Project Abstract:

An ontology is “a formal, explicit specification of shared conceptualization”, where conceptualisation is an abstract model of some phenomenon in the world. Ontologies were created to facilitate the sharing of knowledge and its reuse. In the areas of Semantic Web and data integration, ontology matching is one of the important steps to resolve semantic heterogeneity. Manual ontology matching is very labour intensive, time-consuming and prone to errors. So development of automatic or semi-automatic ontology matching methods and tools is quite important. This project applies machine learning with different similarity measures between ontology elements as features for ontology matching. An approach to combine string-based, language-based and structure based similarity measures with machine learning techniques is proposed. Logistic Regression, Random Forest classifier and Gradient Boosting are used as machine learning methods. The approach is evaluated on two datasets of Ontology Alignment Evaluation Initiative (OAEI).

Introduction:

Typically an ontology matching technique is a combination of much different type of matchers operating at various abstraction levels such as structure, semantic, syntax, instance etc. An ontology matching technique which employs matchers at all possible abstraction levels is expected to give, in general, best results in terms of precision, recall and F-measure due to improvement in matching opportunities and if we discount efficiency issues which may improve with better computing resources such as parallel processing. Ontology matching is a solution to the problem of semantic heterogeneity in the integration and sharing of information. It consists in establishing mappings between entities which semantically belong to different ontologies.

Methodology-

The input data is a list of ontology pairs and the true alignment between them. A model and its parameters are also selected. The names of objects, the names of parents and full paths are retrieved, and the similarity measures are calculated. First, a dataset is created for the classes, then for properties, and after that the datasets are combined.

The input is a true alignment, two lists of entities and the type of input entities. Each entity from the first list is mapped to each entity from the second list. Then, if a pair of entities is contained in the true alignment, then the pair is assigned label “1”, otherwise - label “0”. Also, each pair indicates the type of entity (either “Class” or “Property”) because the same model was used to map classes and properties. Then all pairs are combined into one dataset. Further, the model is trained on the created dataset with either logistic regression or random forest with the selected parameters. We will test the model with the test datasets we have created.

Implementation-

The Dataset we have used is a set of ontologies about Bibliographic references from Benchmark test library. Ontology #101 is the reference ontology. Other ontologies (#102-#103, #301- #304) are compared with the reference ontology.

Dataset 1

It is a set of ontologies about Bibliographic references from Benchmark test library. Ontology #101 is the reference ontology. Other ontologies (#102-#103, #301- #304) are compared with the reference ontology. Dataset has 7 ontologies and 6 true alignments: 3 alignments for training and 3 alignments for testing.

ONTOLOGIES:

* 101.rdf
* 102.rdf
* 103.rdf
* 301.rdf
* 302.rdf
* 303.rdf
* 304.rdf

TRAIN ALIGNMENTS:

* 101-301.rdf
* 101-102.rdf
* 101-103.rdf

TEST ALIGNMENTS:

* 101-302.rdf
* 101-303.rdf
* 101-304.rdf

Dataset 2

It consists several ontologies from Benchmark test library and all ontologies from Conference track of OAEI. Conference track contains 16 ontologies, which dealing with conference organization, and 21 true alignments. Dataset has 27 ontologies and 26 alignments: 8 alignments for training and 18 alignments for testing. The pairs of entities from each pair of ontologies and their alignments are extracted.

ONTOLOGIES:

* confof.owl
* edas.owl
* conference.owl
* myreview.owl
* sigkdd.owl
* linklings.owl
* pcs.owl
* paperdyne.owl
* openconf.owl
* 205.rdf
* cmt.owl
* 204.rdf
* iasted.owl
* 238.rdf
* crs\_dr.owl
* micro.owl
* ekaw.owl
* confious.owl
* 301.rdf
* 101.rdf
* cocus.owl
* 304.rdf

TRAIN\_ALIGNMENTS:

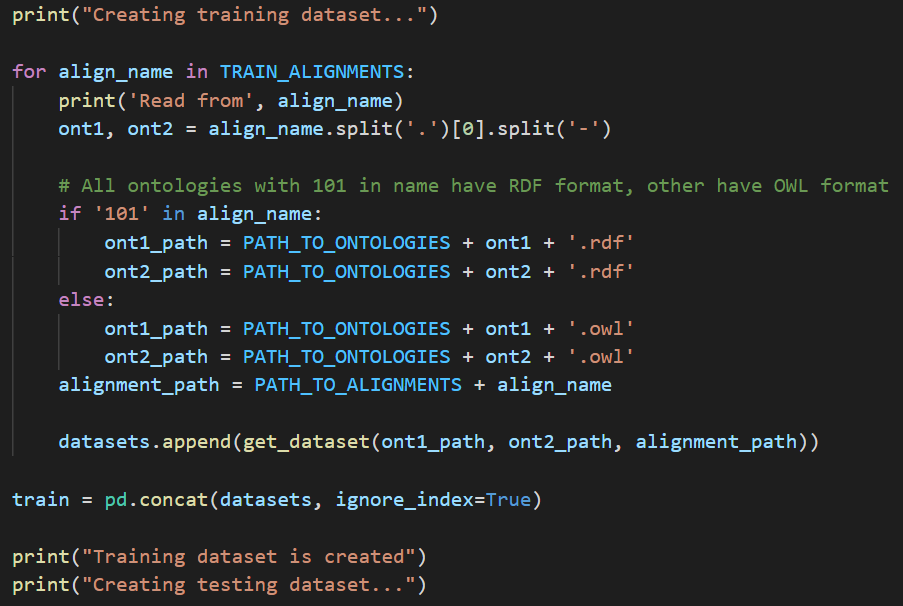
* cmt-conference.rdf
* conference-iasted.rdf
* edas-ekaw.rdf
* 101-204.rdf
* 101-205.rdf
* 101-238.rdf
* 101-301.rdf
* 101-304.rdf

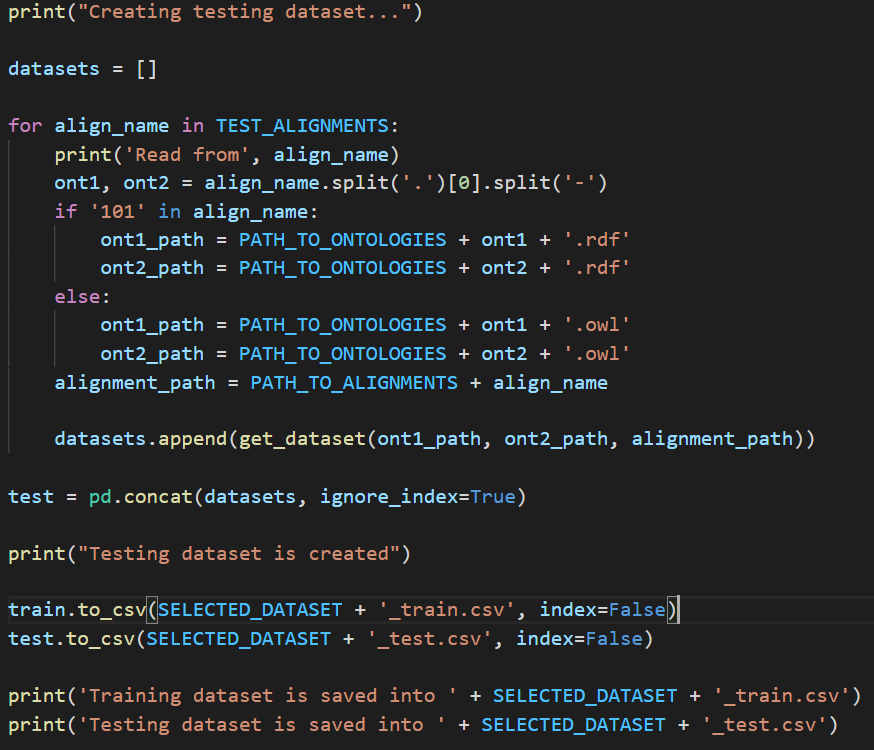
TEST\_ALIGNMENTS:

* conference-edas.rdf
* cmt-sigkdd.rdf
* edas-sigkdd.rdf
* ekaw-sigkdd.rdf
* cmt-edas.rdf
* conference-sigkdd.rdf
* confof-edas.rdf
* confof-iasted.rdf
* conference-confof.rdf
* cmt-confof.rdf
* conference-ekaw.rdf
* cmt-ekaw.rdf
* confof-ekaw.rdf
* iasted-sigkdd.rdf
* cmt-iasted.rdf
* edas-iasted.rdf
* ekaw-iasted.rdf
* confof-sigkdd.rdf

The approach was implemented using Python 3.9. This language is widely used for implementation of machine learning workflows and possesses a lot of useful program libraries.

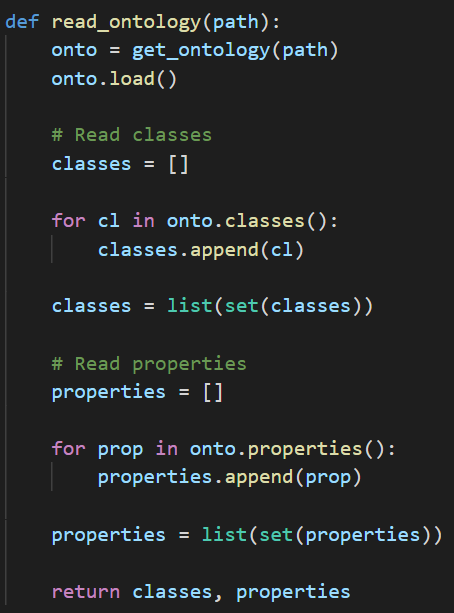
Ontologies are represented as RDF/OWL files. The owlready2 library was used for syntactic parsing of ontologies. Alignments are defined in RDF format. For parsing alignments, the BeautifulSoup library was used. As implementation of logistic regression machine learning techniques sklearn library is used. Dataset is formed as a dataframe of the pandas library. To evaluate F-measure, the Alignment API library was used. Computation experiments: training of machine learning models and the selection of their parameters were performed at the Hybrid high-performance computing cluster. WordNet dictionary is taken from the nltk library. Word2vec model was trained on GoogleNews news. N-gram implementation is taken from the ngram library. Similarity measures based on edit distance implementation is taken from the editdistance library.

Creating Train and Test datasets:  


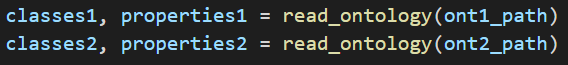


**Python functions and classes used**

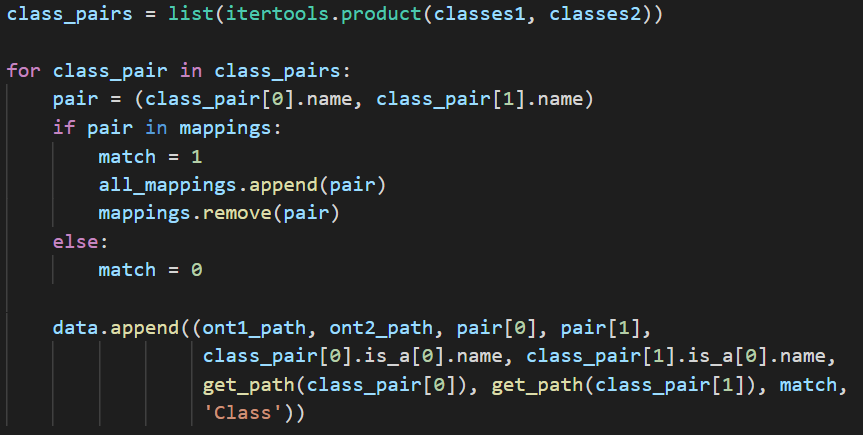
**For example:**



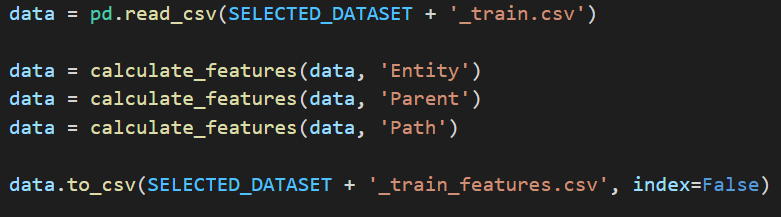
Parse ontologies:



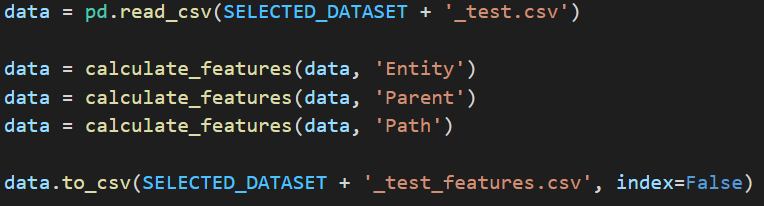
Generate pairs of classes:



Calculate features for training dataset:



Calculate features for testing dataset:



The best parameters for the models were selected by the brute force method (a grid of values was created for each parameter): the models were trained on all combinations of parameters and the model with the best F-measure value using threshold was selected.

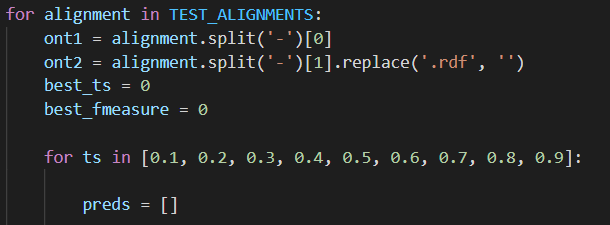
For logistic regression, the following parameters were selected: inverse of regularization strength, weights of classes and norm used in the penalization.

For random forest the number of trees in the forest, the maximum depth of the tree, the number of features to consider when looking for the best split and class weights were selected.

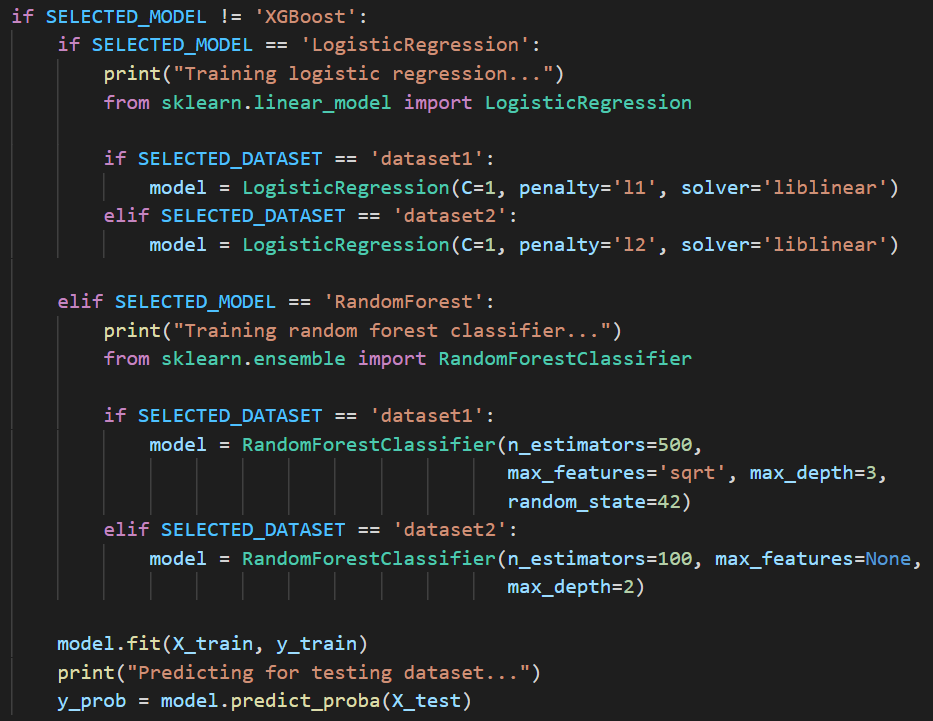
After training and searching for the best parameters, a threshold was selected with the highest F-measure value for each alignment.

The values of F-measure for each alignment will be presented.

Choose best threshold:

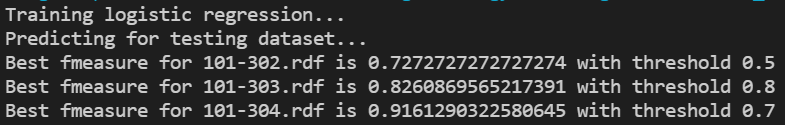


Training Model:

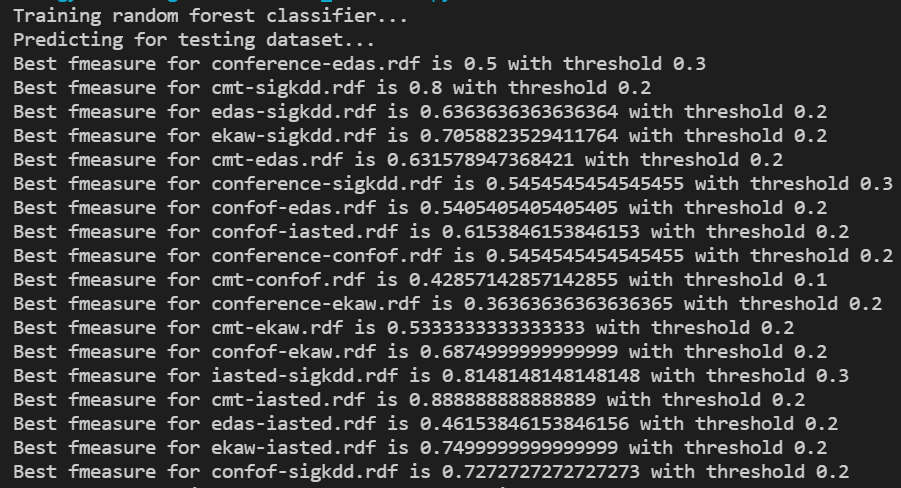


Evaluated results:

Logistic regression-



Random forest-



Conclusion:

We combined string-based, language-based, and structural-based similarity measures using different machine learning models and apply them for ontology matching problem. The approach is implemented and evaluated using datasets selected from Ontology Alignment Evaluation Initiative (OAEI). Due to the large number of similarity measures, there is hope that there is a potential for a more universal use of the approach. Universality refers to the applicability of the different subject areas. It is necessary to test the approach on ontologies with other subject areas. After training and searching for the best parameters, a threshold will be selected with the highest F-measure value for each alignment. The values of F-measure for each alignment will be presented.