# Extensible and Scalable Entity Resolution for Financial Datasets Using RLTK

Yixiang Yao
yixiangy@isi.edu
Information Sciences Institute
Marina del Rey, CA

Pedro Szekely
pszekeley@isi.edu
Information Sciences Institute
Marina del Rey, CA

Jay Pujara
jpujara@isi.edu
Information Sciences Institute
Marina del Rey, CA

**ACM Reference Format:**
Yixiang Yao, Pedro Szekely, and Jay Pujara. 2019. Extensible and Scalable Entity Resolution for Financial Datasets Using RLTK. In *The 5th Workshop on Data Science for Macro-modeling with Financial and Economic Datasets (DSMM'19), June 30, 2019, Amsterdam, Netherlands.* ACM, New York, NY, USA, 1 page. https://doi.org/10.1145/3336499.3338008

## 1 INTRODUCTION

The problem of entity resolution is a well-studied task that spans decades of research. However, real-world applications, such as matching entities across financial datasets, present many challenges that make entity resolution difficult. The FEIII 2019 challenge presents a complex entity matching task that includes millions of entities with signficiant name and location ambiguity, heterogeneous source structure, and many domain-specific attributes. To address these challenges, we have developed the Record Linkage Toolkit[1] which provides an easy-to-use, feature-rich framework enabling scalable entity resolution and rapid prototyping of novel task-specific features. RLTK implements most prominent sequence and set-based similarity measures, efficiently performs blocking using external sorts to dramatically scale the number of entities, and computes evaluation metrics that provide insight into entity resolution performance. In this abstract, we highlight several challenges from the FEIII challenge and how they are addressed by RLTK.

## 2 CHALLENGES

- **Schema**: Each dataset is structured differently, e.g., addresses in a single column versus split across many columns
- **Representation**: Abbreviations, prefixes, suffixes, directional and ordinal specification vary, e.g. I-5 W/S 100FT SW OF CASTELLA INT B
- **Ambiguity**: False pairs look more similar than true pairs, e.g., Kingsburg Buddhist Church vs Kingsburg Church of Christ.
- **Scale**: The FEIII challenge datasets have between 6K and 4M entities. A full, pairwise similarity comparison would generate over a trillion comparisons
- **Errors**: Datasets contain many typographical errors and missing information, e.g., zip codes, special characters

RLTK provides a comprehensive system for handling these challenges.

| Task | Cmps | Cmps (Blocking) | Baseline | Matches |
|---|---|---|---|---|
| CCR vs SFO | 23B | 63M | 139 | 761 |
| CCR vs LIQ | 353B | 350M | 3530 | 15,821 |
| CCR vs TAX | 716B | 808M | 14,354 | 40,673 |
| SFO vs LIQ | 575M | 1.4M | 75 | 376 |
| SFO vs TAX | 1.2B | 1.7M | 39 | 79 |

**Table 1: Preliminary results**

## 3 APPROACH

The RLTK API provides a common abstraction of each dataset. Using this abstraction allows the schema properties of each record to be mapped into a common feature space, allowing comparisons of records without a per-task schema mapping. Common utilities assist in normalization tasks, such as identifying the components of an address. These utilities use conditional random field (CRF) models for segmentation and error correction. RLTK implements a vast library of similarity functions that can be tuned to specific domains (e.g., addresses) or are broadly applicable to a large class of strings.

The primary concern for many real-world tasks is scale. Large-scale entity resolution systems may require deployment on a distributed computing cluster that can be costly to operate and maintain. RLTK is optimized to handle cluster-sized workloads on a single machine. One innovation to support this scaling is the abstraction of different data storage choices into an RLTK "Adapter" which allows a simple method to assign features to different storage subsystems (e.g., memory or disk) based on the size and performance. In addition, RLTK supports a set of efficient blocking operations, which group candidate entity pairs together while eliminating the vast majority of comparisons and boosting precision. Using multiple blocking operators, RLTK maintains high recall. RLTK also has a human-friendly parallel processing module, with which few lines of code can maximize computing resource utilization.

To demonstrate the efficiency of RLTK-based entity resolution, we used blocking operations on the FEIII Challenge dataset. Table 1 contains an analysis of the raw comparisons required for a naive pairwise algorithm, the number of comparisons made by RLTK, the number of exact-match pairs and the number of possible matches.

[1] https://github.com/usc-isi-i2/rltk

*DSMM'19, June 30, 2019, Amsterdam, Netherlands*
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6823-0/19/06…$15.00
https://doi.org/10.1145/3336499.3338008