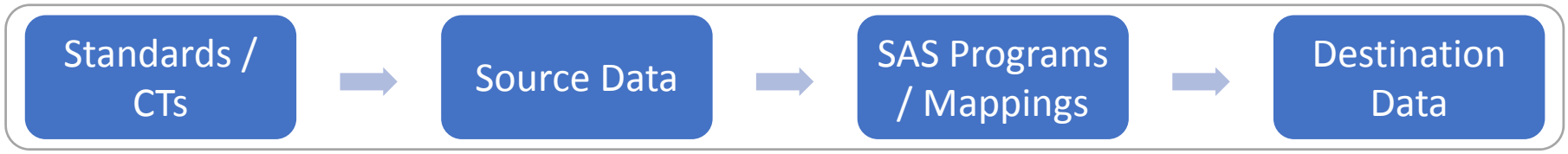
GUF 2018 - Espace Palissy, Boulogne-Billancourt

Data Mapping using Machine Learning

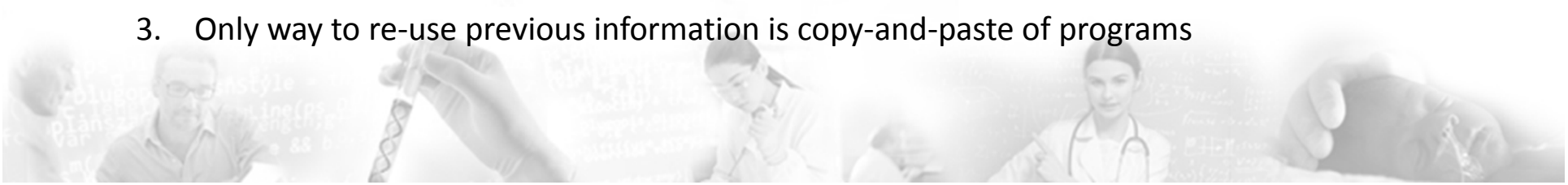
Stijn rogiers, Global Health and Life Sciences Practice, SAS
Member of Europe CDISC Coordinating Committee (E3C)

May 2018

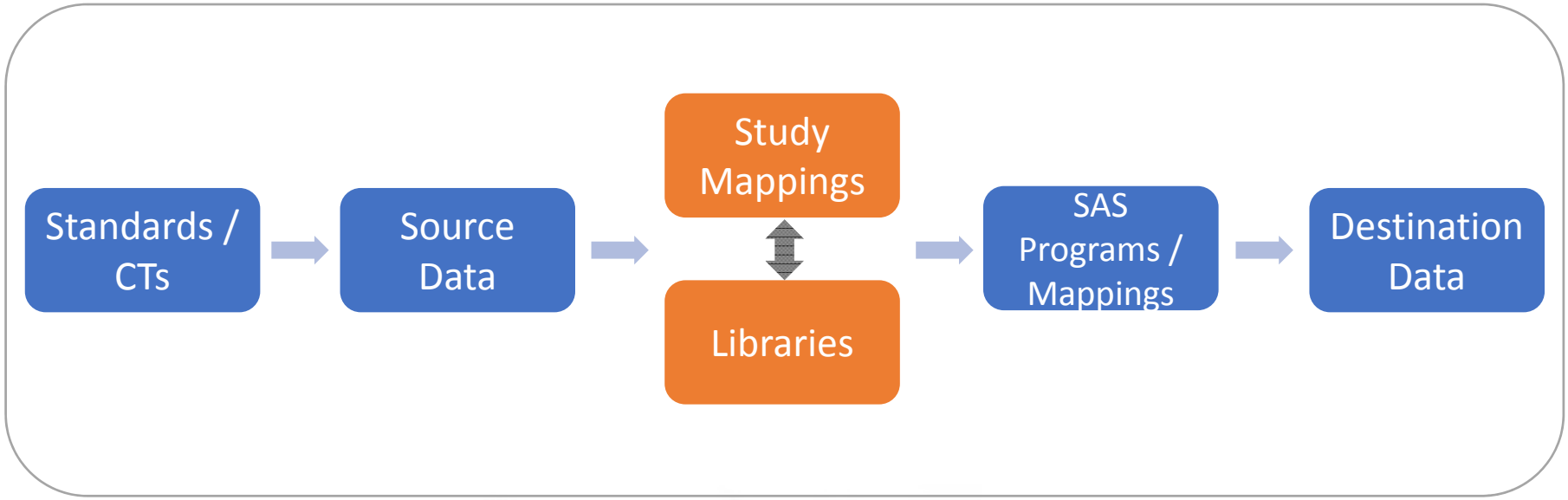
Problem definition



- Source Data is mapped to Destination Data based on Standards for Clinical Studies
- Standards are guidelines rather than instructions, leaving room for individual interpretation
- Mapping is currently done in individual SAS programs which:
  1. Lacks collection of central metadata
  2. Leads to inconsistencies in way mapping is done across different studies
  3. Only way to re-use previous information is copy-and-paste of programs



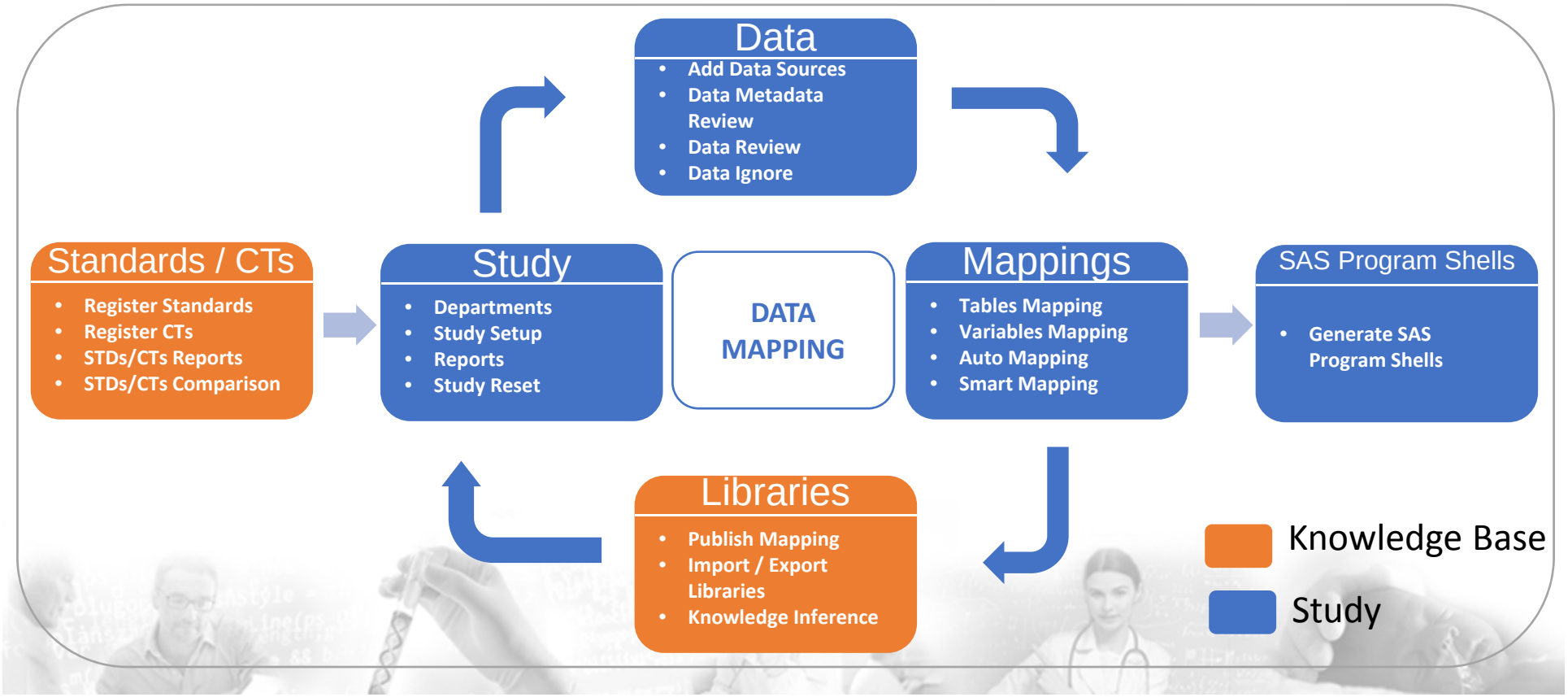
Data Mapping Solution



Data Mapping

- **Libraries module** - Collection of Mapping Rules in Central Database as Metadata
- User-Interface to **collect Data & Standards Metadata** and allow user to define rules for mapping
- Provide **ability to Re-Use Mapping rules** for Future Studies
- **Auto Mapping** (Targeting Destination) - One-to-One mapping Rule between Source Data and its variable, to Destination Data and its variable
  - e.g. adverse.patid ➔ AE.USUBJID
- **Smart Mapping** (Targeting Source) - Similar Variables mapped in past provide guidelines to map new variables
  - e.g. adverse.ptid ➔ AE.USUBJID
  - Suggested Mapping based on adverse.Patid➔AE.USUBJID
- Provide ability to **generate SAS programs**

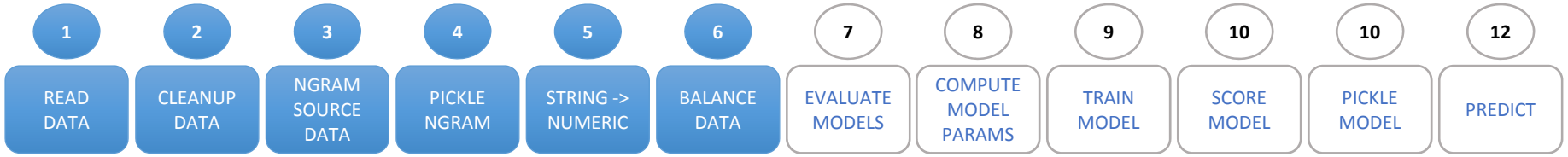
Process Flow



Target Mapping Types

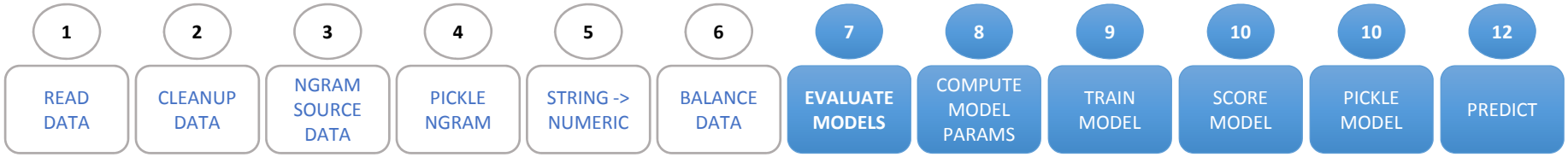
MAPPING TYPE	DESCRIPTION	SOURCE			DESTINATION		
TABLES	1-1 MATCH (Top Similarity Value)	DATASET			DOMAIN		
VARIABLE (AUTOMAP)	1-1 MATCH (Top Similarity Value)	DATASET	VARIABLE		DOMAIN	VARIABLE	
VARIABLE (SMART MAP)	1-3 MATCH (>0.25 Similarity)	DATASET	VARIABLE		DOMAIN	VARIABLE	
CONTROLLED TERMS	1-1 MATCH (Top Similarity Value)	DATASET	VARIABLE	VALUE	CONTROLLED TERM		VALUE

Design Flow



STEP	DESCRIPTION	MODULES	PROCESS
1	Read Data	Pandas	Read data into Dataframe
2	Cleanup Data	Pandas Dataframe	Drop NaN Records, convert to lower case, combine columns, Filter categories based on MIN count
3	NGRAM Data	NGRAM	NGRAM data using Character ngram with limit of 2
4	Pickle NGRAM	Pickle	Ngram pair of (Source,Destination) value and pickle it
5	String to Numeric Value conversion	TfidfVectorizer, MinMaxScaler	Generate Dictionary with TF-IDF values and re-scale it using MinMaxScaler
6	Balance Data	SMOTETomek	Balance data to handle under-sampling and over-sampling

Design Flow (cont'd)



STEP	DESCRIPTION	MODULES	PROCESS
7	Evaluate Models	sklearn	LogisticRegression, DecisionTreeClassifier, RandomForestClassifier, GaussianNB, MultinomialNB, OneVsRestClassifier (LinearSVC), OneVsRestClassifier (SGDClassifier)
8	Compute Model Params	sklearn	GridSearchCV
9	Train Model	sklearn	Fit model
10	Score Model	sklearn.metrics	Classification_report – Precision, Recall, F1 Score
11	Pickle Model	Pickle	Save the trained model in pickle file
12	Predict	sklearn	Predict based on trained model

Tables Mapping: Data Sample

- Word_ngram created for all counts of dest_var
- Minimum count of 3 is required for Model class categories

dsname	dsname_to
adverse	AE
ae	AE
AE	AE
conmeds	CM
adverse_event	AE
cm1	CM
Concomittant_meds	CM
ecg	EG
Electrocardiogram	EG
EG	EG
Chemsitry	LB
Hematology	LB
Preg	LB

source_var	dest_var	Model_ngram	Word_ngram
adverse	AE	ad dv ve er rs se	(ae,adverse)
ae	AE	ae	(ae,ae)
AE	AE	ae	(ae,ae)
conmeds	CM	co on nm me ed ds	(cm,conmeds)
adverse_event	AE	ad dv ve er rs se e_ _e ev ve en nt	(ae,adverse_event)
cm1	CM	cm m1	(cm,cm1)
Concomittant_meds	CM	co on nc co om mi it tt ta an nt t_ _m me ed ds	(cm,concomittant_meds)
ecg	EG	ec cg	(eg,ecg)
Electrocardiogram	EG	el le ec ct tr ro oc ca ar rd di io og gr ra am	(eg,electrocardiogram)
EG	EG	eg	(eg,eg)
Chemsitry	LB	ch he em mi is st tr ry	(lb,chemistry)
Hematology	LB	he em ma at to ol lo og gy	(lb,hematology)
Preg	LB	pr re eg	(lb,preg)

Tables Mapping: Top Model Scores

Model	Definition	Score	Precision	Recall	F1-Score	Support
NB	CalibratedClassifierCV(GaussianNB())	0.772727272727	0.81	0.77	0.77	22
MNB	CalibratedClassifierCV(MultinomialNB(alpha=0.5, fit_prior=True, class_prior=None))	0.727272727273	0.75	0.73	0.72	22
LR	CalibratedClassifierCV(LogisticRegression(C=1))	0.727272727273	0.72	0.73	0.70	22
OVR_SVC	CalibratedClassifierCV(OneVsRestClassifier(LinearSVC(random_state= RANDOM_STATE)))	0.883333333333	0.89	0.88	0.88	22

Classification Report for OVR_SVC

	precision	recall	f1-score	support
ae	1.00	1.00	1.00	3
cm	1.00	0.67	0.80	3
dm	1.00	1.00	1.00	1
ds	0.00	0.00	0.00	1
eg	1.00	1.00	1.00	1
ex	1.00	1.00	1.00	1
fa	1.00	1.00	1.00	1
ho	0.00	0.00	0.00	1
ie	0.25	1.00	0.40	1
lb	1.00	1.00	1.00	3
mh	1.00	1.00	1.00	1
pe	1.00	1.00	1.00	4
vs	1.00	1.00	1.00	1
avg / total	0.88	0.86	0.85	22

Tables Mapping: Model Test Results

- Test Data

```
# Define TEST Data and Expected Outcomes
X_test = np.array(['adverse2', 'conmed', 'chemistry', 'follow', 'electrocardiac', 'inclusion'])
y_test = np.array(['AE', 'CM', 'LB', 'FA', 'EG', 'IE'])
```

- Test Result

```
Predictions: ['ae' 'cm' 'lb' 'fa' 'eg' 'ie'] Expected: ['AE', 'CM', 'LB', 'FA', 'EG', 'IE']
             Model_Matched_Term  Model_Similarity NGram_Matched_Term  NGram_Similarity
Search_Term
adverse2                     ae          0.717276                 ae          0.583333
chemistry                    lb          0.515414                 lb          1.000000
conmed                       cm          0.703553                 cm          1.000000
electrocardiac               eg          0.699650                 eg          0.521739
follow                       fa          0.683542                 fa          1.000000
inclusion                    ie          0.537428                 ie          1.000000
```

- Use combination of Model + NGRAM Similarity to predict output values

Variables Mapping: Data Sample

- Word_ngram created for all counts of dest_var
- Minimum count of 3 is required for Model class categories

dsname	varname	dsname_to	varname_to
adverse	SUBJID	AE	USUBJID
aedata	PATID	AE	USUBJID
adverse	startdt	AE	AESTDTC
aedata	stdt	AE	AESTDTC
adverse	relation	AE	AEREL
ae_info	relinfo	AE	AEREL
hema	ptid	LB	USUBJID
chem	ptid	LB	USUBJID
conmed	patid	CM	USUBJID
cmmeds	patient	CM	USUBJID

source_var	dest_var	model_ngram	word_ngram
adverse.subjid	ae.usubjid	ad dv ve er rs se e. .s su ub bj ji id	(ae.usubjid,adverse.subjid)
aedata.patid	ae.usubjid	ae ed da at ta a. .p pa at ti id	(ae.usubjid,aedata.patid)
adverse.startdt	ae.aestdtc	ad dv ve er rs se e. .s st ta ar rt td dt	(ae.aestdtc,adverse.startdt)
aedata.stdt	ae.aestdtc	ae ed da at ta a. .s st td dt	(ae.aestdtc,aedata.stdt)
adverse.relation	ae.aerel	ad dv ve er rs se e. .r re el la at ti io on	(ae.aerel,adverse.relation)
ae_info.relinfo	ae.aerel	ae e_ _i in nf fo o. .r re el li in nf fo	(ae.aerel,ae_info.relinfo)
hema.ptid	lb.usubjid	he em ma a. .p pt ti id	(lb.usubjid,hema.ptid)
chem.ptid	lb.usubjid	ch he em m. .p pt ti id	(lb.usubjid,chem.ptid)
conmed.patid	cm.usubjid	co on nm me ed d. .p pa at ti id	(cm.usubjid,conmed.patid)
cmmeds.patient	cm.usubjid	cm mm me ed ds s. .p pa at ti ie en nt	(cm.usubjid,cmmeds.patient)

Variables Mapping: Top Model Scores

Model	Definition	Score	Precision	Recall	F1-Score	Support
NB	CalibratedClassifierCV(GaussianNB())	0.8	0.81	0.80	0.78	60
MNB	CalibratedClassifierCV(MultinomialNB(alpha=0.5, fit_prior=True, class_prior=None))	0.666666666667	0.75	0.67	0.67	60
LR	CalibratedClassifierCV(LogisticRegression(C=1))	0.83	0.85	0.79	0.83	60
OVR_SVC	CalibratedClassifierCV(OneVsRestClassifier(LinearSVC(random_state=RANDOM_STATE)))	0.85	0.89	0.85	0.86	60

Classification Report for OVR_SVC

	precision	recall	f1-score	support
ae.aerel	1.00	1.00	1.00	3
ae.aeser	1.00	1.00	1.00	2
ae.aestdtc	1.00	1.00	1.00	3
ae.aetert	1.00	1.00	1.00	2
ae.usubjid	1.00	1.00	1.00	2
cm.cmdosu	1.00	1.00	1.00	2
cm.cmroute	1.00	0.60	0.75	5
cm.usubjid	1.00	1.00	1.00	4
dm.age	1.00	1.00	1.00	3
dm.arm	1.00	1.00	1.00	2
dm.dmdtc	1.00	1.00	1.00	2
dm.height	0.00	0.00	0.00	2
dm.investigator	1.00	1.00	1.00	2
dm.race	1.00	1.00	1.00	2
dm.sex	1.00	1.00	1.00	1
dm.siteid	0.67	1.00	0.80	2
dm.subjid	0.67	1.00	0.80	2
dm.usubjid	1.00	0.33	0.50	3
dm.weight	0.00	0.00	0.00	3
ex.dose	1.00	1.00	1.00	5
ex.freq	1.00	1.00	1.00	2
ex.route	0.00	0.00	0.00	0
ex.usubjid	1.00	1.00	1.00	1
ex.visitnum	1.00	1.00	1.00	2
lb.usubjid	1.00	1.00	1.00	3
avg / total	0.89	0.85	0.86	60

Variables (auto) Mapping: Model Test Results

- Top Match with highest similarity
- Test Data

```
# Define TEST Data and Expected Outcomes
X_test = np.array(['adverse.ptid', 'hema.subjid', 'conmed.patid', 'dm.birthdate','exposure.dosage'])
y_test = np.array(['AE.USUBJID', 'LB.USUBJID', 'CM.USUBJID','DM.AGE', 'EX.DOSE'])
```

- Test Result

Search_Term	Model_Matched_Term	Model_Similarity	NGram_Matched_Term	NGram_Similarity
adverse.ptid	ae.usubjid	0.412617	ae.usubjid	1.000000
conmed.patid	cm.usubjid	0.699372	cm.usubjid	1.000000
dm.birthdate	dm.age	0.682097	dm.age	0.578947
exposure.dosage	ex.dose	0.684286	ex.dose	0.684211
hema.subjid	lb.usubjid	0.474403	lb.usubjid	1.000000

- Use combination of Model + NGRAM Similarity to predict output values

Variables (Smart) Mapping: Model Test Results

- If Similarity >0.9, show 1 match , If Similarity <0.9, show Top 3 match
- Test Data

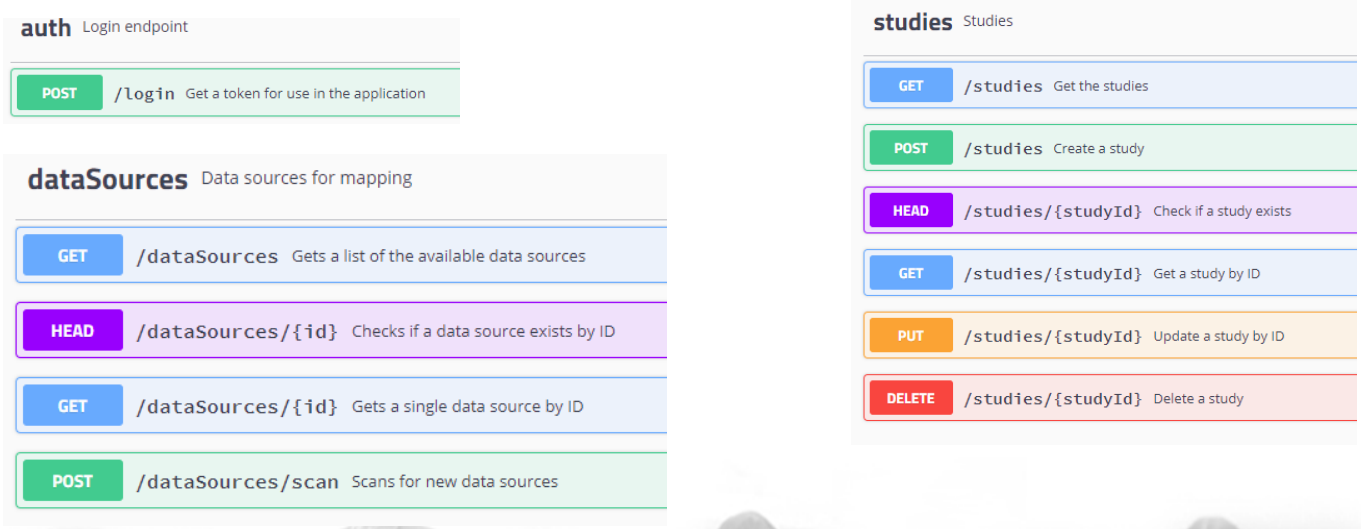
```
# Define TEST Data and Expected Outcomes
X_test = np.array(['adverse.ptid', 'hema.subjid', 'conmed.patid', 'dm.birthdate','exposure.dosage'])
y_test = np.array(['AE.USUBJID', 'LB.USUBJID', 'CM.USUBJID','DM.AGE', 'EX.DOSE'])
```

- Test Result

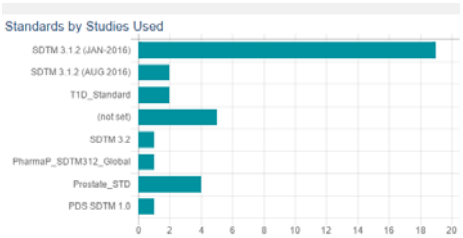
```
Predictions: ['ae.usubjid' 'lb.usubjid' 'cm.usubjid' 'dm.age' 'ex.dose'] Expected: ['AE.USUBJID', 'LB.USUBJID', 'CM.USUBJID', 'DM.AGE', 'EX.DOSE']
       Search_Term Model_Matched_Term  Model_Similarity NGram_Matched_Term  NGram_Similarity
0     adverse.ptid         ae.usubjid          0.412617         ae.usubjid          1.000000
1      hema.subjid         lb.usubjid          0.474403         lb.usubjid          1.000000
2     conmed.patid         cm.usubjid          0.699372         cm.usubjid          1.000000
3     dm.birthdate             dm.age          0.682097             dm.age          0.578947
4     dm.birthdate         dm.usubjid          0.022968             dm.age          0.444444
5     dm.birthdate          dm.subjid          0.021280         dm.brthdtc          0.444444
6  exposure.dosage            ex.dose          0.684286            ex.dose          0.684211
7  exposure.dosage         dm.usubjid          0.022682            ex.dose          0.684211
8  exposure.dosage          dm.subjid          0.020918           ex.route          0.434783
```

- Use combination of Model + NGRAM Similarity to predict output values

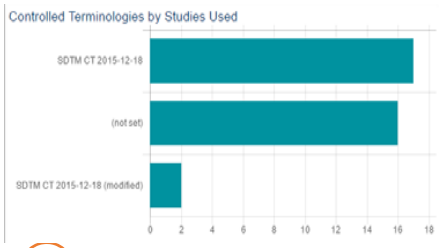
REST API: Ability to integrate with 3rd party Metadata Repositories



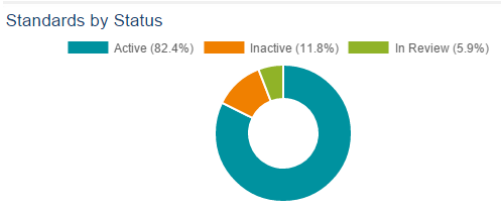
Reports/Metrics



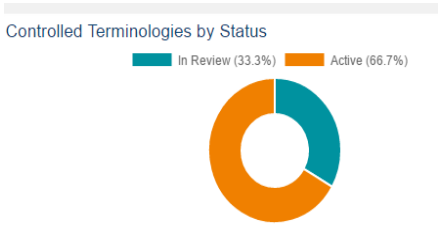
1 Standards By Studies used



2 CT By Studies used



3 Standards by Status



4 CTs By Status

Data Mapping

Thank you !

Sandeep.Juneja@sas.com

Nathan.Asselstine@sas.com

Stijn.rogiers@sas.com

https://www.linkedin.com/in/stijnrogiers/

Twitter @StijnRogiers

https://www.sas.com/en_us/company-information/innovation.html