Data Mapping using Machine Learning

Daniel Choi, Eric Bolender, Stijn Rogiers
Daniel.Choi@sas.com Eric.Boledner@sas.com Stijn.Rogiers@sas.com

Introduction

Mapping raw data to standards is one of the most challenging processes in the healthcare industry. Reusing or reapplying the information collected during mapping processes from previously mapped studies and building upon that knowledge inference is the most important part of the mapping process. Most companies struggle with building knowledge inferences and reapplying them through the data process efficiently. In addition, sometimes, there are multiple tools/programs required to go through the full cycle of data mapping. Therefore, it is difficult for the standard user to know all the different versions and tools, and use them correctly throughout the data mapping process.

The SAS Data Mapping Tool is a web-based tool that provides a user-friendly interface for everything from mapping raw data to generating SDTM standards (including domain templates). Simple User Interface (UI) and click-away concept design provides access to all the required information on a single screen. Auto-mapping and smart-mapping features in the SAS Data Mapping Tool, which are based on knowledge inference derived from machine learning algorithms, reduce time and effort for the user. This leads to improvements in quality, efficiency, and consistency.

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
4. No easy way to publish mapping metadata into different libraries.

Solution

To solve this problem, we have created an application that follows the flow shown below:

Components

Standards/CTs
- Define Standards and Controlled Terminologies. (ex SDTM, ADaM or company specific)
Study
- Define metadata such as Standards/CTs to be used.
Data
- Define Data Sources.
Mappings
- Provides ability to map source to destination data through these steps: data transform, table/variable mapping, and table/variable transform. This is where auto/smart mapping (Machine Learning) will be applied.
Mapping Collections
- Ability to publish mapping metadata into different libraries.
Output
- Generate SAS Datasets/Programs

Machine Learning Design Flow

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Modules</th>
<th>Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Read Data</td>
<td>Pandas</td>
<td>Read data into dataframe</td>
</tr>
<tr>
<td>2</td>
<td>Cleanup Data</td>
<td>Pandas</td>
<td>Drop NaN Records, convert to lower case, combine columns, filter categories based on MIN count</td>
</tr>
<tr>
<td>3</td>
<td>NGRAM Data</td>
<td>NGRAM</td>
<td>NGRAM data using Character ngram with limit of 2</td>
</tr>
<tr>
<td>4</td>
<td>Pickle NGRAM</td>
<td>Pickle</td>
<td>Ngram pair of (Source,Destination) value and pickle it</td>
</tr>
<tr>
<td>5</td>
<td>String to Numeric Value conversion</td>
<td>TfidfVectorizer, MinMaxScaler</td>
<td>Generate Dictionary with TF-IDF values and re-scale it using MinMaxScaler</td>
</tr>
<tr>
<td>6</td>
<td>Balance Data</td>
<td>SMOTETomek</td>
<td>Balance data to handle under-sampling and over-sampling</td>
</tr>
<tr>
<td>7</td>
<td>Evaluate Models</td>
<td>sklearn</td>
<td>LogisticRegression, DecisionTreeClassifier, RandomForestClassifier, GaussianNB, MultinomialNB, OneVsRestClassifier (LinearSVC), OneVsRestClassifier (SGDClassifier)</td>
</tr>
<tr>
<td>8</td>
<td>Compute Model Params</td>
<td>GridSearchCV</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Train Model</td>
<td>sklearn</td>
<td>Fit model</td>
</tr>
<tr>
<td>10</td>
<td>Score Model</td>
<td>sklearn.metrics</td>
<td>Classification_report – Precision, Recall, F1 Score</td>
</tr>
<tr>
<td>11</td>
<td>Pickle Model</td>
<td>Pickle</td>
<td>Save the trained model in pickle file</td>
</tr>
<tr>
<td>12</td>
<td>Predict</td>
<td>sklearn</td>
<td>Predict based on trained model</td>
</tr>
</tbody>
</table>

Variables Mapping Sample and Scores

After word_ngram created for all dest_var

Results From Machine Learning

If similarity > 0.9, show 1 match, else if similarity < 0.9, show Top 3 match

Use combination of Model + NGRAM similarity to predict output values.