

التنبؤ بحركة المرور ضمن المدن الذكية باستخدام خوارزميات التجميع والتعلم العميق

م. خالد قندقجي د. ماهر عباس د. أسماء شعاع

المخلص

تعد المدن الذكية من أهم تطبيقات انترنت الأشياء، حيث تتميز بتنوعها فهي عبارة عن تكامل مجموعة من التطبيقات المختلفة كنظم التنقل الذكية، نظم كشف التلوث وغيرها. تساهم نظم التنقل الذكية وبمساعدة تقنيات الذكاء الصناعي في تنظيم حركة المرور واكتشاف الازدحام مما يوفر المعلومات اللازمة لسائقي المركبات والمشاة في معرفة حالة الطرق، الحوادث الموجودة واختيار المسارات الأفضل للتنقل. قمنا في هذا البحث باستخدام مجموعة البيانات المقدمة من مشروع citypulse وذلك بهدف تطوير نموذج هجين يستخدم خوارزميات التجميع لتحديد الطرق المتشابهة بحركة المرور ومن ثم استخدام البيانات الناتجة في عملية التنبؤ، وبيّنت الدراسة تفوق خوارزمية المحولات على غيرها من خوارزميات التعلم العميق الأخرى، فيما تساهم خوارزميات التجميع في تحسين الدقة، تقليل الزمن وتخفيف الكلفة الحسابية اللازمة لبناء النماذج اللازمة للتنبؤ بحركة المرور المستقبلية من خلال اكتشاف نماذج الحركة بين الطرق المختلفة. وأظهرت خوارزمية DBSCAN أداءً أفضل من باقي الخوارزميات الأخرى في عملية تحسين التنبؤ لاعتمادها على تجميع البيانات بالاعتماد على الكثافة وقدرتها على فترة النقاط الشاذة مما يحسن عمل أداء خوارزميات التعلم العميق.

الكلمات المفتاحية: حركة المرور، التنبؤ، المدن الذكية، خوارزميات التجميع، تحليل البيانات.

م. خالد قندقجي طالب دراسات عليا (دكتوراه) كلية الهندسة المعلوماتية قسم هندسة الشبكات والنظم الحاسوبية جامعة حمص

د. ماهر عباس استاذ مساعد في كلية الهندسة المعلوماتية قسم هندسة الشبكات والنظم الحاسوبية في جامعة حمص

د. أسماء شعاع مدرس في كلية الهندسة المعلوماتية قسم هندسة البرمجيات ونظم المعلومات
جامعة حمص

Traffic Flow Forecasting in Smart Cities using Clustering and Deep Learning Algorithms

Abstract

Smart cities represent one of the most significant applications of the Internet of Things (IoT), characterized by their diversity and integration of multiple systems, such as intelligent transportation and pollution detection systems. Intelligent transportation systems with help of artificial intelligence (AI) techniques play a crucial role in traffic regulation and congestion detection, providing real-time information to drivers and pedestrians about road conditions, accidents, and optimal route selection.

In this study, we used a dataset provided by the CityPulse project to develop a hybrid model that employs clustering algorithms to identify roads with similar traffic patterns and leverages their data for predictive modeling. Our findings demonstrate that Transformer-based algorithms outperform other Deep Learning methods, while clustering techniques contribute to improving accuracy, reducing computational time, and lowering the cost of building traffic flow prediction models by uncovering movement patterns across different roadways.

Notably, the DBSCAN algorithm exhibited superior performance in enhancing prediction accuracy compared to other clustering methods. This can be attributed to its density-based clustering approach and its ability to

filter out outliers, thereby improving the effectiveness of deep learning models in traffic flow forecasting.

KEYWORDS: traffic flow, predicting, smart cities, clustering algorithms, data analysis.

1- مقدمة:

أدى الانفجار السكاني الهائل لزيادة متطلبات وسائل النقل والتي بدورها أدت للكثير من المشاكل كالازدحام المروري، الحوادث وزيادة التلوث. يؤثر الازدحام المروري بشكل عام على حركة المرور حيث يُسبب التأخير وإضاعة الوقت بالإضافة لتعطيل عمل بعض المركبات الأخرى كسيارات الشرطة والإسعاف، مما يؤدي إلى خسائر بشرية ومادية للدول تصل لمعدلات عالية حيث وصلت لـ 80% لدى البعض [2].

ساهم ظهور شبكات انترنت الأشياء (Internet of Things) بتغيير نمط حياة الأفراد حيث باتت هذه الشبكات جزءاً لا يتجزأ في مختلف تطبيقات الحياة. حيث تستخدم في المدن الذكية (Smart Cities)، المجال الصحي وغيرها [30]. ومما ساعد على انتشار هذه الشبكات ظهور الذكاء الصناعي وتطبيقاته القادرة على تحليل البيانات الناتجة عن هذه الشبكات [4]. تعد نظم التنقل الذكية (Intelligent Transportation Systems) ITS من أهم تطبيقات شبكات انترنت الأشياء حيث تستفيد هذه النظم من البيانات التي تولدها هذه الشبكات من خلال تحليلها وذلك باستخدام تقنيات تعلم الآلة (Machine Learning) والتعلم العميق (Deep Learning) لتأمين متطلبات التنقل الآمن عن طريق تحديد المسار الأفضل، إخطار السائقين بالحوادث والازدحام المروري، تنظيم إشارات المرور، تزويد السائقين بمعلومات تخص مواقف السيارات [14]، اكتشاف الشذوذ في حركة المرور [19] وغيرها من التطبيقات الأخرى [9]. حيث تستفيد هذه النظم من

شبكات انترنت الأشياء لتحصيل البيانات المختلفة ضمن المدينة كعدد المركبات، سرعة الحركة، حالة الطقس، التلوث ضمن الجو، معلومات مواقف السيارات وغيرها.

2- مشكلة البحث:

تتأثر حركة المرور ضمن طريق ما بمجموعة من العوامل كحركة الطرق الأخرى، سرعة حركة المركبات، التلوث وغيرها. الأمر الذي يجعل دراسة تأثير هذه الخواص أمراً هاماً في عملية التنبؤ. تتطلب نظم التنقل الذكية نماذج ذات دقة تنبؤ عالية من جهة وذات كلفة حسابية أقل من جهة أخرى الأمر الذي يتطلب بناء نماذج هجينة تقلل كلفة العمليات الحسابية الخاصة بالنماذج اللازمة للتنبؤ بحركة المرور. وذلك بهدف تحديد أوقات الذروة والفراغ ضمن الطرق المدروسة بهدف تحسين حركة المرور ضمن المدينة.

3- هدف البحث:

يهدف البحث إلى دراسة تأثير حركة الطرق المتشابهة على عملية التنبؤ في حركة المرور ضمن الطريق المدروس، ومن ثم دمج النماذج السابقة مع خوارزميات التجميع بهدف إنشاء نماذج أكثر دقة في عملية التنبؤ وأخف كلفة حسابية تتناسب مع بيئة تطبيقات انترنت الأشياء.

4- أهمية البحث:

تساعد نظم التنقل الذكية في تقديم العديد من الفوائد التي يمكننا الاستفادة منها كعملية ضبط زمن إشارات المرور، تنظيم حركة المركبات ضمن المدن وتخفيف الازدحام ومساعدة الجميع للوصول للهدف بأسرع وقت. تساعد عملية الاختيار الأمثل لطرق استقبال البيانات من الحساسات، معالجة وتخزين البيانات وتحليلها على تقديم هذه الوظائف بسهولة ويسر مما يساهم في تحقيق الاستفادة من هذه النظم.

5- السلاسل الزمنية (Time Series):

تُعرف السلسلة الزمنية بأنها مجموعة من نقاط البيانات المرتبة زمنياً، بتعبير آخر هي عبارة عن تسلسل زمني من العينات التي تخص المتحول المدروس [13]، يتم تسجيل هذه النقاط خلال فواصل زمنية متساوية (كل دقيقة، ساعة، يوم...). وتستخدم في الكثير من التطبيقات اليومية في حياتنا كتسجيل استهلاك الطاقة الكهربائية، مراقبة درجة حرارة الجو وغيرها الكثير. [13]. وتتكون السلاسل الزمنية بشكل عام من المركبات التالية:

- 1- قيم السلسلة الزمنية: تمثل السلسلة الزمنية المسجلة.
- 2- الاتجاه (Trend): يمثل التغير الحاصل في السلسلة الزمنية ويمكن أن يكون يتصاعد أو يتنازل حسب طبيعة السلسلة المدروسة.
- 3- الفصلية أو الموسمية (Seasonality): تلتقط هذه المركبة التغيرات الدورية للسلسلة خلال فواصل زمنية ثابتة.
- 4- الخطأ (Residual): تعبر عن الأخطاء الحاصلة وتسمى بـ (White Noise). وهذه المركبة لا يمكن استخدامها كدخل لعمليات التنبؤ، التحليل.

6- الدراسات المرجعية:

6-1 النماذج التقليدية المعتمدة على خوارزميات الانحدار (Regression):

قام الباحثون في [14, 18] بالتنبؤ بحركة المرور المستقبلية باستخدام خوارزميات الانحدار وقدمت هذه الدراسات معدل خطأ مرتفع من أجل التنبؤ بالحركة فيما حسنت الدراسة [18] معدل الخطأ الناتج في بعض الخوارزميات بإضافة خواص أخرى للتنبؤ كسرعة الحركة التي تساهم أيضاً في تقليل حجم النافذة للحصول على نماذج أفضل وبكلفة حسابية أقل.

6-2 نماذج التعلم العميق المعتمدة على الشبكات المتكررة (RNN):

توجد العديد من الدراسات التي تستخدم هذه الفئة من الشبكات في تحليل السلاسل الزمنية كما في [2] التي اقترحت نموذجاً للتنبؤ بالتلوث ضمن المدن الذكية ويحتوي النموذج المقترح على عدد من الطبقات التي تبدأ بمعالجة البيانات ثم مجموعة من طبقات شبكة الذاكرة الطويلة قصيرة الأمد (LSTM) والتي تدرس العلاقة بين سلاسل الدخل وأخيراً طبقة الخرج التي تمثل قيم التنبؤ. حقق النموذج السابق تحسناً على دقة التنبؤ من خلال تقليل الخطأ الناتج بنسبة وصلت لـ 36% من أجل معيار RMSE عن نماذج RNN التقليدية، لكن يعاني النموذج المقترح من كلفة حسابية عالية حيث يحتاج لضعف زمن تنفيذ خوارزمية RNN وهو غير مناسب مع طبيعة تحليل بيانات شبكات انترنت الأشياء. واقترح الباحثون في [3] دراسة مجموعة من النماذج المستخدمة في التنبؤ بالقيم المستمرة كشبكات LSTM، الغابات العشوائية (Random Forest) بالإضافة لخوارزمية التعبئة (Bagging) والتي حققت أقل معدل خطأ من باقي الخوارزميات السابقة، تعاني النماذج المستخدمة من قيم خطأ عالية كما تم استخدام الخوارزميات السابقة بشكل عام دون ضبط قيم البارامترات لتقليل معدل الخطأ.

6-3 نماذج أخرى معتمدة على خوارزميات التجميع، التعلم العميق أو كليهما:

واقترح الباحثون في [4, 5] نظام تنقل ذكي معتمد على مرحلتين الأولى استخدام خوارزميات التجميع لتحديد الطرق المتشابه بشكل ديناميكي حيث قارنت الدراسة بين خوارزميات K-medoids، التجميع الهرمي، التجميع المعتمد على الكثافة والانتشار التقريبي التي أظهرت أداءً أفضل من الخوارزميات السابقة، ومن ثم قامت الدراسة السابقة بدراسة تأثير التنبؤ بحركة المرور ضمن طريق بالاعتماد على شبكات LSTM بالاعتماد على عملية التجميع السابقة وأظهرت الدراسة تحقيق معدل خطأ قليل. لم تقم الدراسة السابقة بالتنبؤ بحركة المرور بشكل واسع واقتصرت على بيانات يوم واحد للاختبار كما يجعل الأسلوب الديناميكي لهذا النموذج مشكلة خاصة مع استخدام شبكات انترنت الأشياء التي تعاني في بعض الحالات من مشاكل في الاتصال. واقترح

الباحثون في [6] نموذجاً للتنبؤ بحركة المرور بالاعتماد على تجميع الطرق ضمن بيان فرعي بالاعتماد على زمن التنقل بين الطرق والطريق الأقصر ومن ثم استخدام هذه البيانات مع الزمان والمكان لتنفيذ عملية التنبؤ. يعاني النموذج السابق من التعقيد ولا يحقق معدل خطأ قليل مقارنة بدراسات أخرى. وقام الباحثون في [7] بمقارنة أداء مجموعة من خوارزميات التجميع على السلاسل الزمنية باستخدام عوامل تشابه كالمسافة الإقليدية والفرق الزمني الديناميكي (DTW (Dynamic Time Warping)) وأظهرت الدراسة تحقيق عدد عناقيد أقل من الدراسة [4] حيث تم تجميع البيانات على شكل فواصل زمنية مختلفة على شكل أيام، ساعات، و 30 دقيقة وركزت الدراسة على خوارزميات التجميع فقط. واقترح الباحثون في [8] بناء نظام تنقل ذكي معتمد على خوارزميات التجميع والتنبؤ بحركة المرور باستخدام مزيج ما بين الشبكات العصبونية التلافيفية (CNN (Convolutional Neural Networks)) وشبكات LSTM، تستفيد الدراسة من استخدام التعلم المتقل (Transfer Learning) لتخفيف عبء التدريب. ولا يحقق النموذج المقترح نتائج تنبؤ جيدة من أجل معدل الخطأ. يوضح الجدول [1] ملخص التقنيات ومجموعات البيانات المستخدمة في الدراسات السابقة:

الجدول (1) ملخص الدراسات المرجعية السابقة

المؤلفين	الخواص المستخدمة	الخوارزميات المستخدمة	سلبيات النموذج
Shouaib et al [14]	الغازات المنبعثة (pollution)	Regression + DL	مكلف جداً بسبب حجم النافذة الكبير للتنبؤ
Kondakji et al [18]	الغازات المنبعثة + سرعة الحركة	Regression only	أخف من الدراسة السابقة بسبب استخدام خواص أخرى في عملية التنبؤ.

مكلف جداً.	RNN only	الغازات المنبعثة	Bello et al [2]
مكلف جداً ودقة قليلة	Regression + LSTM	عدد مركبات الطريق المدروس	Alkarim et al [3]
تعتمد على القيم المتولدة من شبكات IoT بشكل لحظي مما قد يعطل عمل النموذج عند حدوث فقدان بيانات	Clustering + LSTM	عدد مركبات الطرق المتشابهة	Wang [4,5]
معقد ومكلف جداً ويحقق أداءً ضعيفاً مقارنة بدراسات أخرى	محولات + شبكات البيان العصبونية Graph Neural Networks (GNN)	زمن التنقل ضمن الطرق وإحداثيات الطريق	Han et al [6]
قامت بتقييم خوارزميات التجميع دون استخدام الناتج في عملية التنبؤ.	Clustering only	عدد مركبات في الطرق	Toshniwal et al [7]
رغم تخفيف عبء التدريب إلا أن النموذج يعاني من نسبة خطأ عالية. بالإضافة لعدم تقييم أداء خوارزميات التجميع	Clustering + CNN + LSTM with Transfer Learning	عدد المركبات وسرعة الحركة	Zou et al [8]
يحقق النموذج دقة عالية في التنبؤ، ويخفف الكلفة الحسابية بما	Clustering + RNNs + Transformers	عدد المركبات في الطرق المتشابهة	Ours

يتناسب مع شبكات انترنت الأشياء.			
------------------------------------	--	--	--

7- مواد وطرق البحث:

يستعرض هذا القسم توصيف مجموعة البيانات وخوارزميات التجميع والتعلم العميق المستخدمة في الدراسة وآلية تقييم كل منها للحصول على أفضل نتيجة تنبؤ.

7-1 مجموعة البيانات المستخدمة في الدراسة:

تم استخدام مجموعة البيانات من مشروع [9] citypulse EU FP7 لمدينة Aarhus في الدنمارك، وذلك باستخدام مجموعة من الحساسات يبلغ عددها 136 حساس مثبتة في بدايات ونهايات الطرق في المدينة بهدف تجميع بيانات تخص عدد المركبات التي تعبر الطريق، متوسط سرعتها بالإضافة لنسب بعض الغازات في الجو كـ (CO₂, NO₂, ...) خلال فواصل زمنية تبلغ 5 دقائق، يتم تخزين البيانات ضمن ملفات بلاحقة CSV. لكل من حركة المرور مع سرعة الحركة وملفات أخرى لنسب الغازات في الجو لكل طريق عن طريق استخدام بنية سحابية تربط الحساسات مع بنية التخزين وذلك لمدة شهرين متتابعين. بلغ عدد التقارير المتولدة 449 تقرير لكافة الطرق ضمن المدينة وذلك عن طريقة بنية لتجميع البيانات. يوضح الجدول [2] عينة من بيانات حركة المرور المسجلة باستخدام الحساسات.

الجدول (2) عينة من مجموعة بيانات حركة المرور المستخدمة [9]

REPORT_ID	avgSpeed	vehicleCount	TIMESTAMP
158324	50	5	2014-08-01T07:50:00
158324	50	6	2014-08-01T07:55:00
158324	60	4	2014-08-01T08:00:00

يشير حقل report id لمعرفة يخصص الطريق المدروس، كما ويوضح الجدول [3] عينة من بيانات التلوث المستخدمة والتي يتم جمع قراءتها أيضاً باستخدام الحساسات:

الجدول (3) عينة من مجموعة البيانات التلوث المستخدمة [9]

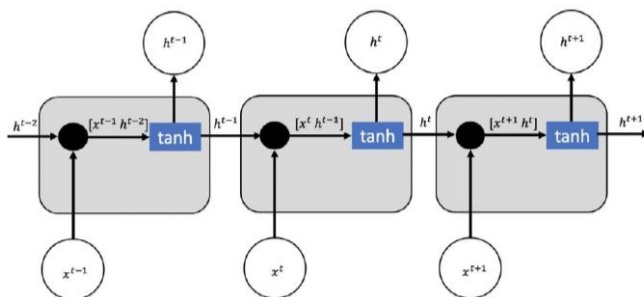
REPORT_ID	latitude	longitude	NO ₂	SO ₂	CO ₂	PM	O ₃	TIMESTAMP
158324	56.2317	10.10498	87	44	49	94	101	2014-08-01T07:50:00
158324	56.2317	10.10498	86	47	48	97	106	2014-08-01T07:55:00
158324	56.2317	10.10498	85	42	49	95	107	2014-08-01T08:00:00

كما وتحوي مجموعة البيانات السابقة معلومات أخرى تخص الطقس، الأحداث في المدينة والمناسبات الاجتماعية والفعاليات ضمنها وحالة مواقف السيارات.

7-2 خوارزميات التعلم العميق المستخدمة في الدراسة:

7-2-1 الشبكات العصبونية المتكررة:

تستخدم الشبكات العصبونية المتكررة (Recurrent Neural Networks) RNN لتحليل البيانات المتسلسلة كمعالجة اللغات الطبيعية والسلاسل الزمنية [10]، وتعتمد على مبدأ مشاركة القيم السابقة في عملية التنبؤ ضمن طبقات الشبكة مما يساهم في عملية تحسين التنبؤ عن الطرق التقليدية الأخرى [1]. يوضح الشكل [1] بنية شبكة RNN التقليدية:



شكل (1) بنية شبكة RNN التقليدية [10]

7-2-2 شبكات الذاكرة الطويلة قصيرة المدى LSTM:

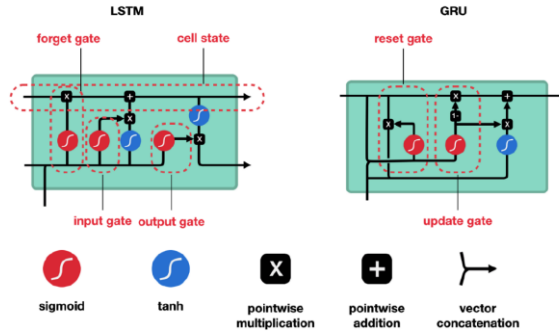
تعاني شبكات RNN التقليدية من مجموعة من المشاكل أهمها مشكلة العلاقة بين القيم السابقة للبيانات والقيم الحالية على المدى الطويل [17]، ولذلك ظهرت هذه الشبكات والتي قامت بتطوير الوحدات السابقة بإضافة خلية ذاكرة إليها مع وحدات لقراءة، نسيان وإضافة قيمة إلى هذه الخلية [1]. تحتوي كل وحدة LSTM على البوابات التالية:

- 1- بوابة الدخل (input gate): تحدد كمية المعلومات التي نريد قراءتها من الدخل الحالي والحالة السابقة [1].
- 2- بوابة النسيان (forget gate): تساعد هذه البوابة تحديد كمية المعلومات التي نحتاج لنسيانها من الذاكرة [1]، حيث نقوم بتطبيق تابع تفعيل على المدخلات وعندما يكون خرج هذا التابع مساوي لـ (1) فإنه يتم تمرير قيمة الخلية السابقة بالكامل وإلا لن تتم عملية النقل [17].
- 3- بوابة الخرج (output gate): تحدد هذه البوابة كمية المعلومات التي يجب استخدامها من القيمة الحالية لتوليد خرج البوابة [1].

3-2-7 وحدات إرجاع البوابة (GRU Gated Recurrent Units):

تعتبر هذه الوحدات مبسطة عن LSTM حيث تستخدم بوابتين فقط بدلاً من ثلاثة وتقوم بتمرير المعلومات المناسبة ضمن قيمة الحالة بدلاً من خلية الذاكرة [1]، وتتكون من:

- 1- بوابة إعادة التعيين (reset gate): تحدد مقدار المعلومات المخفية الواجب اعتبارها ضمن قيمة الحالة عند الوسمة الزمنية الحالية [1].
- 2- بوابة التحديث (update gate): تحدد بوابة التحديث المعلومات التي يجب تجاهلها والمعلومات الجديدة التي يجب تخزينها بتعبير آخر تساعد في تحديد المعلومات التي يجب نقلها للمستقبل [17]. أخيراً تعتبر وحدات GRU أقل كلفة حسابية من وحدات LSTM التي تتفوق من ناحية الدقة على حساب الكلفة. يوضح الشكل [2] بنية كل من وحدتي LSTM و GRU:



شكل (2) وحدات LSTM, GRU [1]

4-2-7 المحولات (Transformers) [1, 15]:

تعتبر المحولات (Transformers) من أهم نماذج التعلم العميق التي صُممت في البداية لمعالجة اللغة الطبيعية، لكنها أثبتت فعاليتها مؤخراً في التعامل مع السلاسل الزمنية. تتميز المحولات

بقدرتها على فهم العلاقات المعقدة بين عناصر السلسلة، حتى لو كانت بعيدة عن بعضها في الترتيب الزمني.

تعاني نماذج RNN بأنواعها المختلفة من عدم قدرتها على تحديد العلاقة بين القيم السابقة البعيدة في السلسلة مع القيم المستقبلية مما يضعف أداء هذه الخوارزميات وهنا تأتي ميزة المحولات التي تتجاوز هذه المشكلة من خلال:

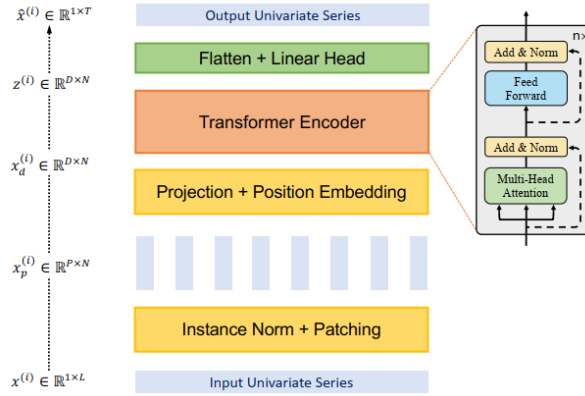
- عدم اعتمادها على الترتيب التتابعي، بل تنتظر إلى السلسلة كاملة دفعة واحدة. بتعبير آخر تقوم المحولات بإضافة معلومات زمنية للمدخلات من خلال ما يعرف بالترميز الموضعي (Positional Encoding) مما يساهم في تعريف النقاط مع اللحظة الزمنية الخاصة بها على عكس خوارزميات RNN التي تعتمد على الترتيب المتتابع للبيانات.
- قدرتها على التركيز على الأجزاء الأهم من السلسلة عبر آلية الانتباه (Attention).
- تعمل بكفاءة عالية على الأجهزة الحديثة لأنها تُمكن من التوازي في المعالجة.

يتكون نموذج المحول من مكونين رئيسيين وهما:

1. المشفر (Encoder): يأخذ بيانات السلسلة الزمنية ويحوّلها إلى تمثيل داخلي غني من خلال عدة طبقات.
2. المفكك (Decoder): يأخذ التمثيل الناتج من المشفر ويستخدمه لتوليد التنبؤات المستقبلية خطوة بخطوة.

كما وتستخدم المحولات ما يعرف بالانتباه الذاتي (Self-Attention) والذي يعتبر من أقوى المميزات الموجودة في المحولات والتي تتمثل في قدرتها على التركيز على النقاط المهمة في السلسلة، بغض النظر عن موقعها. على سبيل المثال، عند التنبؤ بدرجة الحرارة غدًا، قد يولي

النموذج اهتماماً خاصاً لنمط حدث قبل أسبوعين إذا كان مشابهاً للحالة الحالية. يوضح الشكل [3] المخطط العام لنماذج المحولات.



شكل (3) البنية العامة لنموذج المحولات من أجل السلاسل الزمنية [15].

7-3 خوارزميات التجميع وعوامل تقييمها المستخدمة في الدراسة:

تُستخدم خوارزميات التجميع بهدف تشكيل مجموعات من نقاط البيانات من خلال وضعها ضمن عناقيد (Clusters) بحيث يكون التشابه بين النقاط الموجودة في نفس المجموعة عالياً ويكون التشابه أقل مع نقاط البيانات الموجودة في المجموعات الأخرى [10]. وتقسم هذه الخوارزميات لـ:

7-3-1 خوارزميات التقسيم (Partitioning Algorithms):

من أهم الخوارزميات المستخدمة في هذا النوع هي K-Means, K-Medoids ويكمن الفرق بينهما أن خوارزمية K-Means تختار المراكز الابتدائية بشكل عشوائي مقارنة بالخوارزمية الأخرى التي تختار المراكز من نقاط البيانات الموجودة [10]. كما ويمكننا إعادة تشغيل الخوارزمية بمراكز ابتدائية مختلفة من خلال استخدام معامل (n_init). ثم تقوم الخوارزميتين وبشكل تكراري اسناد النقاط للمراكز الأقرب لها، تليها عملية تحديث المراكز حتى تستقر المراكز. ومن

الخوارزميات الأخرى المستخدمة خوارزمية الانتشار التقريبي (Affinity Propagation) [4]، والفرق بين هذه الخوارزمية والخوارزميتين السابقتين أنها لا تحتاج عدد العناقيد كدخل لها، وهنا تكون جميع نقاط البيانات مرشحة لتكون مركز عنقود بنفس الوقت تبحث عن مركز جديد لها. في كل مرحلة من مراحل عمل الخوارزمية تقوم كل نقطة أو غرض من مجموعة البيانات بحساب المسؤولية (Responsibility) والإتاحة (Availability) والتي تخص كل زوج من النقاط الموجودة ضمن مجموعة البيانات. تعبر المسؤولية بين نقطتين $r(i, k)$ عن وزن يدل على مقدار مناسبة (k) لتكون مركز للعقدة (i) أما الإتاحة بين النقطتين $a(i, k)$ تعبر عن وزن يمثل مقدار مناسبة (i) لتتنتمي للمركز الذي تتواجد فيه العقدة (k). وتكون قيمة العنقود لكل نقطة تمثل القيمة العظمى لمجموع القيمتين السابقتين.

7-3-2 طرائق التجميع الهرمي (Hierarchical Clustering) [10]:

تعمل هذه الطرق على تجميع البيانات بشكل هرمي (بتعبير آخر على شكل شجرة من العناقيد). ويتكون هذا النوع من طريقتين رئيسيتين (Top - Down) أو (Bottom - Up)، ومن أمثلة النوع الأخير خوارزمية Agglomerative Clustering والتي تبدأ باعتبار كل نقطة تشكل عنقوداً مستقلاً عن العقد الأخرى، وبشكل متتالي يتم جمع نقاط البيانات مع بعضها البعض الأقرب فالأقرب بناءً على مقياس تشابه معين حتى يتحقق شرط معين أو تصبح جميع النقاط ضمن عنقود وحيد. وتعتمد هذه الخوارزمية في عملها على نواة تحدد طبيعة التجميع بناءً على القيم العظمى أو الصغرى للتشابه أو القيمة المتوسطة للتشابه وتحدد باستخدام البارامتر (linkage).

7-3-3 التجميع المعتمد على الكثافة (Density Based Clustering) [10]:

تولد الخوارزميات السابقة عناقيد كروية بشكل عام، ولإيجاد عناقيد بإشكال أخرى نلجأ لاستخدام أنواع أخرى من الخوارزميات وأشهرها DBSCAN (Density-Based Spatial Clustering)

(of Applications with Noise) والتي تستخدم مجموعة البيانات ونصف القطر المعبر عن مجال الجوار لكل غرض وعتبة الكثافة. وتعمل الخوارزمية كما يلي:

1- تبدأ الخوارزمية بتحديد النقاط الأساسية وهي النقاط التي لديها على الأقل MinPts من النقاط في نطاق ϵ .

2- إذا كانت النقطة أساسية، يتم توسيع المجموعة عن طريق ضم جميع الجيران الذين هم أيضاً نقاط أساسية أو نقاط حدودية أما النقاط التي لا تنتمي إلى أي مجموعة تُعتبر ضجيج.

8- التجارب والنتائج ومناقشتها:

8-1 تجهيز البيانات:

قبل تطبيق الخوارزميات السابقة لابد لنا من القيام بمجموعة من المراحل للتأكد من أن البيانات خالية من المشاكل وذلك باستخدام مكتبة تحليل البيانات بلغة بايثون (Pandas) [11]، تعد مشكلة البيانات المفقودة من أهم مشاكل بيانات انترنت الأشياء والتي تحدث نتيجة عدم تمكن الحساسات من إرسال البيانات نتيجة مشكلة في الاتصال أو تم إهمال القراءة بسبب ضجيج البيانات. توجد العديد من الحلول لاستبدال القيم المفقودة في مجموعة البيانات وسنقوم باستخدام طريقة الإكمال الخطي (Linear Interpolation) والتي تعتمد على وجود علاقة خطية بين القيمة المفقودة والقيم السابقة واللاحقة لها وأظهرت هذه الطريقة أداءً أفضل من طرق أخرى كما في [18] بحيث تكون قيمة الخطأ ما بين القيم الفعلية والقيم المستبدلة أقل ما يمكن. تساهم عملية إدخال القراءات السابقة للحساسات في تحسين عملية التنبؤ (إدخال عامل الزمن) حيث حسنت هذه الآلية عملية التنبؤ كما في [14,18]. يوضح الجدول [4] شكل مجموعة البيانات من أجل قيمة النافذة تساوي 1 على سبيل المثال، بتعبير آخر بفرض أننا حصلنا على القراءات في لحظة ما (t) ونريد

التنبؤ بحركة المرور في هذه اللحظة فإننا نستخدم قراءات الحساس المسجلة في هذه اللحظة الحالية واللحظة السابقة (t-1).

الجدول (4) التنبؤ بحركة المرور بالاعتماد على حركة الطرق المجاورة

Avg speed	Road 1 (t)	Road 2 (t)	Road 3 (t)	Road 1 (t-1)	Road 2 (t-1)	Road 3 (t-1)	...	vehicleCount (output)
47	10	0	3	4	4	3	...	5
50	19	8	5	10	0	3	...	6
60	3	6	2	3	6	2	...	4

بعد عملية التهيئة السابقة نقوم بتقسيم مجموعة البيانات لمجموعتين (80% من البيانات للتدريب و20% للاختبار على الترتيب وذلك بحسب طبيعة بيانات السلاسل الزمنية). أما من أجل تجميع الطرق المتشابهة ضمن عناقيد (clusters). سنقوم بدراسة تأثير تجميع البيانات بشكل ربع يوم، نصف يوم، يومي لحركة المرور. يوضح الشكل [4] الشكل الجديد للبيانات والذي يحوي متوسط الحركة المكافئ لكل طريق بشكل يومي من أجل الطرق المدروسة على حدا بحيث يكون معرف الطريق هو دليل كل سطر على سبيل المثال وذلك لاستخدامها في عملية التجميع.

day	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0	...
road_id											
158324	5.085062	3.769097	3.335938	5.237847	5.136522	3.366551	3.221354	4.853299	3.748264	3.427951	...
158355	5.261411	4.083333	3.619792	5.904514	5.555652	3.571924	3.428819	5.477431	3.980035	4.010417	...
158386	3.447484	2.450751	2.488715	3.436632	3.615523	2.086207	2.117391	3.280105	2.488294	2.208333	...

الشكل (4) متوسط حركة الطرق اليومية المستخدمة في التجميع

8-2 التنبؤ بحركة المرور بالاعتماد على حركة المرور في جميع طرق المدينة:

8-2-1 ضبط بارامترات النماذج المدروسة:

توجد العديد من البارامترات التي يمكن ضبطها أثناء الدراسة وتم اختيار أهم البارامترات التي تؤثر على أداء الخوارزميات كحجم الإطار (النافذة) التي يجب علينا استخدامها لتوليد البيانات اللازمة للتنبؤ وبعض البارامترات المشتركة كعدد الطبقات، عدد العصبونات وغيرها. تم تجربة فضاء من القيم الممكنة لكل بارامتر بحيث تحقق الخوارزميات أفضل أداء لها. يوضح الجدول [5] البارامترات التي تحقق من خلالها كل خوارزمية الأداء الأفضل لها. وللحصول على هذه القيم قمنا بالبحث ضمن مجموعة من القيم الممكنة لكل من هذه البارامترات والتي تحقق أفضل أداء ممكن لكل خوارزمية.

الجدول (5) البارامترات التي تحقق أفضل أداء لكل من الخوارزميات المدروسة

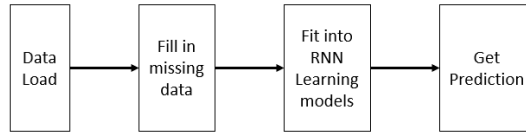
حجم النافذة	حجم التجميع (batch size)	معدل التعلم	عدد العصبونات	معدل dropout	عدد الطبقات المخفية	الخوارزمية
11	32	0.0002	200	0.2	2	LSTM
31	64	0.0002	200	0.1	2	GRU
28	16	0.0001	200	0.2	1	RNN

تساهم طبقة dropout في منع النموذج من موثمة البيانات بشكل كبير (Overfitting) وذلك من خلال إهمال نسبة من خرج الطبقات الأخرى أثناء التدريب مما يساهم في تحسين أداء الشبكات ونقوم بتحديد نسبة العقد التي نريد إهمالها من خلال قيمة احتمال. أخيراً، يوضح الجدول [6] قيم البارامترات الخاصة بخوارزمية المحولات.

الجدول (6) البارامترات المستخدمة من أجل المحولات

المعامل	فضاء التمثيل	عدد رؤوس الانتباه	عدد طبقات المحول	dropout	معدل التعلم	حجم التجميع	حجم النافذة
القيمة	64	4	1	0.1	0.001	64	22

أخيراً يوضح الشكل [5] النموذج العام المستخدم في التجربة الأولى:



شكل (5) المخطط الصندوقي للتجربة الأولى

8-2-2 مقاييس التقييم:

لتقييم أداء هذه الخوارزميات سنستخدم مجموعة من المعايير التي تقوم بتقييم أداء خوارزميات التعلم العميق كما في [14] وهي (RMSE) (Root Mean Squared Error) وهو عبارة عن الجذر التربيعي لمتوسط الفروق بين القيم الحقيقية (O_i) والقيم الناتجة عن التنبؤ (P_i) ويعطى بالمعادلة التالية:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (O_i - P_i)^2} \quad (1) \quad [14]$$

يمثل المعيار السابق الجذر التربيعي لمربع الفروق بين القيم الفعلية والقيمة الناتجة عن التنبؤ وبالتالي يوضح مدى قرب القيم الناتجة عن التنبؤ من القيم الفعلية. كما ويمكن استخدام معامل Mean Absolute Error (MAE) والذي يعبر عن المتوسط الحسابي للقيمة المطلقة للفروق بين القيم الحقيقية والقيم الناتجة عن التنبؤ.

$$MAE = \frac{1}{n} \sum_{i=1}^n |O_i - P_i| \quad (2) \quad [14]$$

أخيراً يمكن استخدام معامل R2 ويعطى بالمعادلة التالية:

$$R^2 = 1 - \frac{\sum_{i=1}^n (O_i - P_i)^2}{\sum_{i=1}^n (O_i - \bar{O})^2}, \text{ where } \bar{O} = \frac{1}{n} \sum_{i=1}^n O_i \quad (3) \quad [14]$$

يسمى المعيار السابق بمركبة التحديد ويوضح المعامل السابق مدى مناسبة البيانات لنموذج الانحدار أو مدى جودة تفسير البيانات الموجودة لدينا باستخدام متحول الدخل وكلما كانت القيمة أكبر يدل على أداء أفضل للخوارزمية.

3-2-8 النتائج:

تحقق المحولات أداءً أفضل من باقي الخوارزميات المدروسة الأخرى بفضل آلية الانتباه والتي تقوم بدراسة السلسلة ككل ويهتم بتموضع النقاط زمنياً مقارنةً بالتتابع المتولد في خوارزميات RNN الأخرى، فيما تحقق خوارزمية RNN أداءً أسوأ من باقي الخوارزميات بسبب معانيتها من التبعيات الطويلة في السلسلة الأمر الذي يضعف أداءها. فيما تحتاج خوارزمية GRU لعدد نوافذ أكثر لتحقيق دقة أفضل وقريبة من خوارزمية LSTM التي تتفوق على الخوارزمية السابقة. يوضح الجدول [7] تقييم أداء الخوارزميات المدروسة من أجل عوامل التقييم المختلفة.

الجدول (7) تقييم أداء خوارزميات التعلم العميق بالاعتماد على بيانات جميع الطرق

R2	MAE	RMSE	الخوارزمية
0.7504	1.6711	2.5734	LSTM
0.7233	1.8027	2.664	GRU
0.6123	2.1558	3.0451	RNN

0.8292	1.3917	2.1395	Transformer
--------	--------	--------	-------------

8-3 التنبؤ بحركة المرور بالاعتماد على حركة المرور في الطرق المتشابهة:

تتكون هذه التجربة من مرحلتين الأولى تشمل تحديد الطرق المتشابهة بحركة المرور وذلك باستخدام خوارزميات التجميع ودراسة تأثير المعاملات وآلية التجميع الزمنية للبيانات. أما في المرحلة الثانية فنقوم باستخدام بيانات الطرق المتشابهة في عملية التنبؤ بدلاً من الاعتماد على بيانات كافة الطرق في المدينة.

8-3-1 ضبط بارامترات خوارزميات التجميع:

لأجل مقارنة أداء الخوارزميات مع بعضها البعض سنقوم باستخدام عدد العناقيد الناتجة عن خوارزمية الانتشار التقريبي ونلاحظ أداء الخوارزميات الأخرى عند تجميع البيانات ضمن نفس العدد من العناقيد. فيما تم استخدام أفضل البارامترات التي تحقق أفضل قيم من أجل مقياس التقييم المستخدم. حيث يوضح الجدول [8] البارامترات المستخدمة من أجل خوارزميات التجميع المدروسة.

الجدول (8) البارامترات المستخدمة لكل من خوارزميات التجميع المدروسة:

المعاملات	الخوارزمية
init= kmeans++, n_init = auto	K-Means
linkage = average	التجميع الهرمي
-	الانتشار التقريبي
eps = 1, min_samples = 10	DBSCAN

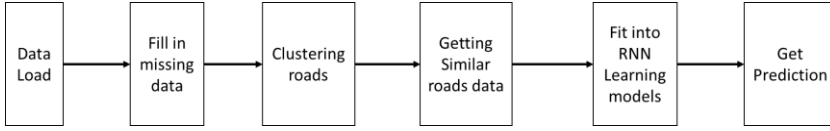
8-3-2 مقاييس التقييم:

يتم تقييم أداء خوارزميات التجميع من خلال عدة عوامل أشهرها معامل Silhouette Coefficient [4] وكلما كانت قيمة المعامل أكبر كلما كانت جودة العناقيد أفضل والذي يحسب كما يلي:

$$s(o) = \frac{b(o) - a(o)}{\max\{a(o), b(o)\}} \quad (4) \quad [4]$$

- $a(i)$: متوسط المسافة بين النقطة والنقاط في مجموعتها.
- $b(i)$: أقل متوسط مسافة بين النقطة وأقرب مجموعة أخرى.

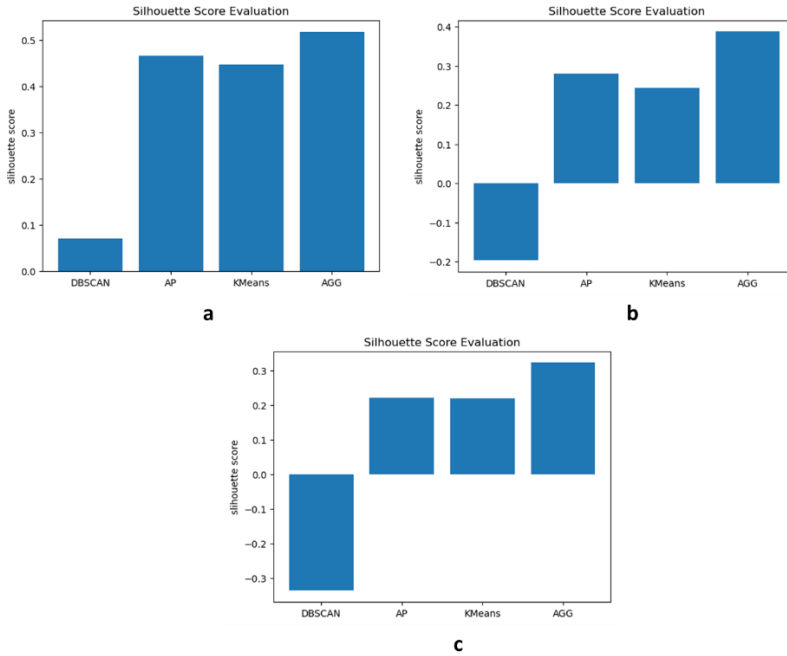
يوضح الشكل [6] المخطط الصندوقي لبناء نموذج للتنبؤ معتمد على خوارزميات التجميع والتعلم العميق.



الشكل (6) المخطط الصندوقي للتجربة الثانية

8-3-3 تقييم أداء خوارزميات التجميع لتحديد نماذج الحركة ضمن الطرق:

يوضح الشكل [7] نتائج أداء كل من الخوارزميات المدروسة عند استخدام عدد العناقيد الناتج عن خوارزمية الانتشار التقريبي:



الشكل (7) تقييم أداء خوارزميات التجميع عند استخدام عدد عناقيد موحد

a. تجميع يومي b. تجميع نصف يوم c. تجميع ربع يوم

نلاحظ أن أداء الخوارزميات يتأثر بحسب طبيعة التجميع حيث تحقق الخوارزميات أفضل أداء لها عند تجميع البيانات بشكل يومي وينخفض مع تجميع البيانات بالاعتماد على متوسط الحركة بشكل نصف يومي أو ربع يومي وذلك بشكل مشابه للدراسة [7] تحقق آلية التجميع بالاعتماد على عدد العناقيد الناتجة عن خوارزمية الانتشار التقريبي أداءً أفضل من استخدام عدد العناقيد المستخدمة في [7] فيما تحقق DBSCAN أداءً أقل بسبب عزلها للضجيج مما يقلل قيمة المعامل مما يزيد من تباعد النقاط عن المجموعات الأخرى. كما وتحقق طريقة تجميع الطرق بناءً على حركة الطرق بشكل يومي ونصف يومي تحسن أداء بعض الخوارزميات عن التجميع الديناميكي

المستخدمة في [4]. سنقوم باستخدام نفس المعاملات المستخدمة في الجدولين [4,5] سابقاً من أجل خوارزميات التعلم العميق.

8-3-4 تحسين عملية التنبؤ بحركة المرور باستخدام خوارزميات التجميع:

سنحدث عن تأثير استخدام بيانات الطرق المتشابهة في عملية التنبؤ بدلاً من استخدام كافة الطرق الموجودة في المدينة، كما توضح الجداول [9, 10, 11, 12].

الجدول (9) ناتج تقييم خوارزمية LSTM مع استخدام خوارزميات التجميع

R2	MAE	RMSE	الخوارزمية (LSTM)
0.7542	1.6773	2.5004	Agglomerative Clustering
0.7513	1.6732	2.5154	Affinity Propagation
0.7292	1.7727	2.6244	Kmeans
0.54336	1.19575	1.8323	DBSCAN

الجدول (10) ناتج تقييم خوارزمية GRU مع استخدام خوارزميات التجميع

R2	MAE	RMSE	الخوارزمية (GRU)
0.7587	1.6763	2.4780	Agglomerative Clustering
0.7554	1.6817	2.4952	Affinity Propagation
0.7127	1.8238	2.7039	Kmeans
0.5503	1.2218	1.8188	DBSCAN

الجدول (11) ناتج تقييم خوارزمية RNN مع استخدام خوارزميات التجميع

R2	MAE	RMSE	الخوارزمية (RNN)
0.7145	1.8495	2.6953	Agglomerative Clustering
0.7087	1.8445	2.7228	Affinity Propagation
0.6493	1.9824	2.9875	Kmeans

0.5407	1.2721	1.8378	DBSCAN
--------	--------	--------	--------

الجدول (12) ناتج تقييم خوارزمية المحولات مع استخدام خوارزميات التجميع

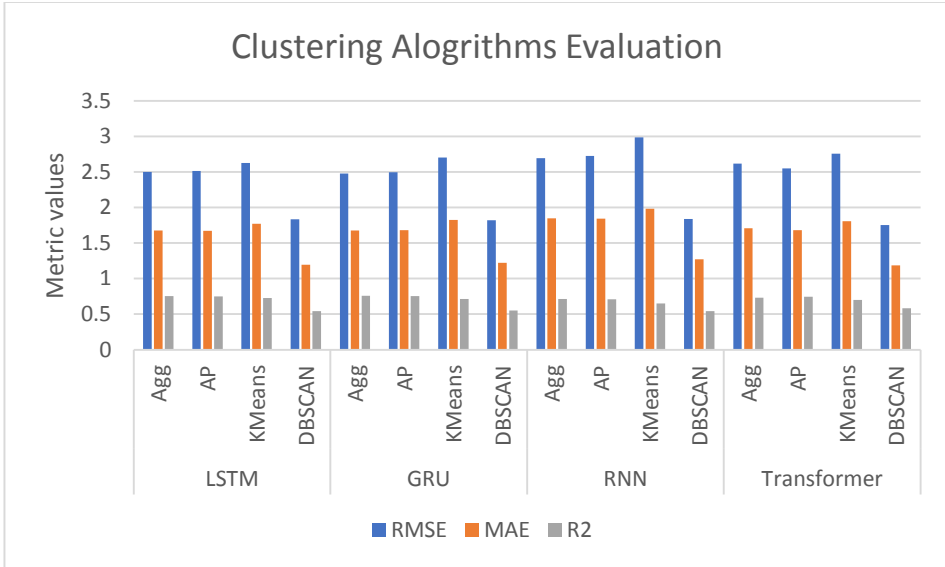
R2	MAE	RMSE	الخوارزمية (Transformer)
0.7307	1.7089	2.6175	Agglomerative Clustering
0.7442	1.6791	2.5514	Affinity Propagation
0.7016	1.8061	2.7555	Kmeans
0.5821	1.1870	1.7530	DBSCAN

يوضح الجدول [13] عدد الطرق المستخدمة في التنبؤ بالاعتماد والناتج عن كل خوارزمية:

الجدول (13) عدد الطرق المستخدمة في عملية التنبؤ بعد تطبيق خوارزميات التجميع

عدد الطرق المستخدمة في التنبؤ	الخوارزمية
448	No clustering
68	Agglomerative Clustering
68	Affinity Propagation
35	Kmeans
265	DBSCAN

يظهر الشكل [8] تقييم أداء تأثير خوارزميات التجميع على أداء خوارزميات التعلم العميق:



الشكل (8) تأثير استخدام خوارزميات التجميع على أداء خوارزميات التعلم العميق

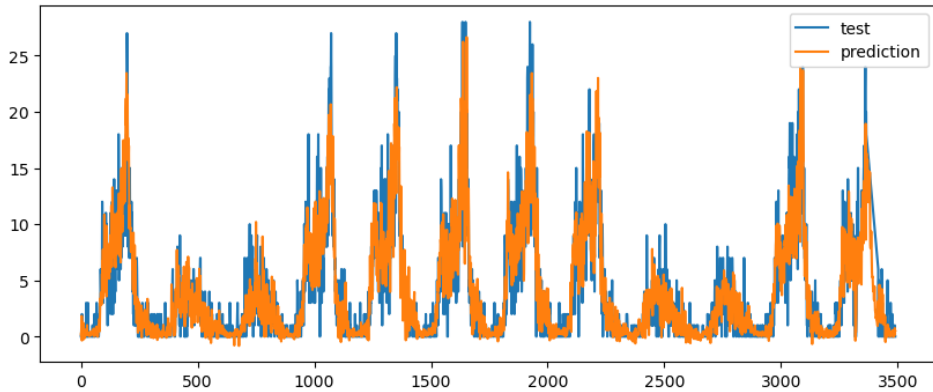
نلاحظ أن استخدام خوارزمية DBSCAN تساهم في تقليل الخطأ الناتج عن التنبؤ لكل من معاملات (RMSE, MAE) بسبب طريقة التجميع المعتمدة على الكثافة على عكس الخوارزميات الأخرى وكذلك قدرتها على التخلص من الضجيج. فيما تحقق الخوارزمية السابقة قيمة R2 أقل من الخوارزميات الأخرى بسبب تقارب البيانات من بعضها البعض مما قلل قيمة التباين في معامل R2 على عكس الخوارزميات الأخرى التي تملك تبايناً أعلى من أجل هذا المعامل بسبب بعض الضجيج وطريقة التجميع المعتمدة على المسافة. كما وتساهم خوارزميات التجميع في تقليل كلفة وتعقيد بناء نماذج التعلم العميق للتنبؤ بحركة المرور مما يتناسب مع تطبيقات انترنت الأشياء. يوضح الجدول [14] كيفية مساهمة خوارزميات التجميع من خلال تقليل عدد البارامترات التي يتم تدريبها من جهة وتخفيف العمليات الحسابية الأساسية على الأرقام الحقيقية (Floating Point Operations) FLOPs وذلك من أجل خوارزمية LSTM:

الجدول (14) تأثير استخدام خوارزميات التجميع على الكلفة الحسابية لبناء نماذج التعلم

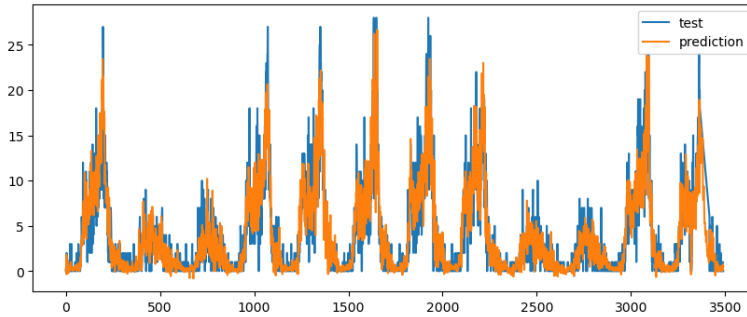
العميق

الخوارزمية	عدد البارامترات التي يتم تدريبها	عدد العمليات الحسابية (FLOPs)	نسبة تخفيف الكلفة الحسابية
No clustering	842,601	296,531,232	-
AGG-Clustering	539,401	189,804,832	35.98%
Affinity Propagation	539,401	189,804,832	35.98%
Kmeans	513,001	180,512,032	39.12%
DBSCAN	697,001	245,280,032	17.28%

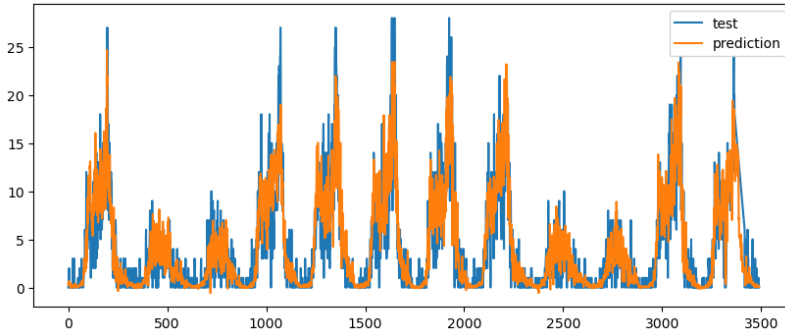
توضح الأشكال [9, 10, 11, 12] شكل الإشارة الناتجة عن استخدام المحولات مع خوارزميات التجميع المختلفة في عملية التنبؤ:



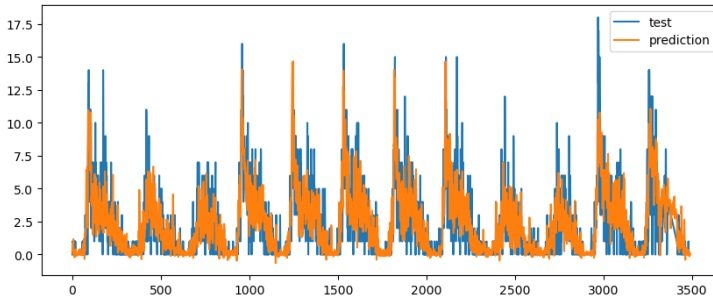
الشكل (9) إشارة التنبؤ الناتجة عن خوارزمية المحولات مع AGG clustering



الشكل (10) إشارة التنبؤ الناتجة عن خوارزمية المحولات مع AP clustering



الشكل (11) إشارة التنبؤ الناتجة عن خوارزمية المحولات مع KMeans clustering



الشكل (11) إشارة التنبؤ الناتجة عن خوارزمية المحولات مع DBSCAN clustering

9- مناقشة النتائج:

تساهم خوارزميات التجميع والتعلم العميق في تحسين جودة التنبؤ بحركة المرور حيث حققت الخوارزميات المعتمدة على المحولات أداءً أفضل من خوارزميات RNN التقليدية، فيما تساعد خوارزميات التجميع على تحسين أداء بعض الخوارزميات من جهة وتخفف تعقيد هذه الخوارزميات من جهة أخرى مع الحفاظ على دقة مقبولة وهذا ما يتناسب مع بيئة انترنت الأشياء التي تتطلب نماذج قليلة الكلفة حسابياً. تحقق النماذج المعتمدة على خوارزمية DBSCAN أداءً أفضل في التنبؤ من الخوارزميات الأخرى نظراً لآلية التجميع المعتمدة على الكثافة بما يشبه حركة المرور في الطرق التي تعتمد على كثافة المركبات في الطريق وإزالة الضجيج وعدم التقيد بشكل محدد للعناقيد الناتجة عن التجميع على عكس الخوارزميات الأخرى التي تفرض تشكيل المجموعات على شكل كروي مما يحقق دقة أفضل في عملية التنبؤ.

10- المقترحات والتوصيات المستقبلية:

توجد العديد من الخواص الأخرى التي يمكن دراستها في مجموعة البيانات وتأثيرها على حركة المرور كبيانات الطقس، المناسبات الاجتماعية في المدينة، مواقف السيارات وغيرها.. كما وتعاني مجموعة البيانات من كونها صغيرة نسبياً لذا يمكن دراسة تأثير النماذج على مجموعة بيانات أكبر وهذا يساهم في تحسين أداء خوارزميات التعلم العميق التي تعتمد على وجود كميات كبيرة من البيانات. أخيراً يمكن بناء لوحة تحكم (Dashboard) لعرض تفاصيل حركة المرور ضمن المدينة وتوضيح اختيار المسار الأقصر بناءً على التنبؤ بحركة المرور المستقبلية.

11- المراجع:

[1] Joseph, M., 2022. Modern Time Series Forecasting with Python: Explore industry-ready time series forecasting using modern machine learning and deep learning. Packt Publishing Ltd.

- [2] BELLO, R.D., SULAIMON, H.A., ABDULRAHMAN, M.L. and AHMAD, M.A., 2020. DEEP RECURRENT NEURAL NETWORK FOR POLLUTION FORECASTING IN SMART CITIES. BIMA JOURNAL OF SCIENCE AND TECHNOLOGY (2536-6041), 4(02), pp.199-205.
- [3] Alkarim, A.S., Al-Ghamdi, A.S.A.M. and Ragab, M., 2024. Ensemble Learning-based Algorithms for Traffic Flow Prediction in Smart Traffic Systems. Engineering, Technology & Applied Science Research, 14(2), pp.13090-13094.
- [4] Wang, Z., Thulasiraman, P. and Thulasiram, R., 2019, July. A dynamic traffic awareness system for urban driving. In 2019 international conference on Internet of things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData) (pp. 945-952). IEEE.
- [5] Wang, Z. and Thulasiraman, P., 2019, November. Foreseeing congestion using LSTM on urban traffic flow clusters. In 2019 6th international conference on systems and informatics (ICSAI) (pp. 768-774). IEEE.
- [6] Han, S.Y., Sun, Q.W., Zhao, Q., Han, R.Z. and Chen, Y.H., 2022. Traffic forecasting based on integration of adaptive subgraph

reformulation and spatio-temporal deep learning model. Electronics, 11(6), p.861.

[7] Toshniwal, D., Chaturvedi, N., Parida, M., Garg, A., Choudhary, C. and Choudhary, Y., 2020. Application of clustering algorithms for spatio-temporal analysis of urban traffic data. Transportation Research Procedia, 48, pp.1046–1059.

[8] Zou, X. and Chung, E., 2024. Traffic prediction via clustering and deep transfer learning with limited data. Computer-Aided Civil and Infrastructure Engineering.

[9] CityPulse Dataset Collection · A collection of semantically annotated datasets for the CityPulse EU FP7 Project <http://iot.ee.surrey.ac.uk:8080/index.html> last access 12/7/2024

[10] Jiawei, H., Jian, P. and Hanghang, T., 2023. Data mining: concepts and techniques. Morgan kaufmann.

[11] VanderPlas, J., 2022. Python data science handbook (No. PUBDB–2023–07774). O'Reilly.

[12] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. and Vanderplas, J., 2011. Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, pp.2825–2830.

[13] Peixeiro, M., 2022. Time series forecasting in python. Simon and Schuster.

[14] Shouaib, M., Metwally, K. and Badran, K., 2023. An Enhanced Time-dependent Traffic Flow Prediction in Smart Cities. *Advances in Electrical & Computer Engineering*, 23(3).

[15] Nie, Y., Nguyen, N.H., Sinthong, P. and Kalagnanam, J., 2022. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*.

[16] م. آلاء السباعي، د. محسن حسين، د. وسيم رمضان، 2022. أنظمة تحليل بيانات الزمن الحقيقي الضخمة الناتجة عن انترنت الأشياء. مجلة جامعة حمص-سلسلة العلوم الهندسية الميكانيكية والكهربائية والمعلوماتية، 44(14).

[17] ميلاد وزان، ترجمة الدكتور علاء طعيمة، التعلم العميق المبادئ والمفاهيم والأساليب 2024، كلية علوم الحاسوب وتكنولوجيا المعلومات، جامعة القادسية.

[18] م. خالد قندقجي، د. ماهر عباس، د. أسماء شعار، 2025. تحليل حركة المرور ضمن المدن الذكية. مجلة جامعة حمص-سلسلة العلوم الهندسية الميكانيكية والكهربائية والمعلوماتية، 47(2).

[19] م. حلا إبراهيم، د. رانيا لطفي، د. أسماء شعار، 2023. تحليل السلاسل الزمنية لاكتشاف الشذوذ ضمن حركة المرور. مجلة جامعة حمص-سلسلة العلوم الهندسية الميكانيكية والكهربائية والمعلوماتية، 45(20).