في الدراسة [12] تصنيف حركة الأحمال بالاستفادة من الأداة (VAE) Variational Autoencoder أداة التشفير المتغير حيث يتم تدريب VAE باستخدام ميزات إحصائية، ويتم استخراج توزيعات الميزات الكامنة للتدفقات في كل فئة خدمة. و يصنف حركة الاستعلام من خلال المقارنة لتوزيعات الميزات الكامنة لحركة الاستعلام مع التوزيعات المستفادة لفئات الخدمة. تم جمع احصائيات تدفقات الشبكة من خدمات الإنترنت في العالم الحقيقي للتدريب والاختبار. تمتعت الطريقة المقترحة بمتوسط دقة 89% .

٣ مشكلة البحث (Research problem):

مع التوسع السريع في تطبيقات الشبكات وتزايد حركة البيانات، تواجه الشبكات تحديات كبيرة في تصنيف التدفقات وتقليل الازدحام. تعد الشبكات المعرفية بالبرمجيات (SDN) بيئة واعدة لتحسين إدارة الشبكات، حيث توفر قدرة برمجية متقدمة لرصد وتحليل حركة البيانات. ومع ذلك، يظل التحدي الأساسي في كيفية استخدام تقنيات الذكاء الاصطناعي والتعلم الآلي لتصنيف الأحمال (مثل تدفقات "Mice" و"Elephant") بكفاءة ضمن هذه الشبكات الحديثة، مما يساهم في تحسين استغلال الموارد وتقليل ازدحام الشبكة. لذا تكمن مشكلة البحث في تطوير نموذج تصنيف يعتمد على ميزات مستخلصة من بيانات الشبكة باستخدام خوارزميات تعلم الآلة، مثل Random Forest، لتوفير قرارات توجيه دقيقة وفعالة في بيئات شبكية ديناميكية تحتوي على أحمال مختلفة.

٤ أهمية البحث وأهدافه (Research objective and importance):

يهدف البحث الى مراقبة حركة الأحمال في شبكة SDN وكشف التدفقات وتصنيفها (أحمال كبيرة وأحمال صغيرة) باستخدام تقنيات التعلم الآلي واختبار خوارزمية التعلم الآلي للمصنف، ثم توجيهها ضمن مسارات الشبكة المحددة سواء للتدفقات الصغيرة أو الكبيرة. تكمن أهمية كشف التدفقات بالتعرف على نمط التدفق وادارة التدفقات لتخفيف الازدحام في الشبكة والمساعدة على الاستخدام الأمثل للموارد.

٥ طرائق البحث ومواده (Research methodology):

استخدم المحاكي (Mininet) وهو محاكي مفتوح المصدر ويوفر الأداة (Miniedit) لرسم طوبولوجيا الشبكة، واخترنا المتحكم (RYU) الذي يدعم كتابة التطبيقات بلغة البايثون. واستخدمت الأداة (Iperf) لإنشاء تطبيق مخدم/زبون(Client/Server) وارسال الرزم بينهما سواء رزم بروتوكول التحكم بالنقل (TCP) (Transmission Control Protocol) وبروتوكول حزم بيانات المستخدم (UDP) User Protocol Datagram. واستخدمت الأداة Wireshark لمراقبة حركة الأحمال في الشبكة وانشاء قاعدة بيانات ليتم تدريب نماذج التعلم الآلي عليها. تم إجراء تدريب واختبار نماذج التعلم الآلي باستخدام برنامج python في بيئة تجربة Jupyter Notebook. ودرسنا عدة بارامترات لتقييم أداء المصنف وهي:

- الإيجابيات الحقيقية (TP): الفئة الفعلية إيجابية، والفئة المتوقعة إيجابية.
- السلبيات الحقيقية (TN): الفئة الفعلية سلبية، والفئة المتوقعة سلبية.
- الإيجابيات الكاذبة (FP): الفئة الفعلية سلبية، والفئات المتوقعة إيجابية.
- السلبيات الكاذبة (FN): الفئة الفعلية إيجابية، والفئة المتوقعة سلبية.

ومن هذه المعاملات يتم حساب معاملات جديدة تعبر بدقة عن أداء نماذج التصنيف: [13]

الدقة (Accuracy) هي عدد المكونات التي تم تصنيفها بشكل صحيح الى العدد الاجمالي لمكونات العينة وتحسب بالعلاقة:

$$Accuracy= \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (1)$$

Precision :مقياس يحدد قدرة النموذج على التنبؤ بالعينات الإيجابية بشكل صحيح ويعطى بالعلاقة:

$$Precision = \frac{TN}{TP+FP} \qquad (2)$$

Recall: ويسمى هذا المقياس أيضا بالحساسية أو بالمعدل الإيجابي الحقيقي، ويعطى بالعلاقة التالية:

$$(3)Recall = \frac{TP}{TP+FN}$$

F1−Score: هي المتوسط التوافقي للدقة والاستدعاء، ويعرف بالعلاقة التالية:

$$F1.score = \frac{TP}{TP+0.5(FP+FN)} \quad (4)$$

التعلم الآلي هو تطبيق للذكاء الاصطناعي (AI) الذي يسمح بتطوير الأنظمة التي تتعلم بشكل مستقل من خلال تحديد الأنماط المعقدة من مجموعات البيانات الكبيرة. من وجهة نظر تشغيلية، يتكون التعلم الآلي من مرحلتين: المرحلة الأولى تتعلق بالتدريب وتتكون من تزويد خوارزميات التعلم الآلي بمجموعة فرعية من مجموعة البيانات المستخدمة (تسمى مجموعة التدريب) والتي يمكن لنموذج النظام التعلم منها، والمرحلة الثانية تتكون من اتخاذ القرار، حيث يمكن للنظام تقدير نتيجة إدخال جديد، بناءً على النموذج المدرب. يتم تصنيف خوارزميات التعلم الآلي على نطاق واسع إلى التعلم الخاضع للإشراف، وغير الخاضع للإشراف، وشبه الخاضع للإشراف، والتعلم التعزيزي [14]. ويبين الجدول التالي تقنيات التعلم الآلي [15]:

جدول (١) تقنيات التعلم الآلي

أمثلة	التدريب	صنف التعلم الآلي
RF،KNN،DT،SVM MLP،LR،GB،	يتم تدريب النموذج باستخدام بيانات تحتوي على مدخلات(Features) وأهداف، (Labels) الهدف هو تعلم العلاقة بين المدخلات والأهداف بحيث يمكن للنموذج التنبؤ بالأهداف الجديدة بناءً على المدخلات	الخاضع للإشراف
PCA،SVD،K-means	يتم تدريب النموذج باستخدام بيانات لا تحتوي على أهداف، الهدف هو اكتشاف الأنماط في البيانات.	غير الخاضع للإشراف
	يتم تدريب النموذج باستخدام مجموعة من البيانات التي تحوي على بعض (الأمثلة المصنفة (مع أهداف) وأخرى غير مصنفة (بدون أهداف)	شبه الخاضع للإشراف

٦ الشبكات المعرفة بالبرمجيات SDN [16]:

تتألف شبكات SDN من الطبقات والواجهات التالية:

٦,١. طبقة التطبيقات (Application Layer): هي الطبقة الأولى في نموذج البنية المعمارية لشبكة SDN وتتكون من الخدمات والتطبيقات التي تقدمها الشبكة للمستخدم.

٦,٢. طبقة التحكم (Control Layer): تحتوي على المتحكم (Controller) الذي يعتبر أساس هذه التقنية، فهو المسؤول عن قرارات التوجيه والادارة والتحكم بكافة وظائف الشبكة وتتصل هذه الطبقة مع طبقة التطبيقات من خلال واجهة برمجة التطبيقات API.

٦,٣. طبقة البنية التحتية (Infrastructure layer): تتكون من اجهزة الشبكة (موجهات-مبدلات) تتلقى الأوامر من طبقة التحكم وتقوم بتنفيذها حيث تتصل هذه الطبقة مع المتحكم عن طريق (OF) وهو البروتوكول الأساسي في عمل تقنية SDNحيث ينظم التواصل بين المتحكم وطبقة البنية التحتية.

٦,٤. واجهات التخاطب الجنوبية (Southbound Interface): توفر بروتوكول اتصال بين طبقة البنية التحتية وطبقة التحكم، وتستخدم من قبل المتحكم لإرسال إعدادات التهيئة ومداخل التدفق وتقوم بالتغييرات على قواعد التوجيه المستخدمة في أجهزة طبقة البنية التحتية، وتنقل وظائف الشبكة إلى طبقة التحكم.

٦,٥. واجهات التخاطب الشمالية(Northbound Interface) : تستخدم من قبل طبقة التطبيقات للتواصل مع المتحكم، وتعد الجزء الأكثر حساسية في بنية شبكاتSDN لقدرتها على دعم وتشغيل مجموعة واسعة من التطبيقات المختلفة وتلعب دوراً أساسياً لمطوري التطبيقات وتوفر واجهة تخاطب مشتركة بين المتحكم وطبقة الادارة.

٧ بروتوكول OpenFlow:

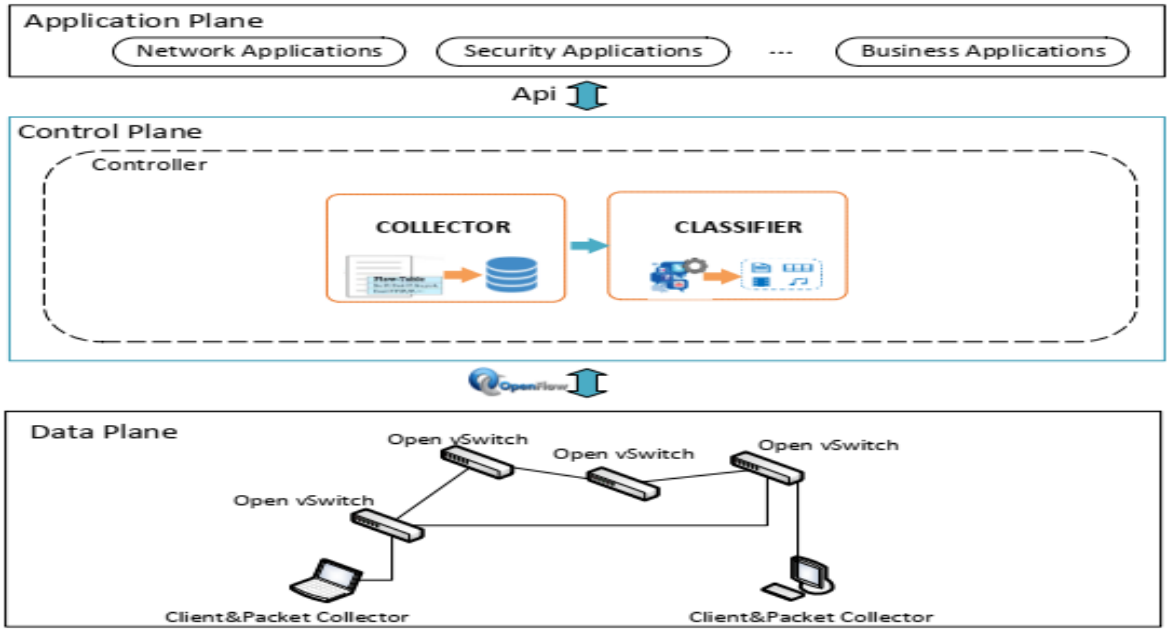
هو بروتوكول الاتصال المباشر بين طبقة التحكم وطبقة البنية التحتية، أي بين المتحكم و أجهزة الشبكة (routers & switches) وأهم ميزاته: تخزين قواعد التوجيه لدى أجهزة الشبكة بجداول التدفق المؤلفة من Flow Entries(مداخل التدفق).ونقل أوامر التوجيه وتحديد المسارات بين SDN OF controller وبين

OF Switch. (إضافة مسار، تعديل. حذف..) أي ينقل Action Or Rule الى مستوى Data plane. [17]

٨ التصنيف في شبكات SDN (SDN Network Classification):

تم في هذه الفقرة تقديم إطارا عاما عن التصنيف في شبكات SDN مبينا في الإطار كل طبقة من طبقات شبكة SDN، كما يوضح الشكل(1) ، حيث تقوم المبدلات (Open Vswitch) الموجودة في طبقة البنية التحتية بجمع تدفقات حركة الأحمال الشبكة وإرسالها إلى المتحكم الموجود في طبقة التحكم عن طريق بروتوكول (OpenFlow). تقوم وحدة التحكم بجمع واستخلاص المميزات الأكثر أهمية عن طريق المجمع (Collector). تم اجراء دراسة لتحديد المميزات الأكثر أهمية لعمل المصنف عن طريق دراسة الارتباط بين هذه الميزات وتحديد إذا كانت هذه الميزة حاسمة في تصنيف التدفق. ثم ترسل وحدة التحكم المميزات المستخلصة إلى المصنف (نموذج تعلم آلي)، والذي بدوره يصنف التدفقات، وعندما يتم توليد نتائج التصنيف، يرسل المصنف جداول التدفق إلى المبدلات. يمكن أيضًا مشاركة نتائج التصنيف مع وحدات أو تطبيقات أخرى من خلال واجهة برمجة التطبيقات الشمالية.

أي أنه بمجرد تصنيف حركة الأحمال، يمكن لوحدات تحكم SDN استخدام هذه المعلومات لاتخاذ قرارات إعادة التوجيه، وتطبيق سياسات جودة الخدمة، وفرض تدابير الأمان، وتحسين أداء الشبكة. من خلال ضبط تدفقات حركة الأحمال ديناميكيًا بناءً على تصنيفها، يمكن لشبكات SDN تحسين كفاءة الشبكة وتقليل زمن الوصول وضمان تجربة أفضل للمستخدم [19,18].



الشكل (١) بنية شبكات SDN وتصنيف حركة الأحمال ضمنها

٩ تدفق البيانات والمعالجة المسبقة: (Data Flow And Pre-Processing) [19]:

٩,١ تدفق البيانات (Data Flow):

يمكن من خلال تحليل حركة المرور على الشبكة لمسؤولي الشبكات تحديد نوع التطبيق أو الخدمة المستخدمة على شبكاتهم. ويساعدهم على فهم أفضل لكيفية استخدام شبكاتهم، وتحديد الازدحام المحتمل، واتخاذ الإجراء المناسب إذا لزم الأمر. عندما يبدأ تدفق البيانات في الشبكة نقوم بالتقاط الحزم باستخدام أداة التقاط مثل Wireshark. ثم يتم فحص الحزم الملتقطة لتحديد أي بروتوكول أو أرقام منفذ مرتبطة، يمكن استخدام تقنيات مختلفة مثل فحص الحزم العميق والتحليل الإحصائي وخوارزميات التعلم الآلي لتصنيف نوع التطبيق أو الخدمة المستخدمة على جزء معين من الشبكة.

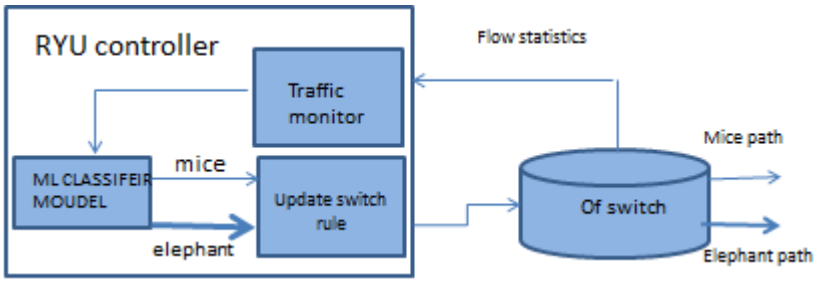
٩,٢ معالجة البيانات مسبقا (Data Pre-Processing):

تعد عملية معالجة البيانات مسبقا خطوة مهمة في أي خوارزمية للتعلم الآلي. فهي تضمن تنسيق مجموعة البيانات بشكل صحيح وأنها جاهزة للتحليل. قبل إدخال مجموعة بيانات في الخوارزميات المقترحة، يجب أن تمر أولاً بعملية معالجة البيانات مسبقا لضمان دقة وموثوقية النتائج. وهذا يسمح لنا باكتساب رؤى أفضل حول مجموعات البيانات الخاصة بنا دون وجود أي مشكلات تتعلق بتنسيقات غير صحيحة أو نتائج غير موثوقة بسبب التنسيق الأولي السيئ للغاية الذي تم إجراؤه مسبقًا.

١٠ النموذج المقترح للعمل (Proposed Business Model):

تم مناقشة التقنية المقترحة وكيفية تصميمها لتحقيق الفعالية القصوى، يتألف هذا النموذج كما هو مبين بالشكل (٢) من:

- مراقب الشبكة (Traffic Monitor) الموجود ضمن طبقة التحكم يقوم بطلب واستخلاص الميزات بعد مراقبة شبكةSDN ويتم ذلك باستخدام برنامج Wireshark.
- نظام تصنيف (ML Classification Module) الموجود ضمن طبقة التحكم يقوم على خوارزمية التعلم الآلي حيث أن نموذج التعلم الآلي هو الذي سيحدد نمط التدفق بناء على المعلومات الواردة ضمن packetin حيث يمرر المتحكم الميزات الى النموذج لتحديد نوع التدفق وارسال القواعد الى طبقة البنية التحتية لفرز البيانات.



الشكل (٢) مخطط العمل للشبكة المقترحة

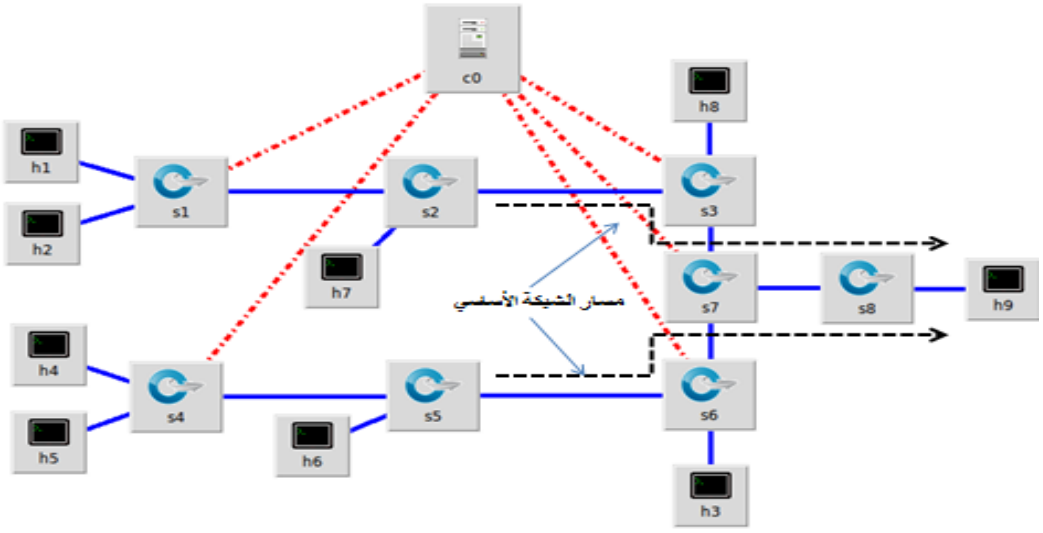
بعد تجميع البيانات أي الحصول على بيانات حركة الأحمال في الشبكة أي الميزات، مثل (عدد الحزم، البروتوكول، المدة الزمنية معدل النقل، حجم الحزمة الأولى.....). تتم معالجة البيانات (تنظيفها وتحضيرها للتدريب بإزالة البيانات غير المفيدة أو المكررة) ثم نستخدم هذه البيانات التي قمنا بتجهيزها لتدريب نموذج التعلم الآلي الذي قمنا باختياره. ومن التحديات التي يمكن أن تواجهنا أن عملية جمع البيانات وبناء

dataset تكون معقدة (لأن تشغيل الشبكة ومراقبتها على الحاسب الشخصي يستحوذ على جزء كبير من الموارد وهذا يؤدي لضعف الأداء). ومن أجل بناء قاعدة البيانات تم تشغيل الشبكة مرات عديدة وارسال تدفقات من احجام مختلفة ومراقبة حركة الأحمال باستخدام برنامج Wireshark وحفظها في ملفات على وحدة التخزين الداخلية للحاسب، كما يجدر الإشارة الى أن نموذج التعلم الآلي يحتاج لقدرة لحسابية كبيرة ليقوم بعملية التدريب، وهذا يحتاج لاختيار وموازنة بين حجم قاعدة البيانات ودقة النموذج فكلما زاد حجمها زادت الدقة ولكن تتطلب قدرة كبيرة لإتمام عملية التدريب.

سيناريو العمل (Scenario of work) :

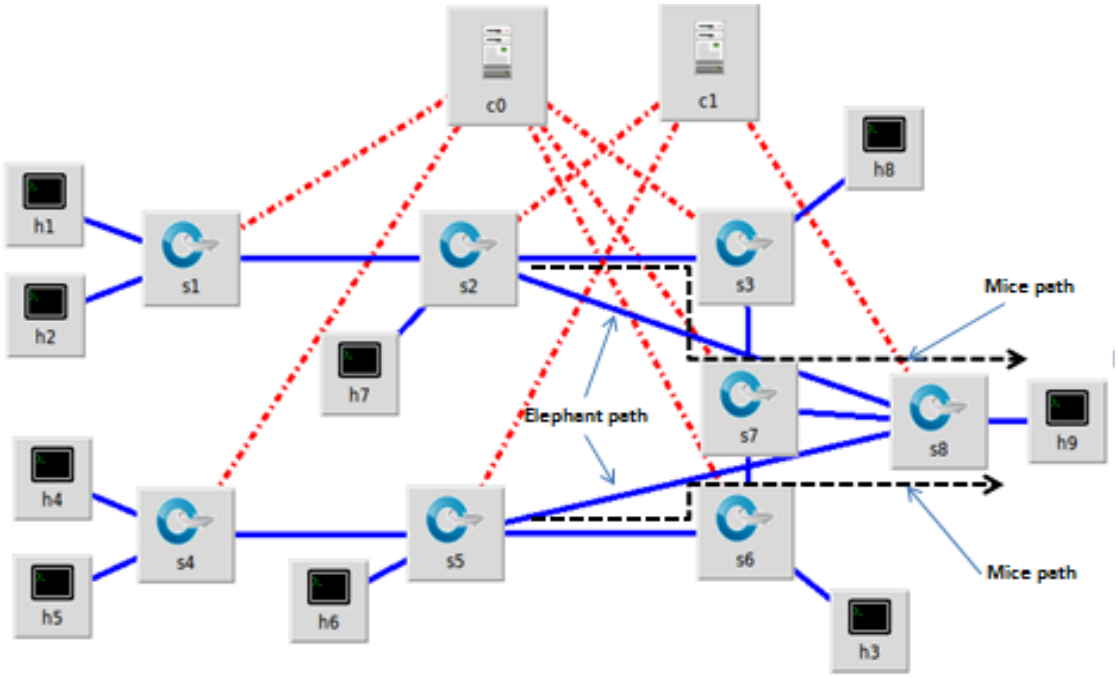
تم بناء شبكة SDN تتألف من متحكم (Controller) ومبدلات (Switchs) وطرفيات (Hosts) تم بتشغيل برنامج mininet وتشغيل المتحكم RYU واستخدام الأداة miniedit لنرسم طبولوجيا الشبكة، نرسل التدفقات باستخدام الأداة (Iperf) حيث يتم ارسال أحمال متنوعة ونعتبر كل Host محطة طرفية تتولد فيها مجموعة من التدفقات أو تمر عبرها من محطة مجاورة. وتمر كافة الأحمال الكبيرة والصغيرة ضمن مسار الشبكة الأساسي كما هو موضح بالشكل (٣):

.



الشكل (٣) طوبولوجيا الشبكة المقترحة

و لتخصيص مسارات محددة في الشبكة بحيث تكون هذه المسارات مخصصة لتمر عبرها التدفقات الكبيرة (المسار من s2 الى s8 والمسار من s5 الى s8) بحيث لا يتم اشغال الشبكة الأصلية بهذه التدفقات الكبيرة والتي من الممكن أن تكون مقاطع فديو أو أي تدفق ذات حجم كبير وندع مسارات الشبكة الأصلية لتمر عبرها بقية التدفقات لتصل كافة التدفقات الى نقطة المراقبة الممثلة ب (H9) حيث يقوم المتحكم بإرسال تعليمات الى المبدلات من اجل تحديد المنافذ لإرسال الأحمال بتحديد المسار لأي تدفق وارد سنقوم بإضافة متحكم آخر للشبكة والذي يحقق نموذج التعلم الآلي بعد تدريبه على تصنيف الأحمال بناءً على قاعدة بيانات تم انشائها على ذات الشبكة أي تطبيق النموذج المقترح في الشكل (٢) على الشبكة ونصل المبدلات (s2،s5،s8) بهذا المتحكم كما هو مبين بالشكل(٤).



الشكل (٤) طبولوجيا الشبكة بعد تحديد مسارات للتدفقات الكبيرة

بعد تأسيس الاتصال بين المتحكم c1 والمبدلات s8، s5،s2 وتبادل الرسائل، أي تدفق وارد الى المبدلات يتم البحث في جدول التدفق عن تطابق مع القواعد أو المداخل الموجودة ويتعامل معها كالتالي:

عند التطابق يتم التعامل مع التدفق وفق القواعد الموجودة وفي حال عدم التطابق يرسل التدفق الى المتحكم ليعطي التعليمات للمبدلات باتخاذ القرار المناسب. باستخدام برنامج wireshark (هو اداة مفتوحة المصدر تستخدم لمراقبة وتحليل حركة الأحمال ضمن الشبكات) تتم مراقبة حركة الأحمال ضمن الشبكة من حيث (المصدر، الوجهة، حجم التدفق، المدة الزمنية للتدفق، المنفذ....) حيث يتم التقاط حزم البيانات التي تمر عبر واجهة الشبكة حيث يمكنه التقاط جميع الحزم أو حزم معينة بناءً على معايير محددة، تم عرض تفاصيل كل حزمة، بما في ذلك معلومات مثل (عنوان المصدر، عنوان الوجهة، نوع البروتوكول.....) وأي بيانات إضافية، ويدعم مجموعة من البروتوكولات، مما يسمح بتحليل حركة الأحمال في الشبكة عبر بروتوكولات مختلفة مثل TCP،open flow، UDP، HTTP، DNS وغيرها. ويوفر إحصائيات مفيدة مثل عدد الحزم المرسلة والمستلمة، وأوقات الاستجابة...) ويوفر مجموعة من الفلاتر مثلا يبين الشكل (٥) مراقبة تدفقات البروتوكول TCP :

Endpoints: any

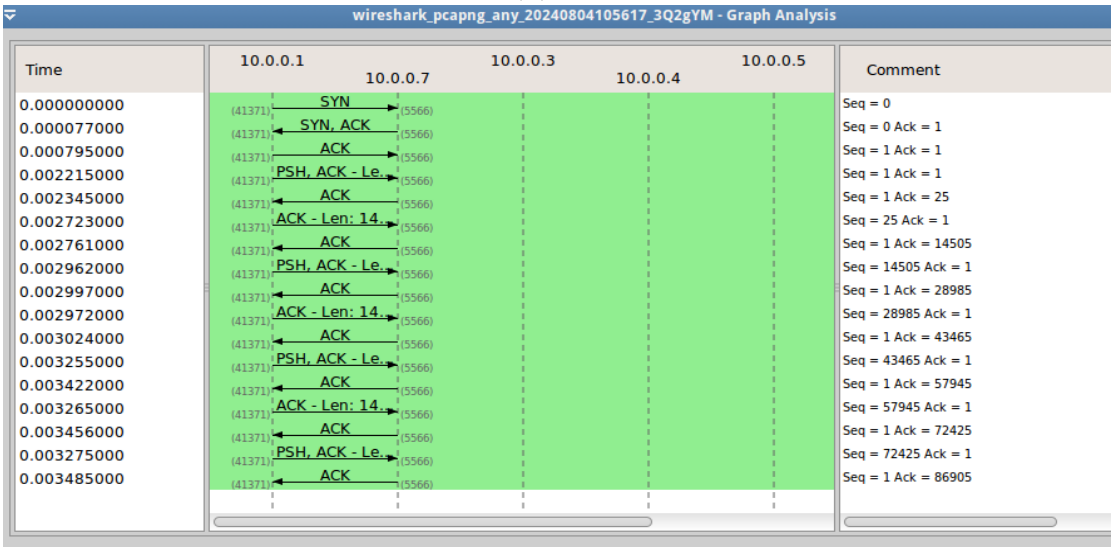
Ethernet | Fibre Channel | FDDI | IPv4: 3 | IPv6 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 12 | Token Ring | UDP | USB | WLAN

TCP Endpoints

Address	Port	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes
10.0.0.1	34099	11 580	319 483 228	8 850	319 297 040	2 730	186 188
10.0.0.3	5566	54 619	1 396 272 768	13 307	910 504	41 312	1 395 362 264
10.0.0.2	37983	886	18 996 992	625	18 977 996	261	18 996
10.0.0.2	37984	1 301	7 246 840	809	7 213 276	492	33 564
10.0.0.1	34102	10 267	225 844 348	8 076	225 694 320	2 191	150 028
10.0.0.2	37986	1 620	6 341 828	975	6 297 828	645	44 000
10.0.0.2	37987	690	10 453 040	563	10 444 396	127	8 644
10.0.0.1	34105	12 831	326 927 312	9 775	326 718 236	3 056	209 076
10.0.0.2	37989	686	12 730 544	557	12 721 772	129	8 772
10.0.0.2	37990	551	12 582 724	459	12 576 468	92	6 256
10.0.0.2	37991	1 928	42 000 456	1 519	41 972 636	409	27 820
10.0.0.1	34109	12 279	413 665 456	9 104	413 448 296	3 175	217 160

الشكل (٥) احصائيات بروتوكول TCP

و يوفر البرنامج مراقبة حركة الأحمال والرسائل المتبادلة بين الطرفيات والسيرفر كما هو موضح بالشكل (6):



الشكل (٦) الرسائل المتبادلة في الشبكة

بعد تشغيل الشبكة الموضحة بالشكل(٣) لفترة زمنية ومراقبة حركة الأحمال باستخدام برنامج Wireshark يتم حفظ النتيجة كملف (pcap) ضمن البيئة الافتراضية التي بنيت ضمنها شبكة SDN. ليتم نقله بعد ذلك وتصديره الى بيئة (Windows)، يتم التعديل عليه وحفظه كملف (csv) حيث نعيد تشغيل الشبكة أكثر من مرة لينتج لدينا مجموعة بيانات مناسبة للتدريب، ثم نجمع الملفات التي نتجت من مراقبة الشبكة وبالتالي نكون أنشأنا قاعدة بيانات (Dataset) وأصبحت جاهزة لتدريب نموذج التعلم الآلي. حيث تم بناء مجموعات البيانات من عمليات التقاط حزم الشبكة، حيث تتضمن كل حزمة: بيانات الحمولة والبيانات الوصفية مثل (عناوين IP المصدر والوجهة والبروتوكول ورقم المنفذ....). من أجل تدريب هذه النماذج بدقة على مجموعة البيانات هذه، يجب تضمين مجموعة متنوعة من الميزات التي يمكن أن تساعد في تحديد الأنواع المختلفة من الحزم المرسلة عبر الشبكة. ونضيف تسميات الفئة التي تشير إلى حجم المحتوى الموجود داخل حزمة معينة، والفئة للحمل الذي قمنا بتحديده بناءً على العتبة التي تميز بين الأحمال الكبيرة والأحمال الصغيرة، وتظهر ضمن dataset التي قمنا بحفظها ميزات كل حزمة من هذه الحزم. يوضح الجدول (٢) بعض ميزات قاعدة البيانات التي قمنا باستخدامها:

جدول(2) ميزات قاعدة البيانات المستخدمة

Field name	Description
datapath_id	معرف المبدل
Flow_id	معرف التدفق
ip_src	عنوان المصدر
ip_dst	عنوان الوجهة
ip_proto	البروتوكول
flow_duration_sec	مدة التدفق بالثانية
packet_count_per_s	عدد الحزم كل ثانية
Label	1 for elephant،0 mice
First_pkt_size	حجم الحزمة الأولى
Avg_pkt_size	متوسط حجم الحزمة
Pktrate	معدل الحزم

يوجد حالتي تصنيف اما Elephant أو Mice وبالتالي يعتبر تصنيفاً ثنائيا، سنقوم باختبار احدى خوارزميات التصنيف الآلي وهو مصنف (RF) الغابة العشوائية: وهو تقنية تعلم خاضعة للإشراف (Supervised Machine Learning Algorithms) يمكنها التعامل مع مشكلات التصنيف والانحدار [20]. وتم اختياره بسبب قدرته على التعامل مع مجموعات البيانات الكبيرة والتعامل مع القيم المفقودة، وتعتمد على المعلومات المكتسبة حيث ان كل شجرة قرار في هذا التصنيف ستقوم بتصنيف نفس المشكلة، وسيتم تحديد النتيجة النهائية من خلال الأخذ بعين الاعتبار غالبية النتائج. تم استخدام لغة Python ومكتبة scikit–learn من أجل حزم نموذج التعلم الآلي.

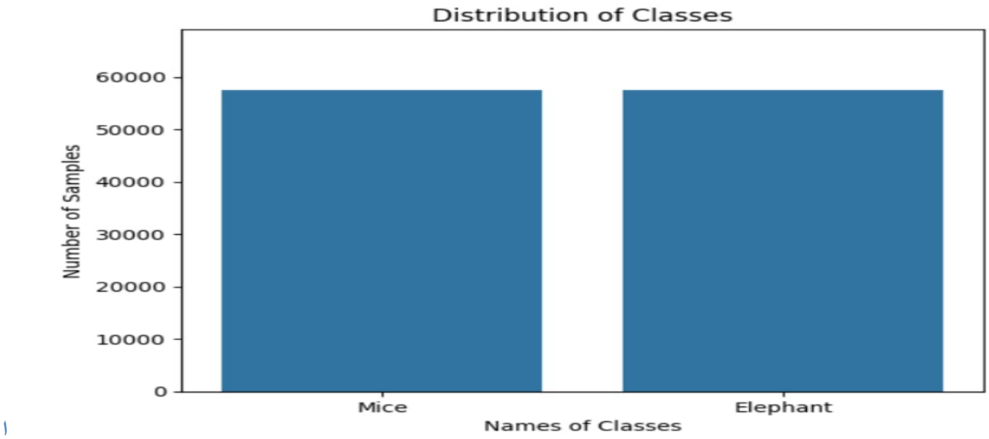
١١ النتائج والمناقشة Result And Discussion:

بعد أن تم تدريب نموذج التعليم الآلي، تم اختبار النموذج المدرب على قاعدة بيانات الاختبار، ويبين الجدول (٣) نتائج تدريب واختبار نموذج Random Forest model:

جدول (٣) نتائج اختبار وتدريب النموذج

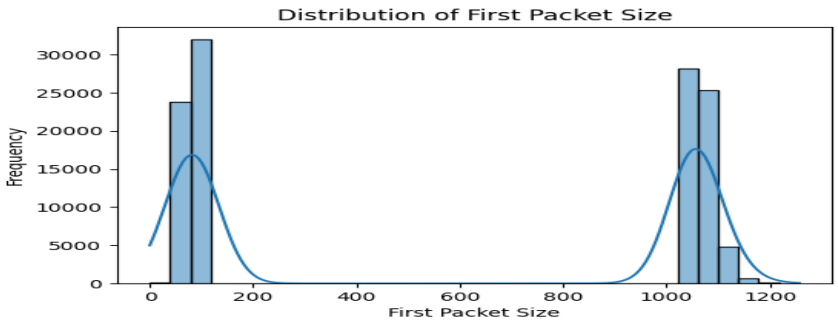
Training Set Evaluation:	Test Set Evaluation:
Accuracy: 0.9927	**Accuracy: 0.9919**
Precision: 0.9962	**Precision: 0.9949**
Recall: 0.9891	**Recall: 0.9887**
F1 Score: 0.9927	**F1 Score: 0.9918**

يشير الشكل (٧) الى توزيع الأصناف المتوازن في قاعدة البيانات المستخدمة والمكونة من أكثر ١١٥٠٠٠عينة:



الشكل (٧) توزيع الفئات

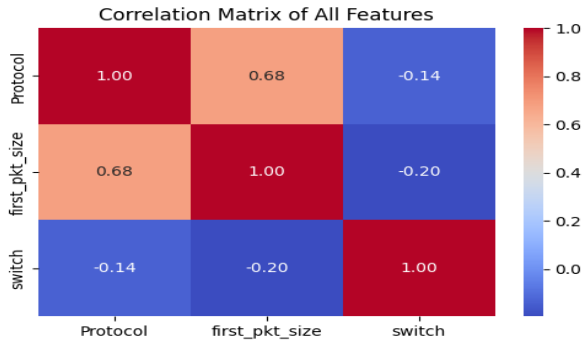
و بملاحظة أن توزيع الفئات متوازن تقريبا، مما يشير إلى أن مجموعة البيانات تم تعديلها لتجنب مشكلة عدم التوازن بين الفئات، وهو أمر جيد لتدريب النماذج. من أجل الاستفادة من تصنيف نموذج التعلم الآلي وتحسين دقة النموذج ومساعدة المتحكم على اتخاذ القرار في المسار المناسب لحمل الشبكة بمجرد ورود الحمل الى المتحكم وقبل نهاية كل زمن التدفق. تم اضافة ميزات للتدفقات وهي (حجم الحزمة الأولى من التدفق ونوع البروتوكول) وللتأكد من صحة اختيارنا واهمية اختيار هذه الميزات كما هو موضح بالشكل (٨):



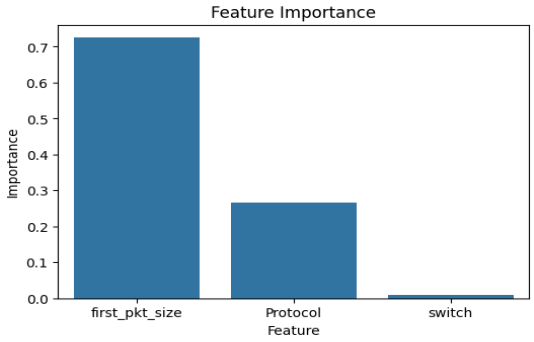
الشكل (٨) توزع حجم الحزمة الأولى

يُظهر التوزيع ذروتين بارزتين، مما يشير إلى وجود مجموعتين رئيسيتين من البيانات مرتبطة بالفئات المختلفة(Mice) و(Elephant) هذا التوزيع يعزز أهمية هذه الميزة في التفريق بين الفئات.

وباستخدام مصفوفة الارتباط بين الميزات نلاحظ وجود ارتباط مرتفع نسبيا بين"Protocol" و "First_pkt_size"بقيمة 0.68، كما هو موضح بالشكل (9) مما يشير إلى أن اختيار البروتوكول يمكن أن يؤثر على حجم الحزمة الأولى. ومن ناحية أخرى، هناك ارتباط ضعيف بين "switch" وبقية الميزات، مما قد يشير إلى أن هذه الميزة ليست حاسمة في تصنيف الفئات.

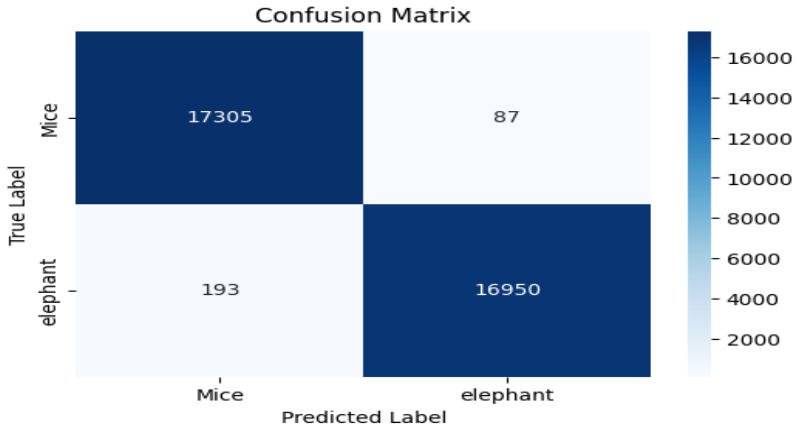


الشكل (٩) مصفوفة الارتباط بين الميزات



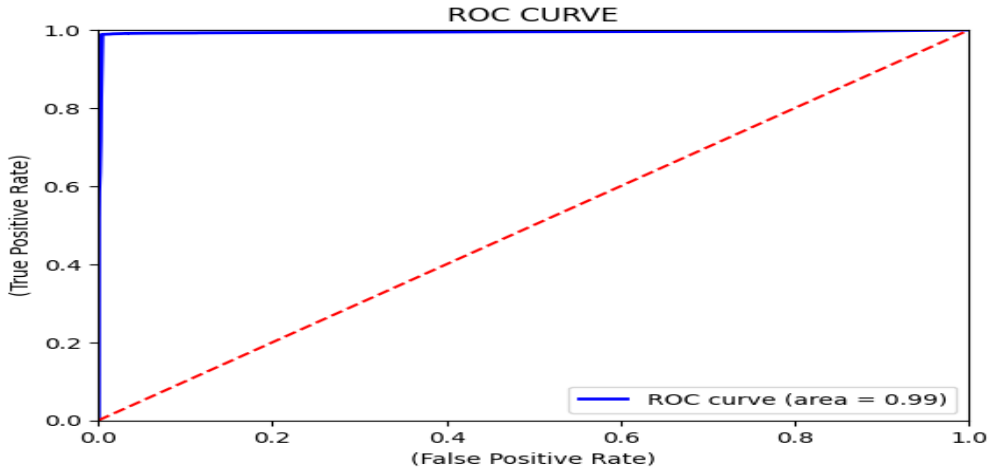
الشكل (١٠) أهمية الميزات

تمت مقارنة أهمية الميزات بالشكل (١٠) حيث تأكدت أهمية اختيار حجم الحزمة الأولى ونوع البروتوكول. بعد اختيار المميزات سيتم اختبار نموذج التعلم باستخدام مصفوفة الارتباك حيث أظهرت المصفوفة أن النموذج يقوم بعمل جيد في تصنيف كلا الفئتين، مع عدد قليل من الأخطاء 87 خطأ في تصنيفMice كـ elephant و193 خطأ في تصنيفelephant كـ Mice يشير هذا إلى أن النموذج دقيق وفعال في التفريق بين الفئتين كما هو موضح بالشكل (١١):



الشكل (١١) مصفوفة الارتباك

ويظهر منحني(ROC) Receiver Operating Characteristic Curve أداء النموذج في التمييز بين الفئات من خلال عرض العلاقة بين معدل الإيجابيات الخاطئة ومعدل الإيجابيات الحقيقية. وتعبر المنطقة الواقعة تحت المنحني عن قدرة المصنف على التمييز بين الفئات الايجابية والسلبية كما هو مبين بالشكل(12):



الشكل (١٢) منحني ROC

ومع وجود AUC بقيمة 0.99، يظهر النموذج قدرة ممتازة على التمييز بين الفئتين، مما يشير إلى أن النموذج متين ويمكن الاعتماد عليه في بيئات العمل.بعد كل هذه الاختبارات للنموذج نقوم بحفظ النموذج كملف sdn_traffic_classifier_model_v2.pkl حيث أن ملفات pkl هي ملفات بايثون يمكن تحميلها لاحقا لاستعادة البيانات الأصلية منها، ونقوم بتطبيق هذا النموذج على المتحكم RYU (المتحكم c1 في طبولوجيا الشبكة المقترحة) عن طريق استدعاء مكتبة pickle بالتالي يصبح المتحكم قادرا على تحديد المسار اللازم للتدفق بناء" على المعلومات التي يقدمها نموذج التعلم الآلي أي بمجرد ورود حمل كبير يقوم المتحكم بإعطاء التعليمات للمبدلات S5،S2 بتوجيه الأحمال الكبيرة في المسار(S2,S8) والمسار(S5,S8) وباقي الأحمال في المسار الأساسي للشبكة.

١٢ الاستنتاجات والتوصيات (Conclusion and Recommendation):

أظهرت النتائج أن استخدام نموذج التعلم الآلي المدرب على بيانات حقيقية مأخوذة من الشبكة يحقق أداءً متميزاً في تصنيف حركة الأحمال الشبكية إلى فئتي "Mice" و"Elephant". يعتمد هذا النموذج بشكل كبير على ميزتي "first_pkt_size" و"Protocol"، مما يتيح لوحدة التحكم القدرة على تحديد المسارات بشكل سريع وفعال، مع الاستفادة من تقنيات التعلم الآلي في تحسين عملية التصنيف.

بناءً على ذلك، يمكن اعتماد هذا النموذج في الأنظمة الحقيقية لتصنيف حركة الأحمال الشبكية بكفاءة عالية، مما يسهم في تحسين إدارة الموارد وتقليل ازدحام الشبكة. تشمل التوصيات توسيع نطاق الدراسة بإدراج ميزات إضافية لتحسين الدقة، واختبار النموذج في بيئات شبكية متنوعة لضمان توافقه مع مختلف أنواع التطبيقات والشبكات، وتعزيز استخدام تقنيات التعلم الآلي المختلفة في الشبكات المعرفة بالبرمجيات.

المراجع (Reference):

Eldhai, A.M., Hamdan, M., Abdelaziz, A., Hashem, I., Marsono, M.N., Hamzah, M. and Jhanjhi, N.Z., 2024. Improved Feature Selection and Stream Traffic Classification Based on Machine Learning in Software-Defined Networks. *IEEE Access*.

Daoud, R., Khalifah, J. and Issa, M., 2023 .Using software-defined networking technologies to improve the performance of traditional networks in serving critical data .*Tishreen University Journal-Engineering Sciences Series* ،(٥) ٤٥ ،pp.69-82

Zaw, H.T. and Maw, A., 2019. Traffic management with elephant flow detection in software defined networks (SDN). *International Journal of Electrical and Computer Engineering*, *9*(4), p.3203.

Awad, M.K., Ahmed, M.H.H., Almutairi, A.F. and Ahmad, I., 2021. Machine learning-based multipath routing for software defined networks. *Journal of Network and Systems Management*, *29*(2), p.18.

Zhao, Y., Li, Y., Zhang, X., Geng, G., Zhang, W. and Sun, Y., 2019. A survey of networking applications applying the software defined networking concept based on machine learning. *IEEE access*, *7*, pp.95397-95417

Raikar, M.M., Meena, S.M., Mulla, M.M., Shetti, N.S. and Karanandi, M., 2020. Data traffic classification in software defined networks (SDN) using supervised-learning. *Procedia Computer Science*, *171*, pp.2750-2759.

Pradhan, B., Srivastava, G., Roy, D.S., Reddy, K.H.K. and Lin, J.C.W., 2022. Traffic classification in underwater networks using sdn and data-driven hybrid metaheuristics. *ACM Transactions on Sensor Networks (TOSN)*, *18*(3), pp.1-15.

S Serag, R.H., Abdalzaher, M.S., Elsayed, H.A.E.A., Sobh, M., Krichen, M. and Salim, M.M., 2024. Machine-Learning-Based Traffic Classification in Software-Defined Networks. *Electronics*, *13*(6), p.1108..

Azab, A., Khasawneh, M., Alrabaee, S., Choo, K.K.R. and Sarsour, M., 2024. Network traffic classification: Techniques, datasets, and challenges. *Digital Communications and Networks*, *10*(3), pp.676-692.

0 Eissa, M.E., Mohamed, M.A. and Ata, M.M., 2024. A robust supervised machine learning based approach for offline-online traffic classification of software-defined networking. *Peer-to-Peer Networking and Applications*, *17*(1), pp.479-506.

1 Shafiq, M., Yu, X., Laghari, A.A., Yao, L., Karn, N.K. and Abdessamia, F., 2016, October. Network traffic classification techniques and comparative analysis using machine learning algorithms. In *2016 2nd IEEE International Conference on Computer and Communications (ICCC)* (pp. 2451-2455). IEEE.

2 Jang, Y., Kim, N. and Lee, B.D., 2023. Traffic classification using distributions of latent space in software-defined networks: An

experimental evaluation. *Engineering Applications of Artificial Intelligence*, *119*, p.105736.

Yacouby, R. and Axman, D., 2020, November. Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models. In *Proceedings of the first workshop on evaluation and comparison of NLP systems* (pp. 79-91).

Nunez-Agurto, D., Fuertes, W., Marrone, L. and Macas, M., 2022. Machine Learning-Based Traffic Classification in Software-Defined Networking: A Systematic Literature Review, Challenges, and Future Research Directions. *IAENG International Journal of Computer Science*, *49*(4).

Ren, Z., Wang, S. and Zhang, Y., 2023. Weakly supervised machine learning. *CAAI Transactions on Intelligence Technology*, *8*(3), pp.549-580.

Khalifah، J. and Issa، M. .،*"Classification Of Network Load Traffic By taking advantage of Software Defined Network"* Tishreen University Journal-Engineering Sciences Series 2023،(6)45 ،pp.115-127.

Bao K. *A survey on software-defined network and openflow:* ، Hao Q،Hu F *From concept to implementation*. IEEE Communications Surveys & Tutorials. 2014 May 22;16(4):2181-206.

Nisar K, Jimson ER, Hijazi MH, Welch I, Hassan R, Aman AH, Sodhro AH, Pirbhulal S, Khan S. *A survey on the architecture, application, and security of software defined networking: Challenges and open issues*. Internet of Things. 2020 Dec 1;12:100289

Tonkal, Ö. and Polat, H., 2021. Traffic Classification and Comparative Analysis with Machine Learning Algorithms in Software Defined Networks. *Gazi University Journal of Science Part C: Design and Technology*, *9*(1), pp.71-83.

Mhamdi, L. and Isa, M.M., 2024. Securing SDN: Hybrid autoencoder-random forest for intrusion detection and attack mitigation. *Journal of Network and Computer Applications*, *225*, p.103868.