International Journal of Engineering Inventions
e-ISSN: 2278-7461, p-ISSN: 2319-6491
Volume 13, Issue 1 [January. 2024] PP: 164-174

A Generic Approach to Entity Resolution Mechanisms for Big Data on Real World Match Problems in the Global Oil and Gas Sector

Abdulhafiz Sabo[1] and Jamilu Usman Waziri[2]
[1] *Mathematical Science Department, Bauchi State University Gadau, Bauchi State, Nigeria*
[2] *Mathematical Science Department, Bauchi State University Gadau, Bauchi State, Nigeria*

ABSTRACT: *Complex challenges are facing the global oil and gas industry. Oil prices are dropping due to OPEC production level, US oil boom, and other factors. Many experts believe that prices of oil will remain low for years at equilibrium of around $40-50 (Blumberg, 2018; Walls and Zheng 2018; Azar, 2019). Although 2019 oil price is expected to average at $65 with a further decline at $62 by 2020 (Amadeo, 2019; Kasim, 2019). Also, newly commercial resources are extremely expensive to develop, as massive capital investments are required. This research intends to develop a comprehensive entity resolution framework that has the ability to search across multiple databases with disparate forms, tame large amounts of data very quickly, efficiently resolving multiple entities into one, as well as finding hidden connections without human intervention. Putting in place a system to manage these entities will not only help to better assign resources, but to do so in a more expedient fashion. Although the necessary information is mostly already available within the oil and gas companies, it is spread around different company areas and application. Entity resolution will helps to aggregate these data, identify and exploit connection between entities and offer holistic all-in-one information that can helps to identify and deal with potential risk. We therefore present such an evaluation of existing implementations on challenging real-world match tasks. We consider approaches both with and without using machine learning to find suitable parameterization and combination of similarity functions. In addition to approaches from the research community we also consider a state-of-the-art commercial entity resolution implementation. Our results indicate significant quality and efficiency differences between different approaches. We also find that some challenging resolution tasks such as matching product entities from Opec database are not sufficiently solved with conventional approaches based on the similarity of attribute values.*

Keywords: *Entity resolution, Big Data, framework, Machine learning*

Date of Submission: 10-01-2024 Date of acceptance: 24-01-2024

I. Introduction

Entity resolution (also referred to as object matching, duplicate identification, record linkage, or reference conciliation) is a crucial task for data integration and data cleaning. It is the task of identifying entities referring to the same real-world entity. The high importance and difficulty of the entity resolution problem has triggered a huge amount of research on different variations of the problem and numerous approaches have been proposed especially for structured data. Recent surveys include (Reynolds, 2018; and Zycher, 2018). But due to the high number and diversity of different entity resolution approaches we see a strong need for comparative evaluations of different schemes. To date most entity resolution approaches have been evaluated individually using diverse methodologies, configurations, and test problems making it difficult to assess the overall quality of each approach, let alone their comparative effectiveness and efficiency.

Therefore, the emergence of alternative energy sources is negating old assumptions, and competition for scarce human capital is intense (Reynolds, 2018; Zycher, 2018). Compounding the problem is a lack of operational insight caused by siloed data from disparate sources, and a lack of standardization. Systems require better connectivity, monitoring and control, and process automation (Tafamel 2015; Gervorkyan and Semmler, 2017). Clearly, challenging times for the oil and gas industry have emerged.

To make up for the lost revenue, oil companies need to setup, maintain, and operate in the most cost-effective way possible. There is need to have a structured operating system that will ensure personal safety (enforcing management of change), Environmental safety (Regulatory compliance), and Investment Safety (Sophisticated maintenance planning and correct operational procedures) (Barkhatov and Baranova, 2017). Without an accurate 360-degree view in real time, it's hard to know which fires to put out first (Martin, 2018).

1 Correspondent Author: jamiluuwaziri@basug.edu.ng

One way to solve these problems is by capturing real-time accurate performance and maintenance data at the point of occurrence: platform, rig or field and put this information into the hands of decision makers immediately while still relevant (Ayat, et al., 4014; Gervorkyan and Semmler, 2017).

Figure 2. Entity Resolution Process

Given the amount of data that oil and gas industry manage, they must rely on system that has the ability to obtain data at the point of occurrence, search across multiple databases with disparate forms residing in different locations and can tame large amounts of data very quickly. Developing a system with these possibilities is the key aim of this research.

1.1 Statement of the Problem

Recently, there has been a rapid shift from paper-based documentation to electronic records. This is quite prevalent with:

- Duplicate and incorrect record due to non-existence of sufficient methods for validation or verification during entry processes.
- High possibility that entry will appear in several forms and data input or other types of error will likely occur particularly when integrating large volume of data from various data source.

1.2 Research Justification and Impact

When we hear innovation in oil and gas, our first thoughts might go to hardware—bigger, faster, deeper drilling; more powerful pumping equipment; and bigger transport—or to the "shale revolution"—unconventional wells, hydraulic fracturing, horizontal drilling, and other enhanced oil recovery (EOR) techniques. But, just like any other industry where optimization is important—and due to large capital investment and high cost of error, it's perhaps even more important in oil and gas than in most other industries— there is a huge benefit perhaps an imperative to add more big data, data science skills especially entity resolution, (co-reference resolution and reference reconciliation) and machine learning, into the industry skills mix. skills that oil companies haven't traditionally and broadly had in-house. These (skillsets) when combined with rapid increases in computer processing power and speed, greater and cheaper storage, and advances in digital imaging and processing, will drive innovation and create a rich and disruptive movement among oil and gas companies and their suppliers in Nigeria.

The truth is, the Nigerian oil and gas industry has been dealing with large amounts of data longer than most, some even calling it the "original big data industry. To put that in context, Nigeria National Petroleum Company (NNPC) has twenty (20) subsidiaries engaged in diverse but interrelated operations producing increasingly enormous data (spatial, seismic, operation and nominal) of high resolution and frequency which are being combined with large amounts of historical data—both digital and physical—to create one of the most complex data science problems out there. We are therefore inundated with more and more data that needs to be integrated, aligned and matched before further utility can be extracted. This research is making a novel attempt to solving it. Although one can argue that the industry is a mature and unique one, built on experience and hard-won knowledge, and employing the best heads. They're very good at what they do, and they've been doing it for a long time. This optimization will either fine-tune existing practices or redefined new course of action. Although these improvements in efficiency and productivity may be subtle however can translate into significant economic difference

1.3 Research Aims and Objectives

The research aims to develop a comprehensive entity resolution framework that will generates significant value from oil and gas data. This will be done by examining a variety of attributes and evaluating different matching strategies. In particular, the use of sophisticated matching algorithm, which is able to fuse multiple entity

attributes for the derivation of a new generation of robust matching technologies that will not only be adjustable to different evaluation criteria but able to use collective approach to resolve data ambiguities. To achieve this aim, the project has the following specific objectives:

Research Objectives:

1. Investigate existing entity resolution algorithms and methods, in terms of computing cost and matching strategies.
2. Explore new types of attributes and behaviour that can be used for entity identification.
3. Derive a comprehensive entity resolution framework based on the new types of identity attributes, optimized for distributed computing environments.

II. Literature Review

It is evidently tedious to deal with the problem of entity duplication. As defined by Nigam and McCallum (2019), entity resolution is a process of semantic resolution that establishes whether a single entity is the same when being described in a different way. It identifies those records in one or multiple datasets that refer to the same real-world entity.

According to Avigdor (2014), entity resolution can be very complex due to the special data characteristics of individuality records. First, Entity resolution, especially in the intelligence and law enforcement communities, often suffers greatly from the missing data problem. Missing values, if present in many fields of a record, can present a big challenge for entity resolution techniques (Kopcke and Rahm, 2016; Ya-Kun and Gao, 2016). Second, entity resolution needs to handle not only duplicates caused by entry errors or data ambiguities but also intentional mistakes and deception, which tend to be hidden and concealed (Benjelloun, et al., 2018). For instance, conventional identity resolution methods depends on basic attributes such as identification numbers, names and date-of-birth to detect matches (Omar and Steven, 2017). Even though, these properties are commonly available in most record management system, they are just a simple description of an individual and vary in terms of reliability, integrity, and availability across different system (Whang et al., 2016), as it is much easier to fake these personal attributes to conceal their traces.

Third, entity resolution techniques may need to be adjustable to different evaluation criteria (Izakian, 2018; Rajkumar *et al.,* 2018). For instance, false positives may be less tolerable than false negatives for entity authentication that grants access to a critical facility. In contrast, a high false positive rate may not be a big concern when searching for records related to a crime suspect with limited information. Therefore, accurate entity resolution requires a careful design that considers the special characteristics of entity records (Getoor and Ashwin, 2014).

Furthermore, automated entity resolution results in a probability, not a certainty that records are the same, and entity features and their values vary (Sunter and Fellegi, 2013; Domingos and Elkan, 2016). Likewise, many challenges remain for developing systems that can handle variation, for specifying the business rules relevant to different business needs, and for handling the exploding amount of data available (Winkler, 2014). Systems currently addresses one or two of the issues stated here but no system have yet provided an approach to entity resolution with sufficient flexibility, adequate speed, and perfect understanding.

Existing resolution techniques can be classified into rule-based and machine learning approaches (Dong *et al.,* 2014). There have been several rule-based entity resolution approaches based on matching rules encoded by domain experts. Such exact-match heuristics tend to have high specificity but low sensitivity in detecting true matches, especially when data quality problems such as missing values, entry errors and deceptions are present (Winkler, 2019). Hence, a good resolution technique must support partial-match as well to reduce false negatives (Giang, 2014; Xing, 2019).

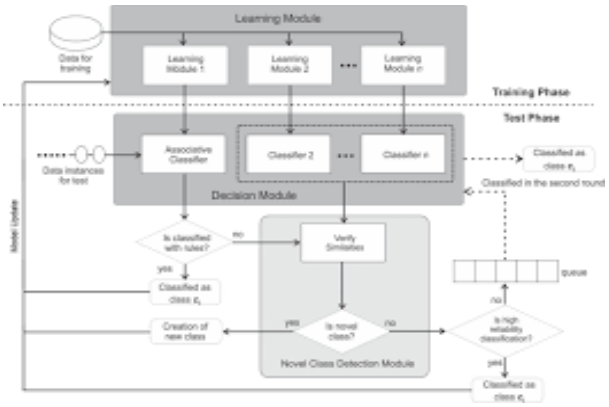
As an alternative to manual rule coding, machine learning can automatically extract patterns from annotated training data with annotated matching pairs and build resolution models for new records (Issa and Vasarhelyi, 2017). Given a pair of records, distance/similarity measures are defined for different descriptive attributes and then combined into an overall score (Koyuturk *et al.,* 2019). If the overall distance (or similarity) score is below (or above) a pre-defined threshold, then the pair should be regarded as a match.

Several recent studies have demonstrated the benefits of utilizing group contextual information in entity resolution (Zhou and Zhou, 2015; Tian, 2016; Zhang et al., 2017). However, most studies lack the guidance of entity theories to construct and examine different types of group attributes for entity resolution (Li Juan, 2018). Entity theories suggest that individual and group identity may complement each other for the purpose of entity resolution (Ripon, 2017; Chuan et al., 2018). Furthermore, existing resolution techniques mainly employ pair-wise comparison when finding matching entity records. Entity resolution studies have shown that resolution accuracy can improve significantly if matching of related entity references is performed in a collective fashion (Han, 2016; Soyemi and Adegboye, 2018). The effectiveness of a collective approach in the context of entity resolution is yet to be examined.

III. Research Methodology

In the research we plan to perform an experimental approach, using researches on latest projects and academic reviews as references for implementing the technical experiment that will accompany our research with the view to Understanding specifics, as well as pertinent general literature on entity resolution will require a solid understanding of matching strategies, and resolution techniques. Therefore, the research project have greatly devoted enough time to the gathering and the studying of the academic and industry specific literatures and recent projects, so that we can have a comprehensive overview of the state of the art in the field of entity resolution. We have investigated existing entity resolution algorithms and methods, in terms of computing cost and matching techniques that is our First Objective. Also we have devoted time to the investigation of new types of quantifiable attributes and behaviour that can be used for entity identification which our Second Objective. And that have form a foundation, for further study and for future study in that field.

The research entailed design and development of the entity resolution framework as the Third Objective of the study. In producing acceptable outputs, the framework uses identified attributes and variables from earlier analysis as input. This framework will be reviewed and improved on during our next research, with a focus on how the framework can be optimized for large distributed computing environments which is a major entity resolution challenge. Furthermore, an advanced review of the developed framework have been carried out to address advances in knowledge and was further improved the original design.



We use the FEVER framework to automatically execute the approaches and to find favorable parameter settings in a comparable way. In particular, we always apply the same blocking method to reduce the search space and use a uniform approach for providing training to the machine-learning approaches. For the approaches not based on machine learning we spend the same effort for optimizing parameters such as similarity thresholds.

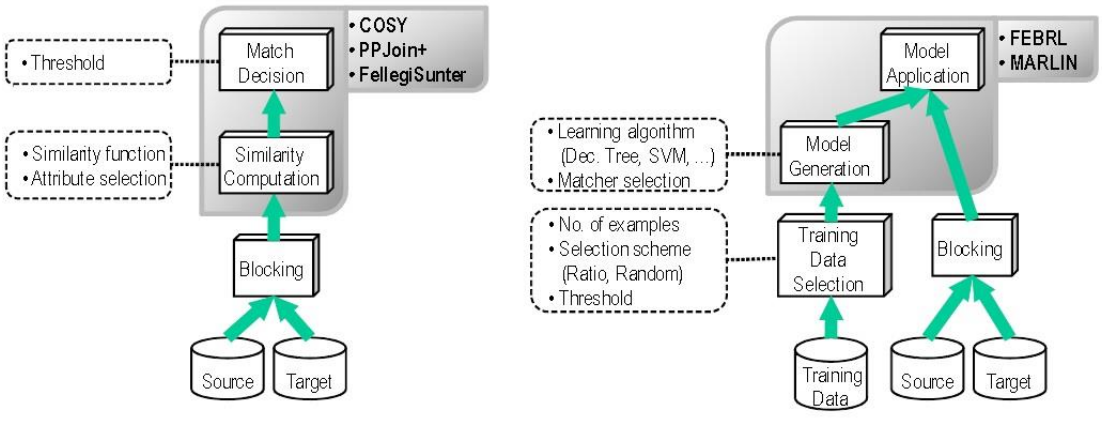


Figure 1. FEVER match workflows for evaluating existing entity resolution approaches

3.1 Evaluation Approach

We use the FEVER platform (Framework for Evaluating Entity Resolution) to evaluate several match approaches for different match tasks. While FEVER has its own library of match algorithms we do not evaluate this functionality here but use FEVER only to evaluate existing entity resolution approaches from the research

community and one vendor. FEVER allows us to automatically execute these algorithms for many different parameter settings in a comparable way as shown below in the following for both non-learning and learning-based match approaches.

3.2 Non-Learning Match Approaches

In FEVER, a match approach is specified by a so-called operator tree or workflow that specifies the sequence of processing steps for determining the match result on two input datasets. Figure 1a illustrates the FEVER operator tree that was applied in our evaluation of non-learning match approaches. For large datasets, it is generally not feasible to exhaustively evaluate the Cartesian product of all input entities. Hence, we first apply a blocking operator to reduce the search space to the most likely matching entity pairs. For comparability, we use a fixed blocking strategy for all non-learning and learning-based match approaches, i.e., blocking is not subject of the evaluation.

The blocking result is input to the non-learning match approaches to be evaluated. In this study all considered match approaches are based on so-called attribute matchers that evaluate the similarity of attribute values based on some similarity function (e.g., an approximate string similarity). The approaches may evaluate only a single matcher (for a specific attribute pair and similarity function) or multiple matchers using different attribute pairs or similarity functions. In the latter case the approaches also need to support a combination of the individual similarities to derive a match decision. In our evaluation, we will always use the same attributes for comparability. Furthermore, all non-learning match approaches apply a threshold-based selection of the matching entity pairs and require the similarity threshold to be provided as a parameter. For the similarity computation and the threshold-based match decision we used the implementation of the following non learning match approaches:

COSY: This is a state-of-the-art commercial system for entity resolution. Unfortunately, license restrictions do not allow us to disclose the name of the system. COSY uses its own similarity function that can be applied on one or several attribute pairs. The most important parameter to be provided is the *overall Minimum Similarity* threshold. An entity pair will be considered a match only if it has a similarity that is greater than or equal to this threshold. Additional *attribute-level similarity thresholds* can optionally be specified for each attribute pair that should be considered in the computation of the entity similarity.

PPJoin+ is a single-attribute match approach (similarity join) using sophisticated filtering techniques for improved efficiency. The approach has two parameters that need to be configured. The parameter *function* determines the similarity function used for the join. We will evaluate both supported implementations for the similarity function (Cosine, Jaccard). The parameter *threshold* determines the threshold for the similarity values above which entities are considered to match.

Table 1. Overview of real-world evaluation match tasks

Source size (#entities)		Mapping size (#correspondences)		
Source 1	Source 2	Full input mapping (Cartesian product)	Reduced input mapping (blocking result)	perfect result
2,616	2,294	6 million	494,000	2,224
2,616	64,263	168.1 million	607,000	5,347
1,363	3,226	4.4 million	342,761	1,300
1,081	1,092	1.2 million	164,072	1,097

FellegiSunter is a non-learning approach from the FEBRL framework [9]. For similarity computation we evaluate three of the similarity measures provided by FEBRL (Winkler, Tokenset, Trigram). The approach has an *lower* and *upper similarity threshold* that can be adjusted. Entity pairs with a similarity above the upper classification threshold are classified as matches, pairs with a combined value below the lower threshold are classified as non-matches, and those entity pairs that have a matching weight between the two classification thresholds are classified as possible matches. For our evaluation, we set the lower threshold equal to the upper threshold as we only want a classification into matching and non-matching entity pairs.

An operator tree typically comprises several operators each having several parameters that need to be specified in order to apply the operator tree to a match problem. FEVER allows a systematic evaluation of operator trees for different parameter settings to help finding a suitable configuration. For this study we limit the number of parameters to be set by applying a fixed blocking approach and manually pre-selecting the attributes to be

evaluated. We further evaluate the existing similarity functions either on one or two attributes of the input datasets. In both cases we have to specify similarity thresholds on the single attribute or combined attribute similarity. For comparability, we evaluate every match approach for a fixed maximum number, *N*, of settings for the threshold parameters. FEVER supports several methods for selecting the parameter values such as manual (user-defined) and random. For this evaluation, we use the sophisticated and effective gradient descent strategy that iteratively refines a parameter setting by considering the quality of previously generated settings.

3.3 Learning-Based Match Approaches

Figure 1b shows the FEVER operator tree applied for the evaluation of learning-based approaches. The execution falls into two phases: model generation and model application. The model generation (left part of the operator tree) requires a training dataset that contains manually labeled correspondences representing matching (similarity value equals 1) and nonmatching (0) entity pairs. The learning algorithm applies the specified matchers to the entity pairs in the training data. The learner then uses the resulting similarity values to automatically determine a match strategy model, i.e., combination and parameterization of the specified matchers to derive a match decision for any entity pair. More details on training selection and model generation will be provided below. The second phase (right part of the operator tree) applies the determined model for the real match task (model application) to match a source and target dataset (or to find duplicates within one dataset).

For model generation, a pre-selected set of matchers is applied to the training data. By comparing similarity values computed by the matchers to the perfect (labeled) match result in the training it is possible to determine (learn) a combination of the most effective matchers and their parameters such as similarity thresholds. In our evaluation we will compare several existing training-based approaches for model generation and application offered by the following frameworks:

- **FEBRL** (Freely Extensible Biomedical Record Linkage) provides a support vector machine (SVM) implementation for learning suitable matcher combinations. For attribute matching we will evaluate the same three similarity measures than for the non-learning matchers studied for FEBRL.
- **MARLIN** (Multiply Adaptive Record Linkage with Induction) offers two string similarity measures (Edit Distance and Cosine) and several learners, specifically SVM and decision trees. The learners can be used in a single step approach or can be employed for a two-level learning approach. For the two-level approach string similarity measures are first trained for every selected attribute so that they can provide accurate estimates of string distance between values for that attribute. Next, a final decision is learned from similarity metrics applied to each of the individual attributes.

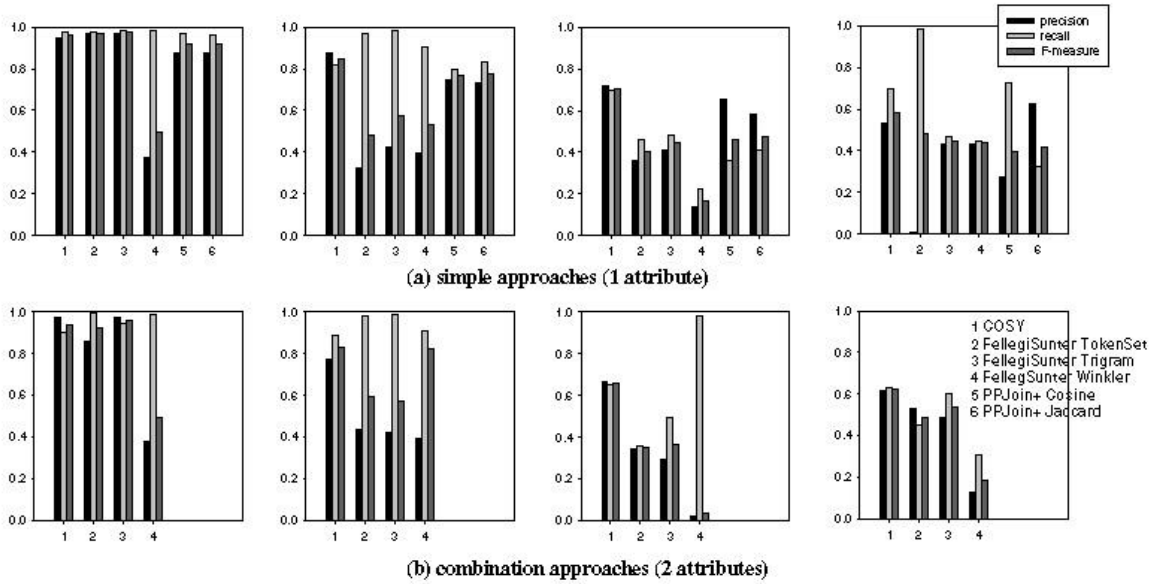


Figure 2: Performance results for non-learning approaches

IV. Results

The effectiveness of machine learning approaches is known to depend on the provision of sufficient, suitable, and balanced training data. On the other hand, the number of entity pairs to be labeled affects the manual tuning effort and should thus be small. To address these issues we build upon our evaluation experiences

reported in (Azar, S. 2019) and only consider entity pairs for labeling for which the similarity exceeds a specified threshold *t*. This ensures that the training is not dominated by trivial non-matching entity pairs that are not useful to find effective matcher parameters and matcher combinations. We further strive at providing both matching and non-matching entity pairs by a training selection approach called *Ratio (r,t)*. It uses a ratio parameter *r* from the range 0 to 0.5 indicating the minimal percentage of both matching and non-matching entity pairs. r=0 corresponds to a random strategy that randomly selects entity pairs with a similarity above the threshold *t*. For r>0 the number of randomly selected entity pairs is reduced so that either the number of matching or nonmatching entity pairs satisfy the ratio restriction. For example,r=0.4 guarantees that at least 40% of all training pairs are either matching or non-matching, i.e., at most 60% are non-matching or matching. By ensuring a minimum number of matching/nonmatching pairs the ratio approach aims at enhancing the discriminative value of the training data for learning effective match strategies. We have extensively evaluated the Ratio training selection approach and found that setting r=0.4 and t=0.4 with TFIDF is a reliable and effective default configuration. Our evaluation for learning-based matching will thus be based on this configuration.

We first present match quality and runtime results separately for non-learning and learning-based approaches. Afterwards we briefly compare the two kinds of matchers with each other. The runtime results are determined for a HP Z400 workstation with 2.66 GHz Intel Quad-Core Processor W3520 and 4GB of RAM running 64-bit Windows 10. The evaluated match approaches are implemented in different languages: PPJoin+ is implemented in C++, MARLIN in Java and FEBRL in Python.

4.1.1 Non-Learning Approaches

Figure 2 shows the match quality (precision, recall, F-measure) results for the four real world match tasks achieved with different non learning approaches. The upper half shows the results for approaches operating on just a single attribute, namely the first attribute listed in Table 1 (publication title for the bibliographic tasks, product name for the e-commerce tasks). The lower half shows the results for approaches combining the similarity for two attributes (the first two attributes listed in Table 1). In both cases we optimized the threshold for the final match decision while all other parameters of the approach were kept constant. Optimization was done with the Gradient Descent approach on a test set of 500 object pairs for each match task. For the FellegiSunter approach from the FEBRL framework we considered three different similarity measures, namely Winkler, TokenSet, and Trigram. FEBRL's FellegiSunter approach sums the logarithms of the single similarities. For the COSY approach it is not clear how similarities are combined.

Table 2 lists the execution times for the considered non-learning approaches for the blocked input as well as the Cartesian product of the considered match tasks. The table shows significant differences between the approaches already for the blocked input. The evaluation of the Cartesian product tests the scalability and leads to huge differences. PPJoin+ and COSY achieved very fast execution times and could even achieve acceptable run times for the Cartesian product. PPJoin+ implements an intelligent pruning of the search space and is uniformly the fastest approach for all match tasks with execution times between less than a second to at most seven seconds. The small increase of at most a factor of 2 for evaluating the Cartesian product proves the excellent scalability of PPJoin+. In this respect it also outperforms COSY that noticeably slows down for the Cartesian product evaluation of Exxon mobile datasets (almost 4 minutes vs. 9 seconds for the blocked input). The considered FEBRL approaches were mostly much slower than COSY and PPJoin+, on the Cartesian products by orders of magnitude. This may be influenced by the Python-based implementation of FEBRL. FellegiSunter using the Winkler similarity turned out to be not only the least effective but also by far the slowest of all non-learning match approaches. On the blocked input, FEBRL with tokenset similarity is almost as fast as COSY.

Table 2: Execution times (in seconds) for non-learning approaches

	blocked	Cartesian	blocked	Cartesian	blocked	Cartesian	blocked	Cartesian
COSY	1	1.6	8.8	224.7	2.7	5.8	6.5	9.6
FellegiSunter TokenSet	2.1	170	10.2	64,057	2.5	17.2	4.6	84
FellegiSunter Trigram	2.5	655	44.7	243,060	25	105	34.1	320
FellegiSunter Winkler	5.7	1,601	164.6	277,200	53.5	364	96.2	1,065
PPJoin+ Cosine	0.4	0.9	3.4	6.9	0.6	2.5	0.5	0.9
PPJoin+ Jaccard	0.4	0.6	3.5	7	0.6	2.5	0.5	0.9
2 attributes								
COSY	35	56	17	434	44	94	28	41
FellegiSunter TokenSet	3	429	17.8	108,896	5.9	43	44	709
FellegiSunter Trigram	3.1	1,512	116	>500,000	58	635	1,940	16,620
FellegiSunter Winkler	7.4	3,602	341	>500,000	135	970	2,833	20,760

4.1.2 Learning-Based Approaches

Figure 3 shows the F-measure results for the four real-world match tasks achieved with different learning-based approaches from FEBRL and MARLIN and different labeling efforts (x-axis). The labeling effort varies between 20 and 500 entity pairs, i.e., we consider only comparatively small training sizes and thus a limited amount of labeling effort. The F-measure results are averaged over 10 runs. All results in Figure 3 refer to matching on the first or the first two attributes listed in Table 1 (types of crude and quantity of crude for the Halliburton tasks, product name and product description for the Chevron tasks) with different similarity functions. Figure 3a shows the results for the SVM learner of FEBRL that was applied for the same three similarity functions (TokenSet, Trigram, and Winkler) as for the non-learning case.

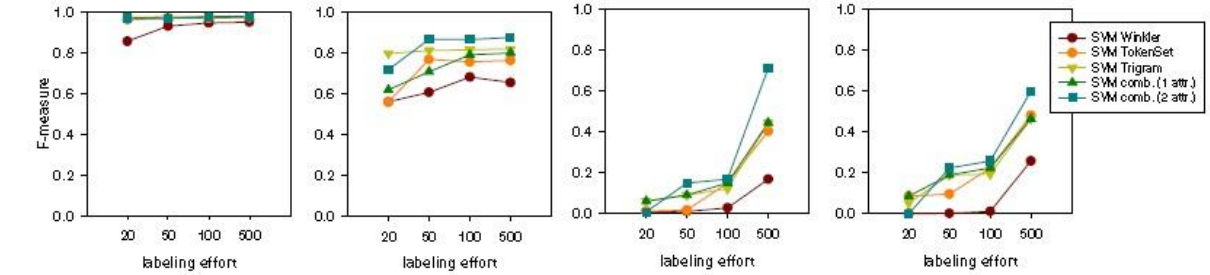
Table 3: Execution times (in seconds) for learning-based approaches

		blocked	Cartesian	blocked	Cartesian	blocked	Cartesian	blocked	Cartesian
FEBRL	SVM TokenSet	3	244	20.0	249,364	8	23	14	124
	SVM Trigram	5	859	79.0	250,920	25	127	46	415
	SVM Winkler	8	2,022	196.5	295,800	62	409	110	1,225
	SVM comb. (1 attr.)	13	2,400	275	>500,000	83	590	154	1,481
	SVM comb. (2 attr.)	99	4,320	482	>500,000	232	1,364	196	36,090
MARLIN	ADTree ED (1)	3	329	76	10,090	22	64	41	161
	ADTree ED (2)	5	582	96	17,427	37	119	57	244
	ADTree Cosine (1)	5	157	71	301	1	89	2	98
	ADTree comb. (1 attr.)	7	951	104	28,476	40	340	95	373
	ADTree comb. (2 attr.)	12	1,553	324	46,501	551	3,456	10,299	41,615
	SVM ED (1)	5	633	117	257,982	28	231	66	333
	SVM ED (2)	7	979	146	445,575	44	192	80	465
	SVM Cosine (1)	4	267	41	7,696	10	143	26	186
	SVM comb. (1 attr.)	9	1,336	157	>600,000	68	552	127	498
	SVM comb. (2 attr.)	20	2,196	375	>900,000	324	3,747	13,768	55,632

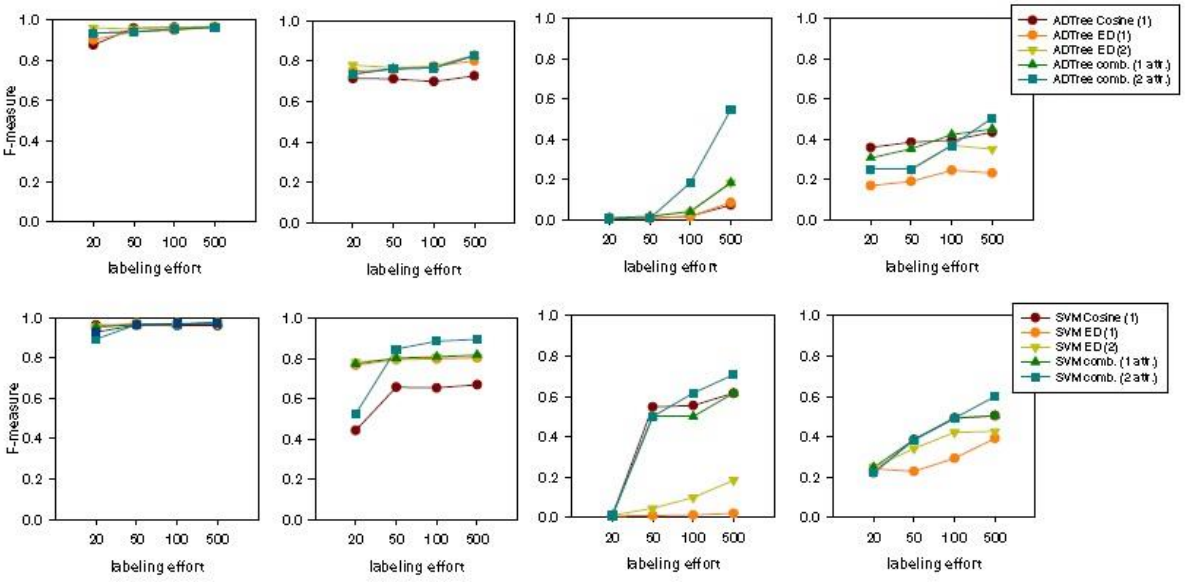
In addition we use the SVM for two combined match strategies using all three similarity measures either on one or on two attributes. Figure 3b shows the results for MARLIN separated by the employed learner, first for MARLIN's decision tree implementation ADTree followed by the SVM results. For both learners we applied the two similarity measures Edit Distance and Cosine. Edit Distance was used in the single-step as well as the two-step learning approach. Cosine was just applied in the single-step approach as it has limitations in the two-step implementation as mentioned by the authors in Kazim, A. 2019. We also tested combined match strategies using the two similarity measures either on one or on two attributes for single-step learning. In total, 15 different learning-based approaches are considered.

For the easy Halliburton match task DBLP-ACM, we observe that both FEBRL and MARLIN are able to achieve stable results already for very small training sizes of 20 labeled entity pairs with all evaluated approaches. For the more challenging bibliographic match task DBLP-Scholar, for both FEBRL and MARLIN the SVM strategies combining several matchers on two attributes perform best and achieve F-measure results of 88-89%. The best one-attribute strategies are the combined SVM approaches and SVM using trigram (for FEBRL) or EditDistance (MARLIN). All approaches have substantial difficulties with the e-commerce match tasks, especially for training sizes smaller than 500 entity pairs. The best match quality is always achieved for the combined strategies using all similarity measures on two attributes, followed by the combined approach on one attribute. This underlines that the learners are able to effectively find a combination of several matchers. The decision tree learner of MARLIN is mostly inferior to the SVM-based results. The SVM learner of MARLIN performs slightly better than the one of FEBRL for smaller training sizes. However, for 500 training pairs both SVM learners perform similarly well and achieve a top F-measure of about 71% for exon Mobile and (only) 60% for Shell Products. From the single similarity approaches FEBRL with trigram and MARLIN with cosine similarity performed best for the Chevron tasks. For MARLIN, the 2-step learning for Edit- Distance was always better than the 1-step approach but still too ineffective for the Chevron tasks. Here the rather long product names and product descriptions tend to favor token-based similarity measures such as cosine, trigram, or the unsupported TF/IDF similarity. There are huge differences between the approaches regarding execution time as can be seen in Table 3. In general, the execution times for the considered learning-based approaches are significantly worse than for the non-learning approaches. Nearly all learning-based approaches do not scale with larger input sets and are unable to match sufficiently fast on the Cartesian product. For the largest match task Halliburton execution times of hours to days are needed, the most effective combined approaches exceeded our

limit of 500,000 seconds. On the blocked datasets, the approach with the fastest execution time for all match tasks is the FEBRL approach with the TokenSet Cosine measure. The combined match approaches on two attributes take the longest time for blocking, too. They are more than a factor 2 slower than the other learning-based approaches and (except for DBLP-ACM) requires execution times in the order of minutes to hours.



(a) FEBRL



(b) MARLIN

Figure 3: Evaluation results for learning-based approaches

4.2.2 Non-learning vs. learning-based

Table 4 shows a brief summary of the maximum F-measure results achieved for each of the considered non-learning as well as learning-based approaches. For three of four tasks the commercial COSY approach performs best for matching on one attribute. However for two match tasks its quality degrades when using two attributes. The learning-based approaches, on the other hand, always improve for matching on two attributes compared to only one attribute underlining their potential to effectively combine different match criteria. SVM learning was most effective and the FEBRL and MARLIN implementations perform similarly well for training size 500. They achieve the top F-measure for three of the four match tasks for matching on two attributes. The learning based approaches from FEBRL perform better than the non-learning FEBRL (FellegiSunter) approach, especially when considering two attributes. The good quality of the learning-based approaches on two attributes comes at the expense of significantly higher execution times. With a single matcher on just one attribute the learning-based approaches could not exploit their potential to combine several matchers and thus turned out to be inferior to the non-learning approaches considering both match quality and execution times. The relatively low match quality for the chevron task asks for further improvements, e.g., by considering additional similarity measures such as TFIDF and/or further attributes and spending more training effort on learning.

V. Conclusion

We presented a comprehensive and comparable evaluation of existing implementations of non-learning as well as learning based entity resolution approaches on challenging real-world match tasks. Our evaluations reveal big differences regarding match quality and execution times. It turned out that the commercial implementation COSY is very effective and efficient for matching on one attribute. However, it was not always able to effectively use more than one attribute for improved match quality. The learning-based match strategies using SVM, on the other hand, outperformed the non-learning approaches for the combined usage of several matchers on more than one attribute. While the SVM approaches effectively solve simple bibliographic match tasks with little training, more training is needed for the challenging e-commerce tasks (500 training pairs in our evaluation). Furthermore, the combined learning-based approaches could only be executed on blocked datasets and required the highest execution times of all match strategies. The best scalability was observed for the very fast single-attribute PPJoin+ implementation which was even faster than COSY and can be applied on the unblocked Cartesian product (execution time of at most 7 s). Hence scalability to large test cases needs to be better addressed in future approaches, especially for learning based approaches. The e-commerce tasks turned out to be quite challenging for all approaches and could not be effectively solved. More sophisticated methods are needed there.

Table 4: Summary of evaluation results (F-measure in %, top values are underlined)

	1 attr	2 attr	1 attr	2 attr	1 attr	2 attr	1 attr	2 attr
COSY	96.2	93.8	84.5	82.9	70.7	65.8	62.1	62.2
FEBRL FellegiSunter	97.6	96.2	57.2	81.9	44.5	36.7	48.4	53.8
PPJoin+	91.9	-	77.8	-	47.4	-	41.9	-
FEBRL SVM comb.	97.3	97.6	81.9	87.6	44.5	71.3	46.5	60.1
MARLIN ADTree comb	96.4	96.4	82.6	82.9	18.4	54.8	45.0	50.5
MARLIN SVM comb.	96.4	97.4	82.6	89.4	54.8	70.8	50.5	59.9

ACKNOWLEDGMENT

This research work was supported by the TEtfund Institutional Based Research (IBR) through the Center for Innovation Research and Excellence (CIRE) of Bauchi State University Gadau, Bauchi State, Under the TETF/DR&S/CE/UNIV/BASUG/IBR/2023/VOL.1 grant allocation.

REFERENCES

[1]. Avigdor, G. 2014. Uncertain entity resolution. Proceedings of the VLDB Endowment 7(13) 1711-1712.
[2]. Ayat, N., Akbarinia, R., Afsarmanesh, H., and Valduriez, P. 2014. Entity resolution for probabilistic data. Information Sciences 277 492-511.
[3]. Azar, S. 2019. Oil prices, US inflation, US money supply and the US dollar. OPEC Energy Review 37(4) 387-415.
[4]. Barkhatov, A., and Baranova, A. 2017. Cost Effectiveness in Oil and Gas Using Data Driven System. Oil and Gas Business (1) 153-177.
[5]. Blumberg, G. 2018. Oil & gas, Kluwer Law International, [Place of publication not identified].
[6]. Chuan, X., Wang, W., Lin, X., Yu, J., and Wang, G. 2018. Efficient similarity joins for near-duplicate detection. ACM Transactions on Database Systems 36(3) 1-41.
[7]. Domingos, S., and Elkan, G. 2016. Query-time Entity Resolution. Journal of Artificial Intelligence Research 30 621-657.
[8]. Dong, Y., Chen, J., and Tang, X. 2014. Unsupervised feature selection method based on latent Dirichlet allocation model and mutual information. Journal of Computer Applications 32(8) 2250-2252.
[9]. Getoor, L., and Ashwin, M. 2014. Entity resolution. Proceedings of the VLDB Endowment 5(12) 2018-2019.
[10]. Gevorkyan, A., and Semmler, W. 2017. Oil Price, Overleveraging and Shakeout in the Shale Energy Sector -- Game Changers in the Oil Industry. SSRN Electronic Journal
[11]. Giang, P. 2014. A machine learning approach to create blocking criteria for record linkage. Health Care Management Science 18(1) 93-105.
[12]. Han, J. 2016. An Approach for Detecting Similar Duplicate Records of Massive Data. Journal of Computer Research and Development 42(12) 2206.
[13]. Issa, H., and Vasarhelyi, M. 2017. Duplicate Records Detection Techniques: Issues and Illustration. SSRN Electronic Journal
[14]. Izakian, H. 2018. Privacy preserving record linkage meets record linkage using unencrypted data. International Journal of Population Data Science 3(4)
[15]. Kazim, A. 2019. Theoretical limits of OPEC Members' oil production. OPEC Review 31(4) 235-248.
[16]. Li Juan, Z., and Xiao, Z. 2018. Detection for Approximately Duplicate Records Based on Fuzzy Comprehensive Evaluation. Applied Mechanics and Materials 397-400 2464-2468.
[17]. Martin, D. 2018. Integration of Multiple Oil and Gas Data Sources for Use in Forecasting Future Rates of Discovery of Oil and Gas. AAPG Bulletin 75
[18]. Nigam, U., and McCallum, K. 2019. SPECIAL ISSUE ON ENTITY RESOLUTION Overview. Journal of Data and Information Quality 4(2) 1-2.
[19]. Omar, E., and Steven, W. 2017. Conventional Identity Resolution Methods: Issues and Trend. The VLDB Journal 18(6) 1261-1277.
[20]. Rajkumar, N., Kishore Kumar, K., and Vivek, J. 2018. Successive Duplicate Detection in Scalable Datasets in Cloud Database. International Journal of Engineering & Technology 7(2.4) 66.
[21]. Reynolds, D. 2018. The Energy Utilization Chain: Determining Viable Oil Alternative Technology. Energy Sources 22(3) 215-226.

[22]. Ripon, K., Rahman, A., and Rahaman, G. 2017. A Domain-Independent Data Cleaning Algorithm for Detecting Similar-Duplicates. Journal of Computers 5(12)
[23]. Soyemi, J., and Adegboye, J. 2018. Database Record Duplicate Detection System using Simil Algorithm. International Journal on Computer Science and Engineering 9(2) 55-61.
[24]. Sunter, I., and Fellegi, L. 2013. Reference reconciliation in complex information spaces. ACM Transactions on Knowledge Discovery from Data 1(1) 5-es.
[25]. Tian, Z., Lu, H., Ji, W., Zhou, A., and Tian, Z. 2016. An n-gram-based approach for detecting approximately duplicate database records. International Journal on Digital Libraries 3(4) 325-331.
[26]. Walls, W., and Zheng, X. 2018. Shale oil boom and the profitability of US petroleum refiners. OPEC Energy Review 40(4) 337-353.
[27]. Whang, S., Menestrina, D., Koutrika, G., Theobald, M., and Garcia-Molina, H. 2015. Entity resolution with iterative blocking. Proceedings of the 35th SIGMOD international conference on Management of data - SIGMOD '09
[28]. Winkler, W. 2019. Matching and record linkage. Wiley Interdisciplinary Reviews: Computational Statistics 6(5) 313-325.
[29]. Xing, Z., Xingchun, D., and Jianjun, C. 2018. A High Accurate Multiple Classifier System for Entity Resolution Using Resampling and Ensemble Selection. Mathematical Problems in Engineering 2015 1-6.
[30]. Ya-Kun, L., and Gao, H. 2018. Efficient Entity Resolution on XML Data Based on Entity-Describe-Attribute. Chinese Journal of Computers 34(11) 2131-2141.
[31]. Zhou, D., and Zhou, L. 2015. Algorithm for detecting approximate duplicate records in massive data. Journal of Computer Applications 33(8) 2208-2211.
[32]. Zycher, B. 2018. Barriers to Alternative Energy Sources. Fuel and Energy 42(2) 129-133.