MASTER THESIS
INFORMATION SCIENCES

Efficacy of Data Cleaning on Financial Entity Identifier Matching

Liping Liu (S4600150)
l.liu@student.ru.nl

Supervisors:
Prof. dr. ir. A.P.de Vries
arjen@cs.ru.nl
Prof. dr. ir. T.P. van der Weide
th.p.vanderweide@cs.ru.nl

July 14, 2016

Contents

1 Summary

This research has been carried out in context of the Financial Entity Identification and Information Integration(FEIII) challenge in finance area. This challenge aims to enhance the available data processing toolkit with new technologies facilitating the integration of financial information. For achieving this, one of the toughest tasks is to identify identical financial entities whenever they are represented differently. For example, a single financial entity can be maintained in different information systems with different names. It will be a significant achievement if such a technology, which has the capability to identify financial entities with 100% accuracy, is developed.

This task can be considered part of the research topic known as Record Linkage. Considering the duration for the research and also the present state of Record Linkage research, we have decided to research on the possibilities to clean the financial data so that the diverse data sets can be, to some extent, unified and thus be the complexity of record linkage reduced.

This research has been initiated with an analysis of the provided data sets from FEIII challenge organizers, including sample ground truth data and also data consisting of financial entity information. This helps to get some insights of the features of the data which afterwards has served as the basis for formulating the data cleaning strategy. After this, we have evaluated the prevalent data cleaning tools and record linkage tools. On the basis of the evaluation result, we have decided to utilize OpenRefine for data cleaning and FRIL for record linkage. At last, the record linkage task has been performed with the tools and the results have been discussed and evaluated afterwards. After the discussion on the research results, we have proved that data cleaning helps increase the recall of the record linkage while has no significant impact on the precision of the record linkage. This experiment also reveals that making an appropriate data cleaning strategy relies not only the data itself but also the domain knowledge. At last, considering the low recall value, we have suggested the research on the matching algorithm and the matching decision model as future work.

2 Introduction

In the financial sector, financial data resides in various information systems such as financial firms' internal systems, regulatory collections, and public websites. Financial data can also be presented across the financial ecosystem in different formats like financial contracts, regulatory filings, news articles, social media, and etc. A single financial entity is very likely represented in different ways. In other words, the mentions or references of a single financial entity might be diverse in the financial data collections. Actors like researchers, industry participants, and regulators, bring together and align financial data from a broad range of sources on a regular basis. Therefore, the resolution of mentions or references to the same financial entity is not trivial for carrying out the alignments of financial entity identification and information integration.

In 2015, a challenge called Financial Entity Identification and Information Integration (FEIII) has been announced. The challenge is jointly organized by the Office of Financial Research, the National Institute of Standards and Technology and the University of Maryland. The goal of the challenge is for information specialists to develop technologies that automatically align diverse financial entity identification schemes from four key regulatory datasets. These technologies aim to improve the efficiency and accuracy of the financial entity identification and information integration, by enhancing the toolkit for people who operate heterogeneous collections of financial entity identifiers [1].

The first task of the FEIII challenge is to identify matching entities, namely the rows indicating the same financial entity, across two of the four files provided by the organizers. Figure 1 is an example of the matching between FFIEC and SEC financial entities. In the figure, four SEC financial entities match with FFIEC financial entity 62110, out of which 866998 and 315123 are true positive matches while 1466052 and 1370965 are false positive matches.

FFIEC_ID	SEC_ID	ROOT	MODIFIER	STREET	CITY	STATE	ZIP	
39420		lorain	national bank	457 broadway	lorain	oh	44052	
	842581	lorain	national bank	457 broadway	lorain	oh	44052-1769	0.98
62110		rockefeller trust company,	national association	10 rockefeller plaza	new york	ny	10020	
	866998	rockefeller trust company,	national association	10 rockefeller plaza 3rd floor	new york	ny	10020	0.99
	315123	rockefeller & co., inc.		10 rockefeller plaza 3rd floor	new york	ny	10020	0.84
	1466052	evercore trust company,	national association	55 east 52nd street 23rd floor	new york	ny	10055	0.81
	1370965	deutsche bank trust company,	national association	280 park avenue floor 6 west	new york	ny	10017	0.78

Figure 1 FEIII task sample: Financial Entity Matching

Regarding the provenance of the data files, please refer to [1]. This task is a challenging one because of the following reasons:

- The complexity of datasets. Over decades, financial regulators have kept different data in a variety of databases. For instance, the address of an organization might be a single field, whereas it could be also broken into multiple fields.
- The data inconsistency. In additional to outright errors and typos, financial entity could be mentioned in datasets differently. For example, the mentions of 'J. P. Morgan' and 'JPMorgan Chase & Co' actually refers to the same financial entity.

- The implicit semantic knowledge of financial entity identifier. For example, the name of financial entity may contain ‘National Association’ or ‘State Bank of’ which indicates the financial organization is state owned.
- The automation of the process. This task requires to develop a technique that can match financial entities of different datasets in an automatic way.

Identification of the same financial entity across different data files is a task lying in the area of Record Linkage. Record linkage refers to the task of finding records referring to the same entity across different data sources when entities do not share a common identifier [2]. Variations of the problem are also known as Entity Resolution and Co-Reference Resolution. This is the essential problem that we need to solve in FEIII challenge. Over the past years, there are already many state-of-art techniques and frameworks developed so as to solve the record linkage problem. It is also commonly acknowledged that no universal techniques or frameworks that are capable of solving the record linkage problem across all industries considering the heterogeneity of data sources. In other words, for a single technique or framework, it is expected to perform differently in terms of effectiveness and efficiency when applied to different cases.

After studying the literatures about Record Linkage and the corresponding state-of-art techniques and frameworks, in general, the whole Record Linkage process can be divided into two steps. The first step is data preprocessing phase and the second step is the record matching phase. Nowadays, most of the data collected are unstructured or semi-structured data. In the first phase of Record Linkage, the input data is transformed and refined into structured or semi-structured data with high quality such that the computer can proceed with next phase without considering the data quality as a main impact on the performance of record linkage. After the data preprocessing phase, it comes to the record matching phase in which the record matcher algorithm, the strategy to combine record matchers and the blocking algorithms is going to be determined. Many existing frameworks have been developed for figuring this out, for details please refer to the survey paper [2].

Both of the two phases are vital parts of Record Linkage. However, regarding the performance of each technique or framework, it is commonly agreed that 20% of the effort is invested on recording matching while 80% on data preprocessing [9]. This implies also that the performance of the Record Linkage technique is greatly impacted by data preprocessing. Record matching consists of three main parts: the algorithms for matching records, the strategy of combining the algorithms so as to complement one another, and the blocking algorithms restricting matching on limited set of records. Many techniques have been developed for each of the three main parts, and this area seems not the most promising for further enhancements. What is more, considering that the four data files provided by FEIII organizers simply consist of structured financial entity data, it is relatively easy to find an existing framework or tool to do record matching. However, regarding data preprocessing in record linkage subject, generally it is a tricky problem because usually the data sources are diverse and have low quality. After studying the literatures on existing data preprocessing techniques, we also found most of the technologies are not automatic or even semi-automatic

approaches, which means human intervention is somehow always involved at a certain point of time during data preprocessing. So, there are many possibilities in data preprocessing areas and we think it is worth carrying out a research on this topic for the given the FEIII challenge.

3 Background

3.1 Issues to be solved

The intention of data preprocessing is to transform the unstructured or semi-structured data into structured ones with high quality. We also call this data cleaning. Three issues need to be resolved to achieve this.

- The first issue corresponds to the problem of schema mapping. Generally speaking, different information systems adopt different data schemas to store information. For instance, the address of the financial entity could be stored in a single field or multiple fields. Therefore, data schema mapping is necessary in order to eventual unify the way how the data is organized or make it clear the matching should take place on which set of fields.
- The second issue is normalization. Initially the data is entered by human, an error-prone process including typos. Things like typos might happen. So it is important to normalize the terms of the data such that the typing errors can be eliminated as a factor impacting the record matching. An example of normalization is to transform 'Americ' to 'America'.
- The last issue is standardization. Standardization helps to standardize terms referring to the same meaning. For example, United States of America, 'U.S.' and 'USA' refers to the same country. Therefore, we standardize the relevant terms into 'USA' as the standard representation of the country. This is very helpful for some record matching algorithms based on string similarity, because different representations of one word harms the precision and recall of the matching results. So, for eliminating or diminishing this negative impact, standardization of terms in data files is an important step.

3.2 Data Wrangling and tools

Now we introduce a term called 'Data Wrangling'. Data wrangling is indispensable for big data analysis. It is a loosely defined process of manually converting or mapping data from one 'raw' form into another format which allows more convenient consumption of the data with the help of semi-automated tools [4]. So we have researched on how much data wrangling can contribute to data cleaning.

Data wrangling aids to understand and gain insights into the datasets you deal with [5]. However, this does not have to be a tedious manual task [5]. A wide variety of data wrangling tools have been developed in academic and industry. In this thesis project, we have compared two data wrangling tools called OpenRefine and Trifacta and decided to use OpenRefine for data cleaning.

3.2.1 OpenRefine

OpenRefine (formerly Google Refine) is a powerful tool for working with messy data: cleaning it; transforming it from one format into another, and extending it with web services and external data [6]. It is an open source desktop application which is now supported by a group of volunteers. OpenRefine is hosted on a local machine, being operated through a web user interface (UI). When starting OpenRefine, it starts a web server and starts a browser to open the web UI powered by this web server. The web UI can be also accessed directly from http://127.0.0.1:3333/ after starting the server. Figure 2 is a simple view of OpenRefine web user interface. It is easy to get started with OpenRefine. The main steps taken for carrying on a data cleaning task on OpenRefine are like below:

- The first step is certainly start the OpenRefine and then go to the web UI by access the address mentioned above to start the clean the messy data.
- Then the user can start to create a project and upload one or more files that you want to clean. OpenRefine supports a wide variety of file formats, varying from tab separated files to JSON and XML. Since OpenRefine is an open source application, support for other formats can be added with OpenRefine extensions.
- The main functions provided by OpenRefine includes:
    - Facets. This function helps to categorize the values of the column and shows also the number of records for each category. Custom facet is also possible by writing and applying transformation expression in General Refine Expression Language (GREL) [21].
    - Text filter. It helps to filter the records by specific value and subsequently proceed with manipulating only the data set required.
    - Edit cells. Transformation of cell values is primarily realized by this option. There are some common transformations like ‘To uppercase’ provided by OpenRefine. Aside from that, user can also write their own expression by GREL to transform the cell values as they want. Another important feature is the ‘Cluster and Edit’ function. This feature helps to find groups of different cell values that might be alternative representations of the same thing.
    - Edit column. With this function, new columns can be created based on the values of the selected column. OpenRefine also allows to create new column with value fetched from web services, for example, can be used for geocoding addresses to geographic coordinates.
    - Transpose. This feature helps to transpose between rows and columns.
- Export result files. OpenRefine allows to export result files or intermediate files anytime during the data manipulation. It supports many file formats like CSV, HTML.
- Export operation history. At the end of the process, the whole set of operations can be extracts with format of JSON, and afterwards applied to other files when needed.

Figure 2 OpenRefine web user interface

3.2.2 Trifacta

Trifacta is a free data preparation application. It installs on a local machine and provides a graphic user interface. Trifacta facilitates data analysis by enabling users to transform complex data into structured formats. Their data wrangling experience is made up of six steps as outlined by Data Scientist Tye Rattenbury. Below each of the steps are explained [7].

- Discovering: This steps helps user to find out what data the dataset has.
- Structuring: The data set can be transformed into a structured format in this step.
- Cleaning: Messy data will be cleaned. For example, standardize the way to represent a country name.
- Enriching: After data has been structured and cleaned, data analyst might need additional information to proceed the analysis. So the data set can be enriched with additional information either from external source or inferred from the existing data set.
- Validating: After a series of data transformation operations, the resulting dataset is necessary to be validated so as to ensure no wrongful transformation carried out.
- Publishing: This is the last step of the data wrangling process. In it, the transformed dataset will be sent the downstream people.

Trifacta main window consists of three main parts:

- Analytic view of the data panel. On this panel, a set of analytic statistics can be viewed with respect a column of the dataset. Figure 3 gives a screenshot of the panel.
- Transformation script panel. Options of change and delete transformation is performed on this panel. It gives a list of transformations applied and also options to download the scripts.
- View of data in rows. The contents of the dataset can be viewed in this panel. Moreover, a preview of transformation results is also available on this panel.

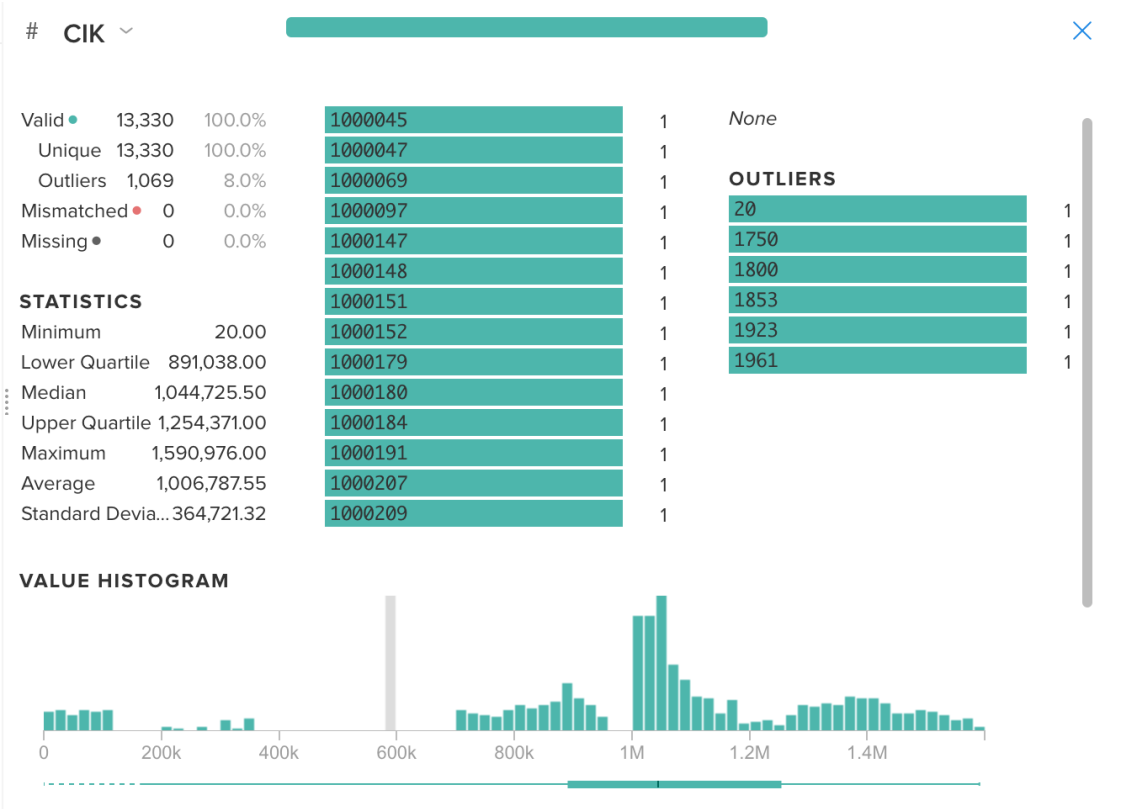

Figure 3 Analytic view of column data

3.3 Record Linkage tool: FRIL

Fine-Grained Records Integration and Linkage (FRIL) supports both record linkage and deduplication. In our case, we only used it for record linkage. FRIL is a very user friendly graphical tool that enables user to match records by just configurations. There are three types of configurations and they are data pre-processing manual configuration, record matcher configuration, and blocking algorithm configuration. The configurations are done sequentially. In this chapter, a brief introduction of FRIL has been given, and for more details about this tool, you could refer to paper 'FRIL: A tool for comparative record linkage' [3].

Overall Architecture

Figure 4 shows the overall architecture of FRIL. A FRIL workflow starts from specifying input data for linking. Next, FRIL can further be configured by specifying options for the search method, the distance function in the attribution comparison module and the decision model. After running the linkage task, the output files contain paired results of records, entities in our case.

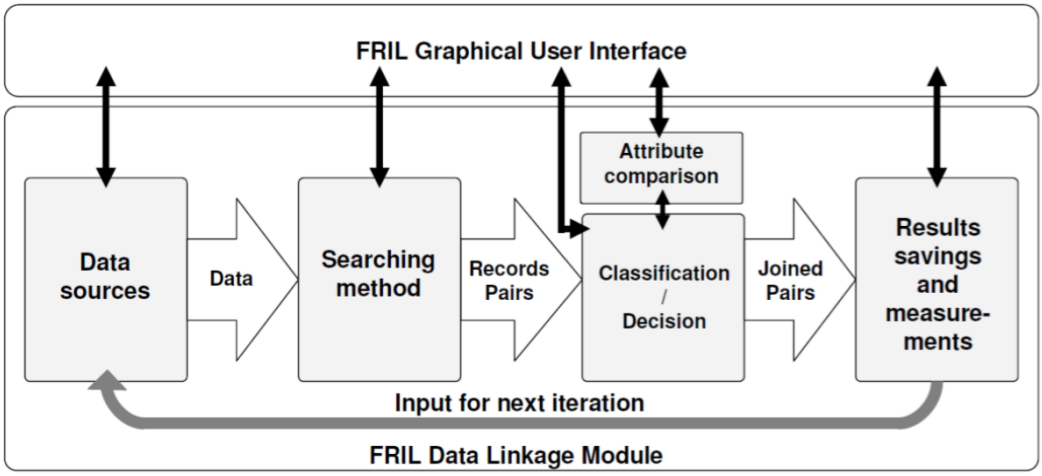

Figure 4 General architecture of FRIL [8]

Main Window

Configuration of FRIL takes place using a graphical user interface. Figure 5 shows the main window of FRIL containing the set of components through which all configurations can be finished.

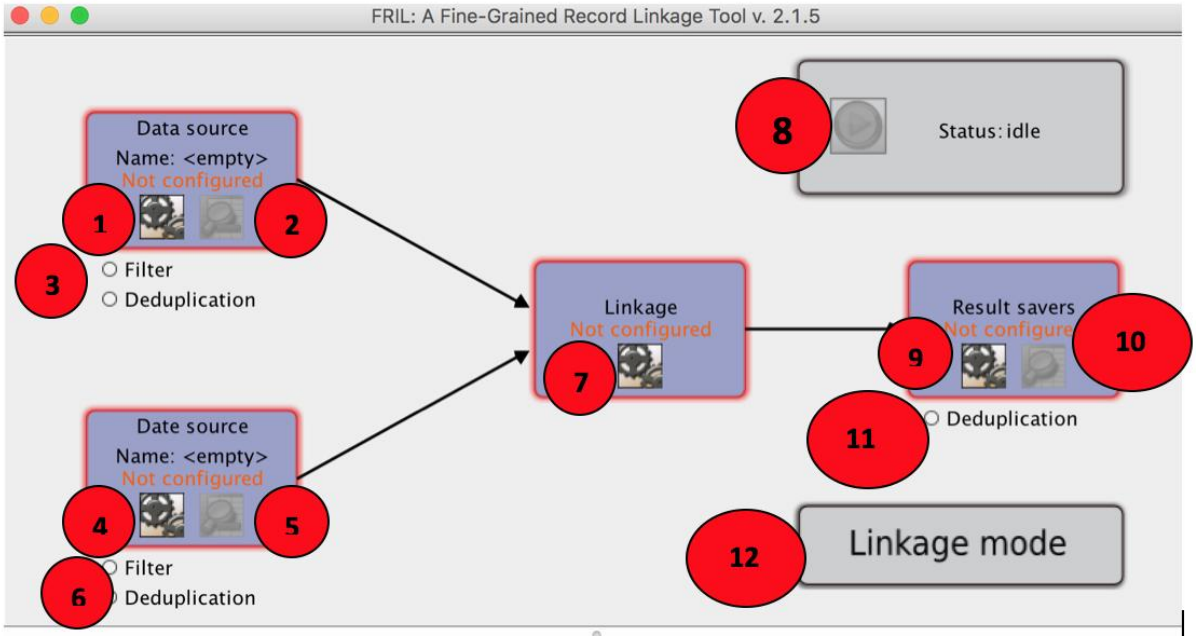

Figure 5 Main window of FRIL [8]

By clicking on 1 and 4, two files can be uploaded as input data for matching. By clicking 2 and 5, the data contents of the uploaded files can be viewed. 3 and 6 indicate of whether the file will be deduplicated and filtered before the matching process starts. This can be configured when uploading input files via 1 or 4. Clicking 7 opens a new window where distance metrics, search method and decision model can be configured. By clicking 9, the file system location for output files can be configured. 10 is an option to view the matching result in a graphical UI within FRIL without opening the result file. The system can also carry out a

deduplication task on the result file by selecting option 11. 12 is an indicator indicating which task FRIL is currently performing. In our project, we will utilize FRIL to complete record linkage task.

Linkage configuration

For carrying out a record linkage task on FRIL, below are the main configurations need to be accomplished:

1. Select fields to be compared from the input files. An example is given in Figure 6.

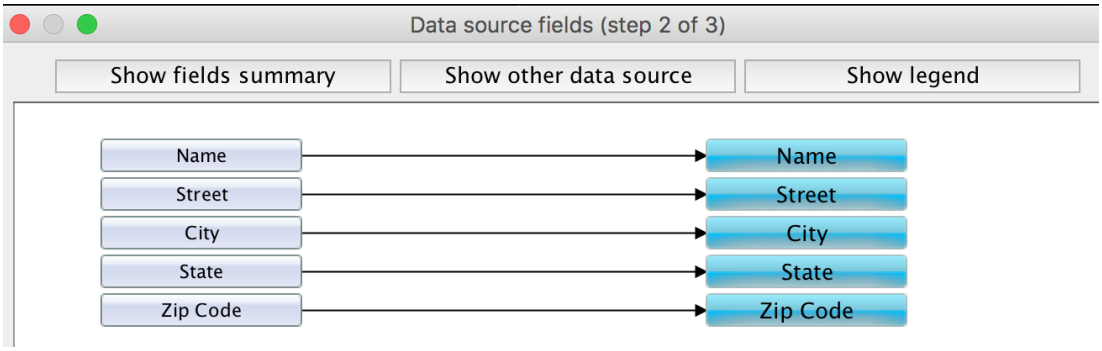

Figure 6 Field selection for matching

2. Configure a distance metric for each pair of fields to decide which matching algorithm is going to be adopted for comparing records (Figure 7). Comparison is weighted, so we also have to configure the weight for each pair of fields, specifying the contribution to the matching score. Finally, the decision model (acceptance level) needs to be configured as well (Figure 8). This conducts FRIL to categorize the pairs of entities into matches or mismatches.

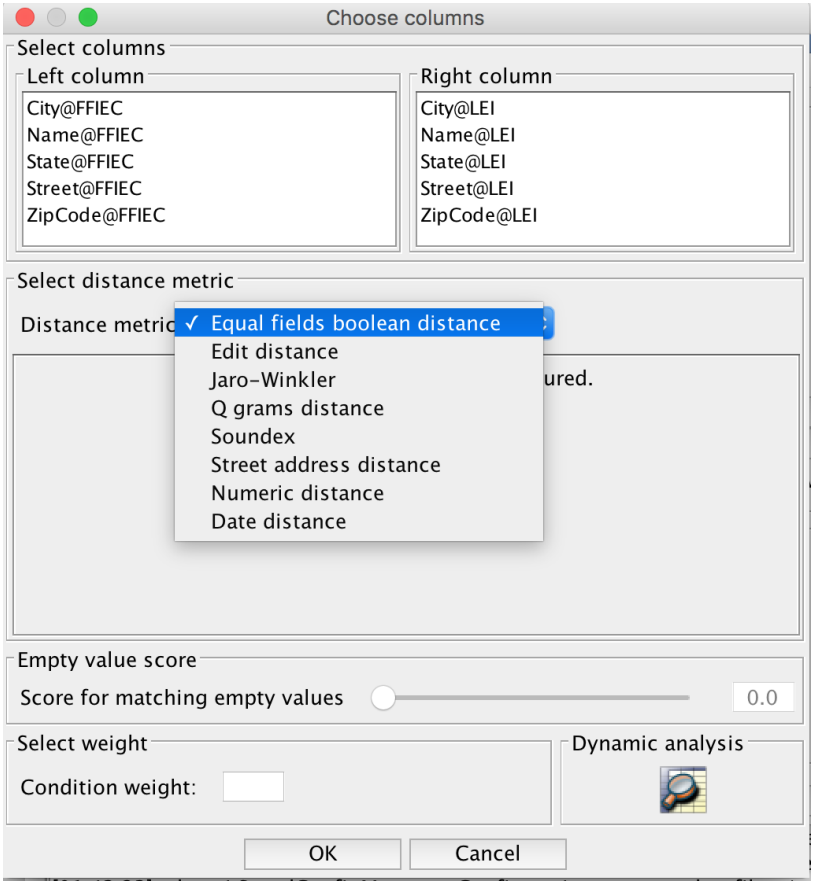

Figure 7 Distance metric configuration in FRIL

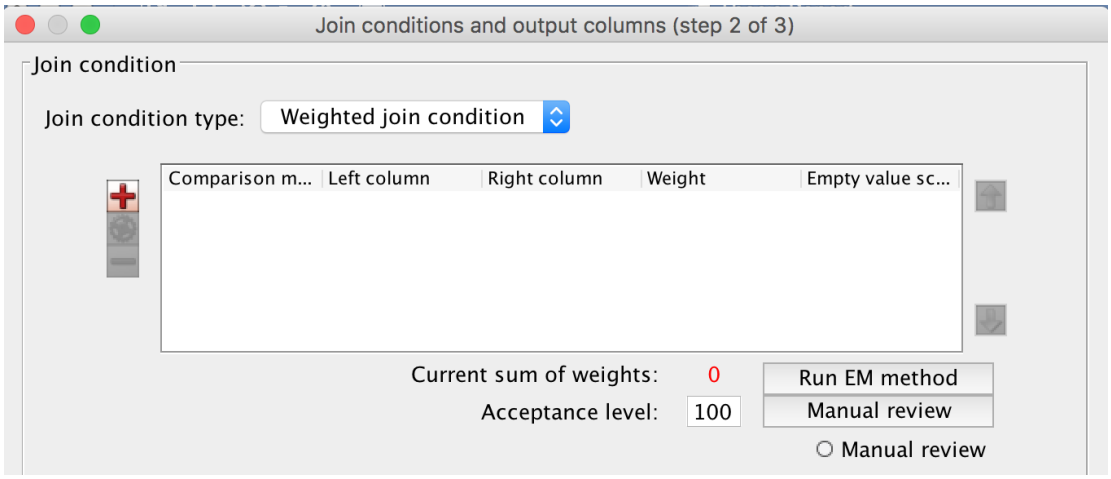

Figure 8 Decision Model in FRIL

Regarding the distance metric algorithm, FRIL offers a variety of options:

- Equal fields boolean distance: input value are treated as raw strings. This distance function returns only two values; 0 is returned if the compared values are different, otherwise 1 is returned.
- Edit distance: input values are treated as raw strings. This function tests how many operations need to be applied to the first string so that it can be converted to the second string.

- Jaro-Winkler: Input values are treated as raw strings. It is suitable for shorter strings, such as person names [3].
- Q-grams distance: input values are treated as raw strings. Both input strings are first divided into q-grams (substrings with length of q). It operates on the set of substrings regardless the order. Therefore, this distance minimizes errors due to switching, for instance, first, second and last names.
- Soundex: input values are treated as raw strings. This function calculates soundex code [3]. It is suitable to minimize misspelling errors when the two words pronounced the same.
- Street address distance: this has not been talked about in FRIL tutorial.
- Numeric distance: Input values are treated as numbers.
- Date distance: Input values are treated as dates.

3. Configure the search method

Search method refer to blocking algorithms for determining which pairs of records to compare. Currently, only Nested Loop Join (NLJ) and Sorted Neighborhood Method (SNM) are implemented in FRIL.

- NLJ: It performs an all to all comparison between two data files and is useful for small input data files [3]. This blocking algorithm guarantees the full set of matches can be found paying the price of huge amount of running time in case the matching is done upon two big input files.
- SNM: It sorts records in both input data files over relevant attributes, and follows by comparing only records within fixed window Wa and Wb of records [3]. This avoids the need to compare each record of one file against the entire data set of the second file and therefore reduce the running time for the matching. However, the window size limits the number of entities to be compared, as a result of which, matches could be lost.

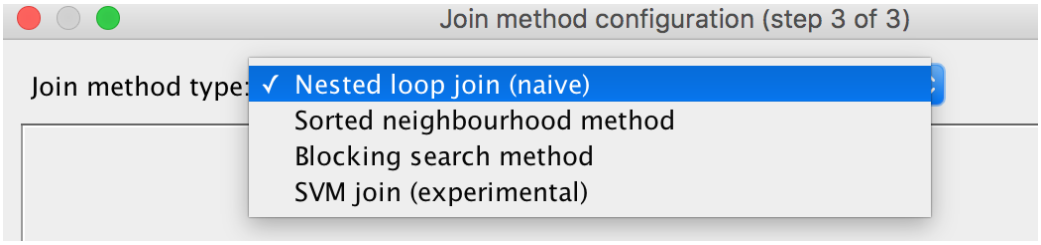

Figure 9 Search method configuration in FRIL

3.4 Data files

In our research, four data files have been used and they are:

- FFIEC data file. This file is released by the Federal Financial Institution Examination Council (FFIEC) and provides information about banks and other financial institutions that are regulated by agencies affiliated with the Council.
- SEC data file. This file is provided by the Securities and Exchange Commission (SEC) and contains entity information for entities registered with the SEC.

- Sample ground truth data file. In this file there are 40 matches along with information on how those matches were adjudicated as true or false matches by a human expert. This file is given by FEIII challenge organizers with the purpose of inspiring the challenge participants.
- Final ground truth data file. The FFIEC challenge organizers wrote a baseline algorithm and generated the record matching result which has also been reviewed by human experts. This file is supposed to be released in July, however, for continuing with our research, the FFIEC organizers released the file to us as a favor. This file is for calculating the precision, recall and F-score of the record matching results.

3.5 Evaluation method

We have adopted an indirect approach to evaluate the performance of data cleaning using data wrangling tools. The main steps of this approach are as follows:

- Cleaned the provided FFIEC and SEC data sets using data wrangling tools. As a result of this, the FFIEC and SEC financial entity information has been transformed into a new format with improved quality.
- Two rounds of record linkage have been carried out. In one round, the cleaned FFIEC and SEC data files were handed over to FRIL as input data source files whose financial entities have been compared and matches; in the other round, the initial FFIEC and SEC data files have been taken over by FRIL as input data files. The same set of configurations of FRIL was applied so as to eliminate the bias from FRIL.
- A set of measurements, including precision, recall and F-score, have been computed for both rounds and they have been compared with each other for evaluating the performance of the data cleaning. Below shows how the three metrics should be computed.
    - Precision = true positive matches / positive matches. This measurement indicates the accuracy of the matching, namely the percentage of the true matches out of the whole set of the returned matches including both true and false matches.
    - Recall = true positive matches / (true positive matches + false negative matches). This measurement tells how many true matches can be returned at last, namely the percentage of the true matches returned out of all true matches either returned or not.
    - F-score = 2 * (Precision * Recall) / (Precision + Recall). Usually we need a trade-off between precision and recall. Both high precision and recall are not achievable. Therefore, F-score is a measurement reflecting the trade-off between precision and recall. It can be used as an indicator of the overall performance of the matching.

4 Research work

4.1 Research question

This thesis project has been carried out in context of the FEIII challenge. As the first phase to solve the financial entity matching problem, this challenge requests the participants to finish a record linkage task between two of the four data files they provided. The files contain limited information about the financial entities, basically restricted to header metadata like name, country, state and post code. In our research, we have made the research questions as below which are supposed to be answered in the end:

- What are the advantages and disadvantages of data preprocessing tools such as OpenRefine and Trifacta in terms of data cleaning, and which of them is more suitable for the FEIII challenge?
- What is the superior strategy to 'clean' the task files with the help of data wrangling tools?
- How much can the performance of FRIL be improved if the files are 'cleaned' before carrying out record linkage?

4.2 Research approach

The FEIII challenge organizers provide four data files originating from different US financial institutions and regulators. We therefore decided to carry out our research using the FFIEC file and the SEC file. The sample ground truth data released by FEIII organizers is on the base of FFIEC and SEC matching.

we studied the provided data files and formulated a general strategy of data cleaning to clean the provided data files. For evaluating OpenRefine and Trifacta, we have studied the two tools with the help of their tutorials and came up with a list of advantages and disadvantages of the tools considering the FEIII challenge. The superior strategy has been finalized during the preprocessing for the provided files because along with the process we have become more familiar with the data. At last, the effectiveness of the data cleaning and its impact on record linkage have been evaluated by comparing the matching results generated out of the original data files and cleaned data files. So the entire research work has been carried out in several steps as follows.

1. Data analysis. This is for two purposes:
    a. Analyze FFIEC and SEC data files and decide which information is vital for and relevant with our research.
    b. Analyze the sample ground truth data and get insights of how a match and mismatch can be adjudicated.
2. Decide a resonable combination of parameter values of FRIL.
3. Formulate a general strategy to clean the data. At this phase, the data analysis results from step 1 have been utilized to make such a cleaning strategy.

4. Evaluation of OpenRefine and Trifacta in terms of data cleaning. The performance of these two tools has been evaluated in terms of human efforts, reusability of transformation scripts and also the considerations from FEIII challenge.
5. Revise data cleaning strategy. Starting with the initial strategy, after finishing each step, we examined the cleaned data set to ensure the work has been done correctly. During the examination, new issues of the data are possible to be detected and this requires the revise of the cleaning strategy if necessary.
6. Compare the matching results before and after data cleaning. By this comparison, we are able to know how much the data wrangle tools contribute to record linkage.

4.2.1 Analysis of the given data

This chapter describes the data analysis process. It includes the analysis of two types of data files:

- Sample ground truth data file
- FFIEC and SEC data files

Initially, we assumed matching was done syntactically. For instance, if the name of the financial entity is very similar syntactically, we assume this is a match. However, the real case is more complex. The sample ground truth data file includes positive matches from the FEIII challenge along with the adjudication from domain expert for true and false positive matches. So we studied the sample ground truth and tried to get some insights about what else has been considered for record linkage from the matches and mismatches in the file.

Analysis carried out on FFIEC and SEC files is for summarizing an initial set of data characteristics of these two files as a preparation for formulating an initial data cleaning strategy.

4.2.1.1 Analysis of sample ground truth data

For evaluating the matching results submitted by the FEIII challenge participants, the organizers wrote a baseline algorithm and generated the final ground truth data file. The final ground truth data file is supposed to be released in July of 2016. However, for giving some insights and also examples of match and mismatch entities to the researchers before the release of the final ground truth data file, the organizers published a sample ground truth data in 2nd Feb, 2016.

The sample ground truth includes 40 matches between FFIEC and SEC files. It includes true positive and false positive matches which have been distinguished by human experts based on their domain knowledge. There are also notes from human experts explaining why the sample match was adjudicated as true positive or false positive. The ratio of true positive and false positive matches is 1:1 as depicted in Figure 10. Figure 11 shows details about the distribution of matches adjudicated with and without the help of domain knowledge.

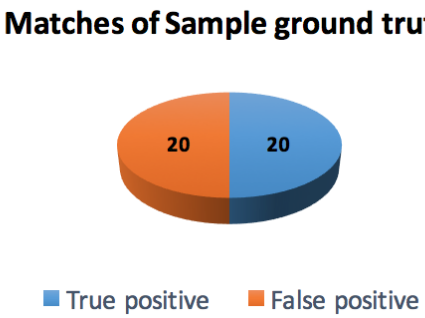

Figure 10 Sample ground truth

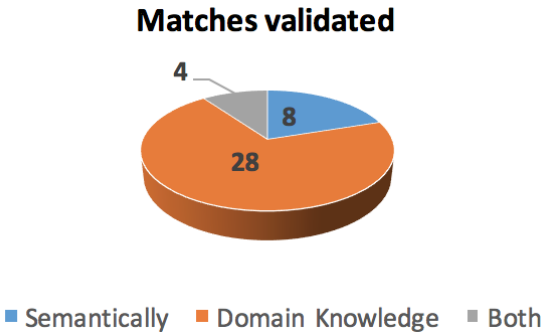

Figure 11 Human expert verification

From Figure 11, we know that 80% of the sample matches were verified by human expert with the help of domain knowledge. To increase our understanding, we classified the domain knowledge that has been used such that we know what is relevant with the matching. After studying the notes, we identified three types of considerations:

- Corresponding FFIEC entity of SEC entity. For some of SEC financial entities, human experts checked their corresponding registration in FFIEC and subsequently compared the institution types. If the institution types are different, the entities are different, no matter how similar their names or addresses are.
- Filing form type submitted by SEC entities. Some SEC entities were categorized as "transfer agent" because of the type of filings submitted by that entity on SEC.
- Registration of the financial entity on the state website. In the US, the financial entity name is guaranteed to be unique within the state while can be reused in other state. So in case the name of the financial entities is the same and they are both registered in the same state, then they can be treated as the same entity.

Certainly it will be great if all information mentioned above can be collected, however, the FFIEC and SEC data files provided by FEIII challenge include basically only entity names and addresses. In other words, we have no straightforward data source containing all the

wanted information. The analysis of the human review leads to the following valuable insights:

- Institution type implies the nature of the financial entities. So if a system could simply interpret that if the entities are with different institution types, then they are different entities.
- Institution type is often reflected in the name of the institution, by words like ‘corp’, ‘bank’ and so on.
- Representations of a single institution type vary. For instance, ‘trust company’ and ‘trust co’ represent the same entity type. Standardization of institution name is essential.
- Registration of institutions with the same name could happen in different states, so state information is very important and should have a high weight in the matching procedure. If two entities are located in different states but with similar names, system should consider them as different entities.
- Zip codes, ‘21874’ and ‘21874-0010’ should be treated as identical because the digits after hyphen do not contribute a lot. In the US, zip codes have 5 digits and each of these corresponds to a different geographical area. In case ‘state’ is missing for any record, we could infer the state from the zip code. Figure 12 shows the constitution of US zip code.

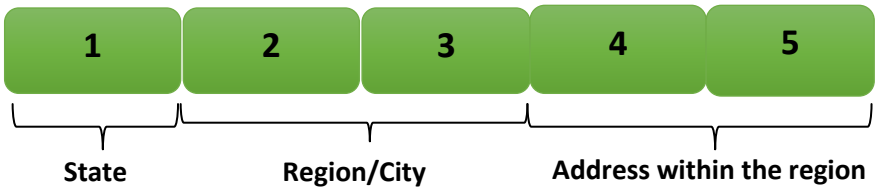

Figure 12 US zip code constitution

From the notes in the sample ground truth, we hypothesized the logic behind the matching done by human expert as follows:

- If the records match exactly and syntactically on name, street, state and zip code, then they are a match.
- If the records match exactly and syntactically on street, state and zip code but not name:
    a. If the name only differs on suffix starting with ‘/’, then they are same record.
    b. If the name only differs on other suffix, then need to check the registrant of SEC in FFIEC and compare the institution type. If the types are different, then they are different records, otherwise they are the same.

We anticipate some difficulties when trying to automate this process.

- Find all the registrants of SEC entities in FFIEC. This is difficult because two reasons. First of all, we did not find a complete set of FFIEC data from the FFIEC Bulk Data Download portal [10]. Secondly, it is essentially another matching task, for which we do not know the exact matching rules.

- Get the exact institution type information for both FFIEC and SEC entities. Considering the first difficulty, collecting the institution type information for all SEC entities becomes unattainable.
- Inference of institution type from entity name is not feasible because it is not 100% guaranteed that a word is only used to indicate institution type. Because of this the extraction of such words from entity name is problematic.

Given the analysis of the sample ground truth data and the difficulties foreseen, we came up with below ideas for optimizing the quality of both FFIEC and SEC data files:

- For suffixes starting with '/', we replace it with space, namely disregard it. For example, remove '/TA' directly because it does not distinguish entities.
- If a zip code value consists of two parts connected by a hyphen, then only keep the part before the hyphen (which is supposed to be a 5-digit numeric string).
- For entities without state information, infer the state of the financial entities if the zip code is not empty.

4.2.1.2 Analysis of the FFIEC and SEC dataset

Considering the provenance of the FFIEC data file and the SEC data file, we understand the former file contains only financial institution information, while the latter one contains not only financial entities but also entities of other industries. Figure 13 and Figure 14 are segments of the FFIEC and SEC entitiess.

IDRSSD	FDIC Certificate Number	OCC Charter Number	OTS Docket Number	Primary ABA Routing Number	Financial Institution Name	Financial Institution Name Cleaned	Financial Institution Address	Financial Institution City	Financial Institution State	Financial Institution Zip Code	Financial Institution Zip Code 5	Financial Institution Filing Type	Last Date/Time Submission Updated On
37	10057	0	16553	61107146	BANK OF HAI	BANK OF HAI	12855 BROAI	SPARTA	GA	31087	31087	41	2014-10-24T16:36:37
242	3850	0	0	81220537	FIRST COMM	FIRST COMM	260 FRONT S	XENIA	IL	62899	62899	41	2014-10-28T12:58:37

Figure 13 A segment of the FFIEC data file

CIK	IRS_NUMBER	CONFORMED_NAME	MinOfFILING_DATE	FORMER_CONFORMED_NAME	FORMER_NAME_CHANGED	ASSIGNED_SIC	B_STREET	B_STREET1	B_STREET2	B_CITY	B_STPR	B_POSTAL
20	221759452	K TRON INTERNATIONAL INC	20080722			3823	ROUTE 55 &	ROUTE 55 &	BOX 888	PITMAN	NJ	08071-0888
1750	362334820	AAR CORP	20080605	ALLEN AIRCR	19700204	3720	1100 N WOO	1100 N WOO	D DALE RD	WOOD DALE	IL	60191

Figure 14 A segment of the SEC data file

By examining the fields of both files and also referring to the data dictionaries [13] [22] of both files, we decided to carry out the matching on name and address information by utilizing corresponding fields of both files:

FFIEC	
IDRSSD	The identifier of financial entity
Financial Institution Name Cleaned	The name of financial entity
Financial Institution Address	The street address of the financial entity
Financial Institution State	The state of the financial entity
Financial Institution Zip Code 5	The Normalized zip code of financial entity

SEC	
CIK	The identifier of entity
CONFIRMED_NAME	The name of entity
ASSIGNED_SIC	The standard industrial classification code of the entity
B_STREET	The street address of the entity
B_STPR	The state of the entity
B_POSTAL	The zip code of the entity

Then we carried out a check on the values of each selected fields of both files and concluded a list of issues detected:

- Function words: These words have very little value in helping adjudicate a match or mismatch. For instance, the word "the".
- Abbreviation: both financial name field and street field sometimes use abbreviation while sometimes use the original words.
- Symbols: symbols appear in name, street and zip code fields.
- Different representations: some words refer to the same thing. For instance, 'U.S.', 'US', 'USA' represents the same country.
- Zip code has different number of digits.

As what we mentioned before, for improving the accuracy of the matching, we need to standardize and normalize the fields that are counted for matching. For example, remove the symbols and function words, unify the representation of a single country, generalize the length of zip code, and so on. This will be elaborated in chapter 4.2.3.1.

4.2.2 Tuning FRIL for Record Linkage

During the data analysis of the FFIEC and SEC data files, we have decided to utilize only name and address (except state) fields for record matching. Considering they are strings, we decided to adopt Edit distance function for name, street and zip code. Equal fields Boolean distance function gives an absolute answer 'Yes' or 'No while our case is on the basis of string similarity. Jaro-Winkler function is designed and best suited for short strings such as person names, however in our case the name and address are long strings. Q-gram distance requires to give the number q to decide how to make substrings. This does not fit our case because there is no way to give the number q. Soundex helps to correct misspelling errors which can also be done by OpenRefine (in chapter 3.2.1). Considering also that spelling mistakes are not the primary issue we have to address in the challenge, so Soundex is not a good candidate. The street address function is not well defined in the documentation, without an explanation of its algorithm, so we do not consider it. Numeric distance and Date distance are very specific to numbers and dates, whereas we deal primarily with string data. For the state field, with the aid of Microsoft excel we did a general check (sorting, filtering) of both

FFIEC and SEC data, and we figured out that the state data is consistent and therefore we adopted Equal fields boolean distance function for comparing that field.

Taking the analysis of the sample ground truth data into consideration, we made the initial set of FRIL parameter values to be configured:

- The weight of the field 'name' should be high, and the initial value is 40% by adopting 'Edit distance' comparison method.
- The weight of the field 'street' is set to 30% and comparison method is 'edit distance'. Because street is also vital information for financial entity.
- The weight of the field 'state' should be high, and the initial value is 20%. What is more, when state is empty, the score is initially set to 0.5. The comparison method for state is 'equal fields boolean distance'.
- The weight of the field 'zip code' is set to 10% and comparison method is 'edit distance'.
- Acceptance level: 70

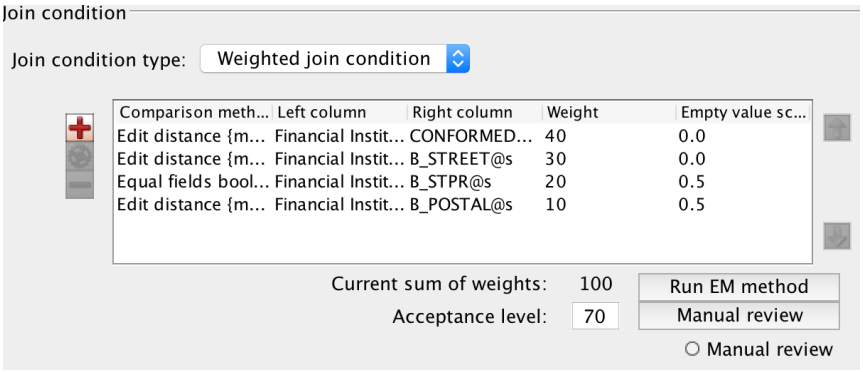

Figure 15 The combination of FRIL parameter values

Aside from the configurations of the distance metric and the decision model, we have decided to employ SNM blocking algorithm for the matching. We tried to run the record linkage upon FFIEC and SEC data files with NLJ blocking algorithm and we found FRIL was still running after almost one hour. Also, as what we mentioned, our research focus on the contribution of the data cleaning to the matching, so we do not aim to get the full set of matches. We configured the window size as the default value 8 proposed by FRIL and sort the records in the input file by state, zip code, name and street address sequentially, because for identical entities, they must have the same state, and similar zip code, name and address.

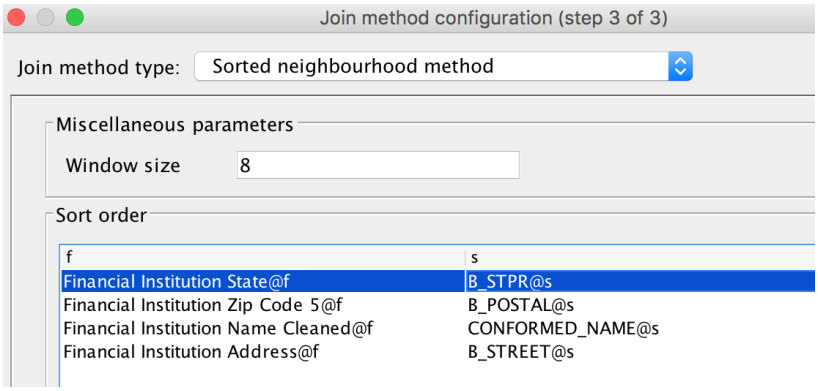

Figure 16 The blocking algorithm for the matching

In the following experiments, we applied the final combination of FRIL parameter values to the record linkage task, and based on the results the precision, recall and F-score have been computed.

4.2.3 The experiment strategy

4.2.3.1 Data cleaning strategy

The aim of the experiments is to clean the FFIEC and SEC data as much as possible so as to improve the precision and recall of record matching result. Hereto, we carry out experiments to:

- Narrow down the scope of the matching by reducing the number of SEC entities so as to limit the matching only for financial entities. As what we mentioned before, the FFIEC data file contains only financial entities, while the SEC data file contains more entities of other industries aside from financial entities. We will restrict the matching to take place only among financial entities, to help eliminate the impact of non-financial entities and thus improve the precision, and also improve the performance of the matching in terms of consumed time.
- Normalize and standardize the information of financial entities including name, street, and zip code. This is for preventing biased matching due to the function words and different representations of an object (like country). Through the normalization and standardization, recall can be increased.

Narrow down the scope of record matching

In SEC data file there is a field 'ASSIGNED_SIC'. The SIC code is the standard industrial classification code of the entities registered in SEC. It is for classifying all industry and therefore indicate the business type of the registrants on SEC. from Figure 17 we could see finance industry has be classified with the range 60-67 (the first two numbers of the SIC code). So the SEC entities with SIC code out of range 60-67 can be removed from the SEC data file.

Figure 17 Industry categorization by SIC code

The unclassified SEC entity comprises a large part of the total number of SEC entities. Without the SIC code, it is unclear how many of the unclassified entities are financial or non-financial entities. Also, until now only a small portion of non-financial entities can be removed from SEC data file. As a solution, we looked into external data sources to collect the SIC code information for unclassified entities. Considering the fact that some SEC entity identifier CIK numbers have been discontinued or even have been reused for other entities, thus for guaranteeing the reliability of the external data source, we have asked SEC web group, who is responsible for questions about SEC public data, for the files containing SEC entities with SIC code. According to the reply from them and after checking the link [12] they offered we eventually found out there was no such public dataset available for the time being. We have also thought about inferring the SIC code by looking for a pattern out of the known financial SEC entities. However, this is a noisy process that risks removing financial entities as well resulting in a reduction on the recall. We have also considered to perform fuzzy record matching between FFIEC and SEC in order to reduce the scope of SEC entities, by setting a low acceptance matching score, yet the same risk remains. Due to the fact that for the FEIII challenge, the performance of record matching in terms of consumed time is not prioritized as the measurements like precision, recall and F-score, we decided to keep the SEC data file with non-financial entities removed as in Table 1 in the experiments.

Normalization and Standardization of SEC entity information

As per the analysis in chapter 4.2.1.2, we constructed the strategy to normalize and standardize the name, street, state and zip code of FFIEC and SEC entities to mitigate the impact caused by abbreviations, diverse representation and function words.

The value of entity name and street fields are be cleaned by:

- Removing function words
- Removing symbols

- Eliminating spelling mistakes
- Substituting words representing entity type by abbreviation

The value of entity state field is cleaned by:

- Populating empty field with correct state value by calling geocoding Ziptastic API with zip code value as input parameter if it is not empty.

The value of entity zip code field is cleaned by:

- Unify the format of the zip code consisting of 5 digits
- Remove fake zip codes.

4.2.3.2 Record matching strategy

For evaluating the contribution of data cleaning, we generated a baseline record matching result before starting the data cleaning. This baseline result file has been compared with the result file generated after data cleaning such that we had concrete evidence to proceed with the discussion on the contribution of data cleaning for the given FEIII challenge. This baseline matching result was generated syntactically by configuring the FRIL in accordance with the set the parameter values we have talked about in chapter 4.2.2. About the matching after data cleaning, we generated 5 matching results:

- Matching result after cleaning Name
- Matching result after cleaning Street
- Matching result after cleaning State
- Matching result after cleaning Zip code
- Matching result after cleaning all four fields.

The purpose of generating multiple matching results is to evaluate and analyze the contribution of each field and also in overall to record linkage. Details are discussed in chapter 4.2.6.2.

4.2.4 Data preprocessing tools adopted

From the wide range of data wrangling tools, we have studied OpenRefine and Trifacta according to the comments from Andy Green on his blog [11]. Both of them are populate and highly recommended data wrangling tools. Nevertheless, we have chosen OpenRefine to clean the FFIEC and SEC data files. This decision has its roots in the analysis of the advantages and disadvantages of the two tools like below:

- OpenRefine enables the user to view the filtered or categorized rows of data and the profiling of the subset of data on the same page, while Trifacta requires the user to switch between ‘Grid’ and ‘Column Detail’ views.
- In terms of filtering, in OpenRefine, only the filtered data visible to the user, and afterwards all transformations applied are only applicable to the filtered data. In Trifacta, the filtered data is highlighted in the whole data set and the transformations for the filtered data are achieved by adding the filtering condition in the transformation expression scripts explicitly. The way how Trifacta works complicates

the user's overview of the data and the incorrect transformations may be applied in case the transformation expression is wrong.

- In OpenRefine, the Cluster and Edit function is very useful to eliminate spelling mistakes, although it is not the primary issue of the FFIEC and SEC data files.
- OpenRefine supports the selective download of transformation scripts in order to apply to other data file as long as the file has the same name of the fields that involved in the transformation. This is also possible in Trifacta by switching data source, yet it requires the data files to have the exact same data schema and selective application of transformation scripts is not possible.

Certainly, Trifacta is a very powerful software for manipulating data. The analytic view by column gives a straightforward option to have an initial idea about the data contents. However, it fits the case more when the analysis of data content is prioritized. OpenRefine has no visualized view of a summation of the statistical feature of column contents, nevertheless it is more favorable when cleaning of messy data is the objective.

4.2.5 Implementation Details

In this chapter, the implementation details of the data preprocessing strategy will be expounded. It describes the implementation details field by filed. We start from the data cleaning on FFIEC file and later on the SEC file by mentioning only the differences.

FFIEC

FFIEC data file contains only financial entities, so there is no need to narrow down the scope. The data cleaning for FFIEC file has been done in OpenRefine. For cleaning each fields of FFIEC data, we first did an analysis on the values of the field by utilizing facet function, and then enhance the data preprocessing strategy we have talked about in chapter 4.2.3.1. Below are the main steps through the whole process:

- Create a OpenRefine project and upload the FFIEC data file with format .CSV.
- Field ZIP CODE
    - Issues:
        - Some zip codes have less than 5 digits
    - Aim: Field 'Financial Institution Zip Code 5' is normalized.
    - Cleaning strategy implemented
        - Transformed zip code to text field because leading zeroes will be removed if it is a number.
        - Added leading zero to zip codes which have less than 5 digits.

- Field STATE
    - Issues:
        - Some state cells have empty value.
    - Aim: Field 'Financial Institution State' is populated.
    - Cleaning strategy implemented

        - Filtered the column and only leave the rows with empty state field.
        - Populated the cell of STATE column by fetching state information via API http://ziptasticapi.com/<zipcode>.

- Field STREET address
    - Field 'Financial Institution Address' is cleaned.
    - Issues:
        - It contains symbols
            - '#' stands for 'number'
            - ',' splits the address units.
            - '&' stands for 'and'
            - '-'separates building number and room number
            - '/' stands for 'or' separates numbers
            - '' appears in S', 'S or go with the name of a street, building etc.
            - '.' appears in abbreviation of a term like ST. (street), N. (north) etc.
        - It contains abbreviation
            - 'Orientation' abbreviation. For instance, N.E. stands for northeast, W. stands for west and so on.
            - 'Road' abbreviation. For instance, ST. stands for street, AVE. stands for avenue and so on.
    - Cleaning strategy implemented
        - First replaced the words that are equivalent with the abbreviation with corresponding abbreviations.
        - Replaced '#' by 'NO' because the symbols is useful as part of street string.
        - Then replaced all remaining symbols by a space
        - At last removed consecutive spaces

- Field NAME
    - Field 'Financial Institution Name Cleaned' is cleaned
    - Issues:
        - Function words. There are words that are useless for contributing the matching. For example, 'AND', 'THE'.
        - Name contains some special words like 'NATIONAL', 'BANK,' 'FINANCE' to indicate the nature of entity.
        - Abbreviation. For example, INC stands for INCORPORATED, CO stands for COMPANY.
        - Multiple representation. The same thing has been represented in different ways. For example, US, U.S., U.S.A refers to the same country.
        - Misspelling. Some words are spelled wrongly. For example, 'Americ' should be a typo of 'America'.

        - '/': Remove this symbols together with the words conjunct with it if there are some. Reason has already been talked about above.
    - Cleaning strategy implemented
        - Copied the 'Name' field to a new field called 'Entity Name'
        - Split the name field into multiple columns each of which contains only one word.
        - Grouped and corrected misspelling words by utilizing the cluster function. Standardization of terms has been accomplished at this step.
        - Faceted each column and got a list of key words indicating the nature of the entities.
        - Deleted the name field and also other fields resulting from the split.
        - Replaced the list of key words by an abbreviation if the word is long.
        - Removed the string starting with '/'.
        - Replaced '#' by 'NO'.
        - Removed all other symbols.
        - Removed all function words.
        - Removed consecutive spaces.

SEC

SEC data file contains not only financial entities but also entities of other industries. And the data is more diverse than FFIEC data, so aside from what we did for cleaning FFIEC data, we have done additional operations on SEC data like below:

- Narrow down the scope of matching. As what we already discussed in chapter 4.2.3.1, we first reduced the number of SEC entities according to the SIC code included in the provided SEC file. Giving the statistics in Table 1, the number of SEC entities has decreased from 129312 to 115186.

Number of entities	129312
Non-financial entities	14126
Unclassified entities	109167
Financial entities	6019

- *Table 1 SEC entities*

- For reusing the data preprocessing scripts, we renamed the columns of SEC file to the same as FFIEC file.
- More issues with Zip code.
    - Hyphen. Some zip codes have hyphen and we trimmed all of then and kept only the part before the hyphen.
    - Fake value. Some zip codes are fake and we blanked out the cells.
    - More than 5 digits. Some of zip codes have more than 5 digits. We simply trimmed the values and kept the first 5 digits.

For both FFIEC and SEC data files, the function words replaced by a space are 'OF', 'THE', 'AND' and 'AT'. The symbols removed from name and street are: '/', '.', '(', ')', ',', '-', '&',

and ‘’’. Table 3 is a list of ‘orientation’ words replaced by corresponding abbreviations. Table 4 contains the list of road words or symbols replaced by corresponding abbreviations. Table 5 includes a list of words have been grouped and therefore standardized. Table 6 shows a list of key words we have identified with corresponding abbreviations for some of the words.

We followed these principles when formulating the cleaning strategy for each of the fields:

- Replace ‘orientation’ and ‘road’ words by corresponding abbreviations. This is because FRIL mainly adopts ‘Edit Distance’ algorithm to compare the similarity of two strings. For instance, string ‘A national association’ and string ‘B national association’ will get a high matching score but are just different entities. The effect is demonstrated by a small experiment depicted in Figure 18 and Figure 19.

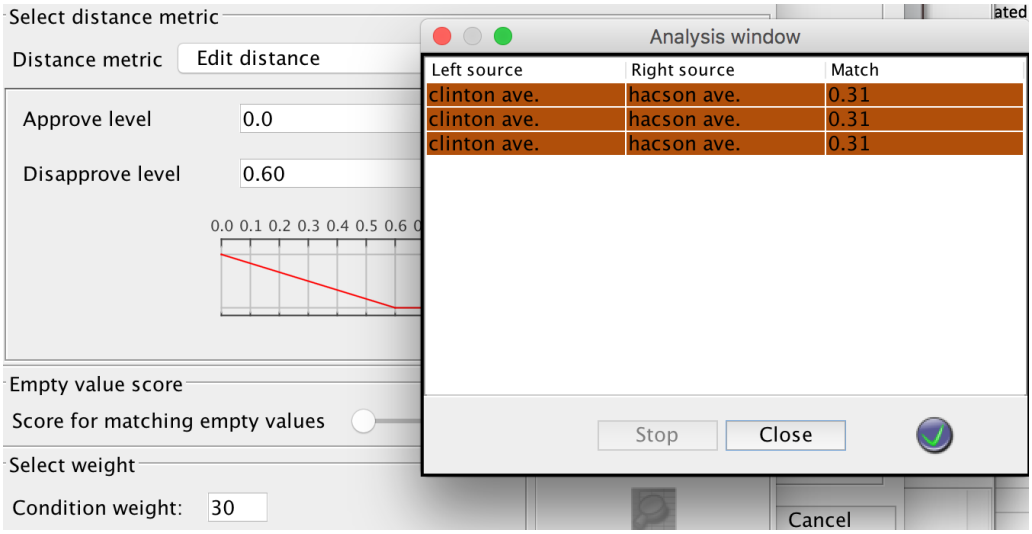

Figure 18 Matching score with abbreviation

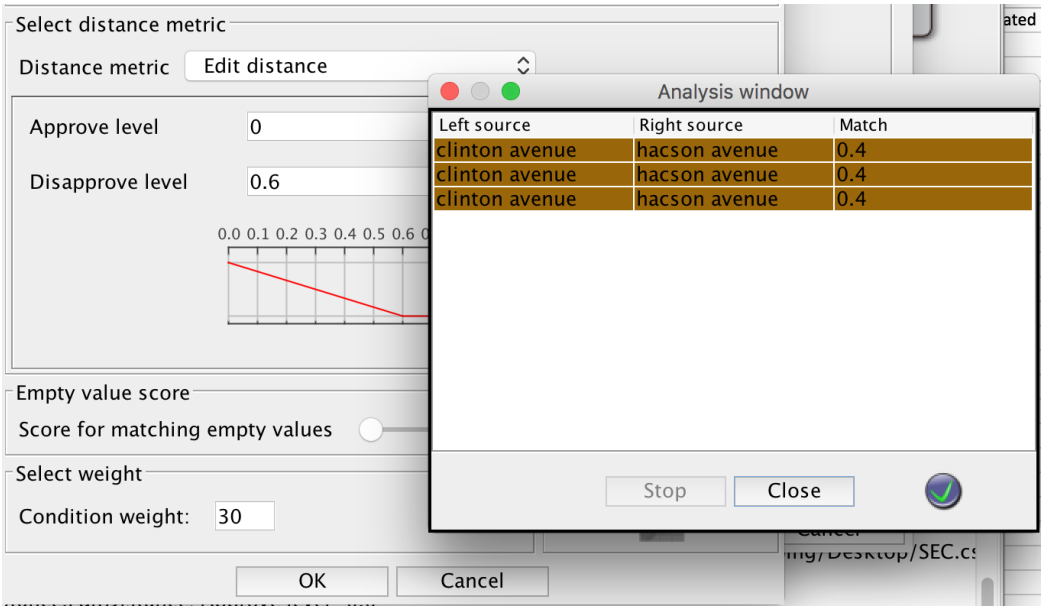

Figure 19 Matching score without abbreviation

- The sequence of steps of the cleaning strategy needs to be followed strictly. For example, removal of symbols needs to be performed after the standardization of words, because symbols might be part of one of the representations.
- To identify the key words implying the nature of financial entities, only pick up frequently used ones. Examine each key word by checking the number of entities of which the word shows the nature. If there is only very limited number of entities, then we abandoned it from the set of key words, otherwise it becomes the candidate. This actually involves domain knowledge, that we acquired from studying the decisions made by experts to construct the sample ground truth.

4.2.6 Results

This chapter describes the results of data preprocessing and also the record matching. The results of data preprocessing have been talked about field by field and the result of record matching have been talked about on the basis of relevant measurements including precision, recall and F-Score.

4.2.6.1 Data preprocessing results

FFIEC

There are 6652 rows of financial entities in FFIEC data file.

Zip Code

- By custom text facet by length, there are 372 financial entities with less than 5 digits. The GREL expression is 'length(value)'.

Figure 20 Custom text facet by length on Zip Code

- The zip code of the 372 entities have been added with leading zeroes so that they were standardized. Now the zip code of all financial entities are with standard format. The GREL expression is '"0"[0,5-length(value)] + value'.

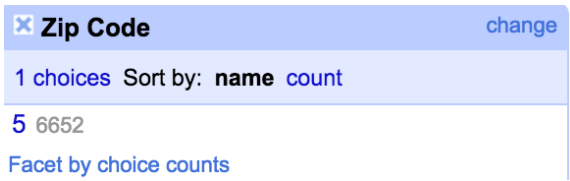

Figure 21 Zip code has all been standardized

STATE

- By customized facet by blank, it showed no financial entity with empty state.

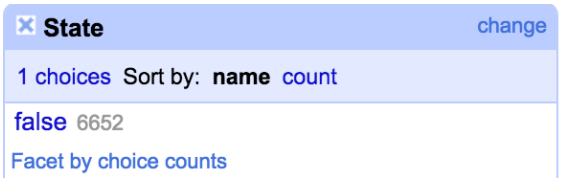

Figure 22 Customized facet with blank on State

- By custom text facet by length of the value, it showed there was one financial entity with state '0'. So there is 1 record has been populated with state value.

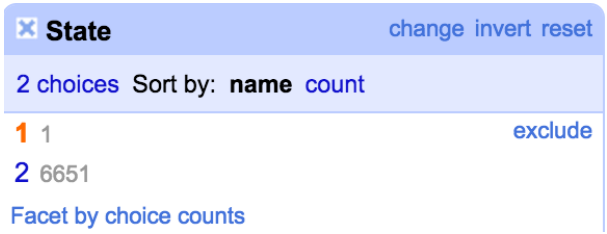

Figure 23 Custom text facet by length on State

STREET

- Replaced all street and orientation words by corresponding abbreviations. As a result, 5475 entities have been processed.
  The GREL expression is like below:
  'value.replace("NORTHEAST","NE").replace("NORTHWEST","NW").replace("SOUTHEAST","SE").replace("SOUTHWEST","SW").replace("EAST","E").replace("WEST","W").replace("SOUTH","S").replace("NORTH","N").replace(" STREET"," ST").replace(" ROAD"," RD").replace(" AVENUE"," AVE").replace(" SQUARE"," SQ").replace(" LANE"," LA").replace(" SUITE"," SU").replace(" PLAZA"," PL").replace("#","NO")'
- Removed all symbols. As a result, 776 entities have been processed. Below is the GREL expression:
  'value.replace("/","").replace(".","").replace(",","").replace("-","").replace("&","").replace("'","").replace("(","").replace(")","")'
- Collapsed consecutive whitespaces. As a result, 53 entities have been processed.

Name

- Replace the list of key words by their abbreviations. As a result, 2955 records have been processed.
  The GREL expression is like below:
  'value.replace("NATION","NA").replace("ASSOCIATION","ASSO.").replace("CAPITAAL","CAP.").replace("COMMUNITY","COM.").replace("CONSTITUTION","CON.").replace("FINANCIAL","FIN.").replace("SECURITY","SEC").replace("FEDERAL","FED").replace("SAVINGS","SAV.").replace("CHARTERED","CHA.").replace("COMPANY","CO.").replace("EXCHANGE","EXC.").replace("COMMERCE","COR.").replace("COMMERCIAL","COR.").replace("INCORPORATIVE","INC.").replace("COOPERATIVE","CORP."). replace("CO-OPERATIVE","CORP.").replace("LIMITED","LTD.")'

- Replaced function words by whitespace. As a result, 2237 records have been processed.
  The GREL expression is like below:
  ‘value.replace(" OF "," ").replace("THE "," ").replace(" AND "," ").replace(" AT "," ")
- Replace symbols by whitespace as what we do to field STREET. As a result, 2334 records have been processed.

SEC

The SEC data file consists of 129312 rows of entities. For reusing the scripts generated during the preprocessing for FFIEC file, we renamed the SEC fields as follows.

Original name	New name
CONFIRMED NAME	Financial Institution Name Cleaned
B_STREET	Financial Institution Address
B_STPR	Financial Institution State
B_POSTAL	Financial Institution Zip Code 5

Zip Code

- Blank out ‘0’ zip code and 23 records have been processed.
- Blank out the fake zip code and 698 records have been processed.
- Add leading zeroes to zip code with less than 5 digits. As a result, 12337 records have been processed.
- Trim zip code with more than 5 digits. As a result, 176 records have been processed.
  The GREL expression for categorizing zip code by length is ‘length(value)’.
  The GREL expression for trimming zip code is ‘"00"[0,5-length(value)] + value’.

STATE

- By customized facet by blank, it showed 1551 entities with empty state and they are populated with state value by calling the API.

Next, for fields ‘STREET’ and ‘NAME’, we simply applied the transformation scripts generated from FFIEC preprocessing. After this we examined both fields and did a further cleaning on them like below.

STREET

Issues:

- symbols still exist: ‘%’

Further cleaning:

- Removed symbol and as a result, 7 records have been cleaned.

NAME

Issues:

- Symbols still exist: ‘!’, ‘$’, ‘*’, ‘?’

Further cleaning:

- Removed symbols and as a result 330 records have been processed.

4.2.6.2 Record linkage results

As we mentioned in chapter 4.2.3.2, six matching results have been generated in total:

- Baseline matching result. this result has been generated from record linkage task upon the un-preprocessed FFIEC and SEC data files.
- Experimental matching results.
    - Matching result after cleaning Name: it was generated only considering the cleaning on Name field.
    - Matching result after cleaning Street: it was generated only considering the cleaning on Street field.
    - Matching result after cleaning State: it was generated only considering the cleaning on State field.
    - Matching result after cleaning Zip Code: it was generated only considering the cleaning on Zip Code field.
    - Matching result after cleaning all fields: it was generated considering the completely cleaned FFIEC and SEC files.

  Both the baseline matching result and the experimental results have employed the same combination of FRIL parameter values proposed in chapter 4.2.2.

As what we mentioned before, for evaluating the performance of the record linkage, FFIEC challenge organizers have provided the final ground truth result. Table 2 is a summary of the relevant statistics of the baseline matching result, five experimental matching results and the final ground truth.

(Ground truth matches = 885)	Positive matches	True positive matches	Precision	Recall	F-Score	Time Consumed (ms)
Baseline result	263	247	0.94	0.28	0.43	39242
Experimental result (Name)	331	304	0.92	0.34	0.50	32716
Experimental result (Street)	274	256	0.93	0.29	0.44	34422
Experimental result (State)	264	248	0.94	0.28	0.43	34822
Experimental result (Zip Code)	270	254	0.94	0.29	0.44	33591
Experimental result (All cleaned)	369	332	0.90	0.38	0.53	20593

Table 2 The summary of Record Matching results

5 Discussion

In the aspect of the time consumed by running the record matching task, from Table 2 all experimental results spent less time than the baseline result. The difference ranges from 4420 to 18649 ms. Especially the time spent on matching the fully preprocessed data file is substantially lower than the time spent on matching the un-preprocessed or partially preprocessed data files. According to our cleaning strategy and the preprocessing results described in chapter 4.2.6.1, two main factors influence the execution time:

- The number of records in each input data file. The identified non-financial SEC entities have been removed from the file and are therefore not considered for matching.
- The number of words in each row of the input files. During the preprocessing, the number of words to be compared for matching has been reduced due to:
    - Removal of function words
    - Removal of symbols
    - Abbreviations of entity types. The entity type key words have been abbreviated and therefore the length of the word become smaller. This has not trivial impact on the matching because the matching algorithm we have chosen is edit distance for which the length of the string values to be compared matters a lot.

We used the final ground truth data provided by FEIII challenge organizers to calculate all the measurement values. In Table 2, it is the summary of all the measurement values. Below we discussed each measurement one by one and also the positive matches and true positive matches:

- Positive matches. These are the matches generated by the FRIL. Compared with baseline result, all experiment results contain more positive matches. This shows to us the cleaning on field value is helpful for generating more matches. Among the experimental results, the numbers of positive matches of the 3rd, 4th and 5th experiments are just slightly more than that of the baseline experiment. On the contrary, the 2nd and the 6th experiments returned much more positive matches.
- True positive matches. The matches are the true matches out of the generated ones. By checking the number of true positive matches, we got to know the experiment results generated more true positive matches. And similar as the number of positive matches, results based on ‘Name cleaned’ and ‘All cleaned’ data files are very noticeable compared with the rest of experimental results.
- Precision. It focuses on the number of true matches out of the generated matches. A high precision value means a high accuracy of the matching result. Precision depends on both the number of true matches and the number of generated matches. In general, the precision values of each matching result just vary a little. This is because for experimental results, they have more true positive matches but also more positive matches. So in terms of precision, we did not see a big contribution from the data cleaning.

- Recall. Unlike precision, recall pays more attention to the number of true matches out of the whole set of generated matches. According to the formula, the value of recall depends on the number of the true matches and also the number of matches. In our case, the number of matches is fixed and that is the number of matches contained in the ground truth file. So recall now turns to only rely on the number of the true positive matches among the generated matches. From Table 2 we see all the experimental results having more true positive matches than what the baseline result has. This is similar with the situation of the 'true positive matches'. The recall of the 3rd, 4th, and 5th result is more or less the same. Nevertheless, the recall of the 2nd and the 6th experimental result is 6% and 10% higher than that of the baseline result. Given this fact, we concluded data cleaning contributes more in terms of recall.

- F-Score. It considers both the precision and recall of the matching to compute the score. The formula is interpreted as a weighted average of the precision and recall. F-Score grades the matching compromising on both precision and recall. According to the formula, the F-score of experimental result is with the situation close to the recall and the true positive matches measurements. But it shows again the data cleaning on field 'Name' did the biggest contribution to the record matching since the F-Score of the 2nd result and the 6th result have very small difference.

When evaluating a matching task, the attention on precision or recall varies in different cases. In our case, since the matching is for identify the identical financial entities and therefore the integration of financial information, so the precision and recall is the same important. Because if the precision is overlooked, the financial information integration is wrongful; if the recall is unnoticed, information integration only applies to part of the available information. So it makes sense to evaluate our case by F-score which considers both precision and recall.

Overall, the matching reaches a high value on precision. This does not owe a lot to data cleaning because the baseline result has also a very high precision. So we think this owes more to the record linkage tool FRIL and the combination of parameter values we have decided. On the other hand, the recall values are low, but since what we emphasized in the beginning that our intention is not achieving a high recall but to evaluate how much data cleaning can contribute, we focus on the increase of the recall after data cleaning. The experimental results support the correctness of our data cleaning strategy and its contribution to the record matching.

6 Conclusion

The entire experiment has been operated on OpenRefine. The more we dealt with the tool, the more practical we felt the tool is. Compared with Trifacta, the user interface of OpenRefine is less fancy but simple and clean. Every function is very accessible without switching from here to there, and this is unlike Trifacta when we operate data on it. OpenRefine allows multiple windows open at the same time, and this is very convenient for users when they want to operate multiple data files. However, Trifacta only permits users to operate a single file and whenever users want to go to the directory of data files, Trifacta forces users to suspend the current data transformation. The most impressive feature of OpenRefine is that all transformation scripts can be easily applied to other data files partially or completely, while Trifacta has no such feature. So in the context of the FEIII challenge and our research, OpenRefine is much more suited than Trifacta.

With OpenRefine, data cleaning can be performed with no trouble. Considering the available features of OpenRefine, we made the data cleaning strategy taking the data analysis into account as well. Data analysis is very essential since it provides insights into the data itself which aids the formulation of the data cleaning strategy. When cleaning the data, it is important to follow the steps sequentially, otherwise wrongful cleaning might take place. Also, the data validation after each step is also very necessary because it helps a lot to avoid wrong data transformation.

From Table 2 we conclude that the data cleaning helps to increase the recall while has no noticeable impact on precision. Also, in terms of the time spent on the matching, data cleaning decreased the time dramatically mainly because the data files have been narrowed down the scope and also the length of the fields have been shortened. The data cleaning removed the function words and symbols interfering the matching, eliminated the spelling errors, and unified the representation of the same object. All these have great impact on the matching score calculation because in our case the primary matching algorithm employed is based on string similarity.

At last, with the help of Data Wrangling tool and provided data files, we have proved that data cleaning increases the efficiency of record linkage in both aspects of the recall and the time spent on the task.

7 Further work

From the experiment results and the measurement values in Table 2, we have concluded that the recall and F-score have been increased a lot. However, the recall value is still low. Which means there are still many matches have not been fetched. The reason is primarily because the FRIL parameter values are not or not well tuned. In our research, due to the lack of training data, we did not tune the FRIL parameter values. The parameter values were just concluded based on the data analysis and our knowledge about the tool. In our research, with this set of parameter values, FRIL generated matching results with high precision but low recall. This also implies that the matching conditions might be over-restricted. So, as the future work, we think it is worth continuing the FFIEC challenge in the aspect of the matching algorithm and the matching decision model.

8 Reference

[1] FEIII organizers, (2015, Dec), Challenge Guidelines and Rules, Available: https://ir.nist.gov/dsfin/guidelines.html

[2] H. Köpcke and E. Rahm, "Frameworks for entity matching: A comparison," *Data Knowl. Eng.*, vol. 69, no. 2, pp. 197–210, 2010.

[3] P. Jurczyk, J. J. Lu, L. Xiong, J. D. Cragan, and A. Correa, "FRIL: A tool for comparative record linkage.," *AMIA Annu. Symp. Proc.*, pp. 440–444, 2008.

[4] F. Endel and H. Piringer, "Data Wrangling: Making data useful again," *IFAC-PapersOnLine*, vol. 48, no. 1, pp. 111–112, 2015.

[5] D. Goldston, "Big data: Data wrangling," *Nature*, vol. 455, no. 7209, pp. 15–15, 2008.

[6] S. van Hooland, R. Verborgh, and M. De Wilde, "Cleaning Data with OpenRefine," in *The Programming Historian*, 2013.

[7] Tye. Rattenbury, (2015, September, 14), Six Core Data Wrangling Activities, Available: http://www.datanami.com/2015/09/14/six-core-data-wrangling-activities/

[8] Pawel. Jurczyk, (2009, July, 08), FRIL Tutorial, Available: http://fril.sourceforge.net/FRIL-Tutorial-3.2.pdf

[9] S. M. Randall, A. M. Ferrante, J. H. Boyd, and J. B. Semmens, "The effect of data cleaning on record linkage quality.," *BMC Med. Inform. Decis. Mak.*, vol. 13, no. 1, p. 64, 2013.

[10] FFIEC Bulk Data Download, Available: http://www.ffiec.gov/nicpubweb/nicweb/DataDownload.aspx

[11] Andy. Green, (2015-04-22), Seven Free Data Wrangling Tools, Available: https://blog.varonis.com/free-data-wrangling-tools/

[12] SEC and Market Data, Available: https://www.sec.gov/data

[13] FFIEC Bulk Data Download Data Dictionary and Reference Guide, Available: http://www.ffiec.gov/nicpubweb/content/DataDownload/NPW%20Data%20Dictionary.pdf

[14] D. G. Brizan and A. U. Tansel, "A Survey of Entity Resolution and Record Linkage Methodologies," *Commun. IIMA*, vol. 6, no. 3, pp. 41–50, 2006.

[15] W. E. Winkler, "Matching and record linkage," *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 6, no. 5, pp. 313–325, 2014.

[16] Z.-M. . Guo and A.-Y. . Zhou, "Research on data quality and data cleaning: A survey," *Ruan Jian Xue Bao/Journal Softw.*, vol. 13, no. 11, pp. 2076–2082, 2002.

[17] T. N. Herzog, F. J. Scheuren, and W. E. Winkler, *Data quality and record linkage techniques*. 2007.

[18] P. Jurczyk, J. J. Lu, L. Xiong, J. D. Cragan, and A. Correa, "Fine-grained record integration and linkage tool," *Birth Defects Res. Part A - Clin. Mol. Teratol.*, vol. 82, no. 11, pp. 822–829, 2008.

[19] Marketwired, "Launch of Trifacta Wrangler Brings Award-Winning Data Wrangling Solution to the Desktop Free of Charge.," *Marketwire (English)*. 2015.

[20] F. Donnelly, G. D. Librarian, and B. C. CUNY, "Processing Government Data: ZIP Codes, Python, and OpenRefine," *Code4Lib J.*, no. 25, 2014.

[21] Github.com, General Refine Expression Language, Available: https://github.com/OpenRefine/OpenRefine/wiki/General-Refine-Expression-Language

[22] EDGAR Header Dump for OFR, (2014-12-19), Available: https://drive.google.com/file/d/0B88Ya9t25EGHRUJjSnBRWHVuMGM/view

9 Appendix

Original	Abbreviation
NORTHEAST	NE
NORTHWEST	NW
SOUTHEAST	SE
SOUTHWEST	SW
EAST	E
WEST	W
SOUTH	S
NORTH	N
N.E.	NE
N.W.	NW
S.E.	SE
S.W.	SW
E.	E
W.	W
S.	S
N.	N

Table 3 Orientation words and their abbreviations

Original	Abbreviation
STREET	ST
ROAD	RD
AVENUE	AVE
SQUARE	SQ
LANE	LA
SUITE	SU
PLAZA	PL
ST.	ST
RD.	RD
AVE.	AVE
NO.	NO
U.S.	US
USA	US
U.S.A.	US
#	NO

Table 4 Road words and their abbreviations

CITY	CITY
CITY,	
(U.S.A.)	USA
(USA)	
PARIS	PARIS
PARIS,	
COMPANY	COMPANY
COMPANY,	
CO.	CO.
CO.,	
COUNTY	COUNTY
COUNTY,	
SOUTH	SOUTH
SOUTH,	
INCORPORATED	INCORPORATED
(INCORPORATED)	
ASSOCIATION	ASSOCIATION
ASSOCIATION,	
ASSOICATION	
CENTER	CENTER
CENTRE	
BANK	BANK
BANK,	
TEXAS	TEXAS
TEXAS,	
CENTRAL	CENTRAL
CENTRAL,	
CAROLINA	CAROLINA
CAROLINA,	
SAVINGS	SAVINGS
SAVINGS,	
INC	INC.
INC.	
VIRGINIA	VIRGINIA
VIRGINIA,	
ARLINGTON	ARLINGTON
ARLINGTON,	
CALIFORNIA	CALIFORNIA
(CALIFORNIA)	
TRUST	TRUST
TRUST,	
NEWTON	NEWTON

NEWTON,	
DELAWARE	DELAWARE
DELAWARE,	
KENTUCKY	KENTUCKY
KENTUCKY,	
FLORIDA	FLORIDA
FLORIDA,	
LOUISIANA	LOUISIANA
LOUISIANA,	
MILLS	MILLS
MILLS,	
PEOPLES	PEOPLES
PEOPLES'	
PEOPLE'S	
CITIZENS	CITIZENS
CITIZENS'	
CITIZEN'S	
INTERSTATE	INTERSTATE
INTER-STATE	
BANKWEST	BANKWEST
BANKWEST,	
MID-AMERICA	MIDAMERICA
MIDAMERICA	
E*TRADE	ETRADE
ETRADE	
TRISTATE	TRISTATE
TRI-STATE	

Table 5 List of words grouped and standardized

Key words	Abbreviation
BANKING	
BANKER	
BANCORP	
BANCO	
ASSOCIATION	ASSO
INC	
NATIONAL	NA
SAVINGS	SAV
CHARTERED	CHA
CO	
COMPANY	CO
FEDERAL	FED

LOAN	LOAN
SSB	
STATE	
CORP	
TRUST	
US	
COMMUNITY	COM
FSB	
GROUP	
INSTITUTION	INS
INTERNATIONAL	INT
LLC	
LTD	
MORTGAGE	MOR
S/B	
COMMERCE	COR
CREDIT	
DEPOSIT	
EXCHANGE	EXC
FINANCE	FIN
FUND	
INVESTMENT	INV
PRIVATE	
SUMMIT	
CHASE	
CAPITAL	CAP
CONSTITUTION	CON
EQUITY	
SECURITY	SEC
UNITED	
FED	
COOPERATIVE	CORP
CO-OPERATIVE	CORP
LIMITED	LTD

Table 6 Key words indicating the nature of entities and their abbreviations

Part of key data cleaning operations as JSON.

Zip code

- Add leading zeroes to zip code which has less than 5 digits.

```
[{
    "op": "core/text-transform",
    "description": "Text transform on cells in column Zip Code using expression grel:\"00\"[0,5-length(value)] + value",
    "engineConfig": {
      "facets": [],
      "mode": "row-based"
    },
    "columnName": "Zip Code",
    "expression": "grel:\"00\"[0,5-length(value)] + value",
    "onError": "keep-original",
    "repeat": false,
    "repeatCount": 10
  }]
```

Street

- Replace orientation and road words by their abbreviations

```
[{
    "op": "core/text-transform",
    "description": "Text transform on cells in column Street using expression grel:value.replace(\"NORTHEAST\",\"NE\").replace(\"NORTHWEST\",\"NW\").replace(\"SOUTHEAST\",\"SE\").replace(\"SOUTHWEST\",\"SW\").replace(\"EAST\",\"E\").replace(\"WEST\",\"W\").replace(\"SOUTH\",\"S\").replace(\"NORTH\",\"N\").replace(\" STREET\",\" ST\").replace(\" ROAD\",\" RD\").replace(\" AVENUE\",\" AVE\").replace(\" SQUARE\",\" SQ\").replace(\" LANE\",\" LA\").replace(\" SUITE\",\" SU\").replace(\" PLAZA\",\" PL\").replace(\"#\",\"NO\")",
    "engineConfig": {
      "facets": [],
      "mode": "row-based"
    },
    "columnName": "Street",
    "expression": "grel:value.replace(\"NORTHEAST\",\"NE\").replace(\"NORTHWEST\",\"NW\").replace(\"SOUTHEAST\",\"SE\").replace(\"SOUTHWEST\",\"SW\").replace(\"EAST\",\"E\").replace(\"WEST\",\"W\").replace(\"SOUTH\",\"S\").replace(\"NORTH\",\"N\").replace(\" STREET\",\" ST\").replace(\" ROAD\",\" RD\").replace(\" AVENUE\",\" AVE\").replace(\" SQUARE\",\" SQ\").replace(\" LANE\",\" LA\").replace(\" SUITE\",\" SU\").replace(\" PLAZA\",\" PL\").replace(\"#\",\"NO\")",
    "onError": "keep-original",
    "repeat": false,
```

```
    "repeatCount": 10
  }]
```


- Remove all symbos


```
[{
    "op": "core/text-transform",
    "description": "Text transform on cells in column Street using expression
grel:value.replace(\"/\",\"\").replace(\".\",\"\").replace(\",\",\"\").replace(\"-
\",\"\").replace(\"&\",\"\").replace(\"'\",\"\").replace(\"(\",\"\").replace(\")\",\"\")",
    "engineConfig": {
      "facets": [],
      "mode": "row-based"
    },
    "columnName": "Street",
    "expression":
"grel:value.replace(\"/\",\"\").replace(\".\",\"\").replace(\",\",\"\").replace(\"-
\",\"\").replace(\"&\",\"\").replace(\"'\",\"\").replace(\"(\",\"\").replace(\")\",\"\").replac
e(\"%\",\"\").
    "onError": "keep-original",
    "repeat": false,
    "repeatCount": 10
  }]
```


Name

- Replace key words by their abbreviation


```
[{
    "op": "core/text-transform",
    "description": "Text transform on cells in column Entity Name using expression
grel:value.replace(\"NATIONAL\",\"NA\").replace(\"ASSOCIATION\",\"ASSO.\").r
eplace(\"CAPITAL\",\"CAP.\").replace(\"COMMUNITY\",\"COM.\").replace(\"CON
STITUTION\",\"CON.\").replace(\"FINANCIAL\",\"FIN.\").replace(\"SECURITY\",\
"SEC\").replace(\"FEDERAL\",\"FED\").replace(\"SAVINGS\",\"SAV.\").replace(\"
CHARTERED\",\"CHA.\").replace
(\"COMPANY\",\"CO.\").replace(\"EXCHANGE\",\"EXC.\").replace(\"COMMERC
E\",\"COR.\").replace(\"INCORPORATIVE\",\"INC.\").replace(\"COOPERATIVE\",
\"CORP.\").replace(\"CO-
OPERATIVE\",\"CORP.\").replace(\"LIMITED\",\"LTD.\")",
    "engineConfig": {
      "facets": [],
      "mode": "row-based"
    },
    "columnName": "Entity Name",
    "expression":
"grel:value.replace(\"NATIONAL\",\"NA\").replace(\"ASSOCIATION\",\"ASSO.\").
replace(\"CAPITAL\",\"CAP.\").replace(\"COMMUNITY\",\"COM.\").replace(\"CO
```

NSTITUTION\",\"CON.\").replace(\"FINANCIAL\",\"FIN.\").replace(\"SECURITY\",\"SEC\").replace(\"FEDERAL\",\"FED\").replace(\"SAVINGS\",\"SAV.\").replace(\"CHARTERED\",\"CHA.\").replace(\"COMPANY\",\"CO.\").replace(\"EXCHANGE\",\"EXC.\").replace(\"COMMERCE\",\"COR.\").replace(\"INCORPORATIVE\",\"INC.\").replace(\"COOPERATIVE\",\"CORP.\").replace(\"CO-OPERATIVE\",\"CORP.\").replace(\"LIMITED\",\"LTD.\").replace(\"COMMERCIAL\",\"COR.\")",
"onError": "keep-original",
"repeat": false,
"repeatCount": 10
}]

- Remove all function words

[{
"op": "core/text-transform",
"description": "Text transform on cells in column Entity Name using expression grel:value.replace(\" OF \",\" \").replace(\"THE \",\" \").replace(\" AND \",\" \") value.replace(\"!\",\"\").replace(\"@\",\"\").replace(\"#\",\"\").replace(\"$\",\"\").replace(\"*\",\"\").replace(\"?\",\"\")",
"engineConfig": {
"facets": [],
"mode": "row-based"
},
"columnName": "Entity Name",
"expression": "grel:value.replace(\" OF \",\" \").replace(\"THE \",\" \").replace(\" AND \",\" \")",
"onError": "keep-original",
"repeat": false,
"repeatCount": 10
}]

- Remove all symbols

[{
"op": "core/text-transform",
"description": "Text transform on cells in column Entity Name using expression grel:value.replace(\"/\",\"\").replace(\".\",\"\").replace(\",\",\"\").replace(\"-\",\"\").replace(\"&\",\"\").replace(\"'\",\"\").replace(\"(\",\"\").replace(\")\",\"\")",
"engineConfig": {
"facets": [],
"mode": "row-based"
},
"columnName": "Entity Name",
"expression": "grel:value.replace(\"/\",\"\").replace(\".\",\"\").replace(\",\",\"\").replace(\"-\",\"\").replace(\"&\",\"\").replace(\"'\",\"\").replace(\"(\",\"\").replace(\")\",\"\")",

```
    "onError": "keep-original",
    "repeat": false,
    "repeatCount": 10
  },
  {
    "op": "core/text-transform",
    "description": "Text transform on cells in column Entity Name using expression
grel:value.replace(\"!\",\"\").replace(\"@\",\"\").replace(\"#\",\"NO\").replace(\"$\",\"\"
).replace(\"*\",\"\").replace(\"?\",\"\")",
    "engineConfig": {
      "facets": [
        {
          "query": "",
          "name": "Entity Name",
          "caseSensitive": false,
          "columnName": "Entity Name",
          "type": "text",
          "mode": "text"
        }
      ],
      "mode": "row-based"
    },
    "columnName": "Entity Name",
    "expression":
"grel:value.replace(\"!\",\"\").replace(\"@\",\"\").replace(\"#\",\"NO\").replace(\"$\",\"\
").replace(\"*\",\"\").replace(\"?\",\"\")",
    "onError": "keep-original",
    "repeat": false,
    "repeatCount": 10
  }]
```