# Wikipedia

375 competitors

## Entities:

**Train**

Page (), Agent (), Access (), Project ()

**Test**

Page (), Agent (), Access (), Project ()

## Data:

Date

Page

Agent

Access

Project

Page Views

**Target:** Page views by date

Benchmarks:

Hybrid of growth adjusted snaive and median of medians over different windows based on selection (39.69671)

Median of medians by weekday/weekend

## Takeaways:

* Neural networks where the big winners – taking up 6 of top 8 placings
  + Different architectures and features across contestants, but similar results
    - GRU, LSTM, FF, CNN
* Features: median-based methods did surprisingly well
  + Medians of medians for different intervals
  + Weekday vs. weekend
  + Winner did not use this feature engineering!! (same for 4th and 6th place)
  + Using lags/ACF values in neural networks to help learn seasonality and prevent forgetting of long term dependencies works -> Similar findings as from M4
* Segmented approach based on kalman filter and median of medians placed 8th
* Very different training & validation approaches did well
  + Basic time split and window-based sampling for training
  + Reduce whole training set to one row per series and use grouped k-fold to predict other series given current fold series
  + TSCV by number 8
* Ensembling or stacking still used in the top
  + Multiple NN’s
  + Muiltiple checkpoints
  + Weight averaging
  + Stacking of models using XGBoost
* Scaling
  + Median scaling
  + Log1p

## Notable writeups

#1st Place Arthur Suilin

* RNN Seq2Seq Model using quarter and week lags
  + Easy to incorporate external regressors
  + Non parametric
  + cuDNN GRU cells
* Features
  + Pageviews log1p (historical???)
  + Agent (OHE)
  + Country (OHE)
  + Site (OHE)
  + Day of Week
  + Quarter ago lag
  + Half year ago lag
  + Year ago lag
  + ACF at lag 365 + lag Quarter (90?)
  + Median pageview
* Preprocessing
  + Normalized features to zero mean unit variance
  + Each pageview series normalized independently: learning patterns across scales
* Training
  + Windows of given context length, sampled at random
  + Trained on smoothed version of SMAPE
* Validation
  + Time-based split
* Ensembling
  + 3 different NN’s
  + 10 different checkpoints
  + Stochastic Averaging of weights -> MA of weights
* Hyperparameter tuning
  + Used SMAC3
  + “Contrary to my expectations, hyperparamter search did not found well-defined global minima.  
    All best models had roughly the same performance, but different parameters.  
    Probably RNN model is too expressive for this task, and best model score  
    depends more on the data signal-to-noise ratio than on the model architecture.”

2nd Place CPMP

* Used matching months one year prior as training set
  + Training set: feature based – 1 obs per time series with historical feature information
* Removed trend by subtracting median of last 112/120 days of train for train & test
* 5-Fold CV with split by pages
  + Median of models on each fold for prediction
  + Used for HPT
* Models
  + GLM trained with Huber loss
    - By site & week in test set
    - Train using same week & site the year before with features generated using medians (9 (S) \* 10 (W) models = 90 models
  + XGBoost with custom objective: MAE + custom size of hessian and gradient
  + Feed forward 4 layer Neural network with custom MAE (clipped)
    - Batch norm, skip connection
    - Multi output: forecasts all 63 days per page at once
  + Stacking: Xgboost on residuals + predictions + features from NN + GLM
* Discarding outliers
  + Used model predictions to remove badly predicted rows (SMAPE > 1.25)
* Features
  + Median based features
    - Median of last week, 2 weeks ago,…, median 8 weeks ago
    - Median of last X weeks (2, 4, 8, 12, 16, 20)
    - For all obs, for weekends only and weekdays only
    - Subtract from historical median
  + Other Aggregations
    - Mean of last week, 2 weeks ago,…, median 8 weeks ago
    - Max of last week, 2 weeks ago,…, median 8 weeks ago
    - Median diff of last week, 2 weeks ago,…, median 8 weeks ago
    - Median diff 7 of last week, 2 weeks ago,…, median 8 weeks ago
    - Median diff of 7day rolling median, 2 weeks ago, ..., median 8 weeks ago
    - Mean diff of 7day rolling median, 2 weeks ago, ..., median 8 weeks ago
  + DoW
  + Week of year
  + Project (OHE)
  + Access and agent for NN but not others (OHE)

3rd Place Thousandvoices

* Features:
  + Medians over last 7, 28, 49, 365 days
  + Lag 365
  + Median visits on same weekday
  + Median of medians
* Linear combination of above features
* NN used to predict linear combination
  + 4 CNN Layers with dilation
  + GRU layer
* NN input is last X days of raw values
* Standardization using median over training window
* No backtesting – train test split over pages

4th Place Chung Ming Lee

* Sliding Window Feed-forward NN
  + 120 period lookback period
  + Two encoders: CNN and direct input
  + DoW
  + WoY
  + Embeddings: project, access, agent

5th Place Nathaniel Maddux

* Weighted Ensemble Based on Series Characteristics (Volume, Yearly Seasonality):
  + Simple median model
  + Polynomial AR
    - Medians in past 2, 4, 8, 16, 32, 64, 128 days
    - Used KNN to (k = 150) to take in medians and predict reg coefficients based on Order 1 -> constant + trend linear model
  + Seasonal series: lag 365 with order 2 median regression
* Seasonality indexes using 35 day week day median (weekday median in last 5 weeks)
  + Subtracted mean of median -> Seasonality indexes
  + Gives 7 features per time series: one for each dow
* Used 10 fold CV
* Training:
  + Random slices with train 480 days and future 62 days
* Ensembling: 10x ensemble?

6th Place sjv

* CNN-Based NN
  + Dilated Causal Convolutions ala Wavenet
* Features
  + Agent -> OHE
  + Project -> OHE
  + Access -> OHE
  + Page id
  + Mean of log page views
* Preprocessing
  + Log transform
* Training/Validation
  + Early stopping

7th Place: Chen

* Weighted avg of NN models
  + NN: 1 LSTM and 2 FC Layers
  + Different objectives: MAE & SMAPE
  + Different preprocessing: raw, log1p, avg standardization
* Does not account for yearly seasonality

8th Place: os

* Kalman filters for high signal series
  + 8 state: Local level and weekly seasonality
  + Handled yearly seasonality differently
* Median of medians for low signal series
  + Fibonacci median grouped by weekday vs weekend
* For spiders:
  + Used median of medians without seasonality
* For non spiders:
  + Test for seasonality with 3 different smoothing params
  + If no seasonality, use fallback
* Used back-testing/TSCV
  + Tried different truncations of data