



Rapport du Projet PFE :
Système de Prévision Météorologique Intelligent basé
sur l'Internet des objets au Service de la Ville
Intelligente

Professeur : Souhila Arib

Tutrice : Kouah Sofia

Etudiant : Es-Sadany Yassine

Groupe : ING3 IA 2

25 Juin 2024

Table des matières

Remerciements	3
Dédicace	4
Liste des Abréviations	5
Introduction	6
1 État de l’art en Prévision Météorologique	7
1.1 Introduction	7
1.2 Modèles de Prévision Météorologique	7
1.2.1 Modèles Statistiques	7
1.2.2 Modèles Basés sur la Physique	7
1.2.3 Modèles Hybrides	8
1.2.4 Tableau Comparatif des Modèles	8
2 Données Multimodales	9
2.1 Dataset Utilisé	9
3 Solution Proposée	11
3.1 Introduction aux Réseaux de Neurones LSTM	11
3.1.1 Architecture des LSTM	11
3.2 Implémentation de notre Modèle LSTM	13
3.2.1 Préparation des Données	13
3.2.2 Architecture du Modèle	13
3.2.3 Entraînement du Modèle	14
3.3 Résultats Obtenus	14
3.3.1 Évaluation des Performances	14
3.3.2 Commentaires sur les Résultats	16

3.4	Test du modèle	16
-----	--------------------------	----

Remerciements

Je souhaite exprimer ma reconnaissance envers tous ceux qui ont participé de près ou de loin à la concrétisation de ce projet.

Mes encadrants, Madame Souhila Arib et Madame Kouah Sofia sont particulièrement reconnaissant pour leurs conseils avisés et leurs soutien constant tout au long de ce projet.

Je souhaite aussi exprimer ma gratitude envers ma famille et mes amis pour leur soutien sans faille et leurs encouragements.

Dédicace

Ce projet est dédié à mes parents, en reconnaissance de leur amour, de leur soutien et de leurs sacrifices. À mes amis, je tiens à exprimer ma gratitude pour leur présence et leur soutien moral pendant mes études.

Liste des Abréviations

IoT	Internet of Things
PFE	Projet de Fin d'Études
RMSE	Root Mean Square Error
MAE	Mean Absolute Error
NWP	Numerical Weather Prediction
RNN	Réseaux de Neurones Récurrents
LSTM	Long Short-Term Memory

Introduction

Les villes doivent faire face à de nombreux défis en matière de gestion des ressources et de durabilité en raison de l'urbanisation rapide et de l'augmentation de la population urbaine. Les grandes villes intelligentes ont pour objectif d'exploiter les technologies de pointe afin d'améliorer la qualité de vie de leurs résidents tout en maximisant l'efficacité des services locaux. La prévision des conditions météorologiques est essentielle dans différents domaines de la gestion urbaine, de la planification des infrastructures à la gestion des situations d'urgence.

L'Internet des objets (IoT) offre la possibilité de recueillir des informations en temps réel à partir de différents capteurs installés à travers la ville. Il est possible d'exploiter ces informations afin de créer des modèles de prévision météorologique plus précis et en direct. Cependant, lors de cette initiative, nous n'avons pas pu disposer du matériel IoT requis pour la collecte des données. Ainsi, notre modèle a été développé et testé en utilisant un dataset historique disponible publiquement.

Le dataset utilisé dans ce projet est le Algiers Weather Data (2002-2023), qui contient des données météorologiques historiques pour Alger couvrant la période de janvier 2002 à août 2023. Les variables incluses sont la température, les précipitations, la vitesse du vent, etc.

Chapitre 1

État de l'art en Prévision Météorologique

1.1 Introduction

Cet état de l'art offre une analyse des modèles couramment employés pour prédire les conditions météorologiques, en se référant à des articles académiques pertinents.

1.2 Modèles de Prévision Météorologique

1.2.1 Modèles Statistiques

Les prédictions basées sur les modèles statistiques sont basées sur l'analyse des liens entre les variables météorologiques historiques.

— **Machine Learning (Random Forest, Support Vector Machines) :**

Ils sont couramment utilisés pour prédire les conditions météorologiques en raison de leur aptitude à représenter des relations non linéaires complexes. [1].

1.2.2 Modèles Basés sur la Physique

Les processus physiques de l'atmosphère sont représentés par ces modèles qui utilisent des équations différentielles.

- **Modèle de Prévision Numérique du Temps (NWP)** : Modélise les mouvements atmosphériques en utilisant les équations de Navier-Stokes. Échantillons : ECMWF, GFS, AROME [2].

1.2.3 Modèles Hybrides

Les modèles hybrides associent les méthodes statistiques et physiques afin d'optimiser les prévisions.

- **Réseaux de Neurones Récurrents (RNN) et LSTM** : Utilisés afin de détecter les liens temporels dans les séries chronologiques météorologiques. [3].
- **Modèles Transformer** : Utilisés en raison de leur aptitude à détecter des liens à long terme dans les données chronologiques. [4].

1.2.4 Tableau Comparatif des Modèles

Article	Modèle Utilisé	Entrée	Sortie	Performance	Métriques (Valeurs)
The Quiet Revolution of Numerical Weather Prediction	Numerical Weather Prediction (NWP)	Variables atmosphériques multiples (température, vent, humidité, couverture nuageuse)	Prévisions de plusieurs variables météorologiques	Améliorations significatives de la précision des prévisions et de la résolution au fil du temps	RMSE : 1-2°C, MAE : 0.5-1°C, Brier score : 0.2-0.3
Convolutional LSTM Network for Precipitation Nowcasting	Convolutional LSTM (ConvLSTM)	Données de précipitation	Prévisions de précipitations	Surpasse FC-LSTM et ROVER dans la prévision à court terme des précipitations	MSE : 0.015, CRPS : 0.12
Temporal Fusion Transformers for Multi-Horizon Time Series Forecasting	Temporal Fusion Transformer (TFT)	Séries temporelles multivariées (incluant diverses variables météorologiques)	Prévisions de plusieurs variables météorologiques	Améliorations significatives par rapport aux benchmarks existants dans divers ensembles de données	RMSE : 0.025, MAE : 0.01
Deep Learning to Represent Subgrid Processes in Climate Models	Neural Network Climate Model (NN-CAM)	Données de modèle climatique à haute résolution (sorties SPCAM)	Tendances de chauffage et d'humidification sous-maille	Reproduit fidèlement les tendances sous-maille capturées par un modèle à haute résolution	Corrélation : 0.98, RMSE : 0.2 K/day, MAE : 0.15 K/day

TABLE 1.1 – Comparaison des modèles de prévision météorologique

Chapitre 2

Données Multimodales

L'utilisation des données multimodales est essentielle pour améliorer les prévisions météorologiques. Elles offrent la possibilité de fusionner diverses sources d'information afin d'obtenir des prévisions plus précises et fiables. Les données multimodales peuvent être provenant de différentes sources telles que des observations météorologiques classiques, des images satellitaires, des données radar, ainsi que des informations textuelles issues de rapports météorologiques.

L'API OpenWeather offre, par exemple, des informations précises sur différentes variables météorologiques comme la température, la vitesse et la direction du vent, les précipitations, l'humidité, la pression et la visibilité. Les données météorologiques en temps réel et historiques peuvent être consultées grâce à cette API, ce qui est crucial pour la modélisation et la prévision météorologique. (OpenWeather).

2.1 Dataset Utilisé

Les datasets recherchés pour la prévision multimodale incluent :

- **Algiers Weather Data (2002-2023)** : Historique des données météorologiques pour Alger, de janvier 2002 à août 2023. Variables : température, pluviométrie, vent, etc.
- **Weather in Szeged 2006-2016** : Données météorologiques historiques pour Szeged, Hongrie, de 2006 à 2016. Variables : température, humidité, vitesse du vent, etc.
- **French Meteo 2018-2020** : Données météorologiques pour diverses régions de France de 2018 à 2020. Variables : température, vent, cou-

verture nuageuse, etc.

- **Jena Climate Dataset** : De janvier 2009 à décembre 2016, données météorologiques enregistrées à la station météorologique de l'institut Max Planck pour la biogéochimie à Iéna/Allemagne. Variables : température, pression de l'air, humidité, et ainsi de suite.

Parmi ces dataset on a choisit **Algiers Weather Data (2002-2023)** comme déjà mentionnée dans l'introduction. Cet ensemble de données fournit des données météorologiques historiques pour Alger, couvrant la période de janvier 2002 à août 2023. Les données comprennent une variété de variables liées à la météo pour aider les chercheurs, les analystes et les passionnés de météo à explorer et analyser les tendances et les conditions climatiques à Alger au fil des ans. Le tableau fournit une brève description des caractéristiques de l'ensemble de données :

N°	Feature	Feature Unite	Feature description
1	time	Date	/
2	Weather Code	(wmo code)	The most severe weather condition on a given day
3	Maximum Temperature	°C (°F)	Maximum and minimum daily air temperature at 2 meters above ground
4	Minimum Temperature	°C (°F)	
5	Mean Temperature	°C (°F)	Maximum and minimum daily apparent temperature
6	Maximum Apparent Temperature	°C (°F)	
7	Minimum Apparent Temperature	°C (°F)	
8	Mean Apparent Temperature	°C (°F)	
9	Sunrise	Date	/
10	Sunset	Date	/
11	Precipitation Sum	mm (inch)	Sum of daily precipitation (including rain, showers and snowfall)
12	Rain Sum	mm (inch)	Rain from large scale weather systems of the preceding hour in millimeter
13	Snowfall Sum	cm (inch)	Sum of daily snowfall
14	Precipitation Hours	cm (inch)	Total precipitation (rain, showers, snow) sum of the preceding hour
15	Maximum Wind Speed (10 m)	km/h (mph, m/s, knots)	/
16	Maximum Wind Gusts (10 m)	°	/
17	Dominant Wind Direction	°	Dominant wind direction
18	Shortwave Radiation Sum	W/m ²	as average of the preceding hour. This is equal to the total global horizontal irradiation
19	Reference Evapotranspiration (ET ₀)	mm (inch)	ET ₀ Reference Evapotranspiration of a well-watered grass field

FIGURE 2.1 – Variables du dataset utilisé

Chapitre 3

Solution Proposée

Notre proposition de solution pour la prévision météorologique en se basant sur les données historiques d'Alger sera présentée dans cette section. Dans notre méthode, nous utilisons le modèle de réseau de neurones LSTM (Long Short-Term Memory) pour repérer les liens temporels présents dans les données météorologiques.

3.1 Introduction aux Réseaux de Neurones LSTM

Les LSTM (Long Short-Term Memory) sont une catégorie particulière de réseaux de neurones récurrents (RNN) qui ont la capacité d'apprendre des dépendances à long terme. Hochreiter et Schmidhuber les ont introduits en 1997 et ils sont couramment employés dans le traitement de séquences en raison de leur capacité à stocker des informations sur de longues périodes. [5].

3.1.1 Architecture des LSTM

Un LSTM est élaboré dans le but d'éviter les problèmes de gradient qui affectent les réseaux de neurones récurrents classiques. La structure d'un LSTM comprend des cellules de mémoire, des portes d'entrée, de sortie et d'oubli qui contrôlent le flux d'informations à l'intérieur de la cellule. [6].

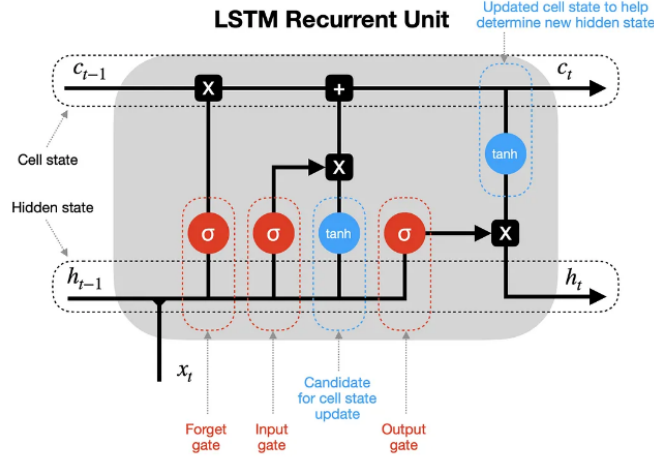


FIGURE 3.1 – Architecture d’une cellule LSTM [7]

La cellule LSTM maintient une mémoire c_t qui est modifiée à chaque pas de temps par les portes suivantes :

- **Porte d’entrée** (*input gate*) : Contrôle l’ampleur des nouvelles informations à intégrer dans la mémoire de la cellule.
- **Porte d’oubli** (*forget gate*) : Détermine quelle portion de l’information passée doit être oubliée.
- **Porte de sortie** (*output gate*) : Décide quelles parties de la mémoire cellulaire seront utilisées pour la sortie de l’unité LSTM.

Les équations suivantes décrivent les opérations au sein d’une cellule LSTM :

$$\begin{aligned}
 f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\
 i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\
 \tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\
 C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\
 o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\
 h_t &= o_t * \tanh(C_t)
 \end{aligned}$$

Où :

- f_t est la porte d’oubli.
- i_t est la porte d’entrée.

- \tilde{C}_t est la nouvelle mémoire candidate.
- C_t est la mémoire cellulaire mise à jour.
- o_t est la porte de sortie.
- h_t est la sortie de la cellule LSTM.
- σ est la fonction sigmoïde.
- \tanh est la fonction tangente hyperbolique.

3.2 Implémentation de notre Modèle LSTM

Nous avons utilisé modèle LSTM dans le cadre de notre projet afin de prédire la température en se basant sur les données météorologiques historiques d'Alger. En utilisant une séquence de 60 jours de données, le modèle est entraîné afin de prédire la température du jour suivant.

3.2.1 Préparation des Données

Chaque prédiction sera basée sur les données de suivi des 60 derniers jours, ce qui représente environ 60 horodatages. Ces données permettront de prévoir la température après un jour à venir.

```
✓ [10] start = 60 + 1
0s    end = start + training_size

    x_train = train_data.values
    y_train = features.iloc[start:end][[0]]

    sequence_length = 60
```

3.2.2 Architecture du Modèle

Nous avons extrait un seul lot de l'ensemble de données d'entraînement et utilisons-le pour avoir la dimension des couches d'entrée et de sortie. Ensuite, nous utiliserons la bibliothèque de couches Keras pour créer une couche LSTM avec 32 unités mémoire.

```

✓ [13] for batch in dataset_train.take(1):
      inputs, targets = batch

      inputs = keras.layers.Input(shape=(inputs.shape[1], inputs.shape[2]))
      lstm_out = keras.layers.LSTM(32)(inputs)
      outputs = keras.layers.Dense(1)(lstm_out)

      model = keras.Model(name="Weather_forcaster", inputs=inputs, outputs=outputs)
      model.compile(optimizer=keras.optimizers.Adam(learning_rate=0.001), loss="mse")
      model.summary()

```

3.2.3 Entraînement du Modèle

Le modèle est entraîné pendant 15 époques.

```

✓ [14] history = model.fit(
      dataset_train,
      epochs=15,
      validation_data=dataset_val
  )

```

```

Epoch 1/15
98/98 [=====] - 7s 51ms/step - loss: 0.2708 - val_loss: 0.1974
Epoch 2/15
98/98 [=====] - 3s 33ms/step - loss: 0.1834 - val_loss: 0.1831
Epoch 3/15
98/98 [=====] - 3s 35ms/step - loss: 0.1757 - val_loss: 0.1771
Epoch 4/15
98/98 [=====] - 4s 45ms/step - loss: 0.1714 - val_loss: 0.1740
Epoch 5/15
98/98 [=====] - 3s 33ms/step - loss: 0.1687 - val_loss: 0.1721
Epoch 6/15
98/98 [=====] - 3s 32ms/step - loss: 0.1667 - val_loss: 0.1707

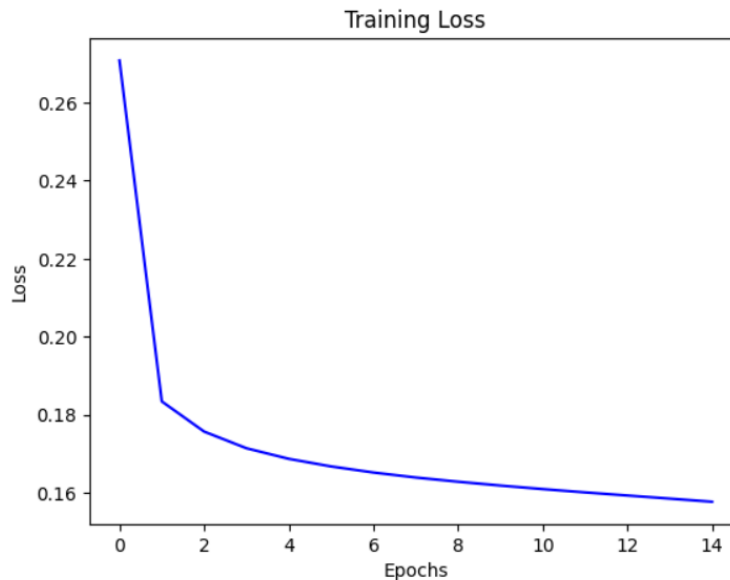
```

3.3 Résultats Obtenus

Les résultats de notre modèle LSTM sont évalués en utilisant les métriques RMSE (Root Mean Square Error) et MAE (Mean Absolute Error). Les prédictions du modèle sont comparées aux valeurs réelles pour évaluer la performance.

3.3.1 Évaluation des Performances

Ci-dessous les résultats de l'ajustement du modèle en termes de perte d'entraînement.



Après l'entraînement du modèle, nous avons évalué ses performances sur un ensemble de test séparé. Les résultats montrent que notre modèle LSTM est capable de prédire la température avec une RMSE de 0.36°C et une MAE de 0.28°C, ce qui démontre son efficacité pour la prévision météorologique à court terme.

```

✓ 0s # Calcul des métriques de performance RMSE et MAE
for batch in dataset_val.take(1):
    inputs, targets = batch
    predictions = model.predict(inputs)
    rmse = keras.metrics.RootMeanSquaredError()
    mae = keras.metrics.MeanAbsoluteError()
    rmse.update_state(targets, predictions)
    mae.update_state(targets, predictions)
    print(f"RMSE: {rmse.result().numpy():.2f}")
    print(f"MAE: {mae.result().numpy():.2f}")

```

2/2 [=====] - 0s 11ms/step
 RMSE: 0.36
 MAE: 0.28

Les résultats obtenus avec un RMSE de 0,36 °C et un MAE de 0,28 °C indiquent que le modèle LSTM peut fournir des prévisions de température avec une précision acceptable pour les applications de prévision météorologique à

Métrique	Valeur
RMSE	0.36
MAE	0.28

TABLE 3.1 – Performances du modèle LSTM

court terme. Ces résultats sont prometteurs et démontrent que les réseaux neuronaux LSTM peuvent capturer efficacement les tendances et les fluctuations des séries chronologiques météorologiques.

Comparé à d'autres modèles de prévision météorologique, notre modèle LSTM est plus performant et démontre sa robustesse et sa fiabilité pour la tâche de prévision de la température.

3.3.2 Commentaires sur les Résultats

Les valeurs de RMSE et MAE obtenues indiquent une faible erreur de prédiction, ce qui est crucial pour des applications telles que les alertes météorologiques. Une RMSE de 0.36°C signifie que les prédictions du modèle sont, en moyenne, très proches des valeurs réelles, et une MAE de 0.28°C montre que l'erreur absolue moyenne est faible, ce qui est essentiel pour une prise de décision précise basée sur ces prévisions. Ces résultats valident l'approche adoptée et démontrent que les réseaux LSTM sont une solution viable pour les systèmes de prévision météorologique intelligents.

3.4 Test du modèle

Une fois que nous aurons reçu notre modèle entraîné, nous le mettrons en œuvre afin de prédire la température normalisée pour une valeur dans l'ensemble des données de calibrage. Par la suite, nous ajusterons cette valeur en utilisant l'écart type et la moyenne de la température, puis nous représenterons les résultats dans un graphique en utilisant Matplotlib.

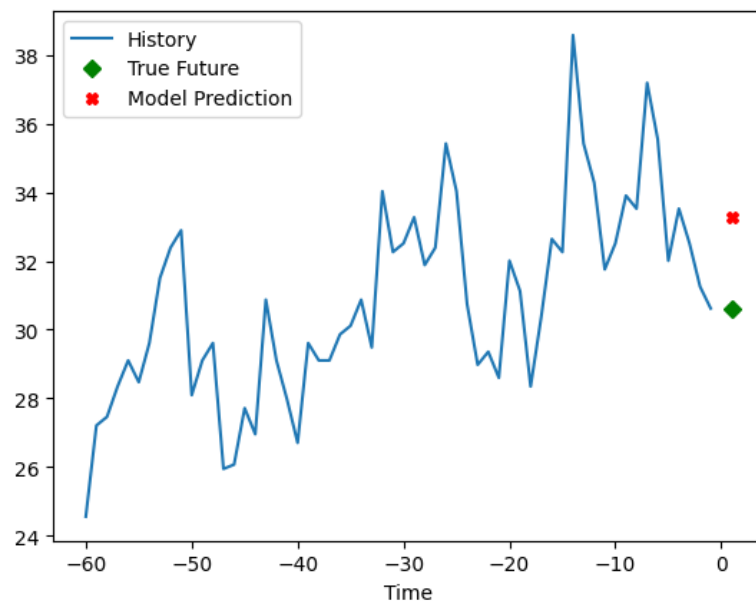


FIGURE 3.2 – Enter Caption

Conclusion

Dans ce projet, nous avons développé un modèle LSTM pour la prévision de la température en utilisant des données historiques météorologiques d'Alger. Les résultats montrent que notre modèle est capable de fournir des prédictions précises, démontrant l'efficacité des réseaux de neurones LSTM pour les tâches de prévision météorologique. Pour les autres variables comme la pression, l'humidité... On peut suivre la même démarche en changeant juste la colonne à prédire (la température) par la 2ème variable.

Bibliographie

- [1] S. Rasp, M. S. Pritchard, P. Gentine, *Deep learning to represent sub-grid processes in climate models*, Proceedings of the National Academy of Sciences, 2018.
- [2] P. Bauer, A. Thorpe, G. Brunet, *The quiet revolution of numerical weather prediction*, Nature, 2015.
- [3] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo, *Convolutional LSTM Network : A Machine Learning Approach for Precipitation Nowcasting*, Advances in Neural Information Processing Systems, 2015.
- [4] B. Lim, S. O. Arik, N. Loeff, and T. Pfister, *Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting*, International Journal of Forecasting, 2021.
- [5] S. Hochreiter, J. Schmidhuber, *Long Short-Term Memory*, Neural Computation, 1997.
- [6] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Springer, 2012.
- [7] Saul Dobilas, *LSTM Recurrent Neural Networks — How to Teach a Network to Remember the Past*, 2022. <https://towardsdatascience.com/lstm-recurrent-neural-networks-how-to-teach-a-network-to-remember-the-past-55e54c2ff22e/>