Convolutional Neural Networks and Temporal CNNs for Covid-19 Forecasting in France

Lucas Mohimont, Amine Chemchem, Francois Alin, Michaël × Krajecki, Luiz Angelo Steffenel

Noname manuscript No.
(will be inserted by the editor)

Convolutional Neural Networks and Temporal CNNs for Covid-19 Forecasting in France

Lucas Mohimont · Amine Chemchem · François Alin · Michaël Krajecki · Luiz Angelo Steffenel

Received: date / Accepted: date

Abstract This paper examines multiple CNN-based (Convolutional Neural Network) models for Covid-19 forecast developed by our research team during the French lockdown. In an effort to understand and predict both the epidemic evolution and the impacts of this disease, we conceived models for multiple indicators: daily or cumulative confirmed cases, hospitalizations, hospitalizations with artificial ventilation, recoveries and deaths. In spite of the limited data available when the lockdown was declared, we achieved good short-term performances at the national level with a classical CNN for hospitalizations, leading to its integration into a hospitalizations surveillance tool after the lockdown ended. Also, A Temporal Convolutional Network with quantile regression was found successful at predicting multiple Covid-19 indicators at the national level by using data available at different scales (worldwide, national, regional). The accuracy of the regional predictions was improved by using a hierarchical pre-training scheme, and an efficient parallel implementation allows for quick training of multiple regional models. The resulting set of models represent a powerful tool for short-term Covid-19 forecasting at different geographical scales, complementing the toolboxes used by health organizations in France.

Keywords Deep Learning · Convolutional Neural Networks · Temporal Convolutional Network · Transfer Learning · Quantile Regression · Covid-19

1 Introduction

The Severe Acute Respiratory Syndrome Coronavirus 2, or SARS-CoV-2, was initially described in Wuhan, China. Its spread is responsible for the Covid-19 disease pandemic, with 188 affected countries, 29,897,412 confirmed cases and 941,363 confirmed deaths (September 17^{th}). This paper focus on the evolution of the French Covid-19 epidemic, which presented an elevated spreading rate at its beginning. Indeed, the first French cases were confirmed on January 24^{th} and

Université de Reims Champagne Ardenne, CReSTIC EA 3804, 51097 Reims, France
E-mail: {lucas.mohimont, mohamed-lamine.chemchem, francois.alin, michael.krajecki, angelo.steffenel}@univ-reims.fr

the 100^{th} case was confirmed on February 29^{th}. The transmission of the virus accelerated in March 2020, following an exponential growth of the number of confirmed cases. This was followed by an exponential growth of hospitalizations, and emergency measures had to be taken to avoid saturation of hospital services. As a result, a national lockdown was adopted from March 17^{th} to May 11^{th}. Schools, non-essentials businesses and public parks were closed. Most outdoor activities and long-distant travels were banned, and a signed form was required for every essential trip (to buy food or medicine, to help a vulnerable family member, etc.). The reproductive number (i.e., the average number of people that would get infected by an already infected person) was estimated to be of 2.9 before the lockdown and 0.67 during the lockdown [43]. While the lockdown was successful in stopping the virus propagation, the death toll was still high with 26,619 confirmed deaths at the end of lockdown (reaching 31,045 confirmed deaths on September 17^{th}). Today, the risk of a Covid19 second wave is important as French confirmed cases are raising since the beginning of August. Indeed, the 7-day moving average since August 28^{th} has been higher than the average of the first wave. For the moment, hospitalizations are increasing at a slower rate than during the first wave mostly because the affected population is younger. Also, the situation has changed with mandatory actions such as the use of face masks in closed areas (and sometimes even in the streets), physical distancing, telecommuting work and massive testing campaigns. The monitoring of the epidemic is crucial because a new strict lockdown would have a highly negative impact on the economy. Indeed, the Organization for Economic Cooperation and Development (OECD) estimated a GPD recession of 11.4% for 2020 without a second wave, and a 14.1% recession in the case of a second wave by Fall 2020 [36].

The scientific community has been very dynamic to provide useful decision tools for epidemic modeling, and not only for contamination or death cases. Providing different projections of the epidemic evolution helps making effective decisions, whose goals are to reduce the number of victims all while avoiding a severe economic recession that would increase poverty, political tension, etc. Multiple French projects have been initiated to model the Covid-19 epidemic and help reduce its harmful effects, often associating French authorities and services at both national and regional scales, public research laboratories and private companies. Most of these projects aim to capitalize on French high-performance computing resources and inter-disciplinary skills related to digital technologies.

Our team at the University of Reims Champagne Ardenne (URCA) is currently involved in several projects related to Covid19 mitigation. At the pharmacology level, the ANR HT-Covid project relies on the ROMEO Supercomputing Center to simulate millions of molecules and protein interactions (molecular docking). The goal is to identify the molecules that are able to inhibit the SARS CoV-2 virus and could lead to a treatment. We may also cite a collaboration with researchers from the French Ministry of Defense, which developed a tool allowing to investigate the impact of sanitary and economic restrictions during the lockdown [14]. Contrarily to the models we present in this papers, that work uses multiple epidemic and economic black-box simulators based on traditional parametric models, with an optimization algorithm to select the best resulting scenario. We are also involved in the Grand-Est region project ECOVISION, whose objective is to create a unique dashboard for Covid-19 monitoring and forecast, by combining multiple prediction models and indicators. This project is partially based on the work presented in this paper, and its goal is to aggregate research projects into an operational decision tool.

Therefore, this paper aims at presenting our experience using deep neural networks (and specially Deep learning) to model and forecast different Covid-19 indicators and at different geographical scales. Machine learning and deep learning models are powerful modeling tools that revolutionize several domains. Contrarily to classical parametric models, modern neural networks do not depend on the knowledge of a given phenomenon, but can use a data-centric approach that can be applied to raw data and still be able to model complex tasks. In several occasions Deep learning proved to be competitive against well known traditional modeling algorithms, as presented in our previous works [1,8].

In this study, we demonstrate how data-driven models can produce excellent predictions. Because Deep learning usually depends on massive input data, it was thought that it could not efficiently model the Covid19 epidemic as the existing data during the epidemic breakout was reduced and fragmented. Nonetheless, our efforts on data processing (including the generation of synthetic data for pre-Covid19 months) proved successful. For instance, our models can perform national forecasts for a 20-30 days window with error rates as low as 1%, in the best cases.

This work was carried on the basis of the official data provided by the French Ministry of Health, and the different models designed by our team were constructed to respond to the actual needs of leading authorities. Hence, the first task we focused was the modeling of confirmed Covid19 cases. This is an important indicator because it represents the transmission of the disease. However, it depends on the number of available tests being conducted, and does not relates directly to the actual burden of the epidemic on the health system. As a consequence, the second task we focused was the hospitalization forecast, at both national and regional scales. This is a more important indicator as a sudden increase in hospitalization can lead to the saturation of hospital services, which may be forced to move patients to other facilities (often in different regions of the country) with available beds and healthy medical staff. To better anticipate the impact on the health structures, we created a deep learning based approach to model a wide set of Covid-19 indicators, such confirmed cases, hospitalizations, hospitalizations requiring artificial ventilation, number of recoveries and number of deaths.

This paper is organized as follows: Section 2 describes related work on Covid-19 epidemic modeling, Section 3 describes the data sources used in this work, how they were used, as well as the computing environment supporting our experiments. Section 4 describes the proposed models and Section 5 describes the obtained results. Finally, Section 6 discusses the results and their implications, and Section 7 concludes this work.

2 Related Works

Epidemic modeling is commonly achieved with compartmental models like *Susceptible - Infectious - Recovered* (SIR) [40]. In the SIR model, the population is divided into three compartments:

- **Susceptible**: the part of the population that can be infected.
- **Infectious**: the part of the population currently infected.

– **Recovered**: the part of the population that recovered from the disease and that is now immune.

Each compartment is associated to a function that represents the evolution of the population. The modeling is performed by solving a system of differential equations. The SIR model can be extended with other compartments like Deceased (SIRD model) or Exposed (SEIR model). Those models need parameters specific to the studied disease like the rate of infection, rate of recovery and rate of mortality. Initial conditions for the compartments population are also needed. In practice, parameters are estimated by fitting the models to the available data. This is achieved with methods for non-linear optimization problems like the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm [34]. Compartmental models can be used for prediction modeling or for a-posteriori analysis.

The need for good initial conditions and a good knowledge of the studied disease is one of the main weakness of this parametric approach. In the case of Covid19, a SIRD model predicted the peak of the Italian first wave for March 21th with 26,000 confirmed cased and 18,000 deaths by the end of the epidemic [17]. Those predictions revealed to be highly inaccurate, as the peak of confirmed cases was achieved on April 19^{th} reaching more than 109,000 cases, and the death toll rose above the predicted 18,000 on April 9^{th}. Similarly, a SEIR model was proposed to predict the second wave in France and Italy [18]. This model was first used to estimate the fraction of the population that got infected during the first wave. This model estimated that 6% of the French population was infected during the first wave. A similar work was proposed by [43], using the data from the Diamond Princess outbreak to estimate the infection fatality ratio in France. A compartmental data was applied to hospital data, estimating that 4.4% of the French population would have been infected by May 11^{th}. Both results are close. However, the uncertainty of the estimate achieved by [18] is high, with estimations ranging from hundreds of thousands to 18 million infections in France. Nonetheless, SIRD models were used to measure the impact of the lockdown on the French epidemic and it was estimated that the reproductive number R0 was divided by 7. In the overall, compartmental models are useful to create complex scenarios and to perform analysis of past epidemics. However, other methods based on statistical and machine learning techniques seem to be more suitable for real-time forecasting [24].

An alternative to compartmental models is a data-centric approach that does not uses pre-determined rules about the disease spreading behavior. Instead, the rules are determined from the data. A death prediction model using a mixture of past predictors was proposed by [44]. The main idea is that the death trend of a country can be represented as a mixture of past death trends from other countries. As a result, good accuracy levels were achieved for up to 10 days forecasts. However, this modeling approach is limited to short-term forecasts. The mixture model needs multiple sequences from different countries that have higher death rates. Those countries are considered to be ahead of time in the epidemic trajectory of the studied country (the one we want to forecast). In practice, the authors have shown that accuracy is low beyond 10 days forecasts because not enough predictors (countries) are available. Therefore, a SIRD model was used as a predictor for longer forecasts.

Genetic Expression Programming was used to create formulas for confirmed cases and deaths evolution in 15 countries [42]. This method seems to be more reliable than LSTM with a higher RMSE and higher R^2. However, it is limited by the used data because only Covid-19 time series are used in the modelling process. A hybrid method using ARIMA and Wavelet-based forecasting was proposed for confirmed cases forecasting [7]. Wavelet-based forecasting was used for error remodeling of the ARIMA model. Good accuracy is shown on the training set. Accuracy for the real-time forecasts in France is lower with almost double RMSE compared to the training set. This can be explained by the French lockdown that started on March 17^{th} and its effects were not visible on the training data.

In this work, we chose to use neural networks as modeling tools. Neural networks have the ability to model complex non-linear patterns and many different architectures are available. One of the most popular architecture for time series analysis is Long Short-Term Memory (LSTM) [23]. It is an improvement of the classical neural network that was developed to solve the vanishing gradient problem. LSTM is suitable for sequence modeling as it is able to model temporal effects. It was used in multiple research works on Covid-19 forecasting, as in the case of Canada [10] or India [47]. Accuracy of 93.4% and 92.67% were found respectively for short and long-term predictions [10]. Similar performances were found by [47], with error percentages ranging from 1.64% to 8%.

LSTM was also compared to ARIMA and Nonlinear Auto-Regression Neural Network on confirmed cases predictions of eight European countries [31]. **In both cases,** LSTM achieved the best accuracy. Other works compared multiple neural network architectures including LSTM, Recurrent Neural Network (RNN), Bidirectional LSTM, Gated Recurrent Unit (GRU [11,12], a variant of LSTM) and Variational Autoencoder (VAE) in order to forecast confirmed cases and recovery predictions [49]. The VAE achieved the best results with a clear margin. It is thought that VAE needs fewer data compared to other neural networks. LSTM can achieve a moderate accuracy but with a lower explained variance compared to the VAE.

Another popular type of neural network is the Convolutional Neural Network (CNN). It became the new state-of-the-art model for image classification in 2012 when a CNN won the ILSVRC-2012 competition on the ImageNet dataset [30]. Although CNNs were first successful on image processing task, they can also be applied on 1D data like Covid-19 time series. CNN are not as popular as LSTM despite potential better accuracy and more efficient training with GPU acceleration [3]. For this reason, CNNs were mostly applied to Covid-19 detection from x-ray images [38,46,2]. Nonetheless, CNNs were used for confirmed cases forecasting in China [25]. In that study, it was compared to LSTM, GRU and to the Multi-layer Perceptron, and CNN achieved a better accuracy. In this work we chose to explore the use of CNNs instead of LSTM, as it presented better results during our preliminary experiments with confirmed cases forecasting.

A hybrid of LSTM and CNN, the ConvLSTM model, was used by [39] for Covid-19 spatiotemporal modeling of confirmed cases. Pixels map for Italy and USA were created from the available datasets and an ensemble of ConvLSTM was trained. The ConvLSTM uses both convolutional layers to process images input and LSTM layers for sequences modeling. The ConvLSTM is an adaptation of CNN for image sequences modelling.

CNN have also been adapted to generic sequence modelling tasks by using causal and dilated convolutional layers. This family of model is called Temporal Convolutional Network (TCN). TCN models are fully convolutional networks that can be applied on sequences on any length. A more complete description of TCN is proposed in section 4. TCN have been applied to different application such as stock price prediction, energy consumption forecasting and for automatic sepsis prediction in hospitals [35,32,29]. Both classical CNN and TCN are applied to multiple Covid-19 modelling tasks. To the best of our knowledge, this is the first time TCN are used for epidemic modelling. Another difference from our work to related work is the development of a multi-level forecast model. Indeed, most works are focused only on national level modeling, but the epidemic crisis management is often operated at regional levels. It is therefore important to provide projections at a local scale. In this work, we preferred to keep the data at a meaningful regional level (the official regional organization of France) during both training and prediction.

3 Datasets

Our work started in mid-March, at the moment the lockdown was declared in France. Several indicators such as French hospitalization data were not yet available. Therefore, an important effort was made on data curation.

3.1 Data Sources

In this work we relied on four datasets, composed from different data sources. They are summarized in Table 1 and were used for the training of the neural networks. Covid-19 data come from different sources like testing laboratories and hospitals. The centralization of French Covid-19 data is a difficult but necessary task that involves hundreds of establishments. Hospitals and testing data are collected and verified by Santé Publique France, a french public organisation for public health surveillance. Those data are openly available and they can be visualized on the official dashboard [19]. Data reliability is important for modelling but some mistakes are expected when many differents establishments are involved. Data anomalies are checked and corrected regularly.

The first dataset was built from the Covid-19 data collected by the Center for Systems Science Engineering at John Hopkins University. They are used for worldwide visualization of the epidemic on an online dashboard and they are available on their Github repository [16]. The Covid-19 time-series data start on January 22^{nd} and are updated every day. The time series are confirmed Covid-19 cases, confirmed recoveries and confirmed deaths. Each sample is corresponding to one geographic area and to one specific date (one sample per day). Data for 188 countries are available. Demographic, economic and health indicators coming from the United Nation were added to the dataset.

The second dataset was built from the official French dataset regarding Covid-19 hospitalizations [20]. It includes time-series of current Covid-19 hospitalizations, current hospitalizations with artificial ventilation, cumulative recoveries and cumulative deaths. The time series are starting from March 18^{th} and are updated every day. They are available at national, regional and departmental levels. External data were added to this dataset. They are demographical indicators coming from National Institute of Statistics and Economic Studies (INSEE) and mobility data extracted by the French mobile operator Orange.

The third dataset is similar to the second one but the Orange mobility data are replaced by the Google mobility data [26]. These mobility data are six indicators of mobility changes from a baseline period in different sectors: grocery and pharmacy, parks, public transports, retail and recreation services, residences and workplaces. Finally, the fourth dataset is built by merging the first dataset and the Google Mobility data.

Most of the development time was dedicated to data curation (data collection, cleaning and preparation) during the beginning of the project. One problem was to find interesting dynamical data for our models. Google have created mobility reports at different geographical scales (national, regional, city, etc.), which include the time series of six mobility indicators described previously. Such variables are useful because population mobility has an impact on the epidemic spreading. Finding relevant dynamical data is difficult because systematic data collection is not always possible for many reasons (ethics, security concern, large amount of processing, etc.).

Table 1 Summary of the training datasets. The starting dates of the time series are shown in the Dataset column.

Dataset	**Target**	**Granularity**	**Features**
Worldwide dataset (Jan 24)	Confirmed cases	Daily / country or region	Demographic, health, international mobility and economic features
French dataset (Mar 18)	Hospitalizations	Daily / departments and regions	Population by age, population density. Static mobility data (changes before and after lockdown) extracted from Orange operator.
French dataset with mobility data (Mar 18)	Hospitalizations, Artificial ventilation, Recoveries, Deaths	Daily / departments and regions	Population, population density, Google mobility data
Worldwide dataset with mobility data (Feb 15)	Confirmed cases	Daily / country or region	Google mobility data, demographic, health, international mobility and economic features

3.2 Data extraction

Data curation was partially performed with Excel, then with the Python library Pandas. Data processing was performed in Python with Pandas and Numpy. Deep learning models were created and trained with Tensorflow and Keras. To adapt the data sources to our needs, we developed some assumptions. Our main assumption for this work is that convolutional neural networks (CNN) are robust enough to learn valuable features from data without much pre-processing. Another assumption is that CNN can benefit from data built from countries with different Covid-19 epidemic evolution.

The datasets contain multivariate time-series of different duration depending on the country or the data source. CNNs require input samples to have the same size. Therefore, time-series were converted into input and target sequences of constant size for supervised learning. One problem was the size of the datasets at the beginning of the project. For french hospitals data, only 12 days of data were available for each department. As data before March 18^{th} was unavailable, the input sequences length were limited to 5 days. Also, the classical CNN proposed in Section 4 was used to estimate the missing values before March 18th. This was done to artificially increase the dataset size. This augmentation was performed by reversing the order of the value in sequences. Hence, the equivalent of ten days of sequences was added to the dataset using this technique. While the datasets were daily updated with new sequences, there were only a few hundreds of entries at the end of March 2020, when we started our project. Today, the datasets count with thousands of sequences available for training the neural networks. Due to the nature of the entries, the datasets are relatively small and can be processed quickly on consumer-grade hardware.

Features are normalized to zero mean and unit variance. The Orange mobility data were originally daily time-series but they had to be simplified to constant features. The Orange mobility features represent the change in mobility before and after the beginning of lockdown. This limitation was caused by the difficulty to obtain frequent update of the data. Google's mobility time series were smoothed by a 14-days sliding-window average to remove noise and weekly seasonality.

4 Developing Models for Covid-19 Forecast

As presented in Section 1, we developed several models to cover different indicators for the Covid19 epidemic. In this section we present these models, and how they were applied to the datasets in order to produce forecasts.

4.1 Convolutional Neural Network for Time Series

Convolutional Neural Networks (CNNs) were designed in the late 1980s to solve image classification tasks. They were successfully applied to computer vision tasks like handwritten digits recognition [13]. CNNs are therefore not a new idea but many limitations have prevented them from a more widespread success in the 1990s. Indeed, CNN training requires powerful computers and a vast amount of data, which were not available at the time. Multi-layer neural networks training were also known to be difficult and other techniques like the Support Vector Machine (SVMs) were successful alternatives. CNN only achieved widespread recognition when a CNN architecture, Alexnet [30], won the ILSVRC-2012 challenge. The goal of this challenge was to achieve the best accuracy on the Imagenet dataset (1000 class tasks) and AlexNet got a 10% improvement over the second-best entry.

CNNs are Multilayer Perceptron (MLP) adapted to image processing. A typical MLP is made of fully connected layers only which is impractical for image inputs as the number of parameters would be too high. For example, an MLP first layer of 32 neurons and 28x28 pixels grayscale image input would have 25,088 parameters (and over 66 million parameters for a full HD image), and fully connected layers do not account for the pixel's neighborhood which limit the ability to learn complex image pattern. The CNN solves this issue by using convolutional layers. Here, the fully connected neurons are replaced by 2D convolutional filters. These filters can be seen as shareable neurons as they are applied in a sliding window manner on the whole image, acting as feature extractors that are reused on every part of the image. The motivation for this is that shapes, textures or objects can be anywhere in the image. Convolutions are followed by a non-linear activation to produce the final output. Each filter produces a feature map and each feature maps are stacked together to be used as input in the next layer. A sub-sampling, or pooling, layer can be used after a convolutional layer to reduce the size of the feature maps. A typical CNN uses a succession of convolutional/pooling layers to produce a robust feature extractor. Fully connected layers are then used for classification with a soft-max function. The parameters of the filters and of the fully connected are both calculated by optimizing a loss function. This is generally performed by stochastic gradient descent and with the back-propagation algorithm [41].

As a result, a trained CNN can be described as a hierarchical feature extractor. The first layers can be used to extract low-level features like edges or lines while the next layers can be used for more complex shapes, textures or object parts [48].

The idea of using CNN for time series processing is not new. It was proposed in the 1990s by the original inventor of CNN Yann LeCun [33]. A CNN for time series processing uses 1D filters instead of 2D filters. Time series can be represented as 1D arrays in the same way images are represented as 2D arrays. Few changes can also be made to solve regression tasks. The main changes are the selection of the appropriate output activation and loss function, that can be, for example, ReLU (Rectified Linear Unit) or the mean-squared error.

4.2 Temporal Convolutional Neural Networks

In addition to the CNN-based architecture for time series presented in the previous section, the literature contains other time series processing architectures that worth being studied. One of them is the Temporal Convolutional Network (TCN) architecture. TCNs use techniques that were first used in Wavenet [37]. This model was initially designed for sound-related predictions, as for example in the case of music or speech synthesis, using raw data and set a new state-of-the-art in Text-to-Speech systems. TCNs use causal dilated convolutional layers, in which the convolutions preserve the time causality. Causal convolutions do not use future values of the input sequence to calculate their activation. The difference between a classical and a causal convolution is illustrated in Figure 1. Causal convolutions are used in TCNs with an increasing dilation rate. The dilation rate is a parameter that can be used to expand the input window of the convolution while keeping the same kernel size (some values of the input are ignored). A higher dilation rate corresponds to a higher receptive field. Therefore, many causal and dilate convolutional layers are stacked to each other to process long sequences. This is illustrated in Figure 2 and Figure 3.

One inconvenient of most TCNs is that they do not have pooling layers to avoid a loss of information. This can be circumvented with the use of a residual skip connection, first introduced in the ResNet models [22]. Skip connections are inserted between the input and the output of the convolutional layers to overcome the vanishing (or exploding) gradient problem.

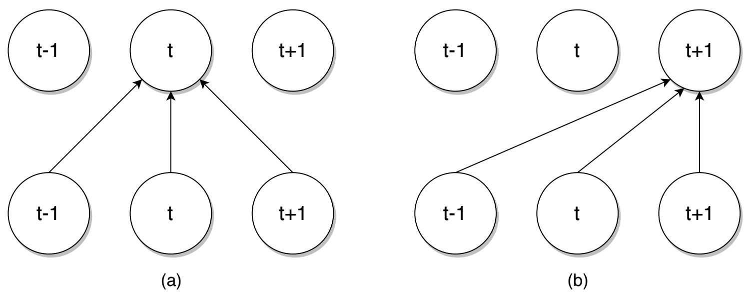

Fig. 1 a) Classical convolution with a size 3 kernel that does not respect time causality. b) Causal convolution with a size 3 kernel [37]

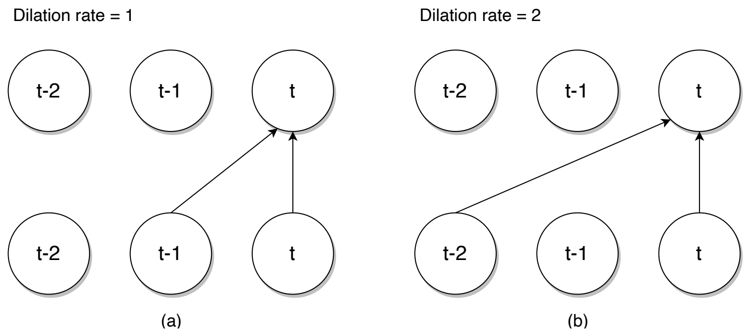

Fig. 2 a) A causal convolution with a dilation rate of 1. b) A causal convolution with a dilation rate of size 2. The receptive field is bigger while having the same number of parameters [37].

4.3 Adapting CNN Models for the Covid19 Epidemic

4.3.1 Description of the proposed CNN

The first proposed model is a classical CNN for time series regression. It was a first attempt at Covid-19 modeling at the beginning of French lockdown in mid-March 2020. The model was therefore limited because the available dataset was small and most of the development time was focused on data curation. The proposed CNN is a one-step-ahead regression model. It takes a 5 days input sequence and produces a forecast for the next day. This limited sequences size was chosen because only few days of data were available at the beginning of the project. Multi-step prediction

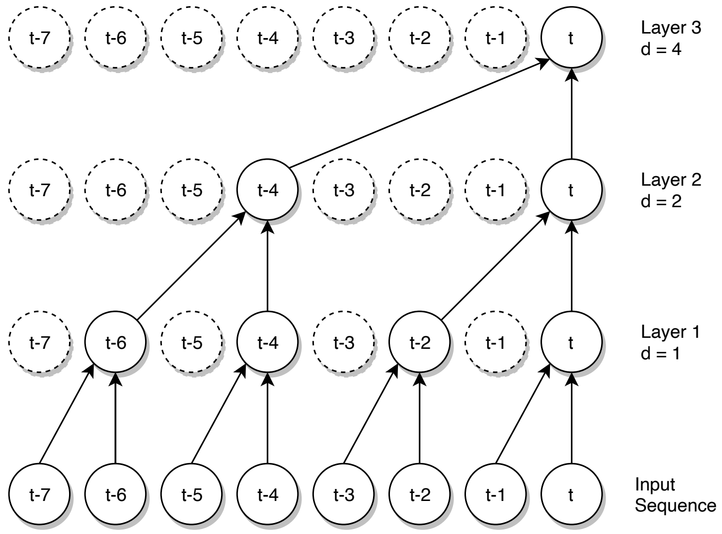

Fig. 3 Stacked causal and dilated convolutional layers. The increasing dilatation rate is used to artificially increase the receptive field while keeping small kernel of size 2 [37].

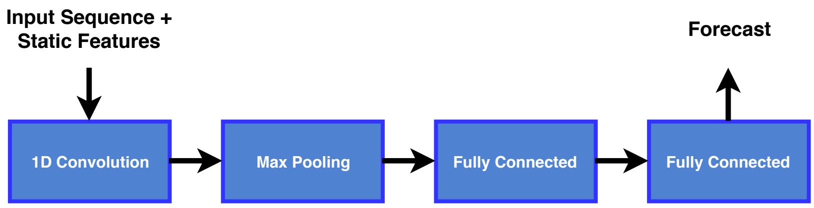

Fig. 4 The proposed CNN for confirmed cases modelling scenarios

is done by shifting the input and by using the one-day forecast as the last value of the input sequence. The CNN uses a single 1D convolutional layer with 2x1 filters followed by a max-pooling layer. Two fully connected layers are then used to produce the final output. The model architecture is described in Figure 4. Another version of this CNN was used. It uses two separates inputs: a convolutional layer for the Covid-19 sequences and a dense layer for the static data. The output of both layers are then concatenated to be used in the final dense layers of the networks. The summary of this CNN architecture can be seen in Figure 5. In this version, dropout was used on the dense input and on the dense hidden layer to reduce the effect of potential over-fitting [45]. A dropout probability of 0.5 was used. Dropout was also used to produce confidence intervals [21]. Both versions of the CNN were trained with the Adam optimizer with a learning rate of 0.0001 and a mini-batch size of 64 [27].

4.3.2 Model for confirmed cases prediction

The first version of the CNN-based model was applied to cumulative confirmed cases forecasting. Three scenarios, corresponding to three independent training,

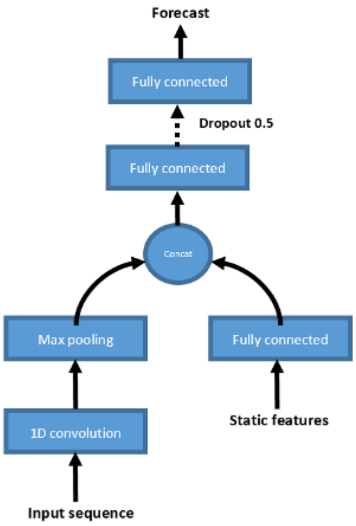

Fig. 5 Multi-input CNN with dropout

were built by filtering the dataset (Dataset 1). The filtering was based on the data available on March 29^{th}. The first, optimistic, scenario used data from countries where the epidemic was stopped. Therefore, this scenario only used data from China and South Korea to train the CNN. The second, compromise, scenario used data from every country that had at least 1000 confirmed cases (including China and South Korea). Finally, the third scenario is the a pessimistic approach to the second scenario where China and South Korea were removed from the training set. The second version of the CNN was applied to daily confirmed cases forecasting. The goal of this model was to create an optimistic scenario of daily increase of confirmed cases. The training was achieved by using decreasing sequences only (corresponding to the countries where the epidemic was receding). Data available on May 8^{th} were used during the training.

4.3.3 Model for hospitalizations prediction

The second version of the CNN was applied to French Covid-19 hospitalizations modeling. This modeling experiment first started in late March when only 12 days of historical data were available (data collection from French hospitals started on March 18^{th}). The training was performed every week to include the new data and projections at both regional and national levels were systematically sent to our contact at the French Ministry of Health. The main problem was to create a model that could be used for surveillance of the epidemic after the end of lock-down on May 11^{th}. Therefore, a baseline model was trained for this purpose. Data available on May 10^{th} was used for training. Only decreasing sequences were kept in the training set. The model was then used for short-term projections. Those projections correspond to a decrease of Covid-19 hospitalizations with a fictional extended lockdown. Projections and observations were compared at both regional and national levels to detect a slow-down of recoveries or an increase in hospitalizations compared to an ideal scenario.

4.4 Adapting TCN Models for the Covid19 Epidemic

4.4.1 Description of the proposed TCN

The CNN-based models proposed in the previous sections have good short-term accuracy but has several limitations. For example, it only uses small sequences of five days because the datasets were too small. Also, more complex models like TCNs are available for longer sequence processing. To circumvent these limitations, we also propose the use of a TCN model. The training of a more complex model was possible because Covid-19 data are updated daily and our datasets were large enough when the development of the TCN model began (mid-May). The TCN proposed here is an adaptation of the conditional TCN introduced by [5], originally applied to financial data forecasting with time series conditioning. This model is built with three blocks. The input block has two separate inputs, one for the main sequences and one for the conditionals sequences. The two separate paths contain a 1D causal convolutional layer as explained in Figure 6. Residual skip connections are used for both inputs. 1x1 convolutional layers can be used in the skip connections to change the number of features maps (to have the same sequence sizes for the addition operation of the skip connection). The central blocks are residual blocks with causal convolutions and an increasing dilation rate (Figure 7).

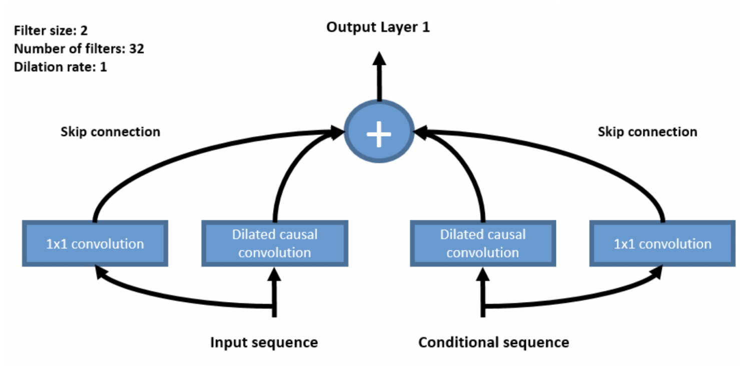

Fig. 6 Input block with 2 separate paths to process the target sequences and the conditional sequences. Residual skip connections are also used with a 1x1 convolution layer to have the same number of features maps before the final concatenation [5].

Multiple central blocks are stacked with an increasing dilation rate. The last central block will have a receptive field that covers the complete sequence length. The output block, from the original publication, is a single 1x1 convolutional layer with one filter per output and followed by a global pooling layer (Figure 8). The TCN proposed in this work uses the same input and central blocks. Each layer have

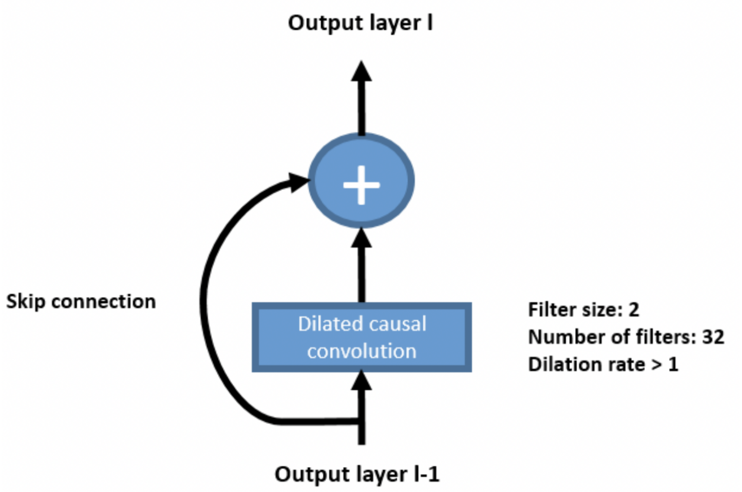

Fig. 7 A central block made of one dilated causal convolutional layer with a residual skip connection. The dilation rate is increased when multiple central blocks are stacked together to process longer sequences [5].

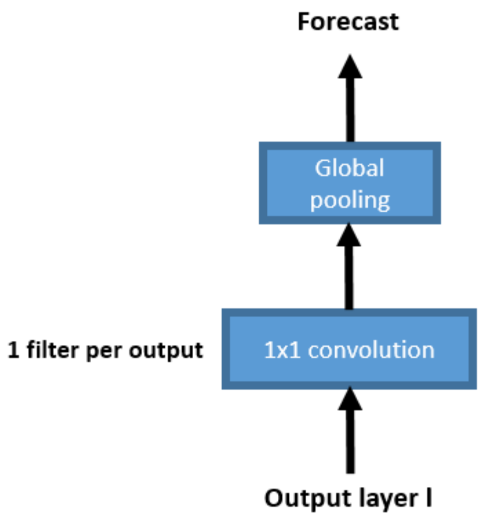

Fig. 8 The original output block with one 1x1 convolution layer and a final global pooling layer to produce a one-step ahead prediction. The 1x1 layer can be adapted for multi-task forecasting by using a filter for each prediction target. Adapted from the figure shown in [5].

32 1D filters of size 2 with ReLU activation. The part that is changing is the output block. It was adapted to produce confidence intervals with quantile regression as proposed in [9, 28]. This is achieved with three independent output blocks with 5%, 50% and 95% quantile losses as explained in Figure 9. The following quantile loss equations were used to train the TCN (Equations 1 and 2):

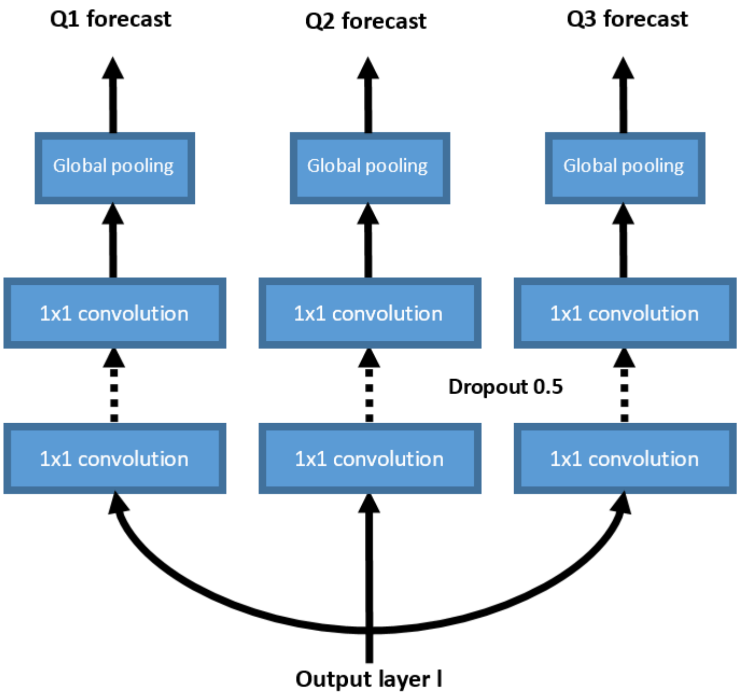

Fig. 9 The quantile regression output block. Each output is similar to the original output block. They used the same feature maps as input and can be adapted for multi-task forecasting by adding more filters to the 1x1 convolutional layers.

$$l(y, \hat{y}, q) = q \times max(0, y - \hat{y}) + (1 - q) \times max(0, y - \hat{y}) \tag{1}$$

$$LOSS = \sum_{(i=1)}^{k} l(y, \hat{y}, q_i) \tag{2}$$

Each output blocks contain a convolutional layer with 1x1 filters with ReLU activation and dropout regularization. This layer acts as a dense layer applied in a sliding window manner on the feature maps produced by the last central block. It is then followed by a final convolutional layer with a single 1x1 filter with ReLU activation and global polling. A multi-task version of the TCN was studied in this work. It is a multi-task model with hard-parameters sharing because there is no output specific to a single target. The whole network is used to produce multiple outputs from different Covid-19 time series. This is motivated because Covid-19 time series, like hospitalizations and deaths, are highly correlated. The multi-task output is achieved by using one filter for each target in the last convolutional layer of the network.

4.4.2 Transfer Learning and Multi-task TCN for regional Covid-19 prediction

The TCN with quantile regression was applied to the four Covid-19 time series of the French dataset 3: current hospitalizations, current hospitalizations with artificial ventilation, cumulative recoveries and cumulative deaths.

Training was first performed independently for each Covid-19 indicators (one-target models): the main input included one of the four Covid-19 time series and the daily variation. The Google mobility data, total population and population density were used in the conditional input. Subsequently, a multi-task TCN was trained with the four Covid-19 time series altogether. This training was performed with the Adam optimizer with a 0.0001 learning rate and a 256 batch size. Early stopping was used for regularization purposes.

We observed that performances at regional and departmental levels were generally lower compared to the national level. Hence, a hierarchical transfer learning scheme was applied to train regional models with improved accuracy. This is illustrated in Figure 10. There, a global (or initial) model is trained with every available data from Dataset 3. Regional models are then trained by using the weights of the initial model instead of random weight initialization. The training at the regional level only uses data from the concerned region.

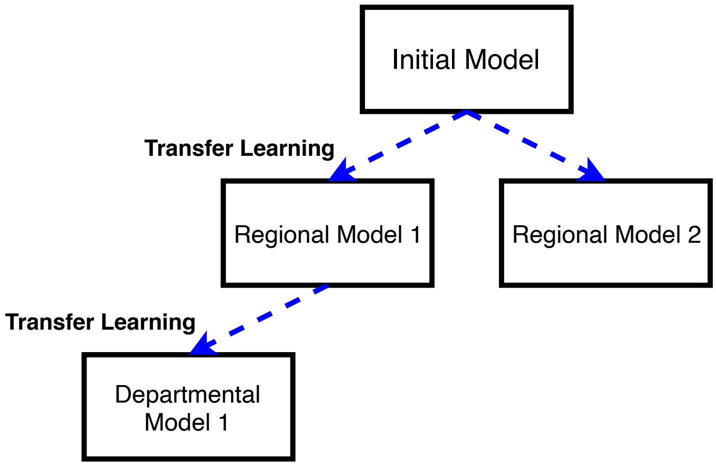

Fig. 10 Example of a hierarchical transfer learning for regional and departmental training

Departmental models can also be trained by following the same technique (the concerned regional model replaces the global model for the pre-training step). In our case, however, the focus was set onto regional levels as the French epidemic is managed at the regional level.

Training multiple models, one for each region, can be highly demanding in computation power. For this step, we mostly relied on the computing resources of the ROMEO Supercomputing Center [1]. ROMEO resources include a mixed CPU-GPU cluster dedicated to HPC and Artificial Intelligence (ranked 249^{th} in

[1] `https://romeo.univ-reims.fr`

the TOP500 list, june 2018), as well as a Nvidia DGX-1 server (8x Nvidia V100 GPUs with 16 GB RAM each and NVLink), specially dedicated to AI.

The training of the initial model was performed on a single Nvidia V100 GPU. Regional models were trained with a multi-thread CPU implementation. Each training was performed on a single thread on an Intel Xeon 40 cores CPU.

4.4.3 TCN for confirmed cases forecasting

The proposed TCN with quantile regression was also applied on French confirmed cases forecasting (Dataset 4) in three different ways:

1. the TCN was trained on French data only for cumulated confirmed cases forecasting,
2. the TCN was trained on Worldwide data for cumulated confirmed cases forecasting
3. the TCN was trained on Worldwide data for daily confirmed cases forecasting

Training were performed with the Adam optimizer with 32, 1024 and 1024 samples per mini-batches and with a learning rate of 0.0001. Early stopping was used as a regularization technique. Negative values (data error) were removed as confirmed cases sequences should only contain positive values.

The three training were performed two times with and without data augmentation. The augmentation technique used is random masking noise (random values of the sequences are set to 0). Training with augmentation were performed by creating 9 new sequences with between 6 and 20 random corrupted values.

4.5 Evaluation Strategies

Classical CNNs showed in Figure 4 and Figure 5 were respectively applied to cumulative confirmed cases predictions scenarios and daily confirmed cases predictions. The second classical CNN was applied to hospitalizations forecasting (Dataset 2). The TCN model was first applied to the modeling of French Covid-19 hospital data (hospitalizations, artificial ventilation, recoveries and deaths) with the Google Mobility reports (Dataset 3). The classical CNN performed poorly when the Google mobility data were used as inputs. The predictions were too sensitive to mobility changes. Finally, the TCN was applied to both cumulative and daily confirmed cases forecasting with worldwide Covid-19 and Google mobility data (Dataset 4). The different experiments are summarized in Table 2. Performances were measured with the normalized root mean-squared-error (NRMSE and the adjusted R-squared as explained in Equations 3, 4 and 5:

$$MSE(\hat{y}, y) = \sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n} \tag{3}$$

$$RMSE(\hat{y}, y) = \sqrt{MSE(\hat{y}, y)} \tag{4}$$

$$NRMSE(\hat{y}, y) = \frac{NRMSE(\hat{y}, y)}{Mean(y)} \tag{5}$$

The TCN transfer learning scheme was evaluated by comparing the sum of regional predictions (which account for the whole country) to country observations. The average regional NRMSE and the weighted average regional NRMSE were calculated to evaluate the improvement of regional forecast performances. Section 5 presents a dedicated analysis for each model described in Table 2.

Table 2 Summary of the 5 modelling experiments

Model	**Target**	**Dataset**	**Training period**	**Forecast period**
Optimistic, compromise and pessimistic scenarios	Total Confirmed cases	Dataset 1	January 22^{nd} to March 29^{th}	March 30^{th} to April 19^{th}
CNN with dropout	Daily Confirmed cases	Dataset 1	January 22^{th} to May 7^{th}	May 8^{th} to July 6^{th}
CNN with dropout	Hospitalizations	Dataset 2	March 18^{th} to May 10^{th}	May 11^{th} to May 24^{th}
TCNs with quantile regression	One model per target (Hospitalizations, artificial ventilation, recoveries, deaths) One multi-task model with the 4 targets	Dataset 3	March 18^{th} to June 5^{th}	June 6^{th} to June 30^{th}
Regional models	Same as TCNs	Dataset 3	March 18^{th} to June 5^{th}	June 6^{th} to June 30^{th}
Worldwide models	Daily or cumulative confirmed cases	Dataset 4	February 15^{th} August 6^{th}	August 7^{th} to September 9^{th}

5 Results and Analysis

This section describes the main results of this paper. A summary of the results is first presented, and a detailed description of the results for each indicator are presented later.

As presented before, our first proposed model for Covid-19 was applied to confirmed cases forecasting. This first model is a classical CNN designed to have a low number of parameters in order to avoid over-fitting. It was trained on worldwide data (Dataset 1) for cumulative confirmed cases modeling with optimistic, compromise and pessimistic scenarios. Results show that confirmed cases evolution followed a different trend compared to projection. Observed data were below the lower predictions from May 8^{th} to June 10^{th} and followed a linear increase. A limitation of this approach is that data filtering must be updated before each training to produce coherent scenarios.

The same model was retrained later when more data were available, providing daily confirmed cases forecasts. Only decreasing sequences were kept during training to produce an optimistic scenario corresponding to an extended lockdown. Results show that the model was too sensitive to extreme values, which produce high daily increase predictions. The model was indeed too optimistic, by predicting a stop of the daily increase by July 2020. Actually, observed data from the end of the lockdown to early July show a linear increase in the cumulative number of confirmed cases.

Confirmed cases is an important indicator because it reflects the spread of the virus in the population, but does not reflect the actual burden caused by the virus on hospitals. Indeed, a massive testing campaign can increase the number of confirmed cases by detecting asymptomatic infected people who don't overload the healthcare system. Instead, Covid-19 hospitalizations is a better indicator because it describes the actual impact of the virus. This indicator is crucial to manage the crisis, as its forecast is directly related to decisions about the number of available beds and staff workers. It is also important to decide when patient must be transferred from a heavily impact region to a less impacted region with available beds.

The classical CNN for confirmed cases forecasting was then applied to hospitalizations forecasting. Training were performed by using French data at departmental, regional and national levels (Dataset 2). Training were performed every week and projections were communicated to the French Ministry of Health. A baseline monitoring model was created by using data available up-to May 11^{th} as training data. Again, we adopted an optimistic scenario corresponding to an extended lockdown. Projections were compared to observed data every week. The decrease of hospitalizations were lower than expected from mid-May to mid-July. This can be explained by the slow-down of hospitals patient recoveries (less people are getting out of the hospitals). These results have shown that modeling can still be performed despite the small size of the datasets.

As seen above, the classical CNN model suffers from many limitations. Performances at the regional and departmental levels were not satisfying and the model was too limited to exploit dynamical data such as Google Mobility Reports. Therefore this work was extended by using a modified Temporal Convolutional Network (TCN) with mobility data (Dataset 3).

The TCN model was successfully applied to five Covid-19 indicators: confirmed cases, hospitalizations, artificial ventilation, recoveries and deaths. In addition, the TCN was successful at different scales: regional predictions could be made from national data, and national predictions were performed from worldwide data.

TCN was first applied to hospitalizations predictions and good performances were achieved at the national level, even of regional level performances were still poor. This was later improved by using a hierarchical pre-training scheme, as presented in Section 4. Finally, the TCN was applied to confirmed cases forecasting and it was found successful at both cumulative and daily confirmed cases modeling (Dataset 4).

The next sections present the detailed analysis for each modeling approach.

5.1 Results for CNN-based Covid-19 modeling

5.1.1 Scenarios for cumulative confirmed cases forecasting

The results presented in this section were obtained with the first classical CNN proposed earlier. The main objective of these scenarios is to model the evolution of the epidemic with the few worldwide data available at the time. From March 30^{th} to April 19^{th} the number of confirmed cases grew from 44,550 to 112,606 with an average of 3,403 daily increase. The daily increase peak was achieved on March 31^{st} with 7,578 new cases. Table 3 contains the performance values for the three scenarios. The optimistic scenario predicted an average of 2,708 daily increase for the same period with a peak on April 3^{rd} with 8,889 new cases. The predicted total cases grew slowly to 98,000 on April 19^{th} with less than 200 daily new cases. The prediction error rate is equal to 9.86% NRMSE for the complete period and 12.94% for the last day of the period. Predictions were optimistic as intended with a final prediction lesser than 14,000 than the observed total cases. The model was optimistic by predicting a slow down to 218 daily increase for the last five days of the periods against 1,600 observed daily increase. The model also predicted a higher peak with a delay of two days.

The compromise scenario is similar to the optimistic predicted peak of 9,467 new cases on April 3^{rd} but with a slower decrease of new daily cases with an average of 811 for the five last days of the period. Predictions grew from 45,570 predicted cases on March 30^{th} and 110,703 on April 19^{th}. The prediction error rate is higher than the optimistic scenario with 12.57% NRMSE but was low for the last day of the period with only a 1.7% error rate. This scenario predicted an average daily increase of 3,257 new cases for the whole period which is close to the observed daily increased (3,403 cases). Predictions for optimistic and compromise scenarios are compared to the real observations in Figure 11 (the pessimistic scenario was omitted for visibility purpose). The pessimistic scenario was different, with a continuous predicted increase in new daily cases. Predictions grew from 44,882 predicted cases on March 30^{th} to 290,864 on April 19^{th}.

Overall, the three scenarios were successful to describe the optimistic, compromise, and pessimistic evolution of the epidemic for the considered period. Optimistic and compromise scenarios were close to observations with the main differences being a higher predicted peak and a higher decrease in new daily cases. This can be seen in Figure 11 with observed data below both scenarios predictions until April 12^{th}.

Table 3 Performances for the confirmed cases modelling scenarios

Evaluation	**Optimistic**	**Compromise**	**Pessimistic**
First prediction	45,873	45,570	44,882
Last prediction	98,830	110,703	290,864
NRMSE	0.0986	0.1257	0.99
R^2	0.85	0.88	0.93

5.1.2 CNN for daily confirmed cases forecasting

Another approach to confirmed daily cases modelling was carried out with the second version of the CNN described earlier. This was performed when more data were available and data from every country were used. Predictions and observations are shown in Figure 12 and performances are shown in Table 4. Dropout was activated at inference time to create 95% confidence intervals. The predicted total confirmed cases grew from about 140,000 cases on May 8^{th} to 155,496 for lower predictions, 164,134 for median predictions and 173,438 for higher predictions on

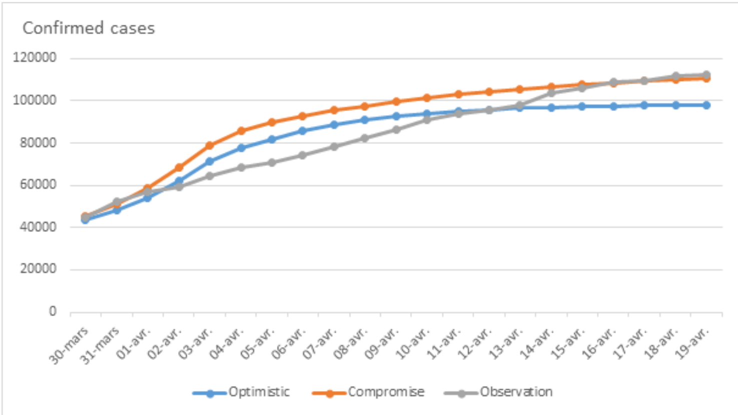

Fig. 11 Optimistic and compromise predictions from March 30^{th} to April 19^{th} compared to observed data

July 6^{th}. In the same period, the observed total number of confirmed cases grew from 138,421 to 168,335. However, observations and predictions have different slopes. The model predicted a daily increase much higher, over 1,800 new cases, that would decrease to a few dozen cases while observation shows a stagnation with 509 daily confirmed cases on average. This much higher predicted daily increase can be explained by the peak in confirmed cases observed on May 6^{th} as raw data were used. Overall, this model was too sensitive to extreme values and it was overly optimistic by predicting a stop of the virus spreading by July 2020.

Table 4 Performances for confirmed cases forecasting

Evaluation	**Lower**	**Median**	**Higher**
First prediction	139,642	140,155	140,579
Last prediction	155,496	161,134	173,438
NRMSE	0.041	0.057	0.099
R^2	0.49	0.53	0.58

5.1.3 CNN for hospitalizations forecasting

The second version of the proposed CNN was also trained to predict the number of hospitalizations cases in France at the country level and regional level. Predictions and observations are shown in Figure 13 and Table 5 contains the performances of the predictions. From May 11^{th} to May 24^{th}, the number of observed hospitalizations in France decreased from 22,115 to 17,021, with an average of 140 daily decreases in the number of hospitalizations. Lower predictions decrease from 20,966 to 17,149, median predictions from 21,554 to 18,587 and higher predictions decrease from 22,115 to 20,292. Observations were inside the confidence intervals for 12 of the 13 regions. Observations for the whole country were close to the lower predictions with 3.2% NRMSE. This model was applied every two weeks on the new data in order to compare predictions and observations and also to detect a potential new wave. Predictions were very close to observations at the country level

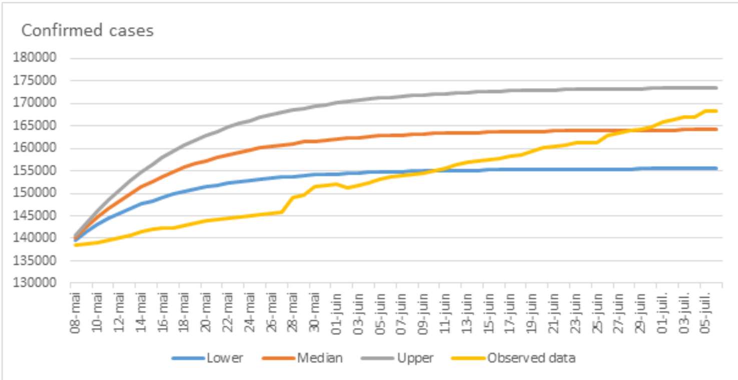

Fig. 12 Predictions from May 8^{th} to July 6^{th} compared to observed data

but not for each region. Predictions tend to be overly pessimistic, with a predicted stagnation or increase for some regions. As we can see in Figure 13, observed hospitalizations from May 11^{th} to May 24^{th} were close to the lower predictions because the virus spreading was stopped and new hospitalizations were low. Observations were closer to the upper predictions from July 6^{th} to July 16^{th} as shown in Figure 14. This can be explained by a slow-down in hospitalized patient recoveries (fewer people are getting out of the hospitals) while new hospitalizations are still observed.

Table 5 Performances for hospitalizations forecasting

Evaluation	**Lower**	**Median**	**Higher**
First prediction	20,967	21,554	22,115
Last prediction	7,149	18,597	20,292
NRMSE	0.032	0.06	0.12
R^2	0.99	0.99	0.96

5.2 Results for TCN-based Covid-19 modeling

5.2.1 TCNs for Covid-19 hospital indicators forecasting

The TCN with quantile regression as described in the previous section is implemented and trained to predict the number of hospitalizations, hospitalizations with artificial ventilation, recoveries, and deaths. These models are compared to the multi-task model that predicts the four targets. NRMSE is calculated for the validation set between median predictions and observations. Country-level performances were compared to regional model performances by calculating the sum of regional predictions that should be close to the national level. The Hospitalization model achieved good accuracy at the country level with 1.2% NRMSE and 0.99 R^2. The sum of regional prediction accurate well with 6.8% NRMSE and 0.98 R^2, observed data were above median predictions and below higher predictions. For

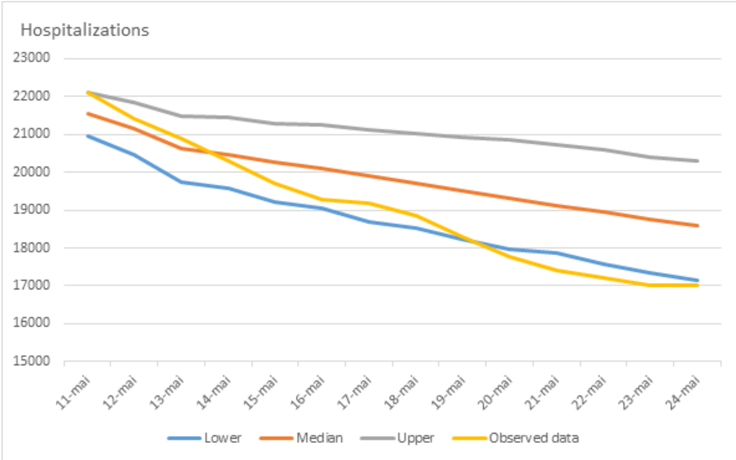

Fig. 13 Predictions from May 11^{th} to May 24^{th} compared to observed data.

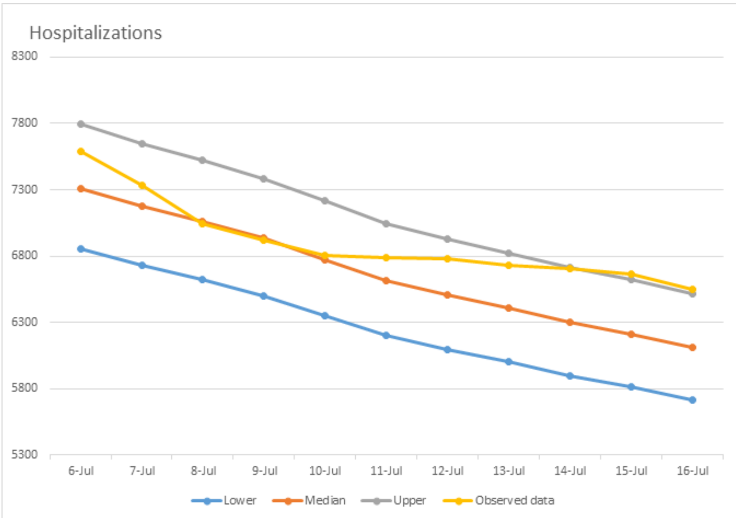

Fig. 14 Predictions from July 6^{th} to July 16^{th} compared to observed data.

both regional and country predictions, the multi-task model creates wider intervals with an upper limit than stagnate (64 daily decreases against 177 for the higher predictions at the country level). In both cases, the observed data are close to the median predictions with 4% RMSE but with lower R^2. Performances are shown in Table 2, and country-level French predictions are plotted in Figure 15.

Similar behavior can be observed with the Artificial Ventilation model. Observed data are close to the lower predictions with 5.6% NRMSE and 0.98 R^2 at the country level. Observation went below the lower limit from June 19^{th} to June 30^{th}. As shown in Figure 16 (country-level French predictions), observed data are close to the median predictions with 5.2% NRMSE and 0.96 R^2 at the regional level. The multi-task model also produced wider intervals that were ex-

Table 6 Performances for hospitalizations predictions with TCNs

Hospitalization models	Training RMSE	Validation RMSE	Validation R^2
One-target – country	0.02	0.01	0.99
One-target – sum of regions	0.03	0.07	0.95
Multi-task – country	0.02	0.04	0.98
Multi-task – sum of regions	0.02	0.04	0.88

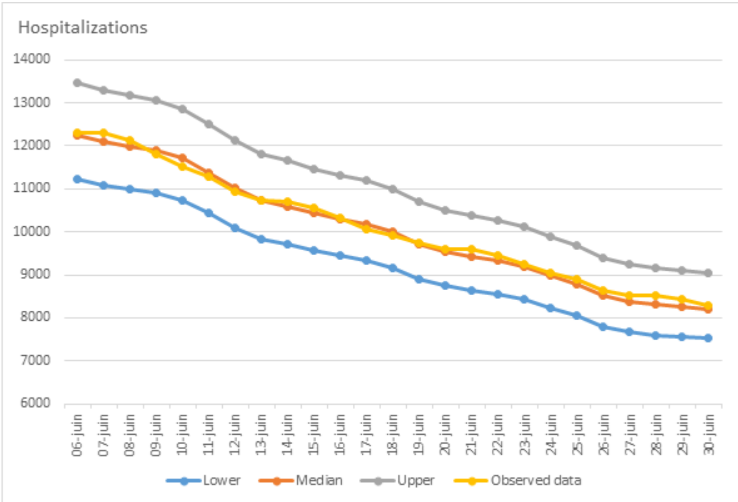

Fig. 15 Hospitalizations predictions from June 6^{th} to June 30^{th} compared to observed data.

pected because artificial ventilation hospitalization is part of the total number of hospitalizations. The multi-task model higher limit has a 2-daily decrease average compared to 16-daily decreases at the country level. The performances of the different models are detailed in Table 3.

Table 7 Performances for Artificial Ventilation hospitalizations predictions with TCNs

Artificial vent. models	Training RMSE	Validation RMSE	Validation R^2
One-target – country	0.03	0.09	0.98
One-target – sum of regions	0.03	0.05	0.94
Multi-task – country	0.06	0.41	0.96
Multi-task – sum of regions	0.05	0.4	0.94

The Recovery model achieved good accuracy at both country and regional level with, respectively, 3.5% NRMSE / 0.98 R^2 and 2.3% NRMSE / 0.95 R^2. The multi-task model improved the performances on both levels with respectively 1.2% NRMSE / 0.98 R^2 and 0.7% NRMSE / 0.99 R^2. The intervals produced by the multi-task model were narrower with a 277 daily increase against 410 daily increase for higher predictions at the country level. It is coherent with the behavior observed for hospitalization because the number of total recoveries is directly linked to the decrease in hospitalization. Performances are shown in Table 4, and country-level French predictions are plotted in Figure 17.

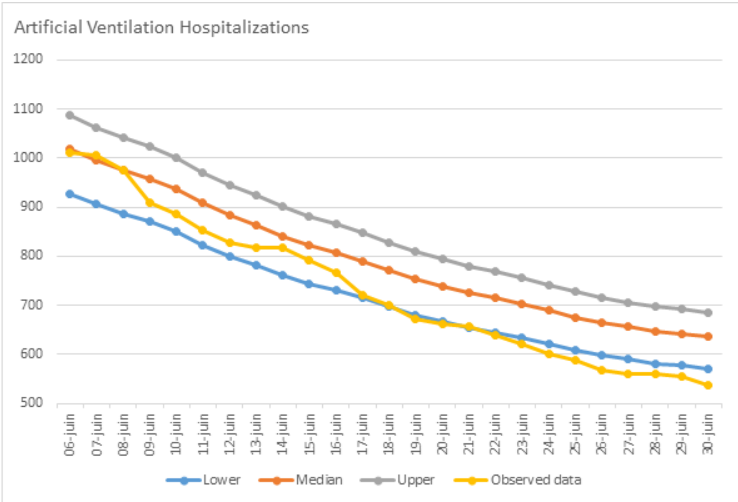

Fig. 16 Hospitalizations predictions from June 6^{th} to June 30^{th} compared to observed data.

Table 8 Performances for recovery predictions with TCNs

Recovery models	**Training RMSE**	**Validation RMSE**	**Validation R^2**
One-target – country	0.01	0.03	0.98
One-target – sum of regions	0.01	0.02	0.98
Multi-task – country	0.01	0.01	0.95
Multi-task – sum of regions	0.01	0.01	0.99

The error rate is low with, respectively, 5.5% and 4.3% NRMSE regarding the predictions of death number by both country and regional model. However, the increasing trend is overestimated with a predicted daily increase of 98 compared to the 21 daily increase observed during the validation period. The predicted increase is lower for the multi-task models, but they produced anomalies with a small decreasing cycle. The predictions trends are still increasing but there are decreasing values that seem to correspond to weekends. This is related to the way data are reported, as fewer deaths are reported on weekends and they are added to the following weekdays. Performances are shown in Table 5, and country-level French predictions are plotted in Figure 18.

Finally, we can say that good performance can be achieved at the country level on the four targets with individual models. The multi-task model can achieve good performances too, with wider or narrower intervals that are coherent with the high uncertainty of the task. However, performances on regional data are not homogeneous among the regions. High accuracy can be achieved in the most impacted regions such as Ile-de-France (Paris region) or Grand-Est (north-east of France), but lower accuracy is observed in smaller regions.

5.2.2 TCNs with transfer learning for regional forecasting

Regarding the hospitalization predictions, better performances were found with regional training with transfer learning with a weighted average of 6.7% NRMSE,

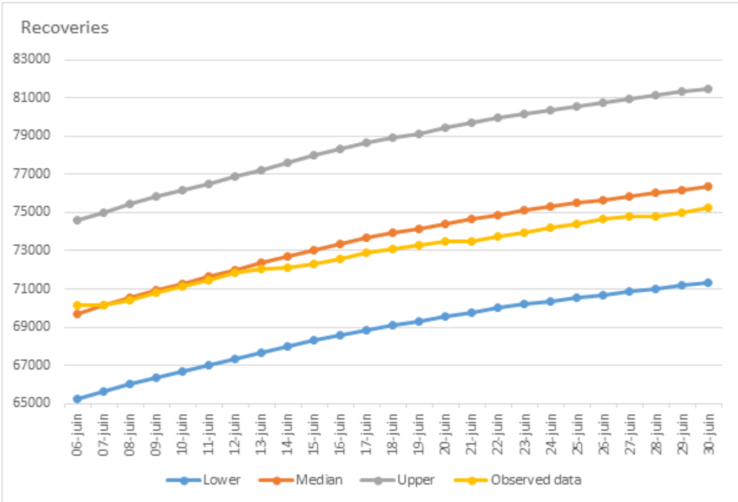

Fig. 17 Recovery predictions from June 6^{th} to June 30^{th} compared to observed data.

Table 9 Performances for deaths predictions with TCNs

Death models	Training RMSE	Validation RMSE	Validation R^2
One-target – country	0.01	0.05	0.95
One-target – sum of regions	0.03	0.04	0.94
Multi-task – country	0.01	0.01	0.95
Multi-task – sum of regions	0.01	0.01	0.50

an average of 15% NRMSE, and 4% sum of predictions NRMSE against 9%, 22%, and 7% with the initial model, respectively. Performances are better for eight of the 13 regions and slightly worse for five regions. The initial multi-task model achieved worse performances with 9.5% weighted average NRMSE, 32% average NRMSE, but presented a better sum of prediction performances, reaching 4%. The regional multi-tasks models achieved the best weighted average performances with 4.7% but with an average RMSE similar to the initial model with 21.5%. The regional multi-task models improved performances in the region that contributed the most to the number of total hospitalization while performances were worse in other regions. All these results are shown in Figure 19.

Similar results with Artificial Ventilation predictions were found. Regional models achieved 8.2% weighted average NRMSE, 20.5% average NRMSE, and 3% sum of predictions NRMSE against 12.3% weighted average NRMSE, 28.5% average NRMSE, and 5% sum of predictions NRMSE. Performances were better on nine of the 13 regions, while four regions presented slightly worse results. Performances were worse on each region with both initial multi-task and regional multitask models, as summarized in Figure 20.

Recoveries NRMSE were lower than 10% for 11 of the 13 regions on both initials and regional models. Performances were improved slightly by the multi-task regional models with 1.3% weighted average NRMSE, 2.7% average NRMSE, and 1% sum of predictions NRMSE against respectively 3.1%, 5.4%, and 2% for the initial model as shown in Figure 21.

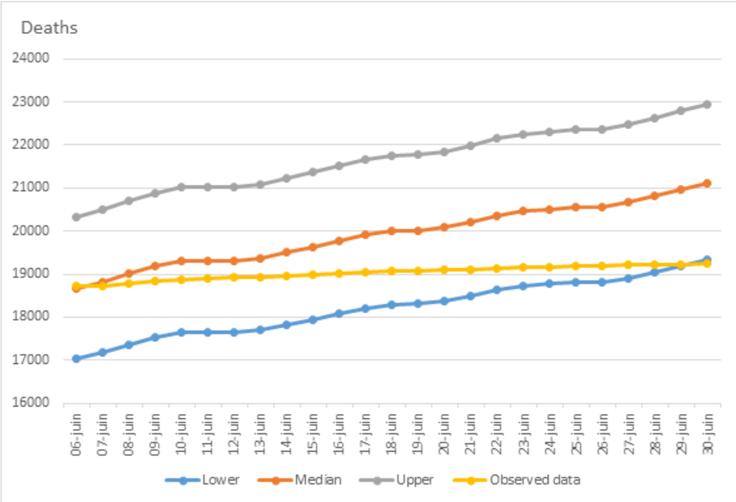

Fig. 18 Deaths predictions from June 6^{th} to June 30^{th} compared to observed data

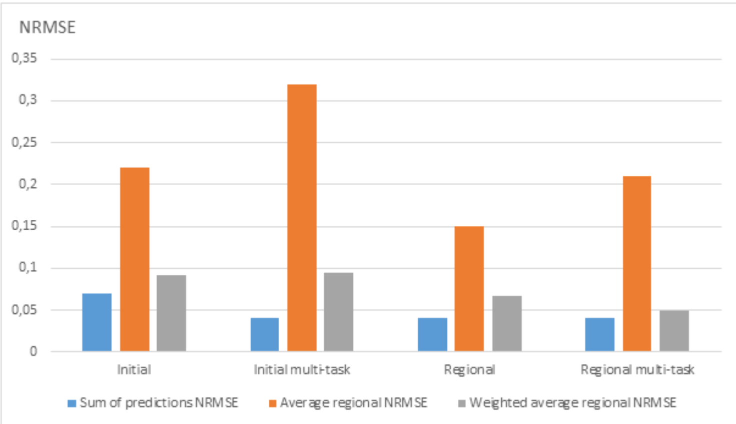

Fig. 19 Performances of the 4 models on hospitalizations predictions

The weighted average NRMSE is slightly lower for deaths predictions with multi-task regional models but decreasing predictions can be found with up to 5 values out of the 25 of the validation periods despite having no decreasing trend on the training set. The one-target regional models' performances are similar to the initial models while the initial multi-task model shows slightly worse performances. One-target initial and regional models show a high correlation with 0.94 R^2 but the predicted daily increase is off with more than 94 predicted daily increase compared to the observed 21 daily increase, all these results are presented in Figure 22.

5.2.3 TCNs for confirmed cases forecasting

The TCN trained on French data only show signs of overfitting with 4.4% NRMSE on the training set and 14.5% NRMSE on the validation set. Also, the projections

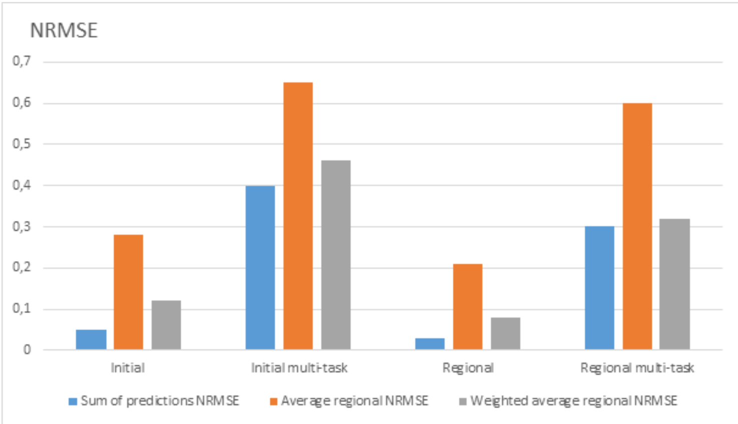

Fig. 20 Performances of the 4 models on artificial ventilation hospitalizations predictions

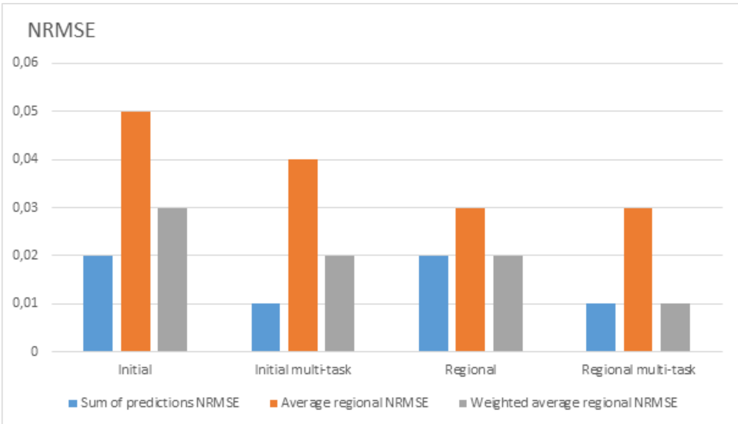

Fig. 21 Performances of the four models on recovery predictions

of this model were incoherent because they were below the values of the input sequences (confirmed cases can only increase or stay constant). The TCN trained with data augmentation had similar performances with 8.2% NRMSE on the training set and a slightly better validation NRMSE of 12.6%. The projections remain incoherent. Performances for this model are shown in Table 10.

The TCN trained on worldwide data for cumulative confirmed cases forecasting achieved 4.1% NRMSE on the training set and 2.7% on the validation set. The performances were of 1% NRMSE for the training set and also 1% NRMSE for the validation in France. The TCN trained for daily confirmed cases forecasting achieved 69.5% NRMSE on the training set and 83.4% NRMSE on the validation set on daily sequences. Performances are shown in Table 11 and predicted values for the validation period are compared to observed data in Figure 23. Performances for France were of 65% on the training set and 29% on the validation set for daily sequences and of 5% and 5% on both training and validation sets for cumulative sequences. Performances are shown in Table 12. The high NRMSE on

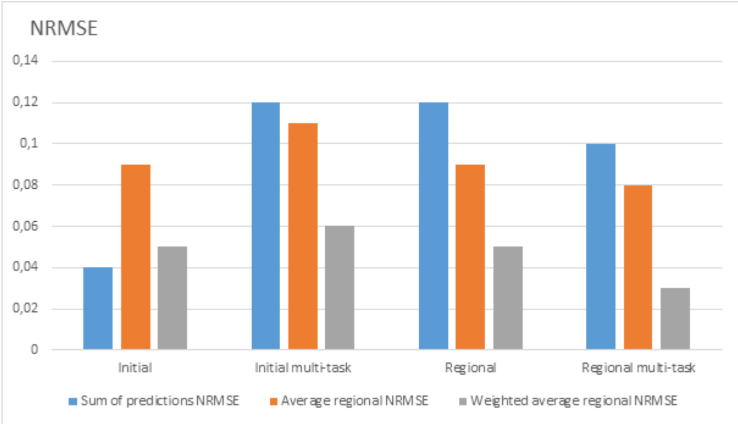

Fig. 22 Performances of the four models on deaths predictions

daily sequences was expected because daily increase of confirmed cases is subject to noise and cyclical patterns related to the organization of Covid-19 test center. However, the actual error on cumulative cases remains low with 5% in France. Data augmentation did not improve the performances of those two models. The performances achieved by both TCN trained with worldwide data show that the proposed TCN can be applied successfully to different Covid-19 related modeling tasks like confirmed cases, hospitalizations, artificial ventilation, recoveries and deaths forecasting. The TCN achieved good performances on both national modeling (confirmed cases forecasting with worldwide data) and regional modeling (hospitalizations forecasting with French data). Finally, the proposed TCN was able to overcome the limitations of the classical CNN proposed in section 4.1.2.

Table 10 Performances for the TCN trained on French data only

Data	Training RMSE	Validation RMSE
France	4.4%	14.5%
France with data augmentation	8.2%	12.6%

Table 11 Performances for the TCN trained on worldwide data for cumulative confirmed cases

Data	Training RMSE	Validation RMSE
Whole dataset (Dataset 4)	4.1%	2.7%
France	1%	1%

6 Discussions and Future Works

The results have shown that CNN and TCN can produce accurate projections of multiple Covid-19 indicators. Best performances for confirmed cases and hospital-

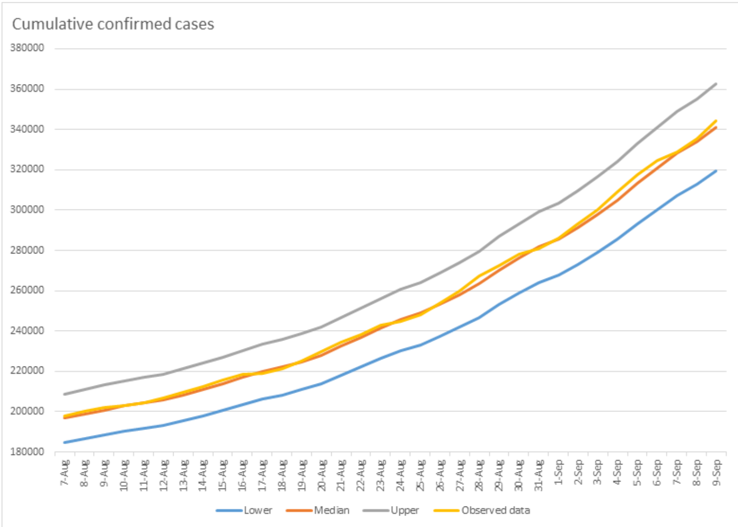

Fig. 23 Predictions and observed data from August 6^{th} to September 9^{th} for confirmed cases forecasting

Table 12 Performances for the TCN trained on worldwide data for daily confirmed cases

Data	**Training RMSE**	**Validation RMSE**
Whole dataset (Dataset 4)	70%	83%
France (daily)	65%	29%
France (cumulative)	5%	5%

izations of the proposed CNN and TCN are compared in Figure 24. TCN seems to achieve good accuracy, but this comparison is limited because the error rates were calculated for different periods. Also, TCN benefited from more data. Overall, TCN can achieve 1% error rate, or less, in the best case. Confirmed cases from September 11^{th} to September 24^{th} have been computed, as seen in Figure 25, and high accuracy was achieved with only 0.5% NRSME.

Training time/cost is a legitimate question that is often overlooked. The size of the datasets and the size of the models, both CNN and TCN, are small enough for the training to be fast on consumer-grade hardware. The training of the TCN can be achieved in about 20 minutes on high-end Nvidia Tesla V100 GPU. The training can also be performed in a mid-range GPU, like the Nvidia GTX 1050, in a reasonable time. The problem of the training cost is more visible in the proposed transfer learning scheme for regional modeling, as each region needs to be trained on different data and a single region dataset is too small to benefit from GPU acceleration. A simple technique to reduce the training time is multi-threaded CPU parallelization. Sequential training takes about 6 hours while parallelized training takes about 45 minutes on two Intel Xeon E5-2698 v4 (20 cores each). This is a costly solution, about 6500$ for the two CPUs. However, sequential training can still be performed in a few hours with cheaper hardware.

As future work, we aim at overcoming some limitations of the current models. For example, the actual TCN model uses constant mobility data for conditional

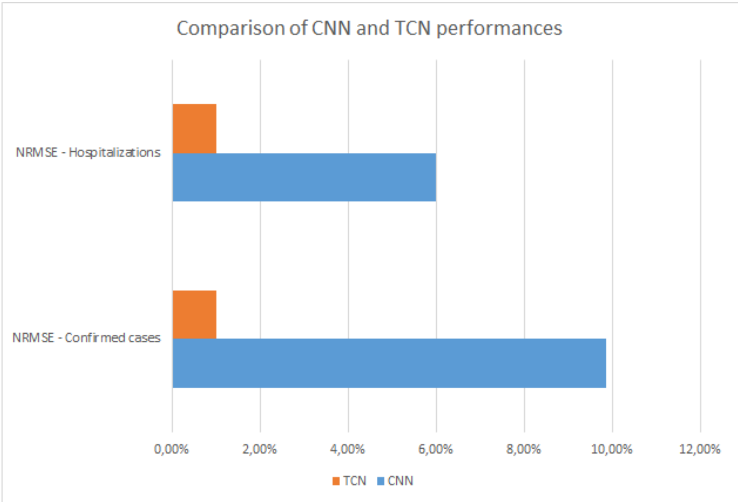

Fig. 24 Comparison of CNN and TCN error rates on confirmed cases and hospitalizations.

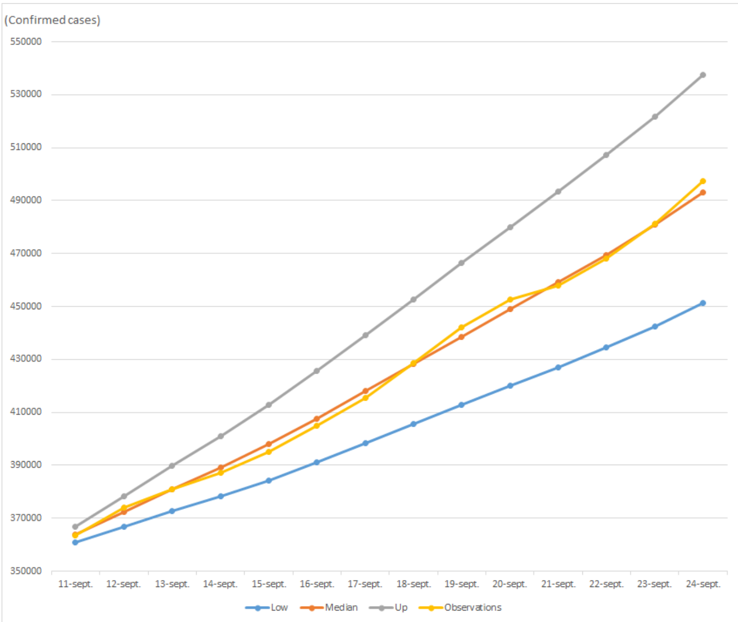

Fig. 25 Comparison of predicted confirmed cases and observed data from September 11th to September 24th in France.

forecasting. Ideally, mobility data would be selected to create optimistic and pessimistic scenarios by setting mobility constraints in the conditional input of the model. One problem is the availability of dynamic data. Many factors have an impact on the evolution of the epidemic and cannot be monitored in real-time. Those factors include the effectiveness of sanitary measures. People's behavior has changed since the first outbreak in March-April. Masks are mandatory in closed spaces, telecommuting is recommended, bars and restaurants can be closed earlier on evenings, universities have limited places etc. Another important dynamical data is regional mobility. It could be used to create projection maps of the virus spreading. An approach using ConvLSTM to generate disease spreading maps have been proposed by [39]. But they did not use mobility data.

The modeling tool GleamViz uses mobility data like commuting networks and air travel but they use compartmental models only [6]. This tool is also designed for a-posteriori analysis or scenario explorations. We believe that a hybrid solution can be designed where, for example, compartmental models could be used to generate artificial datasets. This synthetic data would include many scenarios based on different sanitary constraints (an idea proposed by [4]). Another hybrid solution would be to use deep learning to estimate the parameters of a compartmental model, as proposed by [15] to model the US epidemic. This approach reached good results, predicting that the number of confirmed cases would reach 5 million on August 7^{th}, a number that was finally attained on August 5.

Another development front is related to the multi-task model, which did not improve performances and show signs of over-fitting on two targets (Artificial ventilation hospitalizations and deaths). We believe that this naive multi-task model with hard-shared parameters can be improved with soft-shared parameters and regularization techniques. Constraints would also be necessary to be sure that the predicted values are coherent (eg.: the number of death cannot decrease, artificial ventilation hospitalizations should be lower than the total hospitalizations, etc.). Finally, it is important to ensure that sufficient data is available to feed data-driven models. This implies that data-driven models are less adapted to predict the first wave of an epidemic unless automatic health data collection systems are implemented at the regional or the establishment level. This should be debated as automatic health data collection poses many security and ethical concern regarding the use of data, and is subjected to strict regulation in France.

7 Conclusion

In this paper, multiple data-driven models were proposed for Covid-19 forecasting in France. The proposed TCN can achieve a 1% error rate at the national level for confirmed cases and hospitalizations predictions (compared to 9 and 5% for the proposed CNN). Competitive performances were also achieved on artificial ventilation hospitalizations, recoveries, and deaths predictions with respectively 9%, 3.5%, and 4.5% error rates. The proposed transfer-learning scheme was able to improve accuracy in 8 regions. Several challenges had to be faced when developing these models. The first challenge was the lack of data at the Covid19 outbreak. Hence, one of our contributions was the design of a model based on classical Convolutional Neural Network capable of providing short-term confirmed cases and hospitalizations forecasts, even when few data is available.

Another challenge was the need for more representative metrics on the advance of Covid19 epidemic. Hence, we trained a Temporal Convolutional Network and achieved good accuracy forecasts for indicators such as confirmed cases, hospitalization, artificial ventilation hospitalization and recoveries.

We also had to deal with different granularity levels for our forecasts, as we had to provide both national scale forecasts and regional/departmental forecasts.

Indeed, the French management of Covid19 is mainly performed at regional scale, with a national coordination for resource deployment and patient transfers when needed. We were able to achieve good national and regional accuracy with improved performances through the use of a hierarchical transfer learning scheme presented in this work.

Our models are now being used in a integrated Covid19 monitoring and forecast dashboard that is being developed for the Grand-Est region with the ECOVISION project. The model can be accessed as an online black-box API with HTTPS queries (access can be provided upon request).

8 Acknowledgements

This work was partially supported by the AI4DI H2020 project[2]. The authors would like to thank Cyril Voisard for his help during the data curation process and for his feedback on the models performances. The authors would also like to thank Frédéric Mauguière and Arnaud Renard for the development of the black-box API for our models.

We also would like to thank the ROMEO Computing Center of the University of Reims Champagne Ardenne, where part of the computation (dataset creation and model training) was performed. The Nvidia DGX-1 server allowed us to accelerate the training steps and compare several model approaches.

References

1. Alin, F., Chemchem, A., Nolot, F., Flauzac, O., Krajecki, M.: Towards a hierarchical deep learning approach for intrusion detection. In: S. Boumerdassi, É. Renault, P. Mühlethaler (eds.) Machine Learning for Networking, pp. 15–27. Springer International Publishing, Cham (2020)
2. Apostolopoulos, I.D., Mpesiana, T.A.: Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. Physical and Engineering Sciences in Medicine **43**, 635–640 (2020). DOI 10.1007/s13246-020-00865-4
3. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling (2018)
4. Behrens, T.: Deep learning and diseases (2018). URL `https://ai-guru.de/deep-learning-and-diseases/`
5. Borovykh, A., Bohte, S., Oosterlee, C.W.: Conditional time series forecasting with convolutional neural networks. arXiv:1703.04691 [stat] (2018). ArXiv: 1703.04691
6. Broeck, W.V.d., Gioannini, C., Gonçalves, B., Quaggiotto, M., Colizza, V., Vespignani, A.: The gleamviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale. BMC Infectious Diseases **11**, 37 (2011). DOI 10.1186/1471-2334-11-37
7. Chakraborty, T., Ghosh, I.: Real-time forecasts and risk assessment of novel coronavirus (covid-19) cases: A data-driven analysis. Chaos, Solitons & Fractals **135**, 109850 (2020). DOI 10.1016/j.chaos.2020.109850
8. Chemchem, A., Alin, F., Krajecki, M.: Combining smote sampling and machine learning for forecasting wheat yields in france. In: 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), pp. 9–14 (2019)
9. Chen, Y., Kang, Y., Chen, Y., Wang, Z.: Probabilistic forecasting with temporal convolutional neural network. arXiv:1906.04397 [cs, stat] (2020). ArXiv: 1906.04397

[2] `https://ai4di.automotive.oth-aw.de/`

10. Chimmula, V.K.R., Zhang, L.: Time series forecasting of covid-19 transmission in canada using lstm networks. Chaos, Solitons & Fractals **135**, 109864 (2020). DOI 10.1016/j.chaos.2020.109864
11. Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder–decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734. Association for Computational Linguistics (2014). DOI 10.3115/v1/D14-1179
12. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. NIPS 2014 Workshop on Deep Learning, December 2014 (2014). URL `https://nyuscholars.nyu.edu/en/publications/empirical-evaluation-of-gated-recurrent-neural-networks-on-sequen`
13. Cun, Y.L., Boser, B., Denker, J.S., Howard, R.E., Habbard, W., Jackel, L.D., Henderson, D.: Handwritten Digit Recognition with a Back-Propagation Network, p. 396–404. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1990)
14. DECOV: Decov gitlab repository. URL `https://gitlab.com/covid\_dia/deconf`
15. Deng, Q.: Dynamics and development of the covid-19 epidemic in the united states: A compartmental model enhanced with deep learning techniques. Journal of Medical Internet Research **22**(8) (2020). DOI 10.2196/21173
16. Dong, E., Du, H., Gardner, L.: An interactive web-based dashboard to track covid-19 in real time. The Lancet Infectious Diseases **20**(5), 533–534 (2020). DOI 10.1016/S1473-3099(20)30120-1
17. Fanelli, D., Piazza, F.: Analysis and forecast of COVID-19 spreading in China, Italy and France. Chaos, Solitons & Fractals **134**, 109761 (2020). DOI 10.1016/j.chaos.2020.109761
18. Faranda, D., Alberti, T.: Modelling the second wave of covid-19 infections in France and Italy via a stochastic SEIR model (2020)
19. France, S.P.: Covid-19 - dashboard - suivi de l'épidémie de covid-19 en france. URL `https://dashboard.covid19.data.gouv.fr`
20. France, S.P.: Données hospitalières relatives à l'épidémie de covid-19. URL `https://www.data.gouv.fr/fr/datasets/donnees-hospitalieres-relatives-a-lepidemie-de-covid-19/`
21. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. arXiv:1506.02142 [cs, stat] (2016). ArXiv: 1506.02142
22. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2016)
23. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997). DOI 10.1162/neco.1997.9.8.1735
24. Holmdahl, I., Buckee, C.: Wrong but useful — what covid-19 epidemiologic models can and cannot tell us. New England Journal of Medicine **383**(4), 303–305 (2020). DOI 10.1056/NEJMp2016822
25. Huang, C.J., Chen, Y.H., Ma, Y., Kuo, P.H.: Multiple-input deep convolutional neural network model for covid-19 forecasting in china. medRxiv p. 2020.03.23.20041608 (2020). DOI 10.1101/2020.03.23.20041608
26. Inc., G.: Covid-19 community mobility report. URL `https://www.google.com/covid19/mobility`
27. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv:1412.6980 [cs] (2017). ArXiv: 1412.6980
28. Koenker, R., Bassett, G.: Regression quantiles. Econometrica **46**(1), 33–50 (1978). DOI 10.2307/1913643
29. Kok, C., Jahmunah, V., Oh, S.L., Zhou, X., Gururajan, R., Tao, X., Cheong, K.H., Gururajan, R., Molinari, F., Acharya, U.: Automated prediction of sepsis using temporal convolutional network. Computers in Biology and Medicine p. 103957 (2020). DOI 10.1016/j.compbiomed.2020.103957
30. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, NIPS'12, p. 1097–1105. Curran Associates Inc., Red Hook, NY, USA (2012)
31. Kırbaş, İ., Sözen, A., Tuncer, A.D., Kazancıoğlu, F.Ş.: Comperative analysis and forecasting of covid-19 cases in various european countries with arima, narnn and lstm approaches. Chaos, Solitons & Fractals p. 110015 (2020). DOI 10.1016/j.chaos.2020.110015

32. Lara-Benítez, P., Carranza-García, M., Luna-Romera, J.M., Riquelme, J.C.: Temporal convolutional networks applied to energy-related time series forecasting. Applied Sciences **10**(77), 2322 (2020). DOI 10.3390/app10072322
33. LeCun, Y., Bengio, Y.: Convolutional Networks for Images, Speech, and Time Series, p. 255–258. MIT Press, Cambridge, MA, USA (1998)
34. Liu, D.C., Nocedal, J.: On the limited memory bfgs method for large scale optimization. Mathematical Programming **45**(1), 503–528 (1989). DOI 10.1007/BF01589116
35. Liu, Y., Dong, H., Wang, X., Han, S.: Time series prediction based on temporal convolutional network. In: 2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS), pp. 300–305 (2019)
36. OECD: OECD Economic Outlook, Volume 2020 Issue 1: Preliminary version. OECD Economic Outlook. OECD (2020). DOI 10.1787/0d1d1e2e-en
37. Oord, A.v.d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., Kavukcuoglu, K.: Wavenet: A generative model for raw audio. arXiv:1609.03499 [cs] (2016). ArXiv: 1609.03499
38. Ozturk, T., Talo, M., Yildirim, E.A., Baloglu, U.B., Yildirim, O., Rajendra Acharya, U.: Automated detection of covid-19 cases using deep neural networks with x-ray images. Computers in Biology and Medicine **121**, 103792 (2020). DOI 10.1016/j.compbiomed.2020.103792
39. Paul, S.K., Jana, S., Bhaumik, P.: A multivariate spatiotemporal spread model of covid-19 using ensemble of convlstm networks. medRxiv p. 2020.04.17.20069898 (2020). DOI 10.1101/2020.04.17.20069898
40. Ross, R.: An Application of the Theory of Probabilities to the Study of a priori Pathometry. Part I (1916). DOI 10.1098/rspa.1916.0007
41. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**(60886088), 533–536 (1986). DOI 10.1038/323533a0
42. Salgotra, R., Gandomi, M., Gandomi, A.H.: Evolutionary modelling of the covid-19 pandemic in fifteen most affected countries. Chaos, Solitons & Fractals **140**, 110118 (2020). DOI 10.1016/j.chaos.2020.110118
43. Salje, H., Kiem, C.T., Lefrancq, N., Courtejoie, N., Bosetti, P., Paireau, J., Andronico, A., Hozé, N., Richet, J., Dubost, C.L., et al.: Estimating the burden of SARS-CoV-2 in France. Science **369**(6500), 208–211 (2020). DOI 10.1126/science.abc3517
44. Soubeyrand, S., Ribaud, M., Baudrot, V., Allard, D., Pommeret, D., Roques, L.: Covid-19 mortality dynamics: The future modelled as a (mixture of) past(s). PLOS ONE **15**(9), 1–9 (2020). DOI 10.1371/journal.pone.0238410
45. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15**, 1929–1958 (2014)
46. Swapnarekha, H., Behera, H.S., Nayak, J., Naik, B.: Role of intelligent computing in covid-19 prognosis: A state-of-the-art review. Chaos, Solitons & Fractals **138**, 109947 (2020). DOI 10.1016/j.chaos.2020.109947
47. Tomar, A., Gupta, N.: Prediction for the spread of covid-19 in india and effectiveness of preventive measures. Science of The Total Environment **728**, 138762 (2020). DOI 10.1016/j.scitotenv.2020.138762
48. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (eds.) Computer Vision – ECCV 2014, pp. 818–833. Springer International Publishing, Cham (2014)
49. Zeroual, A., Harrou, F., Dairi, A., Sun, Y.: Deep learning methods for forecasting covid-19 time-series data: A comparative study. Chaos, Solitons & Fractals **140**, 110121 (2020). DOI 10.1016/j.chaos.2020.110121