The problem that I have been trying to solve is how to effectively deduplicate datasets for companies without Master Data Management (MDM) [Define this]. This report consists of an overview of the problem, the steps I took to solve the problem, and still-existing solution concerns.

**Overview of the Problem**

The need for data deduplication has been around since the 1970s (pcdreams). This could have been caused by inconsistencies (e.g. a person appears with multiple addresses), typos, bad data structures, lack of unique identifiers, or many other human-or-non-human-related issues. For this particular project that I have been working on, data deduplication is needed as the most fundamental function for Customer Data Platform (CDP), a software that aggregates and organizes customer data across a variety of touchpoints (Decker). CDP is primarily used by the marketing department with the purpose to build a 3-D profile of its customers. Thus, without the basic functionality of being able to identify the same customer from different sources, CDP is useless.

**Steps I Took to Solve the Problem**

First, I identified the basic requirements of a solution that may solve our problem. The program has to be cost-effective, able to handle both B2B and B2C [terms should be defined] datasets, accurate, and flexible. B2B and B2C’s datasets are very different from each other; for example, depending on the sources of the datasets, B2C’s datasets can be much larger and contains fewer errors such as datasets from online orders where information is entered by customers themselves. Also, one data set often contains many different fields, such as addresses, phone numbers, names, dates, etc. So the program must be flexible and able to handle different field types.

After identifying the requirements, I began to look into different existing solutions on the market. I first ruled out enterprise solutions, such as SAS Dataflux and IBM qualityStage. They are extremely expensive and come with a bundle of functionality that does not fit the solution I want. And then I crossed off solutions that require third-party’s cloud, such as dedupe.io and PIMcore. After several more steps of eliminations (complicated workflows, inaccurate results, and awful technical support), I ended up with two options, dedupe and AWS [Amazon Web Services] Glue.

Techwave has already been using AWS console and AWS Glue for different purposes, so it is a huge advantage that we have people who know how to use AWS Glue. Other advantages of AWS Glue are its capability to implement with AWS Lake and other AWS solutions, simplicity to use, and its user interface.

However, I do think that dedupe is a solution that suits our needs better. First, either you want to customize your own field types or create new functionalities, Dedupe is much more flexible than AWS Glue as it is a python library. Also, it utilizes machine learning to identify duplicates based on training tailored to the dataset, and clusters the duplicates to provide a more accurate result than AWS Glue (To see the advantage of using AWS Glue, visit step 78 and 79 of [Flavio’s slide](https://vintasoftware.github.io/deduplication-slides/)s).

In spite of Dedupe’s advantages over AWS Glue, it does have a few drawbacks. Dedupe requires a very deep understanding of Python and the library itself to be able to use its customizability to full advantage. It also takes much longer to write the program and has a longer run-time. However, those disadvantages are overshadowed by the accuracy Dedupe provides.

I have included my steps of implementing data normalizations and Dedupe in three separate reports.

**Still-Existing Problems**

Although both Dedupe and AWS Glue can identify duplicates, I still haven’t completely solved the problem yet. There are still some problems that exist.

* There is always a trade-off between [precision and recall](https://en.wikipedia.org/wiki/Precision_and_recall).
  + As good as machine learning is, neither Dedupe nor AWS Glue can achieve 100% accuracy. There need to be additional steps being taken to limit false positives and false negatives.
  + [Dedupe.io](https://dedupe.io/) has additional 3 steps to do this. It provides a great framework.
* Link different datasets.
  + Although Dedupe has a command called csvlink in the CSVDedupe library, it uses an older version of dedupe and cannot be run on large datasets.
* Ability to update the datasets with new data.

This project is designed to solve the issue of duplications in datasets by finding a machine learning alternative to customer Master Data Management for companies without the investments. Although Dedupe and AWS Glue do not completely solve the problem, they, especially Dedupe, have great potential to become a mature solution that can be presented to clients. This solution can become a program that accurately identifies and sufficiently consolidates duplicates from different datasets, and serves as the foundation of a CDP solution.

**The Future**

I recommend choosing Dedupe over AWS Glue because of its flexibility, and of the possibility to build a solution that is fully centered around Dedupe. The developers of Dedupe provides [consulting services](https://dedupe.io/pricing/#consulting) at the rate of $250/hr with a minimum retainer of 10 hours. This is a good option if we are looking for a shorter process and have the budget, especially that Dedupe.io is almost the exact solution we are looking for. Another option is to have someone who is an expert in python to develop an advanced program around Dedupe based on the framework of Dedupe.io. Either one of these two options require to have someone who has great knowledge in python, MySQL, and data architecture.

pcdreams. “The History Of Data Deduplication.” *PC Dreams*,15 May 2016, pcdreams.com.sg/the-history-of-data-deduplication/.

Decker, Allie. “What's a Customer Data Platform? The Ultimate Guide to CDPs.” *HubSpot Blog*, 12 Mar. 2019, blog.hubspot.com/service/customer-data-platform-guide#:~:text=A%20Customer%20Data%20Platform%20(CDP,into%20individual%2C%20centralized%20customer%20profiles.