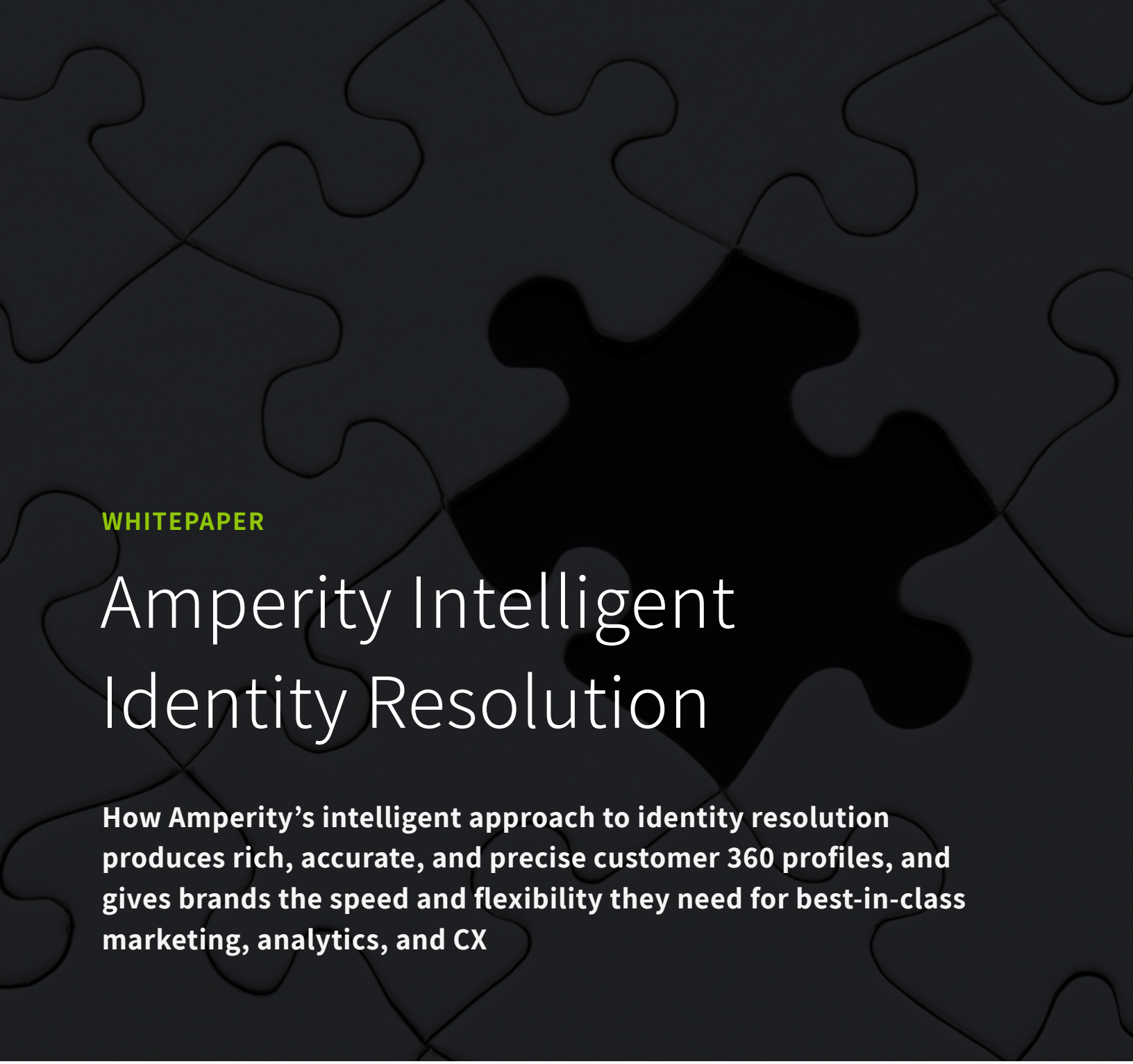


WHITEPAPER

Amperity Intelligent Identity Resolution

How Amperity's intelligent approach to identity resolution produces rich, accurate, and precise customer 360 profiles, and gives brands the speed and flexibility they need for best-in-class marketing, analytics, and CX

There are two phases in the process of building customer 360 profiles. The first is **data preparation** - all of the initial work needed to get data into a single environment and to reformat and prepare it for the matching process. The second is the **identity resolution**, where records and identities are assigned to unique individuals.

This document explains both phases in the process, addressing the common challenges we've seen in the marketplace, and describing Amperity's unique capabilities and why they are **faster** and **more accurate** and **context-specific** than other approaches.

amperity

Data Preparation

Most data scientists acknowledge that the most grueling and time-consuming portion of the machine learning pipeline is data preparation. Bringing data from separate systems into one environment, cleaning it, and normalizing it are all prerequisites to doing anything sophisticated like training or applying machine learning models. Doing this once is hard, but doing it on a recurring basis (weekly, daily, or even faster) is even more complex.

The following are the most common challenges associated with data preparation for the enterprise:

1. Building all connectors for data ingestion - large loads, historical datasets, as well as incremental data loads
2. Data normalization
3. Large data store to put all the data and maintenance of this store
4. Updating connectors when data source schemas change

None of this is rocket science - it's just work. Work that IT teams at enterprise brands generally don't have the time or singular focus to accomplish. The result is that the data science team works with a subset of data from a handful of sources, and the profiles that are returned are incomplete and outdated by the time they are put to use by marketers and analysts.

Amperity is an end-to-end platform, built from the ground-up for the enterprise. Because of this, we handle all aspects of getting your data into our system and preparing it for identity resolution.

AMPERITY COURIER ARCHITECTURE

This starts with Amperity's data couriers, which we build for any and all of your data sources. These couriers are highly configurable to suit your unique data types. Our courier architecture supports a variety of common industry formats including hierarchical, nested array and sparse datasets. Data can be sent in via batches, streaming or even quick 1-offs from users (like one-time survey results or event registration information). Our system was also designed to handle extremely large datasets in a reliable and scalable way. To date, we handle trillions of records on a daily basis, pulling from a variety of different data sources and delivering unified, usable

data for a number of customers each day. With this flexible architecture, we can adapt as you bring in new data sources or your underlying data changes, ensuring all your data is always integrated and usable.

For more details about our data ingestion pipeline, see the whitepaper here.

AMPERITY PII NORMALIZATION

Amperity performs select data normalization, with a focus on PII data (names, email addresses, physical addresses, date of birth). While we don't eliminate data normalization requirements for your team entirely, we do so for the most important subset of your customer data. More on this below.

AMPERITY'S MULTIPLE DATABASES AND TABLES

Amperity provides a cloud-agnostic distributed infrastructure to store your enterprise-scale data. With sources like complete transaction histories, clickstream, and other historical datasets, enterprise brands may require terabytes of storage for trillion of entries. Amperity's flexible approach of multiple databases and tables allows you to organize all your data in the way that best enables all your use cases. And, as your brand grows and changes, your data can grow and change with you - you can generate new databases, tables, and segments at any time, allowing you to experiment with and iterate on your data.

For more details about Amperity's multiple databases and tables, see the whitepaper here.

AMPERITY'S ALERTS AND MAINTENANCE

Amperity automatically detects and alerts when your data is missing, mis-shaped, bigger- or smaller-than-expected. We provide 24x7 internal operations to quickly catch data issues upstream. Often, Amperity is the first to know about outages in your upstream customer data systems. And because Amperity builds, monitors, and maintains all your data pipes into and out of the system, you can count on us as your single vendor for SLA accountability.

If data schemas change, systems are added or replaced, or if unexpected updates are required, the Amperity team is available to make the necessary changes. These changes are all built into the cost of an Amperity license, so it's simple and cost-effective to stay up-to-date with all your new and evolving data assets.

Identity Resolution

The fundamental task that identity resolution is trying to accomplish is to identify the same individual person within and across all data sources that contain customer information. There are two approaches that are generally used for this purpose: the **Unique Identifier** approach and the **Static Rule** approach. In this section we explore both, and highlight their strengths and weaknesses, before delving into the Amperity solution and what makes it better suited to accomplishing this difficult and complicated task.

UNIQUE IDENTIFIER APPROACH

In a perfect world, each individual customer would be represented by the same unique identifier in every one of a brand's datasets. An example of a perfect unique identifier would be a social security number or a right thumb print, because no one shares either of these with anyone else and each person only has one (unlike home addresses, names, or emails).

However, in the real world, few (if any) of a brand's systems have identifiers that are truly unique. In these cases, it's true that a simple primary key/foreign key match is sufficient to definitively connect data about an individual in one table with data about the same individual in another table.

Unfortunately, for the majority of a brand's data, primary keys are not unique to individuals. For example:

ECOMMERCE CUSTOMERS

PK/ID	CUSTOMER NAME	CUSTOMER PHONE	EMAIL ADDRESS
123	Rebecca Scully	425-111-1111	rscully@gmail.com
456	Rebecca Scully	425-111-1111	Rebecca78@yahoo.com

LOYALTY MEMBERS		
PK/ID	CUSTOMER NAME	EMAIL ADDRESS
123	Rebecca Scully	rscully@gmail.com
789	Rebecca Scully	Rebecca78@yahoo.com

Using a simple Unique Identifier to identifying people, one would assume that IDs 123, 456, and 789 represent three distinct customers, despite it being possible that all four records belong to the same individual.

STATIC RULE APPROACH

Data users have long understood that the simple approach outlined above fails when faced with the realities of real world customer data. This is because of the many separate systems in use that were never designed to seamlessly integrate with one another, and the data inputs that are as imperfect as the people entering them.

In an attempt to address these challenges, a more sophisticated, rules-based approach was devised - and this is the mechanism that most legacy solutions have been offering for over a decade.

This is how it works. Revisiting the sample data above, suppose we define the following rules:

IF names are an exact match, **THEN** it's a match

or

IF email addresses are an exact match, **THEN** it's a match

or

IF primary key/foreign key are an exact match, **THEN** it's a match

Applying this rule set illustrates a common problem with this approach. The first rule is sufficient to classify all the records in both tables as one person because there's an exact match on Rebecca Scully. For common names like John Smith, this rule will result in erroneous matches of records that are in fact different people.

To weed out some of these false positives, we could instead define the rules as:

IF names are an exact match **AND IF** email addresses are an exact match, **THEN** it's a match

or

IF primary key/foreign key are an exact match, **THEN** it's a match

Looking at the sample data, this approach is somewhat better, however, now we will have more false negatives. Take an example where the customer's name is very rare, such as Kabir Shahani. In this case two customers with the name Kabir Shahani but different email addresses kshahani@gmail.com and kshahani78@yahoo.com will result in our rules suggesting these are 2 distinct people, despite common sense suggesting that they are the same person.

In addition, data input is error prone. It's extremely common to find examples like Rebeca Scully and Rebecca Scully throughout your data. This could be addressed by doing "fuzzy" matching where we don't look for an exact match of values for a particular field but rather use a more refined algorithm like Levenshtein distance.

As more and more signals for matching are added, more rules have to be hand-coded. Their relationships with each other become more complex as well. Static rules must be strung together in more sophisticated ways than just (Rule A AND Rule B) OR Rule C. Weights have to be assigned to different rules to factor in that some rules are a stronger signal of two records representing the same person, and other rules represent a weaker signal.

Generally, these layers of complexity are captured in a long static rule table. While this can be an effective strategy as a point-in-time solution, they are difficult to maintain as data changes and new data sources are added. The weight-setting is not scientific because the weights are often set with data administrators approximating what "makes sense".

There are two additional things to consider in a typical static rule-set approach:

1. What is being compared?

2. What is the output of the comparison?

What is being compared?

Generally, legacy systems perform pairwise comparisons, meaning that one record is compared with another record. Based on this comparison, a determination is made whether these two records are a match or not. How does this translate when there are more than two records?

Consider this example:

1. Record A and B are compared (using the static rule set above) and match = true

2. Record A and C are compared and match = false

3. Record B and C are compared and match = true

This is conflicting information. Based on this information, should A, B, and C be considered the same person or not? The truth of the matter is that the flaws of pairwise matching with binary yes/no outputs make this question unanswerable in a correct way.

What is the output of the comparison?

In the MDM/static rule set world, when two records are compared, the output of the comparison is a binary (yes/no) answer. While this is simple and easy to understand, important information is lost about whether the quality of the match was high, medium, low, and what records were "near" matches but didn't meet the bar. This can be problematic for two reasons. The first is best represented by the example above.

Imagine if we were comparing A, B and C with each other and:

A and B = strong match

A and C = weak mismatch

B and C = strong match

If the strength of the match and mismatch was known, we would be able to make a more confident determination that A, B, and C are in fact the same person even though A and C, when directly compared, are a weak mismatch.

The second problem is that for specific use cases, it's reasonable to use weaker matches, in other use cases it's only acceptable to use strong matches.

A use case where weaker matches are optimal is product recommendations. Suppose you had a record that contained a name, email address, and zip code, and another with a name, email address, and the person's gender. You want to personalize product recommendations for consumers on your website. If you know they're female, you offer a spa package. If you know their zip code is in an extreme weather area, you offer winter gear.

Let's say the match between the two records is good, but not great. In this use case it's still useful to accept the match and use the gender information associated with the second record for the first. Using a static rules approach, we have no flexibility to process weaker associations, and align them to a given use case.

The Amperity Approach

Amperity's solution to identity resolution was built from the ground-up for the enterprise. Founded in 2016 by customer data matching specialists, we take advantage of the latest advances in machine learning to unlock a never-before-possible approach. This allows for faster, more accurate, and context-aware identity resolution, that allows brands to use more of their data, in a more timely manner, and for a broader variety of use cases.

1. SEMANTIC TAGGING

This first step in the process, semantic tagging, occurs as a part of data preparation, after data from multiple sources have been brought into the system. When feeds are set up, Amperity assesses all fields that contain personally identifiable information (PII), and tags them with semantic labels.

Semantic tagging works like this: a field is represented in the system as a "date". After semantic tagging, the same field is labeled "birthdate". Another example is a field labeled "string". Semantic tagging identifies that field as an "email address". These labels make the data much more usable for identity resolution later on in the process.

When applying semantic tags we consider a wide range of PII data types including first name, last name, any pre-existing primary key/foreign key, email address, shipping and billing addresses, zip codes, phone numbers, age, gender, date of birth, customer type, company, title, and others.

Note that not all these semantic data types need to be represented as distinct fields in the source data for a tag to be applied. For example, “given-name” and “surname” can be derived from “full-name”. We can also derive “gender” from “title” (e.g. Mr., Ms., or Miss) and/or from the “given-name” (e.g. “Mary” represents a female, while “Matt” represents a male). This is important because the more semantic values we leverage, the better our identity resolution will perform.

Sometimes, there is more than one field (derived or source) associated with the same semantic type. Depending on the type, we apply different logic to consolidate the multiple values. For example, “phone” can have different values, so we concatenate them together. “Gender”, on the other hand, can only take a single value. Therefore if “female” is derived from “given-name” and “male” is derived from “title”, we use “null” as the final value.

2. DATA CLEANUP

Next, for all fields with semantic tags, Amperity performs data normalization and cleaning. Data normalization includes string trims, unwanted-character removal, data type conversions, and more. We also normalize some semantic values for easier comparison in the downstream processes.

For example, the address “123 W. Elm St.” is normalized to “123 WEST ELM STREET”, the phone “001-(333)—444-5678 is trimmed to “3334445678”, and the alias email “mona+1234@google.com” is changed back to “mona@google.com”.

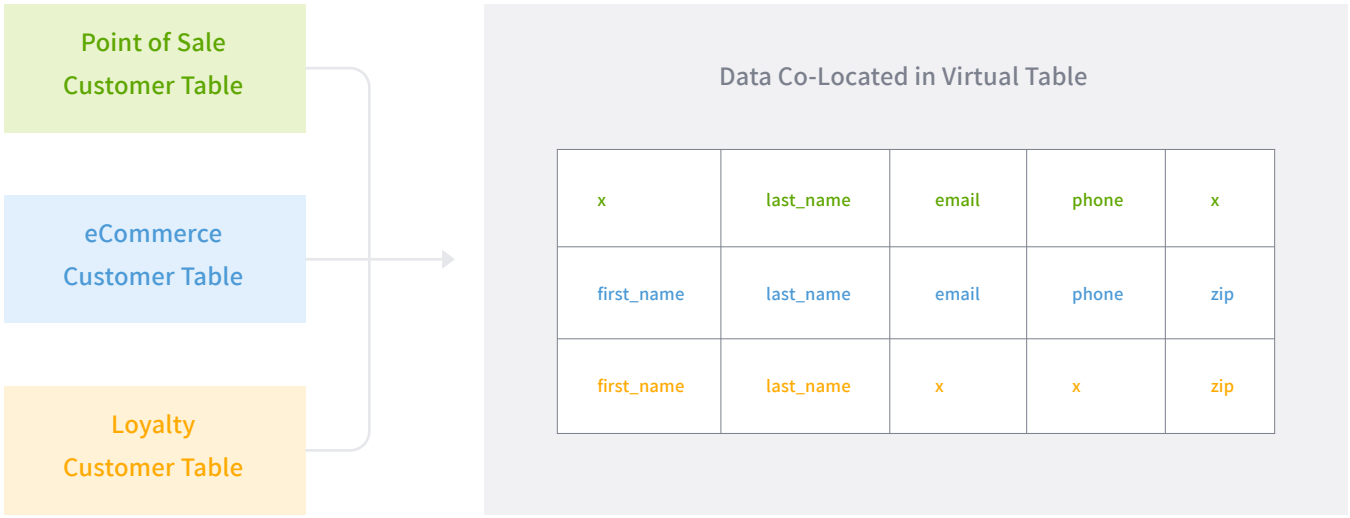
Note that Amperity does not use NCOA data for address normalization because it is illegal to use NCOA data for identity resolution purposes.

As a part of data cleaning, we identify and label common bogus values like “Not Available”, “Null”, “Declined”, and some semantic-specific spam values like the phone number “123-456-7890”, the email address “noreply@abc.com” or the home address “Delivered to the back door”. We are able to accurately identify these common bogus values, having looked at billions of records across customers. We also consider nature of data for specific industries. For example, for a hotel brand, on average more than 1% of its customer records use the hotel address and phone number instead of the individual’s.

3. UNION OF ALL CUSTOMER TABLES

For the next step in the process, we bring together all preprocessed and tagged data into a virtual table. This is a necessary prerequisite to comparing records across all datasets, rather than pairwise matching between tables (the pitfalls of which are outlined above).

Because semantic tags have been added to all relevant PII fields in the input tables, we can now virtually combine or co-locate columns that share the same semantic type. For example, if the eCommerce customer table has a full name column, and the Loyalty customer table has a first name column, the values of those are now virtually aligned.



Semantic fields are aligned and data can now be compared across all datasets instead of pairwise matching between tables.

4. MATCHING

The next step, matching, is at the core of the identity resolution process. This is where we compare records and make connections. There are three key phases to this process: blocking, scoring, and clustering.

It is important to note that the Amperity platform is optimized for enterprise-scale data. Identity resolution on small datasets requires fewer, less sophisticated steps, and many of the techniques are fundamentally different. As we grow, we continue to invest in solving the most difficult identity challenges facing large-scale consumer brands.

Blocking

When performing identity resolution across a vast virtual table containing hundreds of millions of records (or more), each record potentially needs to be compared to every other record. Simple math, however, shows that this process is computationally prohibitive. For example, suppose you have a union dataset containing 20 million records. The number of comparisons will be 20 million x 20 million = roughly 0.4 quadrillion comparisons.

Even if cost was no object, this number of comparisons takes a long time. For effective personalization and targeting, brands needs to use their data in a timely manner - on the scale of hours, not days or weeks. Therefore, to enable identity resolution at this scale, Amperity uses a technique called **blocking**. From a high level, blocking uses simple rules to divide up the entire set of records into smaller “blocks” of records that have a higher probability of matching. Only records within a block are compared. This brings down the computational load considerably, and allows data to be processed rapidly.

DATA SCIENCE DEEP DIVE

During blocking, the complete dataset is divided up into smaller blocks according to “blocking keys”. A blocking key, <a_i, f_i> is a tuple consisting of a semantic attribute ai and a function f_i. The function f_i is usually very cheap to compute. For example, it uses the attribute a_i as input and returns its phonetic encoding, a substring, or itself. A possible blocking key is the concatenated string of the first three characters of the given-name and the first three characters of the surname, which can be represented as <FN, F3> + <LN, F3>. Amperity applies the blocking strategy globally to the entire unioned virtual table so we don’t miss any potential matching across tables, and at the same time, we can harvest the performance gain globally.

An efficient and adaptive blocking strategy is essential to the performance of the matching process. It is an active research area, and Amperity is prototyping and implementing cutting edge techniques like Dynamic Blocking. These investments help us continue to improve the speed and accuracy of our identity resolution for a variety of data types and formats.

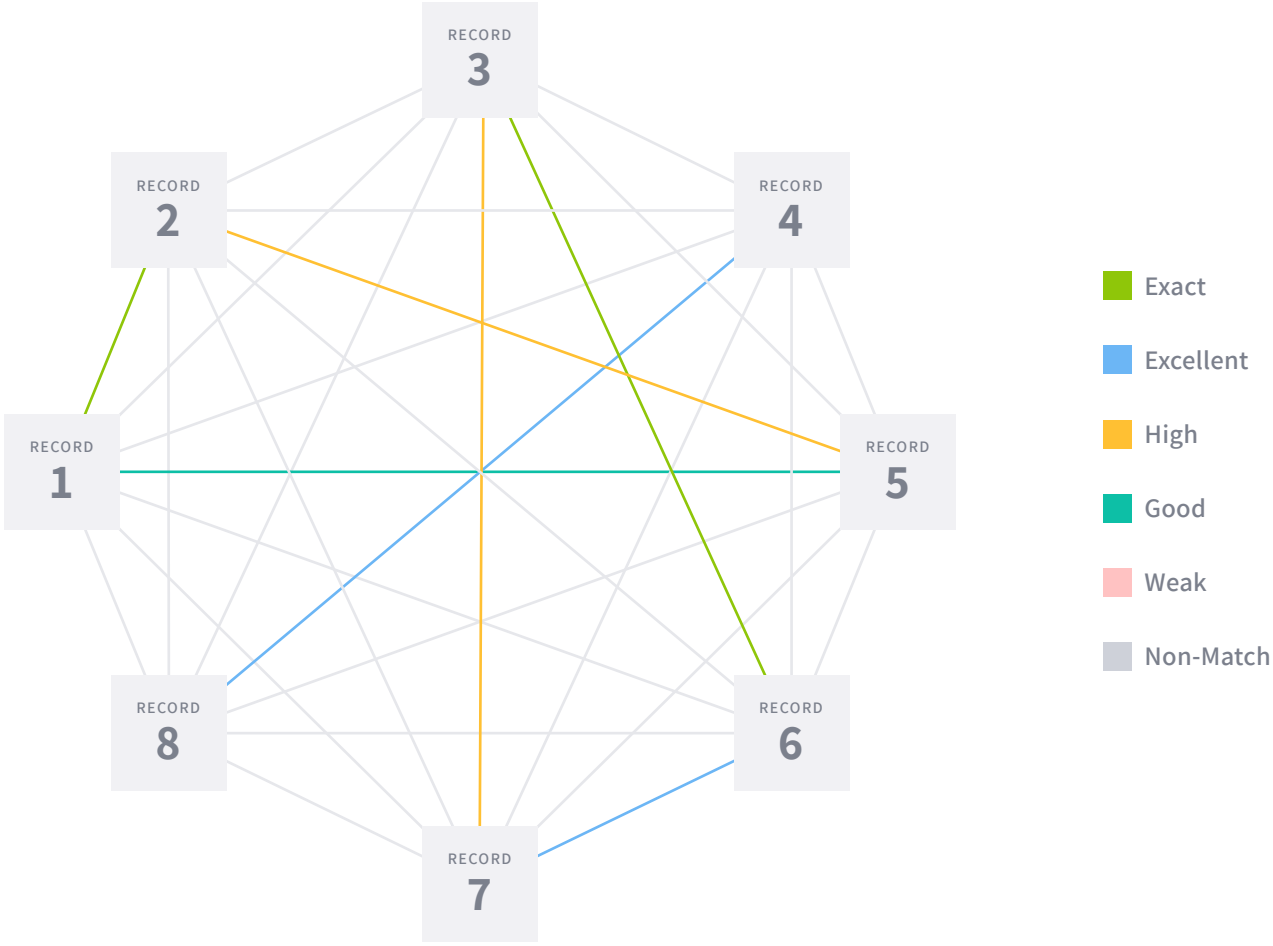
Pairwise comparison and scoring

Next, we compare each pair of records within a block. We use a high precision machine learning model designed by our data science team. Amperity's model was built for the specific purpose of matching customer records, and as such, has many features that take in account the specific nature of various types of customer data. For example, the model considers:

- Exact matches of a loyalty number
- Fuzzy matches of email addresses based on Levenshtein distance
- Probabilistic matches based on the rareness of names in a given region
- Probabilistic matches based on email tokens that combine first name, last name, or birthdate that's elsewhere in the customer record

The final output of this step in the process is one of six classifications for each record pair. These include the following ordinal categories, "non-match", "weak-match", "good-match", "great-match", "excellent-match" and "exact-match". An "exact-match" is a pair of records that we are 100% confident belong to the same individual. A "weak-match" has some indicators that the records belong to the same individual, but the connections are considered weak. The model also classifies "non-matches", where there is a high likelihood that the records do not belong to the same person.

A sample graph showing 28 different pair classifications produced for just eight records within a block; this process is performed over millions of records for a typical enterprise brand



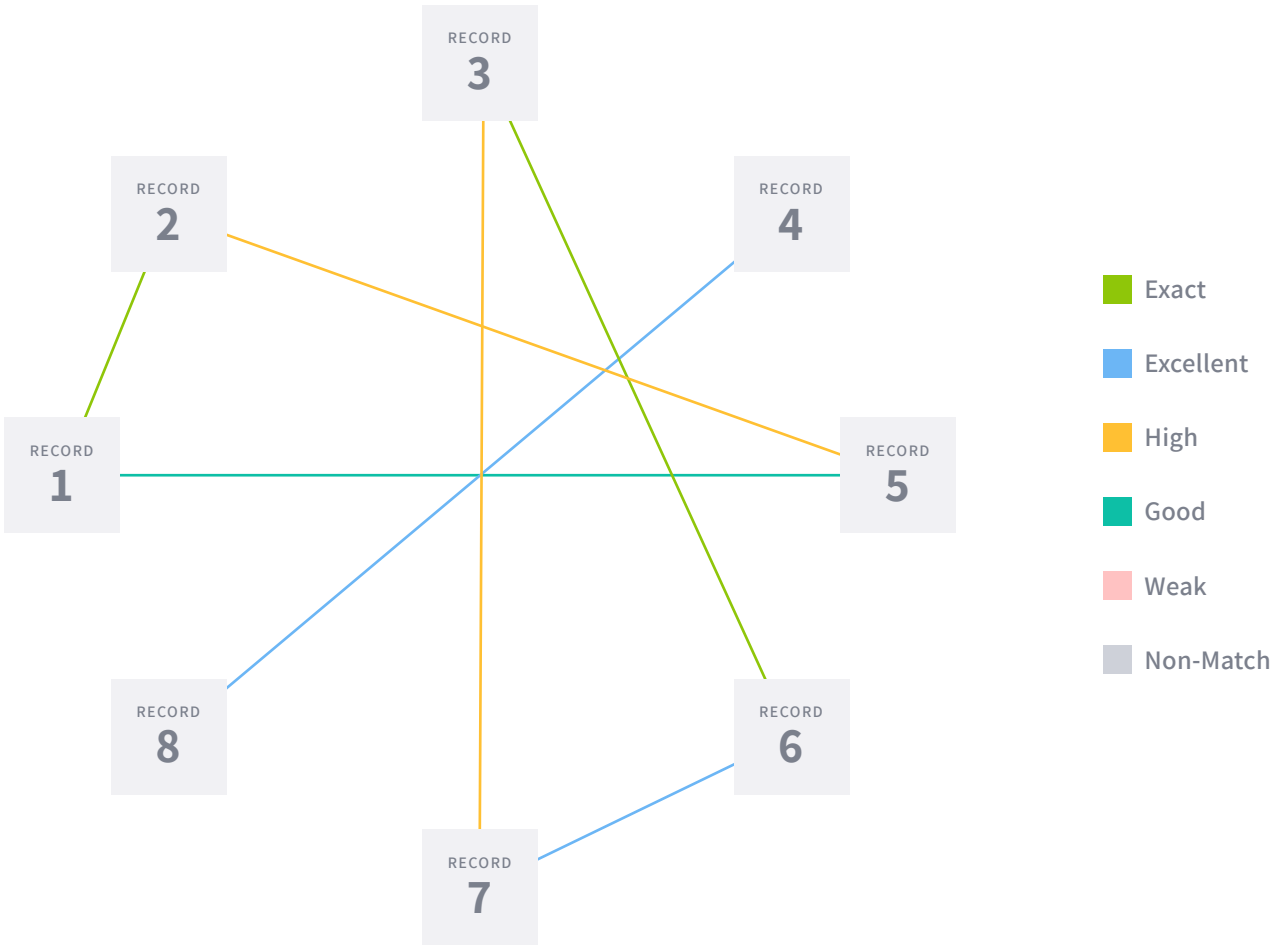
For example, if a block has eight records in it, each record is compared with the other seven records, for a total of 28 classifications. If you look under the hood, you can see the intelligence of the model. Here are some examples:

- records 1 and 2 are an “exact match” because they shared the same loyalty number and SSN
- records 1 and 3 are an “excellent match” because the email addresses were highly similar, and there was an full match on a rare first name and a rare last name
- record 1 and 4 were a “non-match” because their SSNs were different

Clustering

Clustering is the process of deciding which records to include in each customer 360 profile. With a rich graph of connections, records can be divided up in many different ways depending on a brand's unique objectives and constraints.

First, we work with a brand to choose an ideal matching threshold, which defines the minimum quality of a match in order for records to be linked together within a cluster. Lower quality matches can be included, but this will exclusively be due to a transitive connection through a higher quality match.



A threshold of "high" has been chosen, and all matches below this threshold are excluded

At this point, distinct customer profiles begin to emerge from the data. However, sometimes there are conflicts that need to be resolved. For example, a brand wants to keep business travelers separate from customers who travel for personal reasons. But in the sample data below, record 1 is transitively connected to record 5 through their shared connection to record 2. Additional special techniques are required to separate records like these across large datasets.

RECORD	FIRST_NAME	SURNAME	CUSTOMER_TYPE	EMAIL	ADDRESS	LOYALTY
1	Jess	Preston	Business traveler	jesspres12@gmail.com	1000 1st Ave S	X3399-0988
2	Jess	Preston		jesspres12@gmail.com	246 East Elm st.89	X3399-0988
5	Jess	Preston	Personal traveler		246 East Elm st.89	

DATA SCIENCE DEEP DIVE

For the purpose of separating records that have been transitively connected that the brand wants to leave distinct, we can choose from several clustering techniques. These include correlation clustering, which provides faster performance when cluster sizes are very large, or hierarchical clustering using Amperity's proprietary metrics, which have the capacity to separate records when certain semantic fields pose a conflict. By applying the hierarchical clustering with the Amperity metrics, we further partition the cluster {r-1, r-2, r-5} to {r-1}, {r-2, r-5}.

At the end of the clustering process, the system has partitioned the records into sets called Amperity Clusters, where each Cluster corresponds to a single real-world person.

5. AMPERITY ID ASSIGNMENT

After the matching process is done, every Amperity Cluster is assigned a unique ID. Because an Amperity Cluster represents a real-world person with a complete profile, having a stable set of cluster IDs means these people remain trackable over time. However, with new data loaded into the system everyday, previously established clusters may split, merge, rejoin, or even disappear, and net new clusters may also appear. Amperity has developed and implemented a patent pending algorithm to assign cluster IDs after each matching process, so that these cluster IDs can still represent the same real-world people regardless of the dynamics and churns of the cluster structure caused by new information.

Training the Model

Amperity's identity resolution capabilities are primarily driven by supervised machine learning algorithms, which perform best when trained and tuned using large amounts of high quality training data. Unfortunately, high quality training data in the realm of customers is virtually nonexistent. This is because the ground-truth about which records actually belong to the same real-world people is unknown and unknowable. The only way to fully verify which records belong to whom would be to ask individuals, one at a time, to manually update and unify their own records. In reality this is impossible.

Another option is to ask staff to manually label the data by looking at a random set of record pairs and using common sense to distinguish matches. This process is both time-consuming and error-prone. Moreover, randomly generated samples may not be able to fully represent the characteristics of the entire dataset.

To solve these problems, Amperity's data science team has developed a patent-pending algorithm that intelligently extracts a representative dataset from our client's full dataset. The representative dataset includes sample record-pairs that represent all the unique feature combinations (i.e., feature signatures) inherent to the data. We make sure the representative dataset covers record-pairs for all possible feature signatures, which means we capture the full spectrum of the characteristics of the entire dataset.

DATA SCIENCE DEEP DIVE

In the toy example below, we use six features to measure the similarity between records in a dataset, such as first name exact match (“fn-match”), first name approx-match (“fn-approx-match”), last name exact match (“ln-match”), last name approx match (“ln-approx-match”), email exact match (“email-exact-match”), postal exact match (“postal-exact-match”). After blocking and pairwise feature calculation, there are three distinct feature signatures found, 100110, 101000 and 100110, among the four record pairs. So in the representative dataset we include record pairs associated with each feature signature. In reality, our data science team use a much more complex feature set to perform the pairwise scoring, which results in a much larger representative dataset.

CLUSTER-ID	PK	GIVEN-NAME	SURNAME	EMAIL	POSTAL
c - 1	r - 1	Chuck	Sakoda	chuck@amperity.com	
c - 2	r - 2	Yan	Yan	yanyan@amperity.com	
c - 2	r - 3	Yang	Yang	yanyan@amperity.com	98103
c - 2	r - 4	Yan	Yan		98103
c - 3	r - 5	Stephen	Mayles	stephen@amperity.com	98102
c - 3	r - 6	Stephen	Meyles	stephen@amperity.com	
c - 4	r - 7	Ian	Wesley-Smith	ian@amperity.com	
c - 5	r - 8	Derek	Slager		98102

related-pair	fn-match	fn-approx-match	in-match	in-approx-match	email-exact-match	postal-exact-match	feature signature
(r - 2, r - 3)	1	0	0	1	1	0	100110
(r - 3, r - 4)	1	0	0	1	0	1	100101
(r - 2, r - 4)	1	0	1	0	0	0	101000
(r - 5, r - 6)	1	0	0	1	1	0	100110

Once a representative dataset has been extracted, Amperity's data science team labels the data. Any record-pairs that share the same features are bundled together, allowing for efficient and unbiased labeling. The labeled record-pairs are then used to train and tune our clients' models, resulting in exceptional accuracy.

Concurrently, we are also developing semi-supervised learning (or active learning) capabilities to actively capture the unique business logic of our clients. Our proprietary features and client-specific training data are still utilized, but our clients can also train their own customized models for the specialized use cases.

Amperity Identity Resolution Summary

Amperity's intelligent approach to identity resolution produces rich, accurate, and precise customer 360 profiles, and gives brands the speed and flexibility they need for best-in-class marketing, analytics, and CX. Because Amperity was built from the ground up for enterprise-scale identity resolution, Amperity is the most complete and robust solution to unlock customer data for a complete set of diverse use cases.

1. Accurate: All records across all datasets are compared with each other, and sophisticated clustering is applied to avoid transitive inconsistencies. This results in higher accuracy results across all data.

2. Precise: All matches have an ordinal confidence classification instead of binary classification. This means lower-confidence matches can be used for some use cases, and higher-confidence matches for others.

3. Reliable & Robust: A dynamic, machine learning based solution paired with deep investments in training data, lead to powerful and robust identity resolutions capabilities that evolve and improve as data sources change, grow, and are added over time.

4. Fast & Efficient: Data prep and semantic tagging are key parts of the Amperity workflow. We don't force the hard part of machine learning back onto our clients. Instead our identity resolution brings in fresh and raw data, runs on it, and produces results in a matter of hours - leading to rapid and current results every single day, as opposed to one-off match and merge processes.

5. Scalable: Our identity resolution pipeline was built (from day one) for scale. Scaling up to hundreds of millions of records is a fundamentally different technology than scaling to millions of records.

About Amperity

Amperity, the world's first Intelligent Customer Data platform, is revolutionizing customer data unification and management for enterprise brands. Leveraging advanced machine learning-powered identity resolution and the full power of the cloud, Amperity ingests raw data, stitches it together, and forms the richest customer profiles possible. Then, by shaping multiple databases for any downstream system, Amperity syndicates data to the full set of engagement and analytics tools in the precise formats they require. With Amperity, many of the world's most loved brands are unlocking their siloed data, powering their tools with rich customer information, and bringing their best, data-driven marketing ideas to life.

amperity