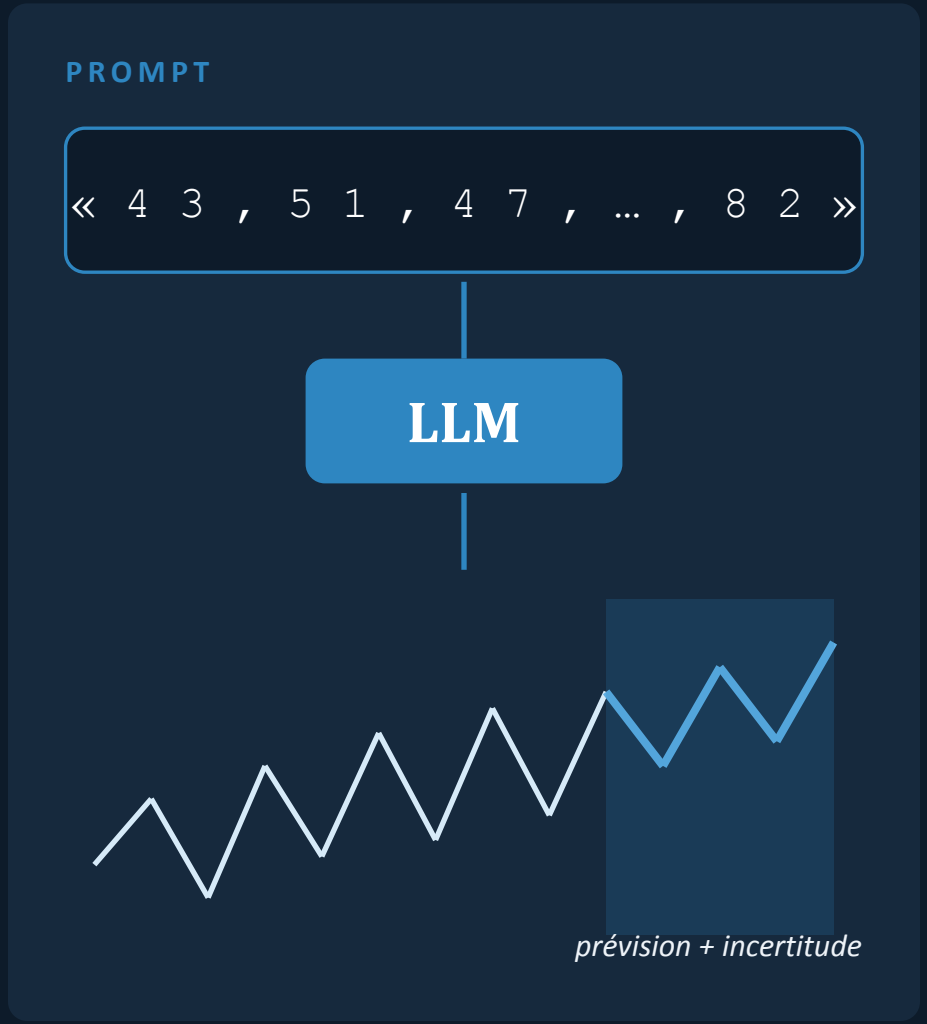
LLMTime

Large Language Models Are Zero-Shot Time Series Forecasters

Gruver, Finzi, Qiu, Wilson · NYU & CMU – NeurIPS 2023

Présenté en binôme par Kévin Feltrin et un coéquipier



Jouons : que vient-il après ?

A $4{,}2 \cdot 8{,}6 \cdot 1{,}0 \cdot 4{,}2 \cdot 8{,}6 \cdot ?$

B $0{,}2 \cdot 1{,}6 \cdot 3{,}0 \cdot 4{,}4 \cdot ?$

À vous : quelle est la valeur suivante ?

Vous venez de prévoir deux séries temporelles

A

4,2 · 8,6 · 1,0 · 4,2 · 8,6 · 1,0

Répétition : le motif revient tous les 3 pas

→ **saisonnalité**

B

0,2 · 1,6 · 3,0 · 4,4 · 5,8

Tendance : on ajoute 1,4 à chaque pas

→ **tendance**

C'est LLMTime : un LLM pré-entraîné prolonge une série écrite en chiffres

Plan de la présentation

1

Introduction

Auteurs, état de l'art, problématique

2

Méthode proposée

Tokenisation, mise à l'échelle, échantillonnage, densités

3

Résultats

Benchmarks, MAE, NLL et CRPS, GPT-3 contre GPT-4

4

Interprétation

Pourquoi ça marche, limites, devenir

5

Reproductibilité

Notre ré-exécution avec un petit modèle

6

Conclusion

Ce qu'il faut retenir

Les auteurs : le laboratoire de A. G. Wilson (NYU)

NG NYU

Nate Gruver*

Doctorat à NYU, dirigé par A. G. Wilson. LLM, protéines, matériaux, séries temporelles.

MF CMU

Marc Finzi*

Doctorat à NYU, dirigé par A. G. Wilson. Biais inductifs, équivariance.

SQ NYU

Shikai Qiu*

Co-auteur à contribution égale. Affiliation indiquée dans l'article : NYU.

AW NYU

Andrew G. Wilson

Professeur. Processus gaussiens, apprentissage profond bayésien, incertitude.

-

Des travaux antérieurs du laboratoire, cités dans l'article, préparent LLMTime

[19] Goldblum, Finzi, Rowan, Wilson (2023)

No free lunch, complexité de Kolmogorov et biais inductifs : base du « biais de simplicité » invoqué pour expliquer les résultats

[6] Benton, Gruver, Maddox, Wilson (2022)

Prévision probabiliste de séries temporelles sur longs horizons : l'incertitude au cœur du problème

[17] GPyTorch · [48] Wilson & Adams (2013)

Processus gaussiens et noyaux spectral mixture : le modèle SM-GP est l'un des baselines de l'article

État de l'art : des modèles statistiques aux LLM



ARIMA, ETS

Un modèle ajusté à chaque série

Réseaux profonds

TCN, DeepAR, N-BEATS, N-HiTS : entraînés par jeu de données

Transformers

Informer, Autoformer, FEDformer : souvent battus par un modèle linéaire

LLM + adaptation

FPT, Meta-Transformer (fine-tuning) ; PromptCast (question-réponse)

LLMTime

Zéro entraînement, encodage soigné, incertitude

Approche	Principe	Ce que LLMTime change
FPT, Meta-Transformer	Fine-tuning d'un modèle de langage pré-entraîné (BERT pour FPT)	Aucun fine-tuning : le LLM est utilisé tel quel
PromptCast	Question en langage naturel sur les valeurs futures	Aucun texte ajouté : les nombres sont encodés et mis à l'échelle avec soin
LLMTime	**Nombres écrits en chiffres, échantillonnage de la suite**	**Prévision probabiliste (incertitude) en zero-shot**

Références : Box & Jenkins 1968 · Hyndman et al. 2008 · Informer, Autoformer, FEDformer · Zeng et al. 2022 · Zhou et al. 2023 (FPT) · Zhang et al. 2023 · Xue & Salim 2023

Problématique : peut-on prévoir sans entraîner ?

Des données hétérogènes

Échelles, fréquences et sources très différentes, parfois des valeurs manquantes.

Pas de pré-entraînement commun

Pas de grand jeu de données unifié : ARIMA ou un modèle linéaire battent souvent le deep learning.

L'incertitude est essentielle

Les observations ne portent qu'une infime part de l'information : une prévision ponctuelle ne suffit pas.

PROBLEMATIQUE

Un LLM pré-entraîné uniquement sur du texte peut-il prévoir une série temporelle sans aucun entraînement (zero-shot), avec une incertitude fiable ?

Principe : écrire la série comme une suite de chiffres transforme la prévision en prédiction du token suivant.

Contributions annoncées

1 · Encoder la série en chiffres

2 · Densités continues à partir de probabilités discrètes

3 · Expliquer : simplicité et répétition

4 · Propriétés : échelle, NaN, texte, limites

LLMTime : la méthode en quatre étapes

1 **Mise à l'échelle et écriture en chiffres**

La série est normalisée, puis écrite chiffre par chiffre, séparée par des virgules.

2 **Le LLM continue le texte**

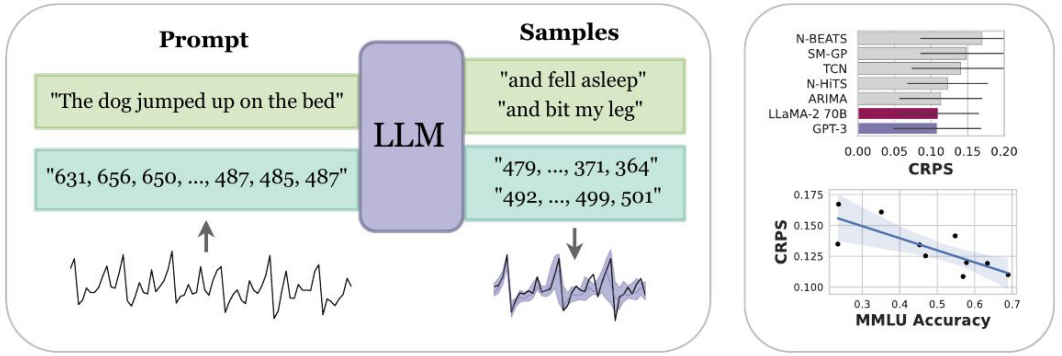
Prédiction du token suivant : environ 20 suites sont tirées au hasard.

3 **Décodage**

Les chiffres redeviennent des nombres, remis à l'échelle d'origine.

4 **Prévision**

Médiane = prévision ponctuelle ; quantiles = incertitude.



Source : Gruver et al. (2023), Fig. 1

Idée clé : une série temporelle a la même forme qu'un texte (une suite d'éléments). Seule la nature des éléments change.

Tokenisation : un token par chiffre

GPT-3 (BPE) `42235630` → `422` | `35` | `630`

LLMTime `4 2 2 3 5 6 3 0` → `4` | `2` | `2` | `3` | `5` | `6` | `3` | `0`

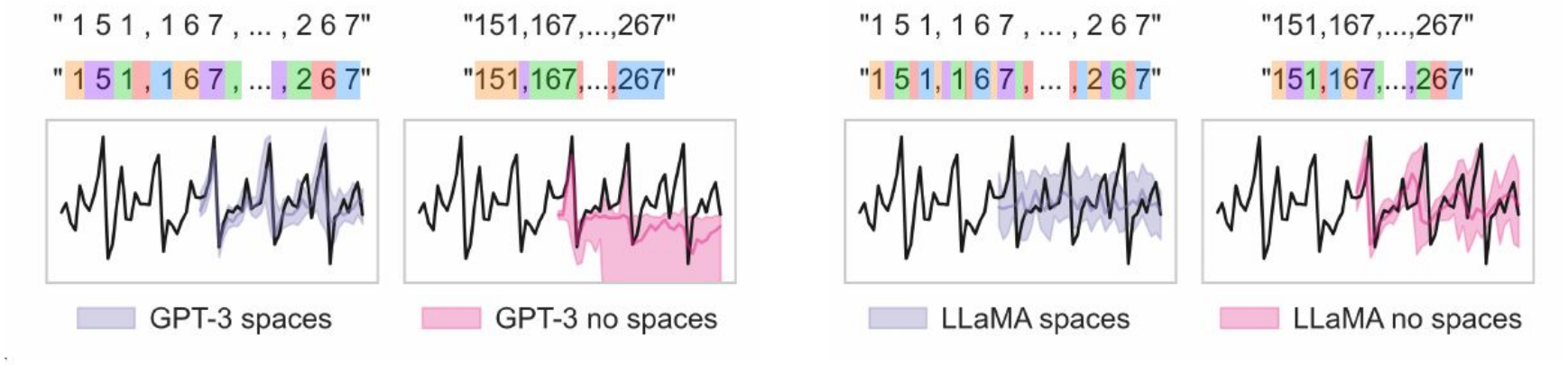
Un chiffre peut changer tout le découpage de GPT-3 ; avec LLMTime : 1 chiffre = 1 token.

Encodage complet (2 chiffres de précision)

`0.123, 1.23, 12.3, 123.0`

`« 1 2 , 1 2 3 , 1 2 3 0 , 1 2 3 0 0 »`

Pas de point décimal · la virgule sépare les pas de temps



- GPT-3 : les espaces forcent un token par chiffre, d'où de bonnes prévisions
- LLaMA-2 : chaque chiffre est déjà un token, les espaces nuisent

Source : Gruver et al. (2023), Fig. 2 (jeu AustralianWine de Darts)

Mise à l'échelle et échantillonnage

Mise à l'échelle



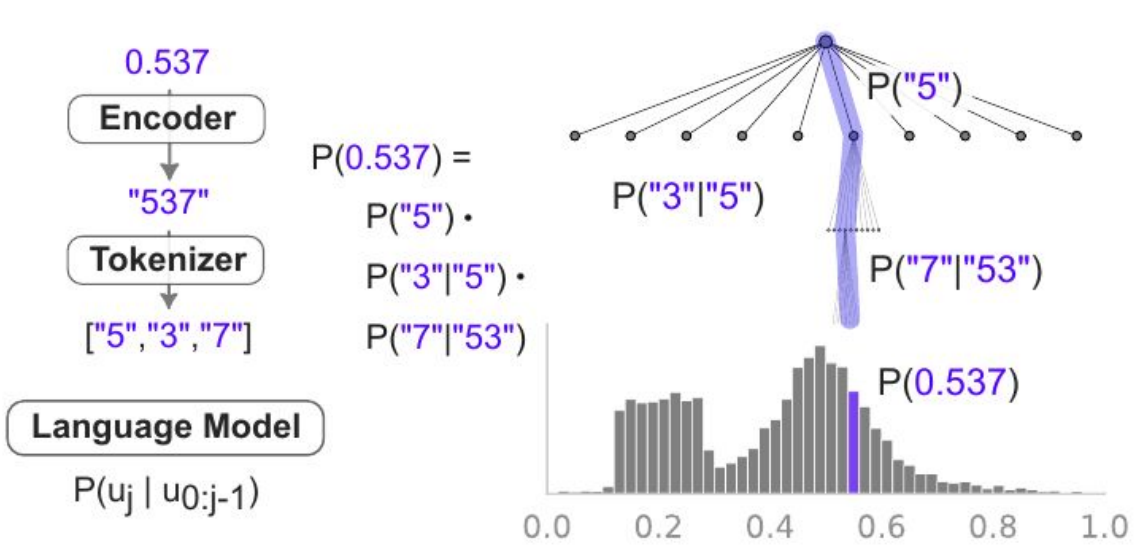
- On divise la série pour que son percentile α vaille 1
- Pas de division par le maximum : le modèle voit des nombres plus longs et sait en produire
- Un décalage β est aussi utilisé ; α et β sont réglés sur la vraisemblance de validation

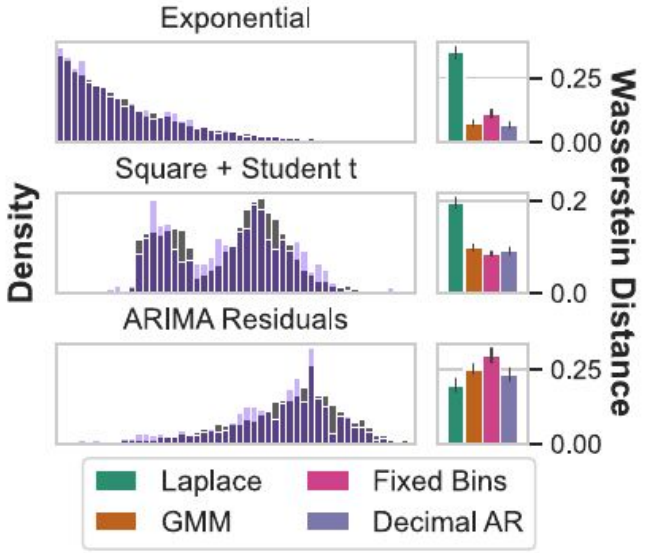
Échantillonnage et prévision

≈ 20	Médiane	Quantiles
suites tirées du LLM	prévision ponctuelle	prévision probabiliste

- À chaque pas de temps, les statistiques des échantillons donnent la prévision
- Contrôle de l'échantillonnage : température, biais logit, nucleus sampling (annexe C)
- Plus on tire d'échantillons, plus la médiane et les quantiles sont stables

Des chiffres aux densités continues



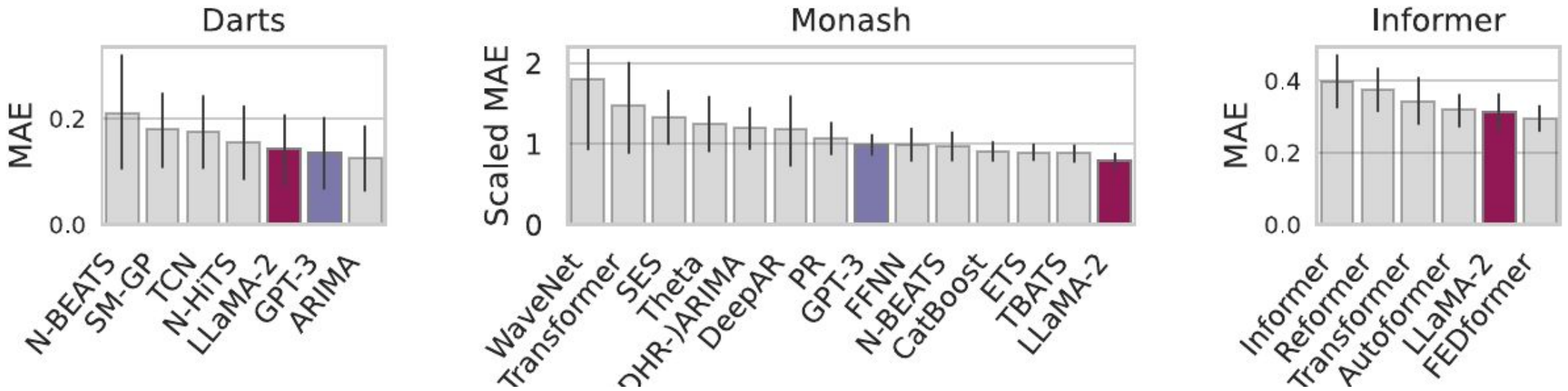


Source : Gruver et al. (2023), Fig. 3

Softmax hiérarchique

P(0,537) = P(5) · P(3 | 5) · P(7 | 53) : chaque chiffre est un softmax sur 0 à 9.

Prévision ponctuelle : un LLM parmi les meilleurs



Source : Gruver et al. (2023), Fig. 4 · LLMTime avec GPT-3 et LLaMA-2 70B (barres colorées)

1er ou 2e en MAE

Sur les trois benchmarks, le meilleur des deux LLM est premier ou deuxième, sans paramètre entraîné.

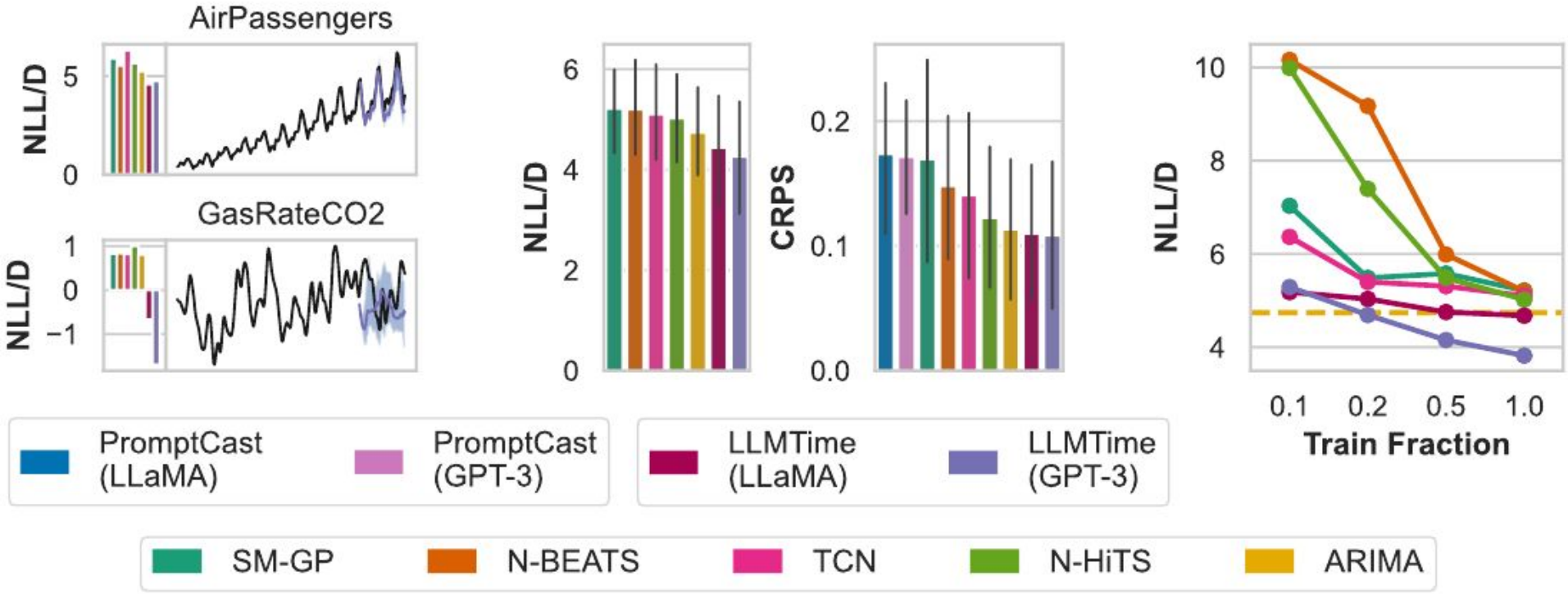
Pas partout devant

ARIMA reste devant sur Darts, FEDformer sur Informer : la domination n'est pas totale.

GPT-3 non évalué sur Informer

Coût des requêtes API : seul LLaMA-2 70B apparaît pour ce benchmark.

Prévision probabiliste : là où LLMTime domine



Source : Gruver et al. (2023), Fig. 5

Meilleurs NLL et CRPS

Meilleur que toutes les baselines en agrégé, et sur presque chaque jeu.

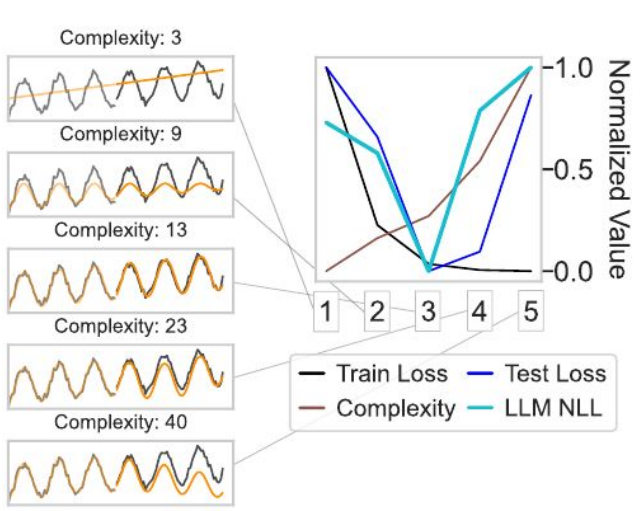
Efficacité en données

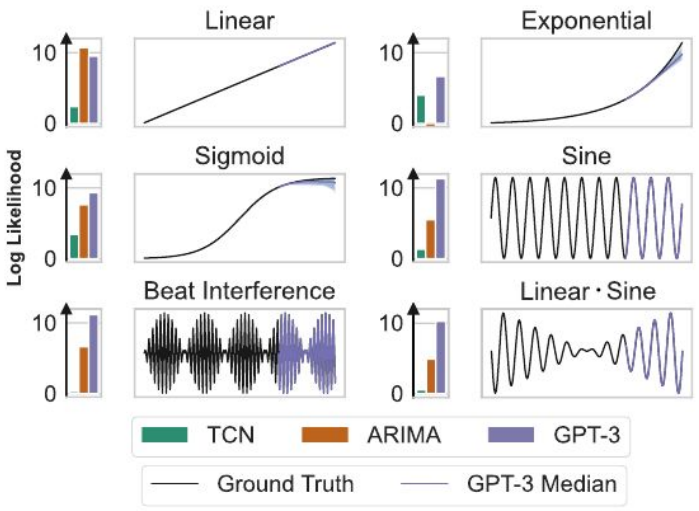
Les concurrents s'effondrent quand on réduit leur jeu d'entraînement ; pas LLMTime.

Face à PromptCast

Nettement meilleur en CRPS et MAE : encodage et mise à l'échelle comptent.

Pourquoi ça marche : des biais qui collent aux séries





Source : Gruver et al. (2023), Fig. 6 · gauche : vraisemblance de GPT-3 selon la complexité de la formule · droite : extrapolation de motifs simples

Simplicité → généralisation

Une règle simple l'emporte sur un ajustement complexe

Répétition → saisonnalité

$x(t) = x(t - T)$: suite A du jeu

Arithmétique → tendances

$x(t+1) = x(t) + c$: suite B du jeu

Aucun entraînement : les biais de simplicité du LLM suffisent.

Ce que les résultats montrent, et leurs limites

Ce que les résultats montrent

- ✔ **Zéro paramètre entraîné :** reste dans le peloton de tête en MAE.
- ✔ **Avantage net probabiliste :** meilleurs NLL et CRPS ; efficace avec peu de données.
- ✔ **Propriétés gratuites :** NaN sans imputation, questions en langage naturel.
- ✔ **La taille ne suffit pas :** alignement et tokenisation comptent (GPT-4 < GPT-3).

Limites

- ! **Fenêtre de contexte :** difficile en multivarié (chaque variable prévue séparément).
- ! **Arithmétique :** faiblesse en calcul et en opérations récursives.
- ! **Coût :** ≈ 20 complétions par prévision ; GPT-3 non évalué sur Informer.
- ! **Fuite de données :** les corpus des LLM ne sont pas publics (annexe B).

Et après ? Ce que LLMTime a lancé

CH Amazon

Chronos (2024)

Même idée (la série en tokens), mais un modèle entraîné sur des séries. Co-signé par A. G. Wilson.

TF Google

TimesFM (2024)

200 M de paramètres pré-entraînés sur des séries : bat LLMTime en zero-shot.

TA NeurIPS

Tan et al. (2024)

Retirer le LLM des méthodes qui le fine-tunent ne dégrade pas les prévisions.

</> GitHub

Une référence

Code ouvert (835 ★), repris comme point de comparaison, par exemple par TimesFM.

Pistes ouvertes par l'article

Contextes plus longs

Plus d'historique et le multivarié : la fenêtre de contexte reste la limite.

Fine-tuning sur séries

Piste proposée par les auteurs, suivie ensuite par Chronos et TimesFM.

Applications

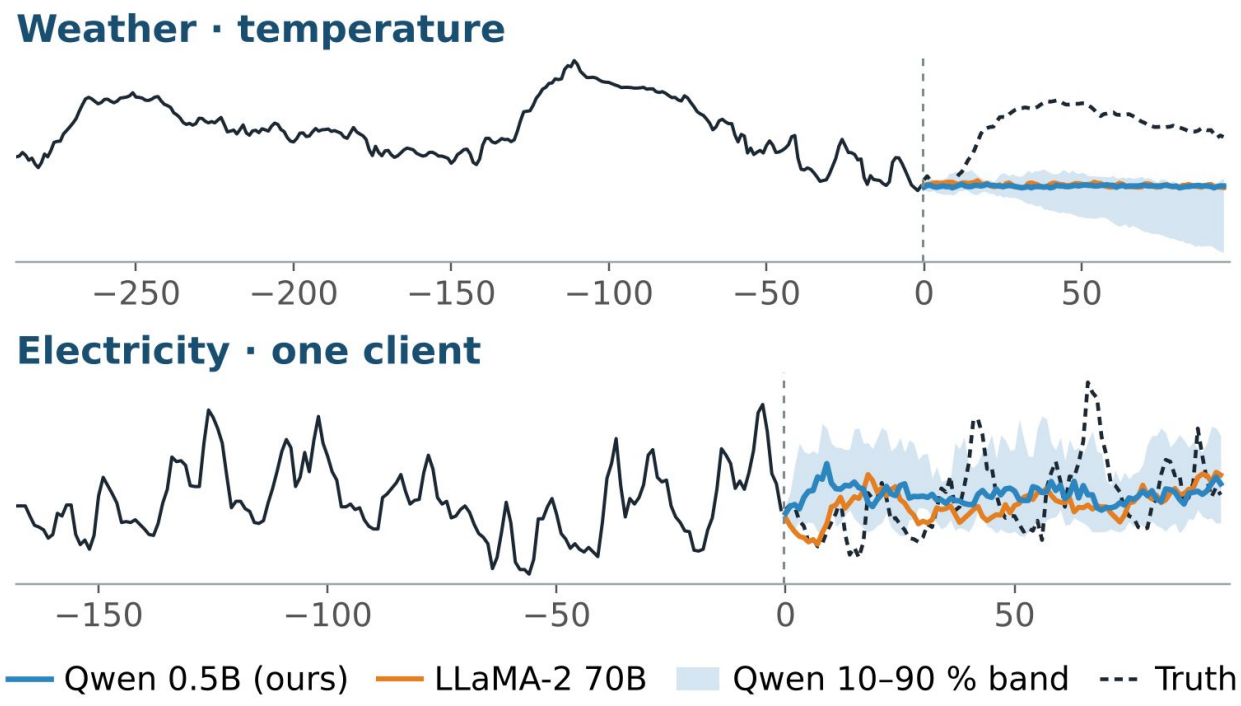
Prévoir sans données d'entraînement : nouvelles séries, petites séries, valeurs manquantes.

Reproductibilité : notre expérience

Mêmes données, mêmes scores

Ré-exécution : Qwen 0,5B sur CPU

GPT-3 et LLaMA-2 70B : hors de portée



Bleu : Qwen 0,5B (nous) · orange : LLaMA-2 70B · bande : intervalle 10–90 % de Qwen · pointillé : valeurs réelles

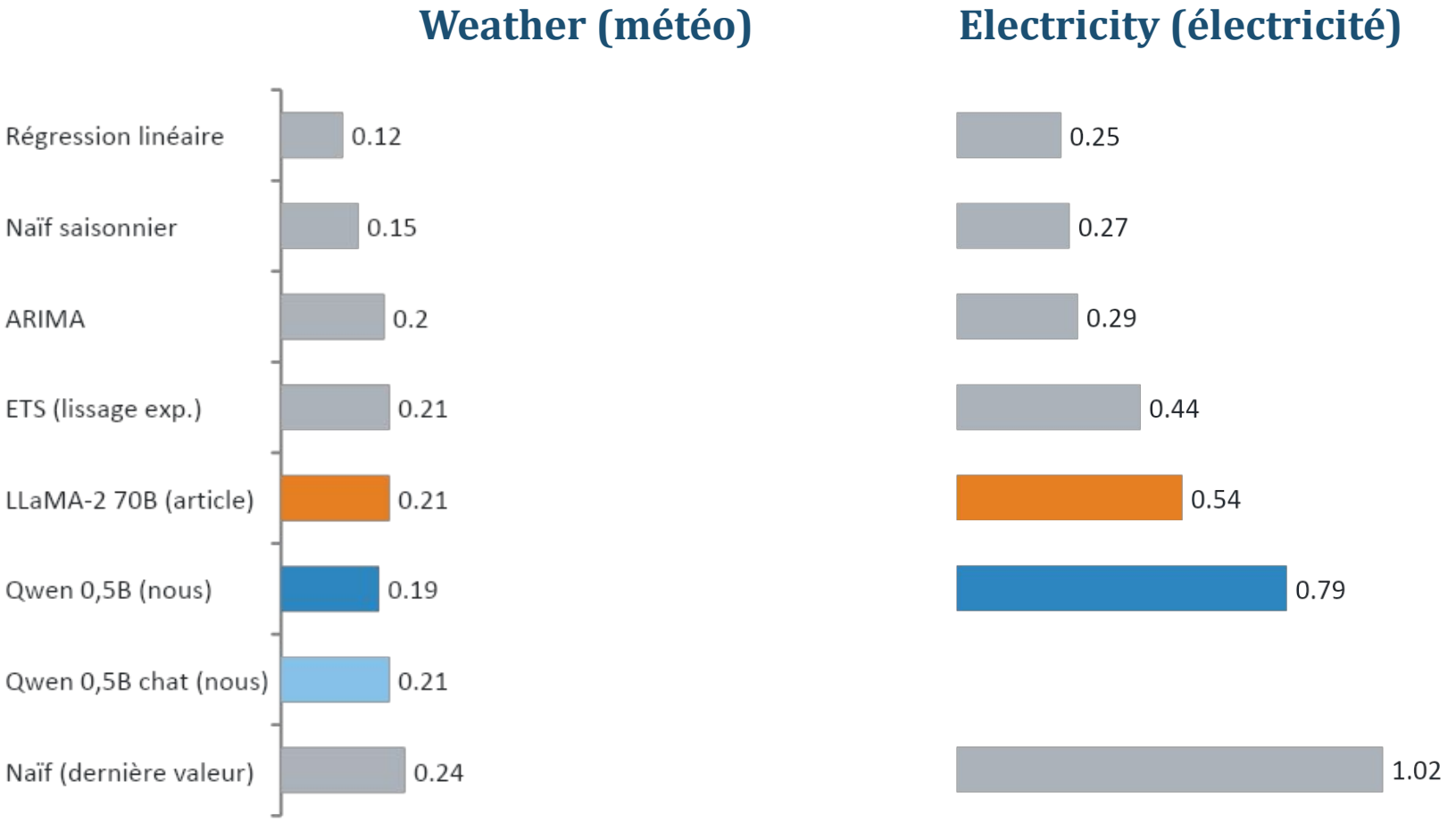
Exemples : une série par jeu de données. Les scores du tableau sont des moyennes sur toutes les séries.

- Weather : les 21 séries · Electricity : 20 séries sur 321
- Test : les 96 derniers pas de chaque série
- Contexte : 2048 tokens (article : 4096)

MAE ↓	Weather	Electricity
Qwen 0,5B (nous)	**0,195**	**0,790**
LLaMA-2 70B [1]	0,214	0,539
Qwen 0,5B chat (nous)	0,208	—
Naïf (dernière valeur)	0,240	1,024

[1] Prévisions publiées par les auteurs, pas relancées par nous (321 séries : 0,498).

Des baselines simples battent LLMTime sur nos données



MAE sur données centrées-réduites, horizon 96, plus bas = meilleur, mêmes fenêtres de test. Régression linéaire : ridge sur 336 retards (Zeng et al., AAAI 2023). Orange : LLaMA-2 70B ; bleu : Qwen 0,5B.

Ce qu'on retient

Top 2 en moyenne seulement

Dans la Fig. 11 de l'article : dernier sur Electricity.

La copie d'hier gagne

Le naïf saisonnier, sans entraînement, bat LLMTime.

Weather ≈ persistance

Les prévisions des LLM restent proches de la dernière valeur.

Conclusion

- Prédire le token suivant sur des chiffres
- Clés : Tokenisation, mise à l'échelle, densité par cases
  Résultats convaincants
- Reproduit, mais des baselines simples gagnent sur nos données
- Ouvre la voie aux modèles de fondation : Chronos, TimesFM

Encodage soigné + LLM pré-entraîné = zero-shot

Merci ! Des questions ?