STATISTICS

IMF Pilot Study on Automated Coding of Economic Time Series

APRIL 16, 2020

Team: Marco Marini, Alberto Sanchez, Alessandra Sozzi, Ayoub Mharzi, James Chan, Patricia Escalante, Yamil Vargas

Outline

- Problem set up
- Presenting the data
- Proposed solution
- Preliminary results
- Main findings and next steps

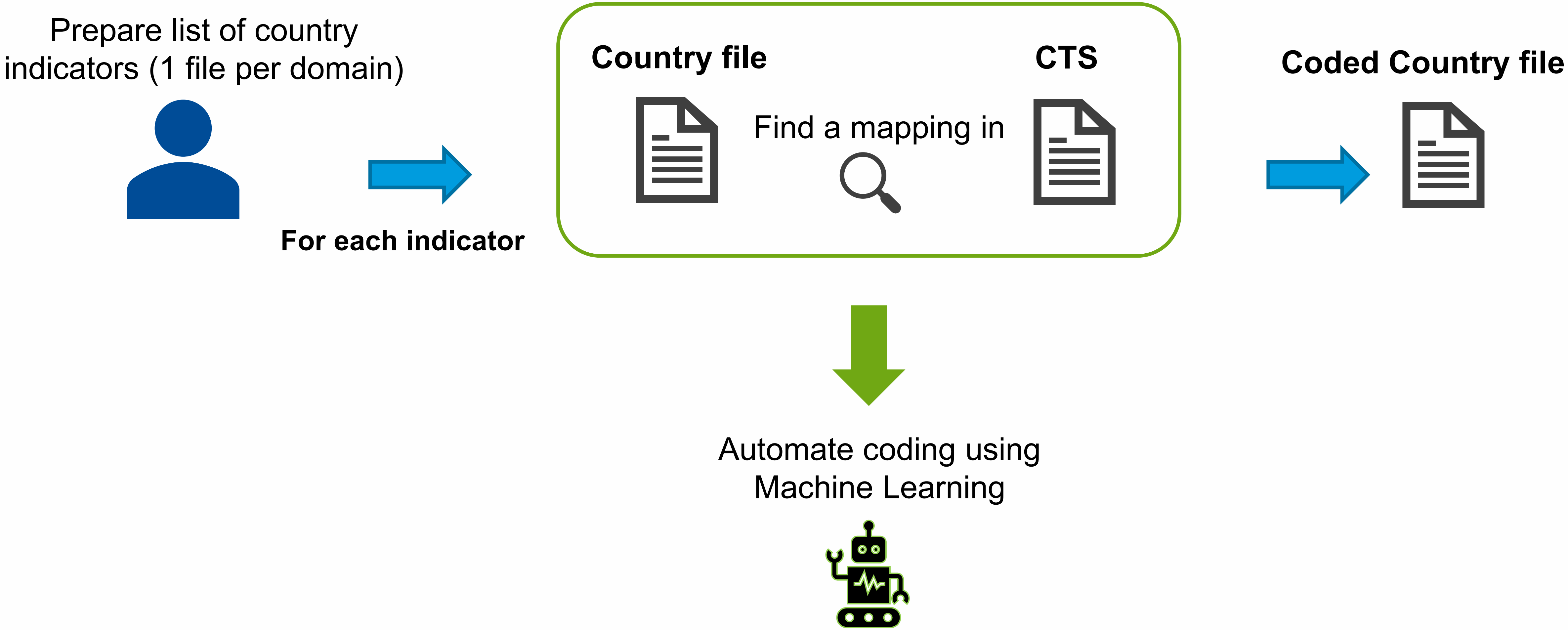
Problem Set Up

- Member countries participating in the Fund's Data Dissemination Standards Initiatives publish economic time series data on their National Summary Data Page (NSDP)

- IMF staff code these series according to an internal Catalogue of Time Series (CTS)
  - Time consuming and cumbersome

- **Objective**: *Create an automated solution to assist with coding*

Current Process



Promising Results from Initial Testing

- Based on our initial results, the best performing machine learning model returns an **80%** accuracy
  - Correct CTS codes generated automatically for **2,898** out of **3,615** indicators

Presenting the Data

- Country indicators that have been already mapped manually

DATASTRUCTURE	IMF:ECOFIN_DSD(1.0)	Datastructure
DATASTRUCTURE_NAME	ECOFIN Data Structure Definition	Datastructure name
DATA_DOMAIN	NAG	Dataset
REF_AREA	AE	Country
COUNTERPART_AREA	_Z	Counterpart area
UNIT_MULT	6	Scale = Million
FREQ	A	Frequency = Annual
COMMENT		Observation status

Descriptor	INDICATOR	BASE_PER	2013	2014	2015	2016	2017
Nominal GDP by Activity	NGDP_PA_ISIC4_XDC	_Z	1432669.89	1480521.39	1315250.5	1311248.3	1405006.8
Agriculture, forestry and fishing	NGDPVA_ISIC4_A_XDC	_Z	9223.06	9468.23	9746.34	10175.82	10721.07

Presenting the Data

- Catalogue of Time Series (CTS)
  - 28,886 codes in CTS

Code	Full Descriptor	Methodology Reference	Sector - Name	Topic - Name
NGDPVA	National Accounts, Activity, Memorandum Items, Gross Value Added, Nominal		National Accounts	Activity
NGDPVAGA	National Accounts, Activity, Memorandum Items, Gross Value Added, of which Government Activities, Nominal		National Accounts	Activity
A_CPC21_0	Economic Activity, Production, By Central Product Classification (CPC) Version 2.1, Agriculture, forestry and fishery products	FAO SEEA AFF; CPC Version 2.1	Economic Activity	Production
ACO_CPC21_0	Economic Activity, Consumption, By Central Product Classification (CPC) Version 2.1, Agriculture, forestry and fishery products	CPC Version 2.1	Economic Activity	Production

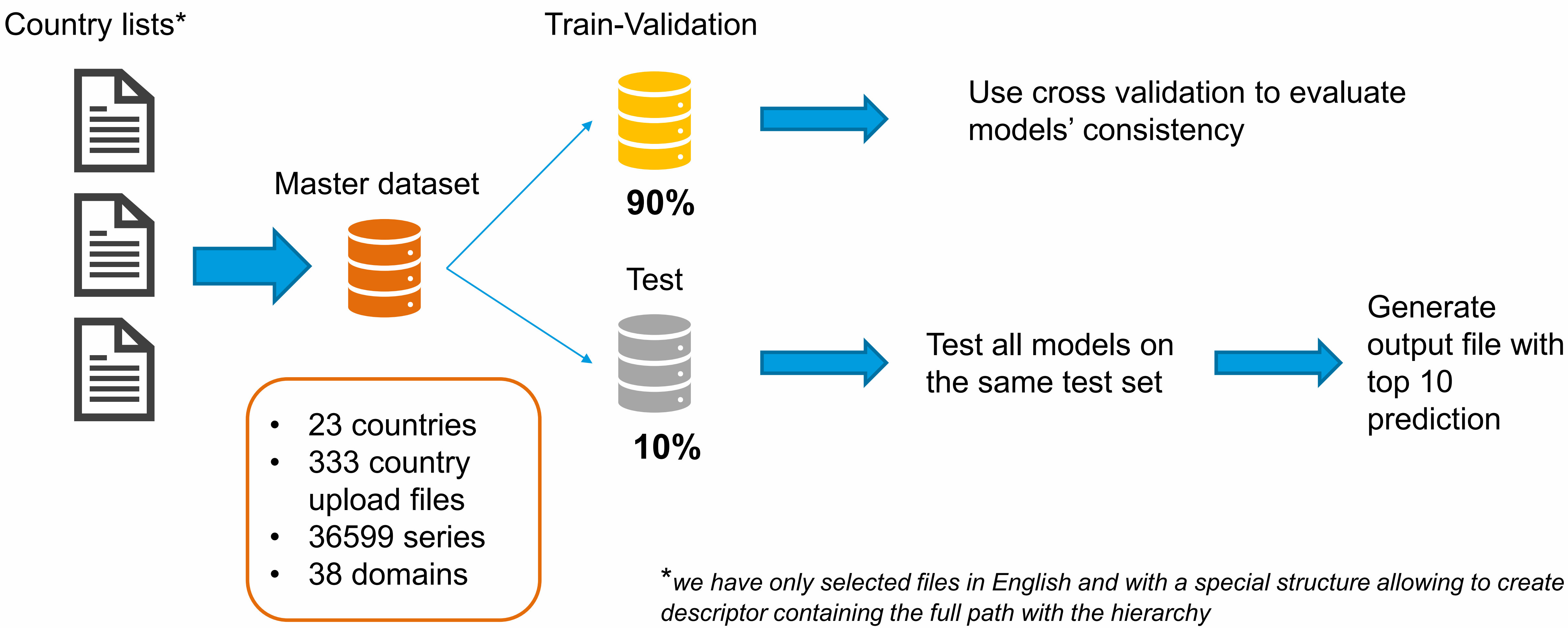
- Extension descriptors

Code	Name
_SA	Seasonally adjusted
_XDC	Domestic Currency

Proposed Approach

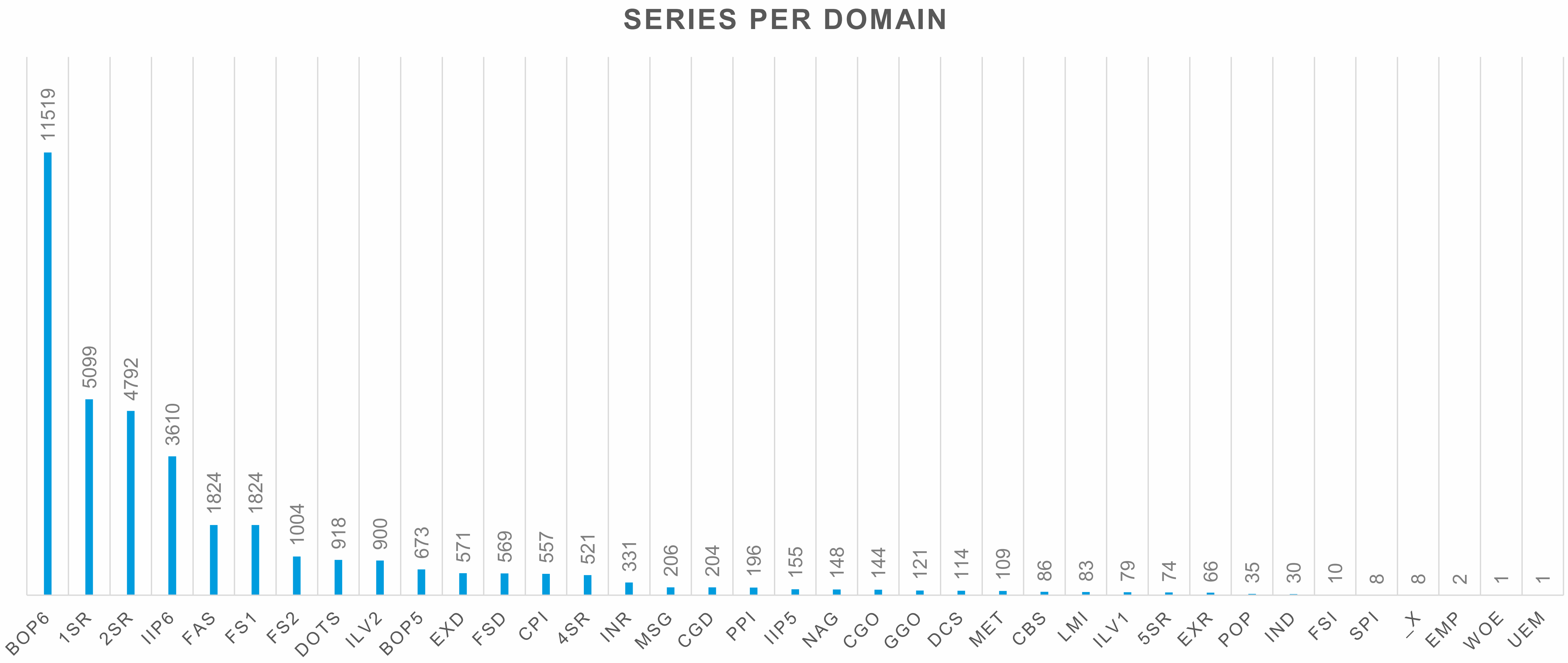
- Supervised Learning Models
  - Logistic Regression
  - Nearest Neighbor
- Feature extraction
  - TF-IDF
  - Word2Vec
- Time series clustering (experimental approach)

Proposed Approach

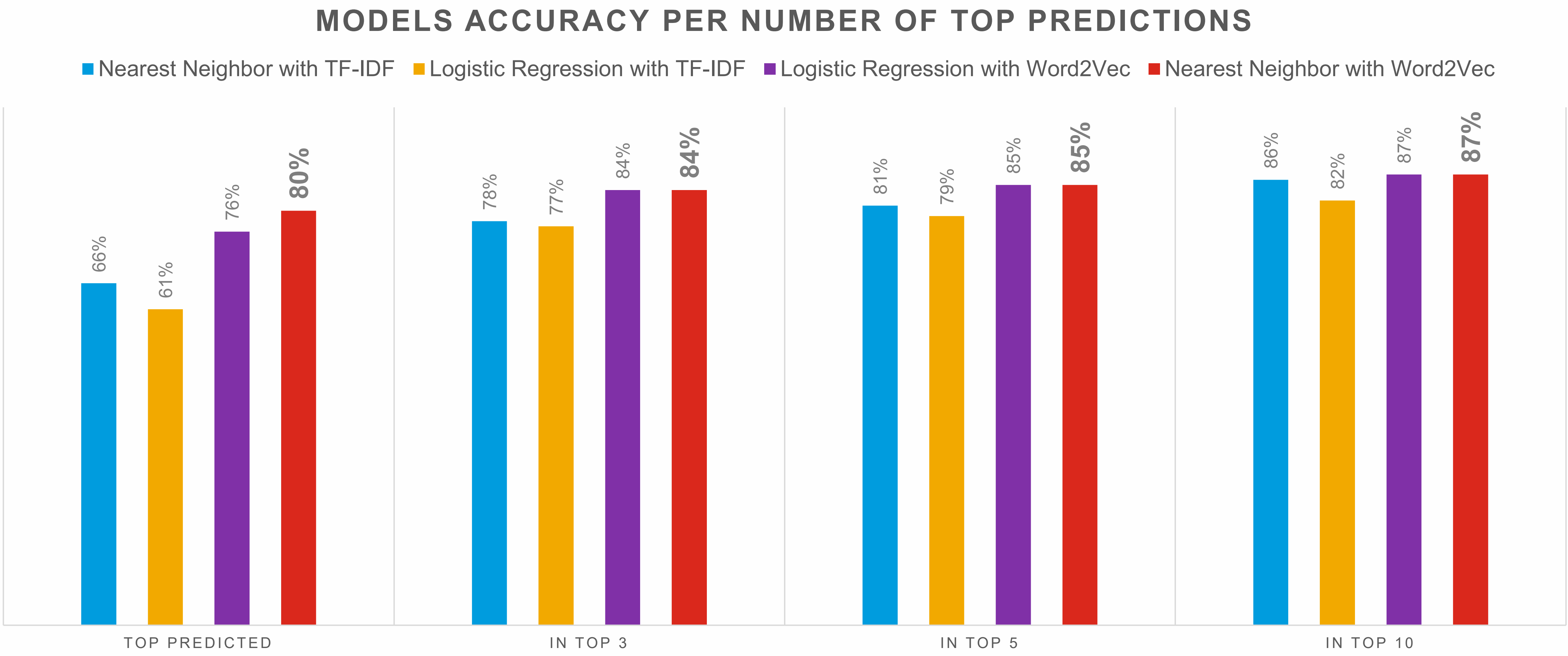


**we have only selected files in English and with a special structure allowing to create descriptor containing the full path with the hierarchy*

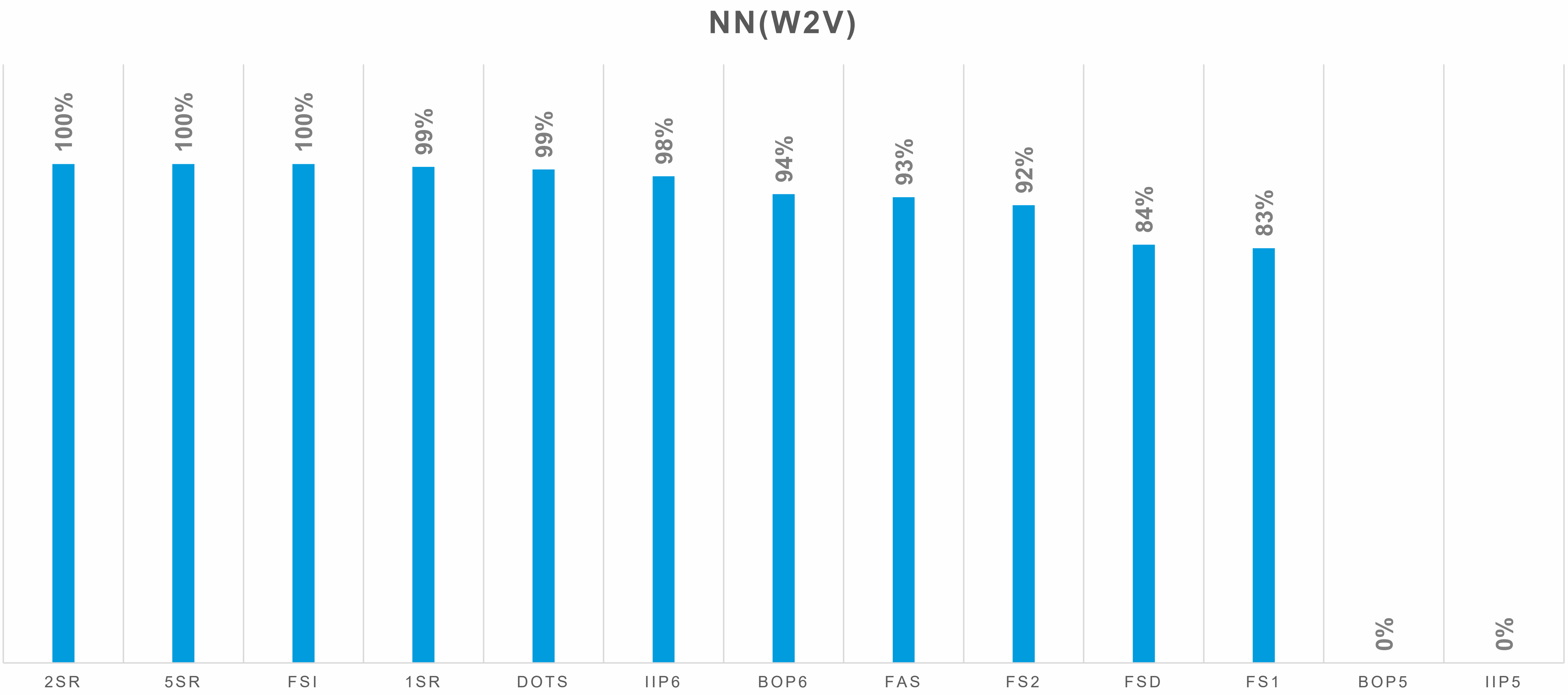
Data Distribution



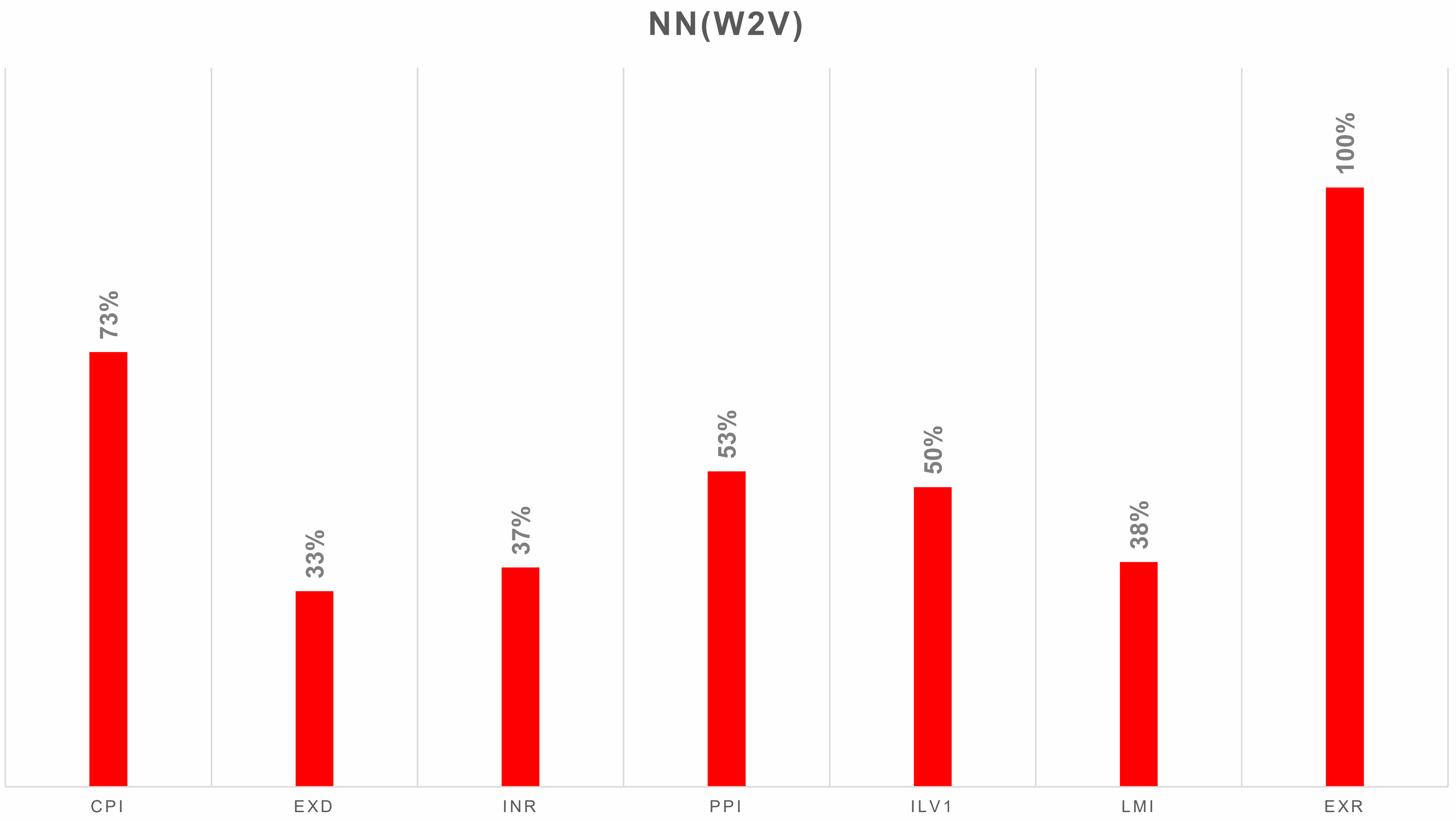
Word2Vec Feature Extraction Provides Better Results than TF-IDF



On Average, Better Performance for Domains with IMF Standardized Report Forms

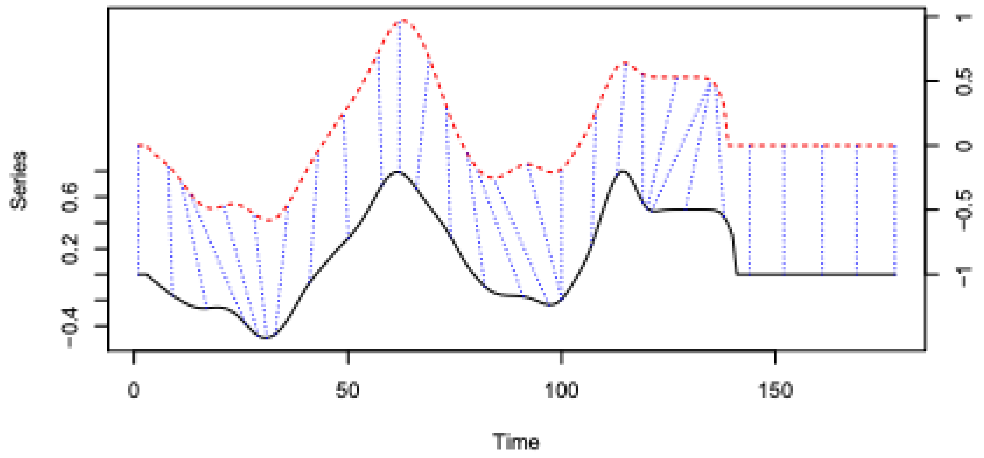


Accuracy Dropping for Non Standardized Reports



Model Descriptions: Time Series Clustering

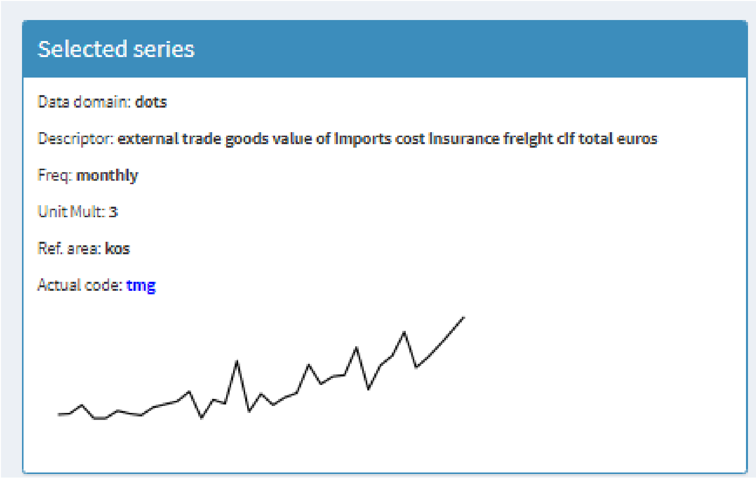
- Use **Dynamic Time Warping (DTW)** to dynamically compare two time series and find the optimum warping path (blue dotted line mapping points of the 2 time series) between them under certain constraints, such as monotonicity

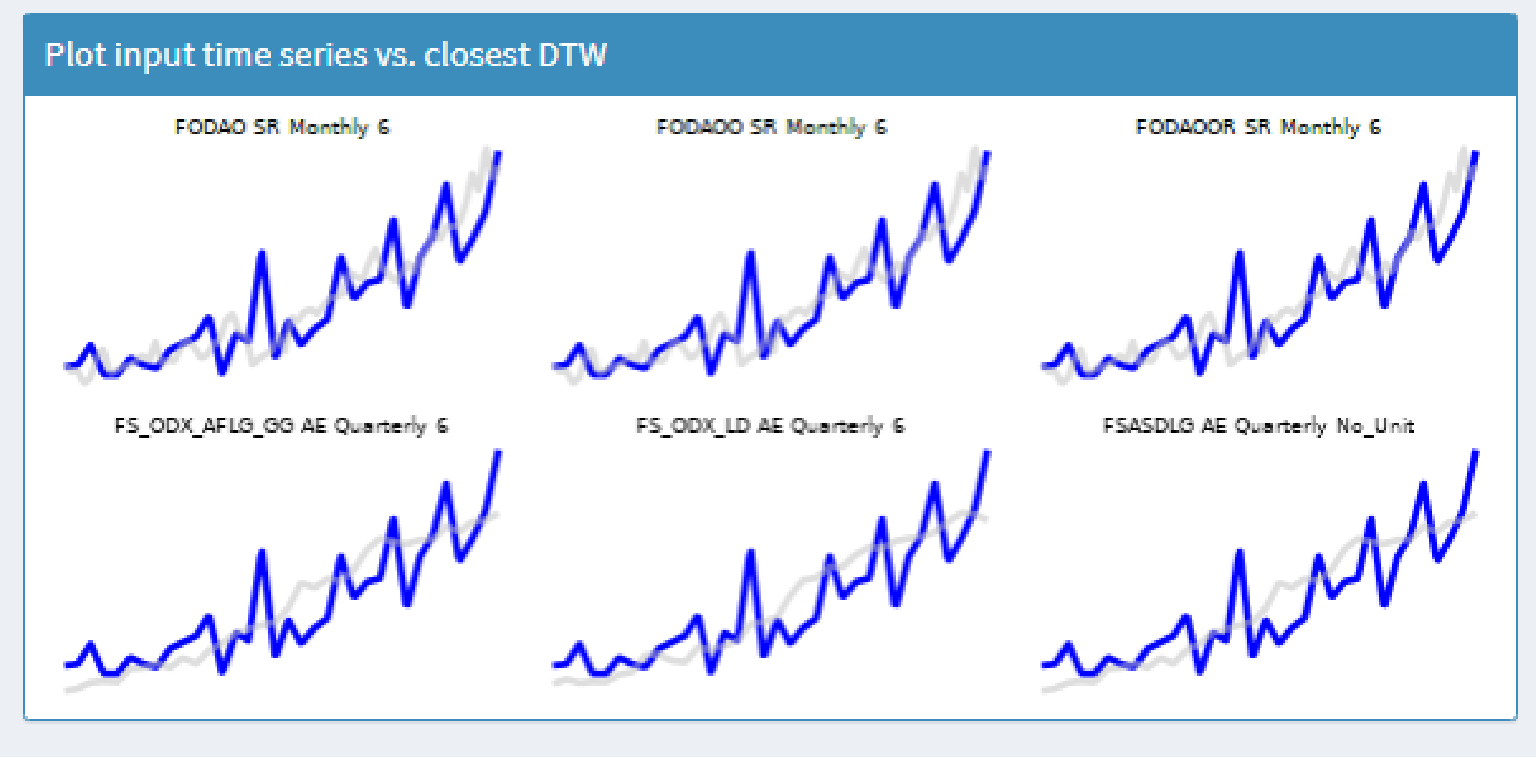


- The idea is, for a given set of time series (our training dataset), we can calculate the closest one to a given new time series using DTW as the distance measure.

Time Series Clustering: A Working Example

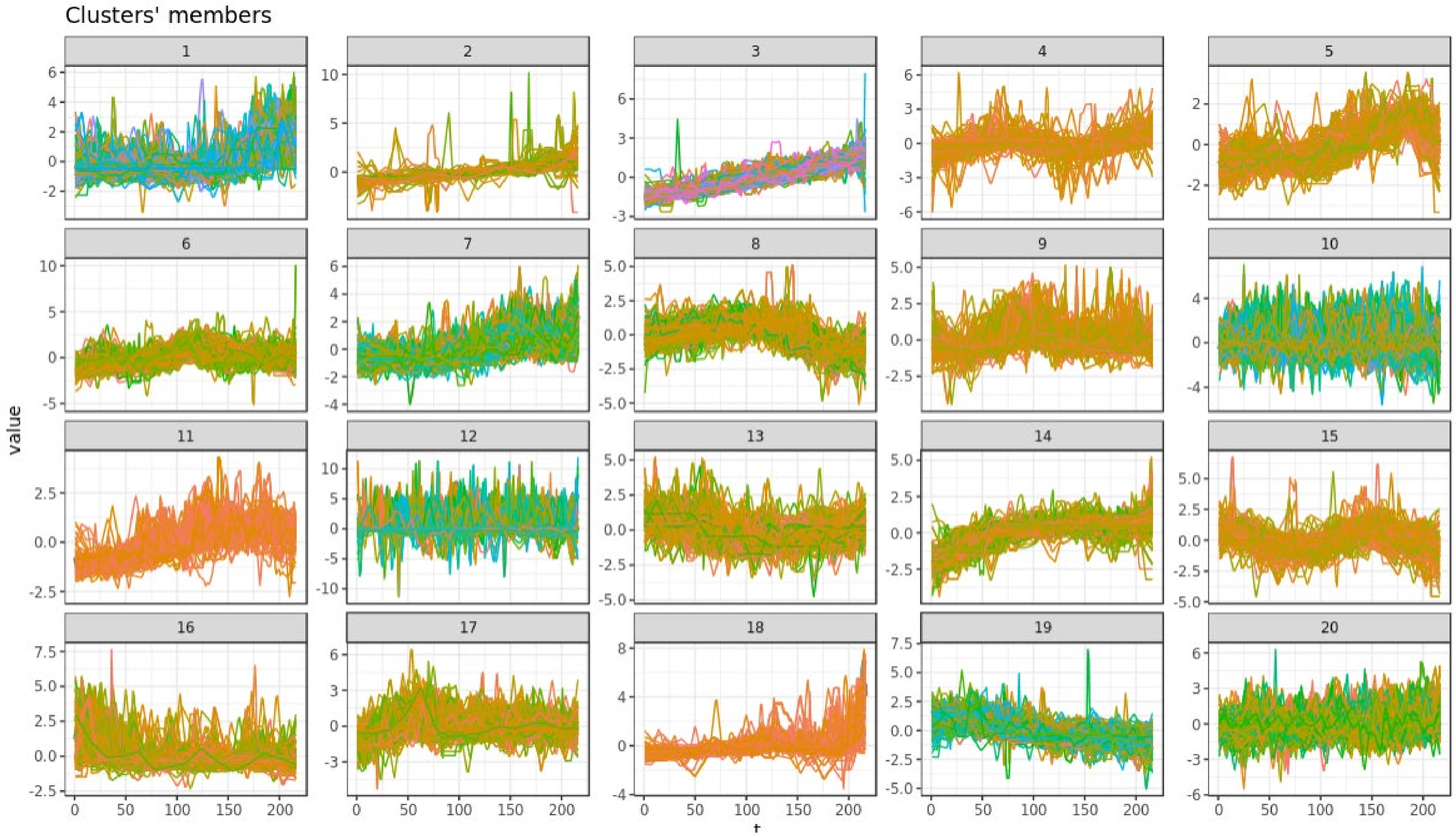
For each new input time series we can calculate the nearest neighbor in the training dataset based on the DTW distance





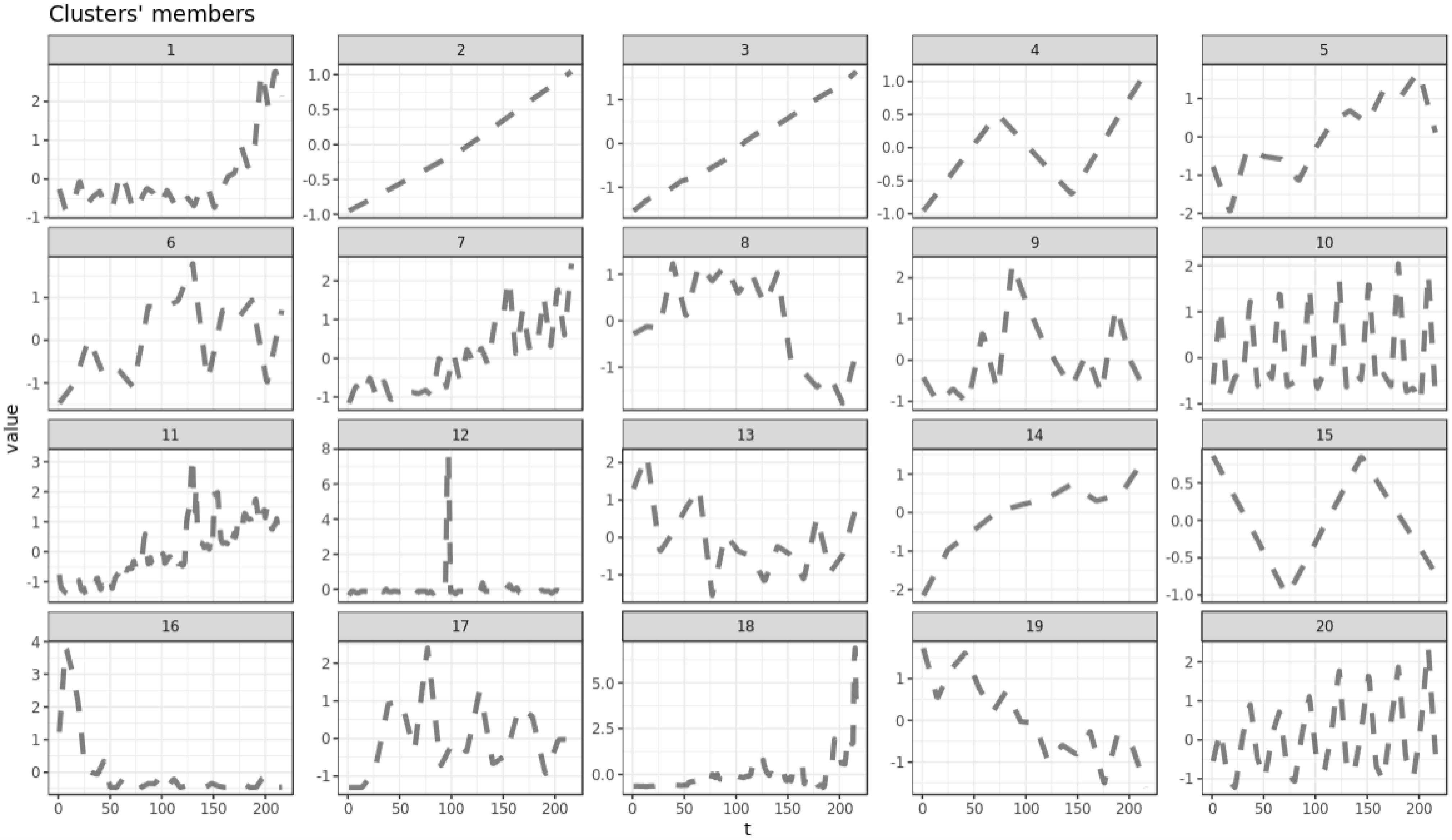
Another Example: Partitional Clustering

We can classify all of our time series in the training data into different types (20). We can use this to complete auto-coding of series by adding the extensions. For example: seasonality, etc.



Another Example: Partitional Clustering

And obtain prototype time series for each cluster



Shiny App

Select series to predict:
Predicts 1st series by default

Domain

1sr

Descriptor

other accounts receivable other accounts receivable other resident sectors

Reference Area

al

Frequency

monthly

Unit Multiplier

no_unit

Predict Code

Selected series

Data domain: **1sr**

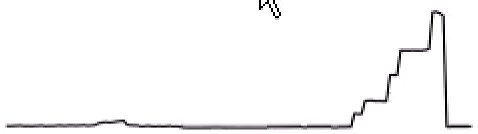
Descriptor: **other accounts receivable other accounts receivable other resident sectors**

Freq: **monthly**

Unit Mult: **no_unit**

Ref. area: **alb**

Actual code: **faafoors**



Top 10 predicted CTS codes

rank	dtw_nn.x	tfidf_lr.x	tfidf_nn.x	w2v_lr.x	w2v_nn.x
1	faafoors(al)(monthly)(no_unit)	faafoors	fodaoor	faafoors	faafoors
2	faafoorssg(al)(monthly)(no_unit)	fodaoor	faafoors	faafoo	fodaoor
3	faafoorss(al)(monthly)(no_unit)	faafoorssr	fodaoor	faafoorssdc	fofaoor
4	tmg(al)(monthly)(6)	faafoorsd	faafoors	fodaoor	faafoono
5	fodafddf(mn)(monthly)(6)	faafo	fodaoor	fofaoor	faafoonimf

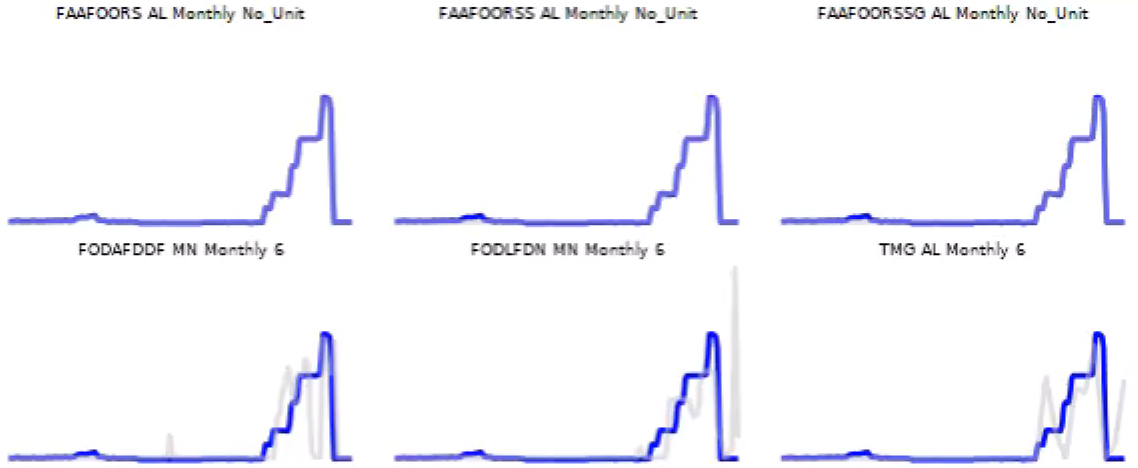
Previous 1 2 Next

Summary

predicted	mean_rank	sd_rank	count_in_top10
faafoors	1.20	0.45	5.00
fodaoor	2.25	1.26	4.00
faafoo	4.00	2.83	2.00
fofaoor	4.67	1.53	3.00
faafoorssr	5.00	2.83	2.00

Previous 1 2 Next

Plot input time series vs. closest DTW



Main Findings from Initial Results

- Data structuring was easier for standardized report forms in order to generate full path descriptors for series
- Time series descriptors contain valuable information to predict codes
  - Better accuracy for standardized report forms since we have more series
- Difficulty of restructuring of the non-standardized report forms impacted the number of series collected and potentially the models accuracy for these series
- Potentially an ensemble of all the models can provide better accuracy
- Overall initial results are promising

Next Steps

- Ingest more data into the master dataset
- Use proper cross validation for parameter selection
- Combine models to improve prediction
- Predict the extensions
- Plan moving the solution to production
  - i.e. Set up quality control thresholds

Questions?

TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency):

- Evaluates the importance of a word in a document (1 descriptor) and in a collection of corpus (descriptors in the entire dataset)
- TF: frequency of a word in the descriptor
- IDF: how many times the word occurs in all of the dataset
- Example:

DATA_DOMAIN	REF_AREA	UNIT_MULT	FREQ	DESCRIPTOR
1SR	AL	No_Unit	Monthly	MONETARY GOLD AND SDRs
1SR	AL	No_Unit	Monthly	MONETARY GOLD AND SDRs, Monetary Gold
1SR	AL	No_Unit	Monthly	MONETARY GOLD AND SDRs, Holdings of SDRs

If we take the word “Monetary” in the second row descriptor

- The TF is: 2(number of occurrences)/5(number of words in the descriptor) = 0.4
- The IDF is: log(3(number of descriptors)/4(number of the overall occurrences)) = -0.12
- The TF-IDF = 0.4*(-0.12) = -0.048

This will be done for each word in each descriptor (besides stop words such as “a”, “the” etc.) Each descriptor becomes a vector of numbers

Sentence embeddings (Word2vec)

- Group of models that tries to represent each word in a large text as a vector in a space of N dimensions (which we will call features) making similar words also be close to each other. One of these models is the Skip-Gram.
- The main idea behind the Skip-Gram model is this: it takes every word in a large corpora (we will call it the focus word) and also takes one-by-one the words that surround it within a defined 'window' to then feed a neural network that after training will predict the probability for each word to actually appear in the window around the focus word.
- Intuition: the model will generate similar vectors for words that share the same context words. For example:
- "fin" and "financial" will be close in this embedding space because both will have "corporations" or "assets" next to them frequently.
- Finally, the vector for the whole sentence (descriptor + unit + domain + freq) is generated by averaging the vectors of the words that form the sentence.

Logistic Regression

- Classification method for multiclass problems, i.e. with more than two possible discrete outcomes. This model tries to predict the probability of assigning a given series descriptor to each of the different CTS codes.
- For any given input series (descriptor + unit + freq + domain) the model will return a probability vector of length the total number of different CTS codes (or labels) in the training dataset.
- Ideally, one of these probabilities for a given input will be close to 1 and we will be able to assign that code to our input series. In general, we will aim at returning the top 10 probabilities for each input series.

Nearest Neighbor

- Classification method for multiclass problems, i.e. with more than two possible discrete outcomes. This model retrieves the K **closest** descriptors in the training data to the new descriptor we want to assign the CTS code.

- For K=1, for any given input series (descriptor) the model will retrieve the closest descriptor in the training data and assign to the input series the same CTS code assigned to the closest descriptor.

- For K=10 the model retrieve the closest 10 descriptor in the training data and ideally we expect the correct code to be the majority of the returned ones, or at least present in the set.

Catalog of Economic Time Series (CTS)

- The Catalog of Economic Time Series (CTS) provides a standard framework for the structure, nomenclature, and coding of economic indicators and times series used within the Fund. It consists of a list of economic concepts and codes commonly used in the Fund as well as a set of coding rules. Standardized codes help improve data management practices and facilitate Fund-wide data sharing.

- The CTS is the authoritative source for economic concept codes used within the Fund. It is the main reference for country desk time series, the World Economic Outlook (WEO) database, the Common Surveillance Database (CSD), and other regional and functional department databases, as well as databases available at