Featured Prediction Competition

Corporación Favorita Grocery Sales Forecasting

Can you accurately predict sales for a large grocery chain?

$30,000
Prize Money

Corporación Favorita · 1,693 teams · 16 hours ago

sjv

4th-Place Solution Overview

posted in Corporación Favorita Grocery Sales Forecasting 15 hours ago

74

First off, congrats to the winners and thanks to all those who contributed to the forum discussions. I'm definitely disappointed with the result, but I might as well share my solution anyways. I'll only discuss my contribution to my team's solution, which is a single model which scores .499 on the public leaderboard.

Model Architecture

We framed forecasting as a sequence to sequence modeling problem, where the encoder reads in a portion of a time series and the decoder sequentially emits estimates of the subsequent 16 values. The encoder and decoder do not share parameters, but they are both similarly parameterized as a stack of ~30 dilated, causal convolutions. Dilated convolutions allowed the receptive field to span the entirety of the 4+ year train period, while still maintaining a small number of parameters (~1 million). To allow the decoder to use future onpromotion values, it was also augmented with a bidirectional LSTM which encoded information from prior and future onpromotion data. The model struggled to accurately predict zero, so we modified the architecture to additionally emit a bernoulli parameter at each timestep (corresponding to the probability that the output was zero) and the

Input Representation

The input to the network consisted of the raw time series values, embeddings of the categorical variables, and some manually engineered features. The manual features included lags, diffs, rolling statistics, date features, and conditioning time series (i.e. average sales for a given product/store/etc.). Promotion data required some special handling, which I'll describe later.

Validation

5% of the time series were held out, and the model was evaluated on random periods from the last 365 training days of these held out series. The purpose of this was to prevent the model from overfitting a validation set which was biased toward any sort of weekly or monthly trends.

Promotion Data

As discussed in the forums, if you naively impute missing onpromotion values with 0, the sales distribution conditioned on onpromotion values will differ in the train and test sets. I didn't find a way to remedy this, but thankfully my teammates were able to come up with a clever solution which improved my score by about .004. The missing values were imputed randomly: 1 with probability p and 0 otherwise. Determining p was a rather tedious task, since we had no data to model it and we had to resort to trial and error leaderboard submissions. We ended up setting p separately for each day, and it was computed as the mean onpromotion rate for each day, scaled by a learnable factor estimated by stochastic leaderboard descent (leaderboard probing...). All credit goes to my teammates for this, and they can probably explain it better than I can.

Source Code

I normally share my source code, but since the core model is nearly identical to one I've already shared, I'll just refer you to that instead: https://github.com/sjvasquez/web-traffic-forecasting

Finally, I'll admit I'm mostly disappointed with Kaggle. I raised concerns about the unseen items in the test set and also about the onpromotion issue earlier in the competition. Ultimately, the competition was a lottery. For the sake of others, I hope Kaggle will begin to demonstrate a moderate level of care in data preparation and show that they understand that whenever they deviate from the assumption that train set is an unbiased sampling of the test distribution, they induce randomness which turns competitions into lotteries. This will be my last Kaggle competition, as my time will be better spent focusing on research from now on. Best of luck to everyone.

Options

Comments (29)

Sort by Hotness

Click here to enter a comment...

Luck Yu • (2nd in this Competition) • 8 hours ago • Options • Reply 1

Hi sjv, big thanks to you. I adapted your model from WTF to this competition. I think your code is organized well. Solution is clean. I suggest everybody to look at your WTF solution.

Same as you, I was bothered by onpromotion data for very long time. My approach is when decoding, not only use the onpromotion on that day. I added more onpromotion information as feature by shifting. i.e. when decoding, not only depends on the onpromotion on predicted day. also depends on onpromotion data on previous days and future day. So, the question transferred to if yesterday, today, tomorrow, day after tomorrow etc is onpromotion or not onpromotion, what about today's unit sales. This idea is triggered by WTF top solution which shifted the history yearly, quarterly data to feed into its RNN. I did also, meanwhile, I thought the onpromotion for past and future could also shifted to give more information.

Thanks again for your sharing code on WTF.

Giba • (8th in this Competition) • 9 hours ago • Options • Reply 3

I agree about bad data preparation in this competition, but unfortunately it's not the first nor the second time it happens. Kaggle really need to do an octopus review of the datasets before launching competitions.

Lingzhi • (5th in this Competition) • 11 hours ago • Options • Reply 3

Thank you so much and good luck with you. I've learned so much from your previous codes.

Rand Xie • (1038th in this Competition) • 11 hours ago • Options • Reply 4

@sjv, so sad that you are leaving Kaggle. I am always impressed by your solutions that apply DL to time series problems. It gives me a lot of inspiration. All the best to your research.

Louis T. • (13th in this Competition) • 12 hours ago • Options • Reply 3

Among many other things, I am most impressed with your clean code. My code always ends up a mess after doing n different kind of experiment. It would be nice if you can share some tips on your workflow, version control and keeping track of experiences.

Sad to see you go, all the best with your research.

ZavodRobotov • 11 hours ago • Options • Reply 1

Similar semi-random shake-ups happens very oftenly last time on Kaggle. +-100 places, and some times +-700 places in average...

sjv • 11 hours ago • Options • Reply 8

Shakeups are expected in certain competitions, and it's generally possible to identify these before entering the competition. If the data was prepared responsibly and the train/test mismatch was handled more carefully, this competition would have had a much more reasonable shakeup. You should not dismiss the shakeup in this competition as an intrinsic part of machine learning competitions -- it was the result of negligence on Kaggle's behalf (though this is not always the case). I don't like to be this critical, but I haven't seen any evidence that Kaggle takes this issue seriously. Frankly, respected academics and AI/ML researchers don't hold my Kaggle performances in high regard at all, particularly because Kaggle results are extremely noisy signals of actual ML competence. Competitions like this one only further this poor reputation.

FengLi • (15th in this Competition) • 12 hours ago • Options • Reply 1

> As discussed in the forums, if you naively impute missing onpromotion values with 0, the sales distribution conditioned on onpromotion values will differ in the train and test sets. I didn't find a way to remedy this, but thankfully my teammates were able to come up with a clever solution which improved my score by about .004. The missing values were imputed randomly: 1 with probability p and 0 otherwise. Determining p was a rather tedious task, since we had no data to model it and we had to resort to trial and error leaderboard submissions. We ended up setting p separately for each day, and it was computed as the mean onpromotion rate for each day, scaled by a learnable factor estimated by stochastic leaderboard descent (leaderboard probing...). All credit goes to my teammates for this, and they can probably explain it better than I can.

So smart... Does this way also help you improve the private LB score? I'm always impressive with your work and all best with your future research.

sjv • 11 hours ago • Options • Reply 1

For some models it helped and for some it did not. Our private leaderboard scores varied drastically between runs, even when the public leaderboard score stayed the same. While in the end it did not work out very well, I'm confident that the approach of using the leaderboard as validation was the optimal strategy from a statistical viewpoint. We have no choice but to fit our models to the public leaderboard when the organizers don't provide any training data that resembles the test set...

Brendan Finan • 10 hours ago • Options • Reply

-2

This comment is made solely for the purpose of getting the "Make a comment" achievement.

plantsgo • (4th in this Competition) • 13 hours ago • Options • Reply

2

Although there is a little regret,congratulate to be a grandmaster.Good luck with your research!

Takuya Akiyama • (17th in this Competition) • 14 hours ago • Options • Reply

1

Your solutions always excite me. Good luck with your research!

ZedYeung • (322nd in this Competition) • 15 hours ago • Options • Reply

0

Big cong!

ZedYeung • (322nd in this Competition) • 15 hours ago • Options • Reply

0

yeah, I think so. It seems that naively put those weired data like having not enough training data to 0 would be better rather than make the model to predict them. The main problem is that this company just simply make the predict dataset from the product of store, item and date. Really disappointed.

Larxel • (98th in this Competition) • 15 hours ago • Options • Reply

0

Thank you for sharing, and congratulations for your result!

K Lin • (55th in this Competition) • 15 hours ago • Options • Reply

0

That's awesome!!!

PZ • (239th in this Competition) • 14 hours ago • Options • Edit • Reply

0

Thank you for sharing !

Lihaoyang • 13 hours ago • Options • Reply 0

Thanks for sharing your excellent solution once again and wish you the best of luck in your future research/study/work and definitely we will miss you ! And one problem I encountered during using your WaveNet framework is that my model can not even converge(I brutely force the model to consume the raw sequence with 210k*1684 and combine it with some embedded features), I guess the problem might be 2/3 of the series are zero and the distribution of the data is not well handled(my dilated rate might not big enough), could you please give me some more advice on how to tackle with your models since handling problems with DL is really exciting.

Arjun Blum • 13 hours ago • Options • Reply 0

Thanks for sharing. Really clever approach with the promotion data.

HuangBingchu • (86th in this Competition) • 12 hours ago • Options • Reply 0

Congratulations! Thanks for sharing the solution.

ZavodRobotov • 11 hours ago • Options • Reply 0

Hope that Kaggle will fix this issue

Wal8800 • (38th in this Competition) • 10 hours ago • Options • Reply 0

Thanks for sharing the solution

changyc14 • (37th in this Competition) • 10 hours ago • Options • Reply 0

Congrats. You were quite resourceful to deal with a variety of issues. I enjoyed reading it. Thanks for your sharing.

Parijat_Kumar • (1321st in this Competition) • 8 hours ago • Options • Reply 0

Thanks for sharing. ALl the best!!

steelrose • (36th in this Competition) • 8 hours ago • Options • Reply 0 thanks for sharing

btw. I totally agree on the "unseen items" issue - I tried a few things but neither worked realy well, the biggest challenge is that you have no chance to know which items are even planned to be sold at a store - I tried item class + store type approach, item class + store cluster, considering only items which are "onpromotion" in the test assuming that that's the planned start of the new items etc. but neither really worked; would love to know if some of the top teams managed this challenge better than the rets of us :)

Frank Pan • (1523rd in this Competition) • 8 hours ago • Options • Reply 0

Would the method described in the following document a potential solution for missing feature data?
http://kennethmarino.weebly.com/uploads/3/7/2/5/37255427/kenneth_marino_goldwater_essay.pdf
That is, to add an additional binary input node corresponding to every feature to indicate if it is missing

Ricko • (118th in this Competition) • 7 hours ago • Options • Reply 0

Thank you so much, @sjv! I will learn a lot from your approach. I would **love** to study your source code for this competition though! Any chance of sharing this? :) Please :D

CPMP • 6 hours ago • Options • Reply 0

Thanks for sharing. I had the intent to run your WTF model here, but could not find the energy (because of the flu). Congrats for your GM title, and hope you'll be back some time here.

I'm not sure I agree with your negative feelings about how this was prepared. In many real life situations I deal with, the data we have now is different from history, for many reasons, mostly because more data is collected now than before. This competition wad a similar flavor, and it does not turn it into a lottery IMHO. It turns it into something more realistic.

Eric Perbos-Bri... • (1262nd in this Competition) • 5 hours ago • Options • Reply 0

Your works on Time Series was referred to me by another Gold medalist here :-D

And congrats on the GM title.

TBH, on the unseen data, I think it's more a case of Favorita DS team being lazy than Kaggle's fault.

They dumped into the Test set all the potential combinations of 'Item_nbr * Store_nbr', while in Real-Life there is no such thing: retailers worldwide use Store Types, Clusters and Assortment Classes to control which 'Item_nbr' will be present in which 'Store_nbr', and not.

Shelf space is a store's most precious asset and is monetized to major Brands (ie. they often pay for guaranteed space in number of "facings", including eye or hand level for max exposure, then send sales reps to control it in stores and take pictures if not enforced and ask for refunds=messy).

Ex: a size 5 store (max size +10,000sqm) has room for 10,000 different SKUs (Stock Keeping Units) on shelves, like 10 types of Coca-Cola formats (from 4*33cl cans to 6*2L PET bottles),

while a size 1 (min size 2000sqm) has room for 2,000 SKUS only, like 4 types of Coke (nothing larger than a single 1.5L PET bottle).

It's carefully decided at HQ level by Marketing/Merchandising/Purchasing and it triggers down the chain via centralized IT to the store floors:

- a size 1 store can NOT order Coke in 6*2L PET Bottles
- it's NOT listed in his purchasing software client
- its logistic platform/warehouse can NOT deliver it
- its employees can NOT print its shelves' labels, btw there's no room for it as shelves facings are full already
- its cashier checkout can NOT scan it.

Even if that store stole a truck of Coke 6*2L Bottles, it could not sell it via the company's centralized IT system.

That 'Item_nbr * Store_nbr', plus probably 1.5M other rows in the Test set could never happen, and Favorita's people at HQ know it very well.

It's Retail 101 ^!^