Using Record Linkage to Create Big Data? How Good Is It?[1]

K. Bradley Paxton, Ph. D.
ADI, LLC
brad.paxton@adillc.net

Abstract

Record Linkage is being increasingly used to create new “big data” collections or improve already existing data. If, for example, a Census data set is correctly linked to a Tax data set, a new or improved data set may result; if, however, the linkage is done incorrectly, then the data set may be made worse. It is important to know how good your record linkage system is performing to optimize it and get the best results for your investment. Record linkage systems, however, are difficult to test.

This paper presents two proven methods for testing record linkage systems: a synthetic data GAMUT (Great Automated Model Universe for Test) for use primarily in the development phase and Production Data Quality (PDQ) in the production phase. These methods were used successfully in the 2010 Census for evaluating forms processing but they are extensible to record linkage and also apply to any IT classification system.

Both methods are cost-effective because they use automation to replace a lot of human effort. Together, the two methods provide “cradle-to-grave” testing and improvement capability for Record Linkage systems.

Finally, we describe a fundamental trade-off between precision and recall to assist in comparing and tuning systems to achieve maximum Record Linkage system value.

[1] Based on a paper given at JSM, Session on Data Quality and Non-Response, Tuesday 5 Aug 2014, Boston, MA

Background

In the course of over twenty years of working with the U.S. Census Bureau and other agencies, ADI, LLC developed two unique methods for testing handprint data capture quality in forms processing. These two methods have been used in both the 2000 and the 2010 Decennial Census programs, and they were very successful and cost-effective.

The first method utilized synthetic data, presented in a form suitable for testing a paper forms processing system, namely as a deck of digitally-printed forms that appear to be real forms written on by real respondents, but were synthetic and created on a computer (A Digital Test Deck™). The major benefit of using these test materials is that the "Truth" is known perfectly, and so data capture results can be precisely scored and subsequently improved.

The second method used an independent data capture engine to derive another opinion, at the field level, about the Truth. When these two fields were compared, we found that over 95% of the fields being compared agreed, which were then placed in the Truth file with extremely high confidence. The remaining 5% had the interesting property that four out of five times, one engine got the correct answer, even though they didn't agree. This enabled a computer-assisted manual process to easily determine the Truth of these fields, leaving less than 1% to be resolved. Of these many of these fields were considered "inconclusive", which meant that a group of intelligent humans could not agree on the right answer, and so these fields were excluded from the scoring process.

Both of these methods, being unique, were patent applied for, and the U.S. Patent and Trademark Office issued both patents (See Refs. 1 and 2).

As we were exploring new uses for these test technologies, we realized that they are both applicable to testing any IT classification system, of which forms processing is but one example. Record Linkage (RL) is another example of a classification system, and testing RL is the topic of this paper.

Since it may not be apparent to the reader that a handprint data capture process is a classification system, the following example may be useful.

As a simple illustration of why forms processing is a classification system, consider the following magnified image of a hand-printed character, scanned by a binary, monochrome scanner at 200 dpi, shown in Fig. 1.

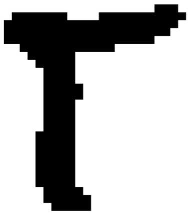


Fig. 1 – A Scanned Hand-printed Character

This character was classified as an upper-case letter "T" by the Optical Character Recognition (OCR) system with confidence 0.93. It helps to show the entire field, as shown in Fig. 2, where the character shown in Fig. 1 is at the end of the field.

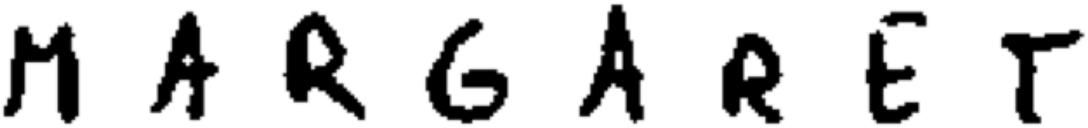


Fig. 2 – A Hand-Printed Field

In this case, the fact that this person was a female and the field was a first-name field was productively used by the OCR system to do the classification of the entire field and get it right, in this case "MARGARET". If you wish to know more about this interesting and complex technology, see Ref. 3.

Now, a Record Linkage engine is also a classification system. In Record Linkage, we are trying to link an entity, say a household, from File 1 containing n_1 entities with another entity in File 2, containing n_2 entities, as shown below in Fig. 3.

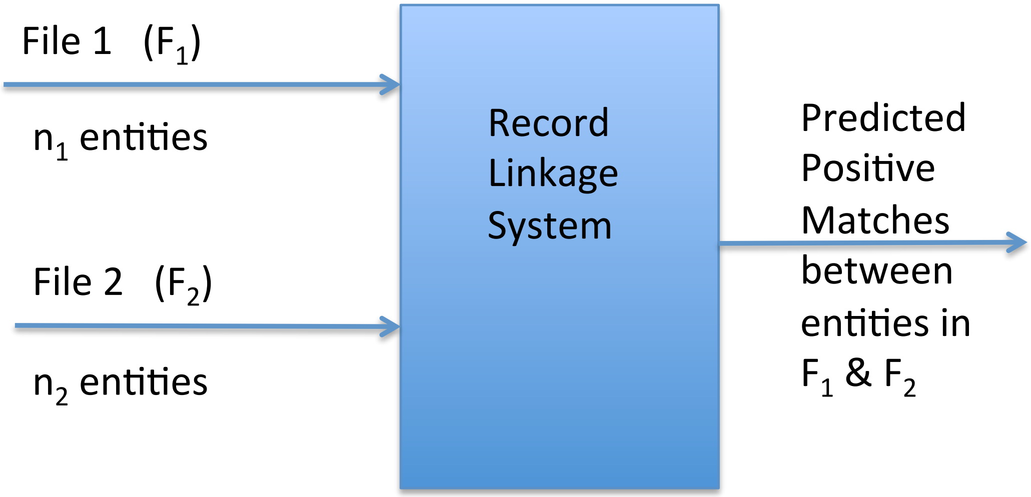


Fig. 3 – A Record Linkage (RL) System

The job of the Record Linkage system is to predict positive matches between the entities in F_1 and F_2, where F_1 might be Census data, say, and F_2 might be Tax data. Again, an example may be useful.

Here are two entities in Fig. 4 buried in a large Census file F_1 and a large Tax file F_2. The RL system might be wondering if these two entities should be linked:

Census - F1			Tax - F2		
HEDLEY		STRIKLAND	H	M	STRICKLAND

Fig. 4 – Two Possibly Linked Entity Names

Because both Strikland and Strickland have the same Soundex Code (S362), and the first name Hedley starts with an H, these two records could possibly match, but the RL engine wouldn't be sure just yet. So, the engine would want to use more data; a birthday would be good, but most tax files don't have birthdays. So let's say the RL engine looks at addresses, as shown below in Fig. 5.

Census - F1			Tax - F2		
HEDLEY		STRIKLAND	H	M	STRICKLAND
8526 NORTH TAHITI LOOP	WASHINGTON	DC	1204 CLEARWATER PLACE	WASHINGTON	DC

Fig. 5 – Is More Data Really Better?

Now we are really confused, so the RL engine might just quit and say there is no match, or it might try adding some more data, like, say, file dates and another person in the household, as shown in Fig. 6 below:

Census - F1	4/1/10		Tax - F2	4/15/11	
HEDLEY		STRIKLAND	H	M	STRICKLAND
8526 NORTH TAHITI LOOP	WASHINGTON	DC	1204 CLEARWATER PLACE	WASHINGTON	DC
JOY		WILLIS	J	S	WILLIS-STRICKLAND

Fig. 6 – Looks like a Match Now

It looks like the Strickland household moved between Census Day 2010 and Tax Day 2011, and the RL engine might decide it's happy now and predict a positive match. If so, then after a step often called "merge and purge", the old Census data (in red on the left in Fig. 7 below) is improved in the green data on the right, and the RL engine has done it's job of correctly classifying two entities as a match and allowing data improvement to result. Of course, if an incorrect match is predicted, then the data could be made worse instead of better, and that's why you need to be able to test and improve such systems; the trouble is, it's hard to do.

Census - F1	as of 4/1/2010		Updated Census File	as of 4/15/2011	
HEDLEY		STRIKLAND	HEDLEY	M	STRICKLAND
8526 NORTH TAHITI LOOP	WASHINGTON	DC	1204 CLEARWATER PLACE	WASHINGTON	DC
JOY		WILLIS	JOY	S	WILLIS-STRICKLAND

Fig. 7 – Improved Data After “Merge and Purge”

Now we will briefly describe how the two testing methods, synthetic data GAMUT and PDQ, can be applied to the problem of testing Record Linkage systems. Either method may be used, but the synthetic data approach is usually used for systems in development, and the PDQ approach is usually used in production, although they can beneficially overlap. Either way, the objective is to measure all four numbers in a confusion matrix, as shown in Fig. 8 below:

		SUT Prediction	**SUT Prediction**	**Row Sums**
		Positive Match	Negative Match	
Data Truth	Positive Match	**TP** cm	**FN** M - cm	M
Data Truth	Negative Match	**FP** m(1 - c)	**TN** N – M - m(1 - c)	N - M
Column Sums		m	N – m	N

Fig. 8 – A Typical Confusion Matrix

Here, TP, FP, FN, and TN are the numbers of true positives, false positives, false negatives, and true negatives respectively. Until you can quantitatively describe all four of these numbers, you haven’t run a good test.

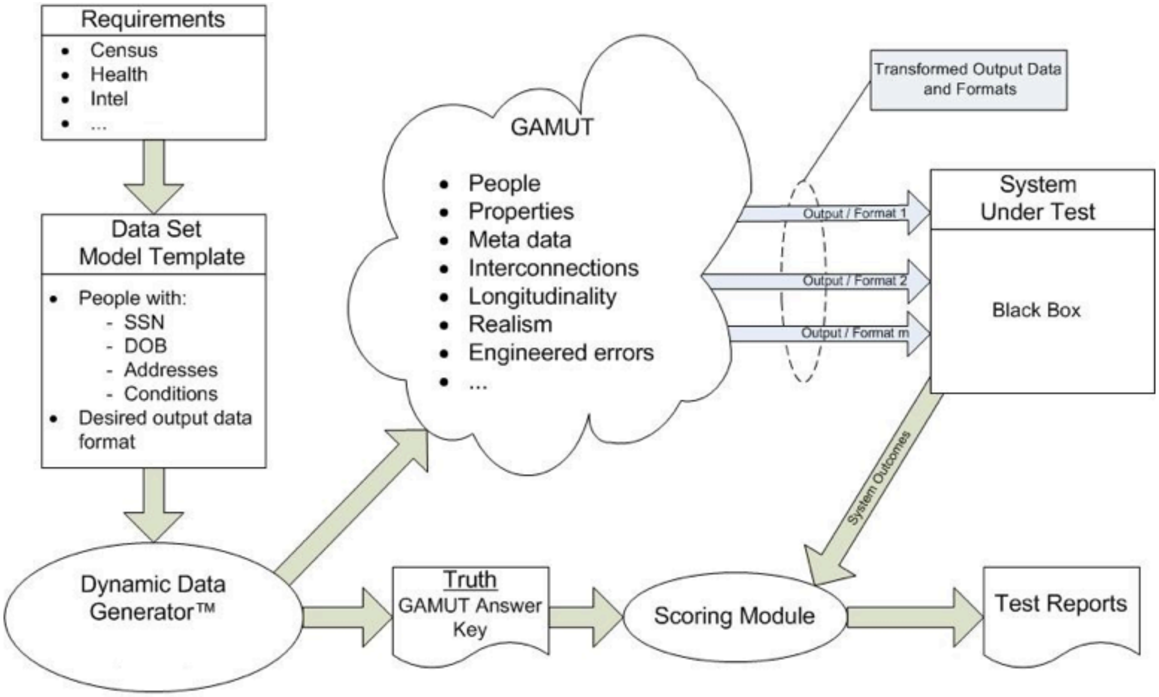
All four of these numbers add to N, the total numbers of possible outcomes of attempting to match all the entities in F_1 with all the entities in F_2, so

$$N = TP + FP + FN + TN \tag{1}$$

A major problem with record linkage testing is that TN is usually so large that it swamps the other three numbers, but there are ways to deal with this problem and still get useful results. In Fig. 8, c is precision, defined later.

Testing Record Linkage with a GAMUT of Synthetic Data

This approach is described in detail in Ref. 4, but is briefly summarized here for completeness. Using our Dynamic Data Generator™ (DDG), it is possible to create a useful model universe we call GAMUT (Great Automated Model Universe for Test), as diagrammed in Fig. 9.



© 2011 ADI, LLC

Fig. 9 - GAMUT: Great Automated Model Universe for Test

We start by considering the System Under Test (SUT) and create a GAMUT, a very large and complex test data set from which we extract data files suitable for implementing the test plan.

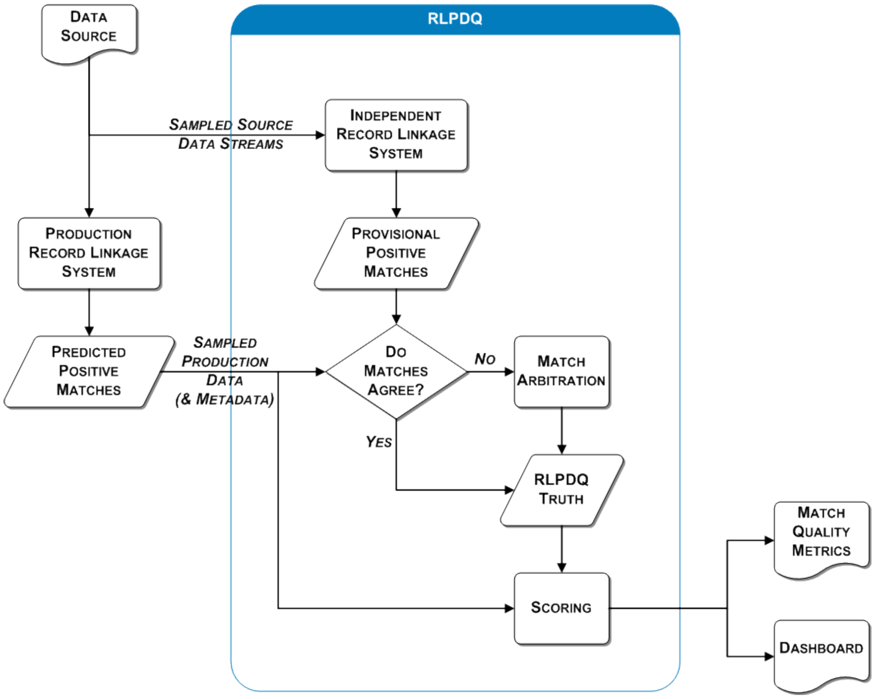
Present testing approaches often use large files of "real" data which are "dirty" and for which the Truth is not well known. Synthetic, yet realistic GAMUT data sets, designed for test, and for which the Truth is known allows for quick, cost-effective, precise testing and quantitative scoring. Both true and false positives may be measured and used to improve and tune systems in development.

The use of synthetic GAMUT testing data can significantly speed up and improve Administrative Records testing, leading to improved system linkage quality and optimal performance. Remember, we don't aim to replace testing with "real" data, but rather to supplement it to speed up the development process to achieve quality software that's scalable and ready for efficient, high-quality production.

Testing Record Linkage Production Data Quality (RLPDQ)

This approach is described in detail in Ref. 5, but is briefly summarized here for completeness. The basic RLPDQ concept is to use an independent RL system that has fundamentally different characteristics and algorithmic approaches than the production RL system, bringing automation to bear on this difficult and costly testing problem.

The data from the two engines is compared, as shown in the block diagram below in Fig. 10:



DATA SOURCE
RLPDQ
SAMPLED SOURCE DATA STREAMS
INDEPENDENT RECORD LINKAGE SYSTEM
PRODUCTION RECORD LINKAGE SYSTEM
PROVISIONAL POSITIVE MATCHES
PREDICTED POSITIVE MATCHES
SAMPLED PRODUCTION DATA (& METADATA)
DO MATCHES AGREE?
NO
MATCH ARBITRATION
YES
RLPDQ TRUTH
MATCH QUALITY METRICS
SCORING
DASHBOARD

Fig. 10 – RLPDQ Block Diagram

The key is to cost-effectively get from “comparison space” which is of order n^2 to “entity match space” which is of order n, where n is the approximate number of entities in a file. This is a huge accomplishment, and great reduces the manual effort needed to test Record Linkage, as illustrated below in Fig. 11, where “entity match space” is the small blue region.

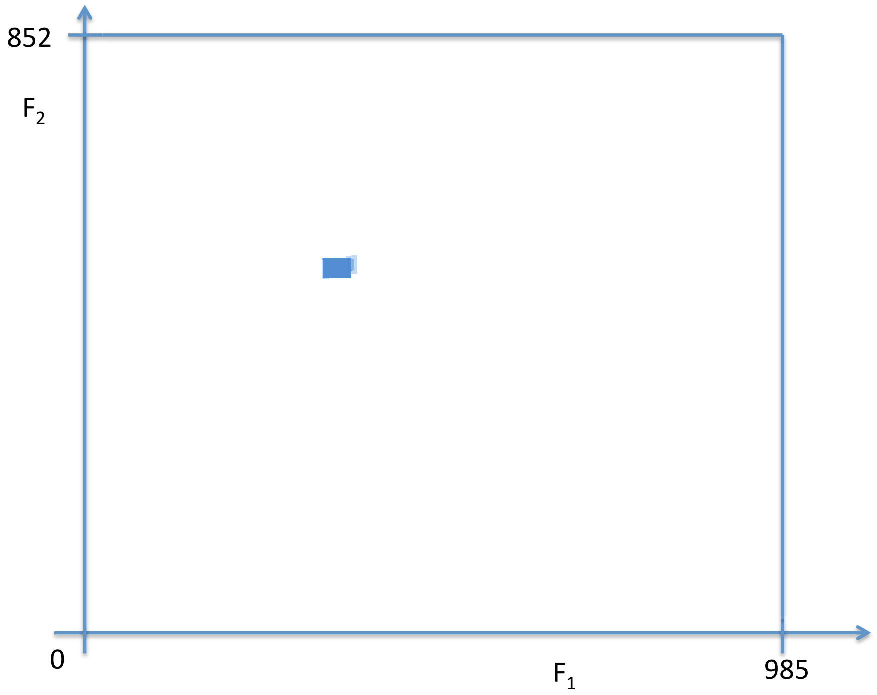


Fig. 11 – Actual Entity Match Space Embedded in Comparison Space

If both approaches to testing are used, then one can enjoy "cradle-to-grave" testing of your RL system, from development through production, as sketched below in Fig. 12:

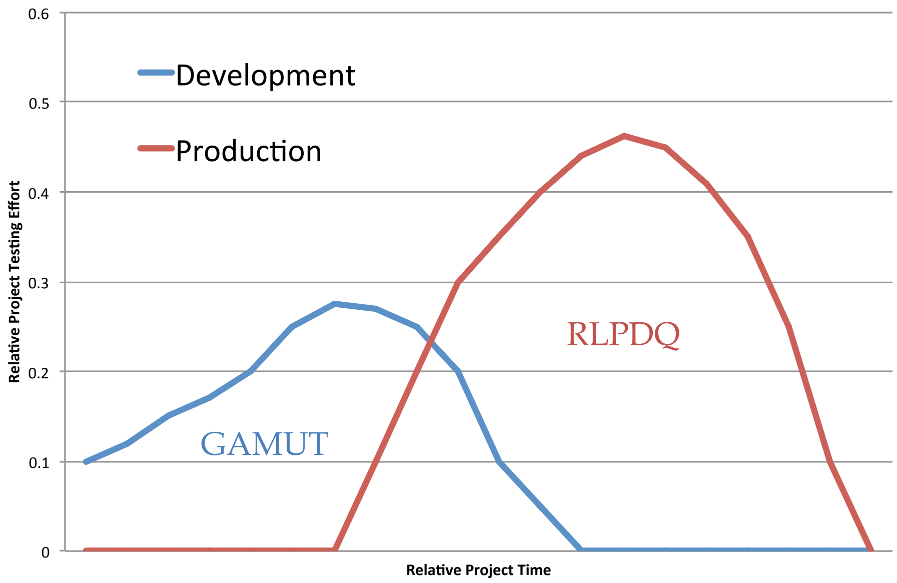


Fig. 12 – "Cradle-to-Grave" Testing

The Precision vs. Recall Trade-off

Now, we assume you have tested your RL system, possibly using one or both of the test approaches mentioned above or one of your own devising, and have derived actual numbers for the elements of the confusion matrix, namely TP, FP, FN, and TN. The number TN usually the largest of these four numbers, so much so that it excessively complicates analysis. One method used to get around this imbalance (see Ref. 6) is to use the metrics precision and recall, defined in our notation below in equations 2 and 3.

Precision: $$c = TP/(TP + FP) \tag{2}$$

Recall: $$r = TP/(TP + FN) \tag{3}$$

Note that neither of these two RL quality metrics uses the large number of true negatives TN, which is advantageous in quantitative analysis of the results.

An interesting paper by Alvarez (Ref. 7) written in the context of information retrieval derives a basic relationship between precision and recall that is clearly applicable to Record Linkage. His basic result, in our notation, is:

$$c(r) = rG/[G(2r - 1) + 1 - A] \tag{4}$$

where precision c is shown as c(r) to indicate the functional relationship to recall (r), and the two new parameters Accuracy (A) and Generality (G) are given by:

Accuracy: $$A = (TP + TN)/N \tag{5}$$

Generality: $$G = (TP + FN)/N \tag{6}$$

The term Accuracy is well known in Record Linkage (Ref. 6). However, the Generality term needs further explanation. It is equal to the number of correct matches in your RL problem divided by the total number of comparisons in comparison space N, and so it is fixed once you decide what files you are trying to match. You usually don't know it's value, but it is usually very small. It is also equal to the probability that you could choose a correct positive match by chance.

Both of these two parameters have the unfortunate property of depending on the large values N and TN. This means that A is usually almost one and G is usually almost zero, however, to facilitate meaningful analysis, I have found it convenient to define a new quantity k as:

$$k = 1 - (1 - A)/G \tag{7}$$

The quantity k is a quality index and is a constant if accuracy A is, since G is always a constant for a given RL problem.

We can now write Eq. (4) more simply as:

$$c(r) = r/(2r-k) \tag{8}$$

This is a basic hyperbolic relationship between precision (c) and recall (r) that is very helpful in understanding test results. It describes the shape of these curves for various values the quality index k. The slope of c(r) is always negative, describing the inherent nature of the trade-off between precision and recall, namely, if recall is increased, precision is decreased, and if precision is increased, recall is decreased, assuming the quality index k is a constant.

We require precision to be less than one, but also greater than ½ for a well-behaved RL system, for if your precision is less than ½ it means your matching algorithm is backwards, and so if you were to invert it, your precision would again be greater than ½. This does not occur in realistic RL systems.

At r = 1, then the precision from Eq. 8 reverts to:

$$c(1) = 1/(2-k) \tag{9}$$

from which we can see that at k = 0, the precision is ½, the worst it can be in actual practice, and so when the quality index k equals 0 it means the worst quality you can have. On the other hand, when k = 1, the precision is one, meaning the best quality you can have, or RL perfection. So k is a very meaningful quality index bounded by zero and one, that is, 0 < k <1.

Note also from Eq. (8) that at r = k, precision becomes one, so all the precision versus recall curves are bounded between k and one on the recall axis. Here is a generic graph for precision versus recall in Fig. 13, showing precision being one at r = k (here k = 0.7), and decreasing as recall increases.

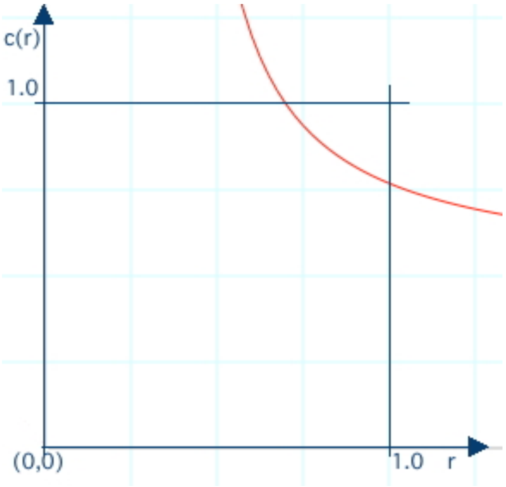


Fig. 13 - Precision Versus Recall Graph

To show how a graph like Fig. 13 can be useful, suppose we did RL testing on two different RL engines, and plotted the precision vs. recall test results [along with two reference points representing the origin (0,0) and perfection (1,1)], as shown below in Fig. 14:

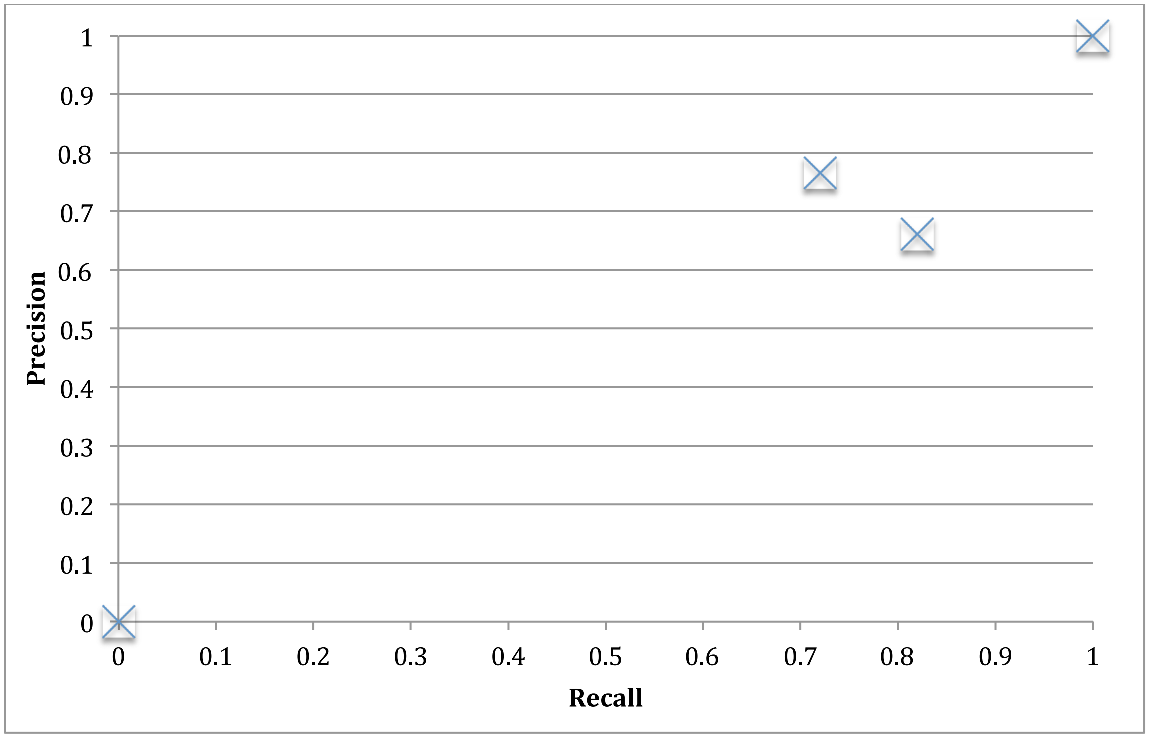


Fig. 14 – Two Data Points Representing Two Different RL Systems

Now, the question is, "Which of the two actual data points represents the best RL system?" One might say that they prefer precision more than they prefer recall, or vice-versa, depending on the way you plan to use the data in your business process. Is it possible that one is somehow better regardless?

One could argue about these questions at length, but I have found it to be helpful to superimpose lines of constant quality index k running from 0.1 to 0.9, say, as a guide, as shown in Fig. 15 below:

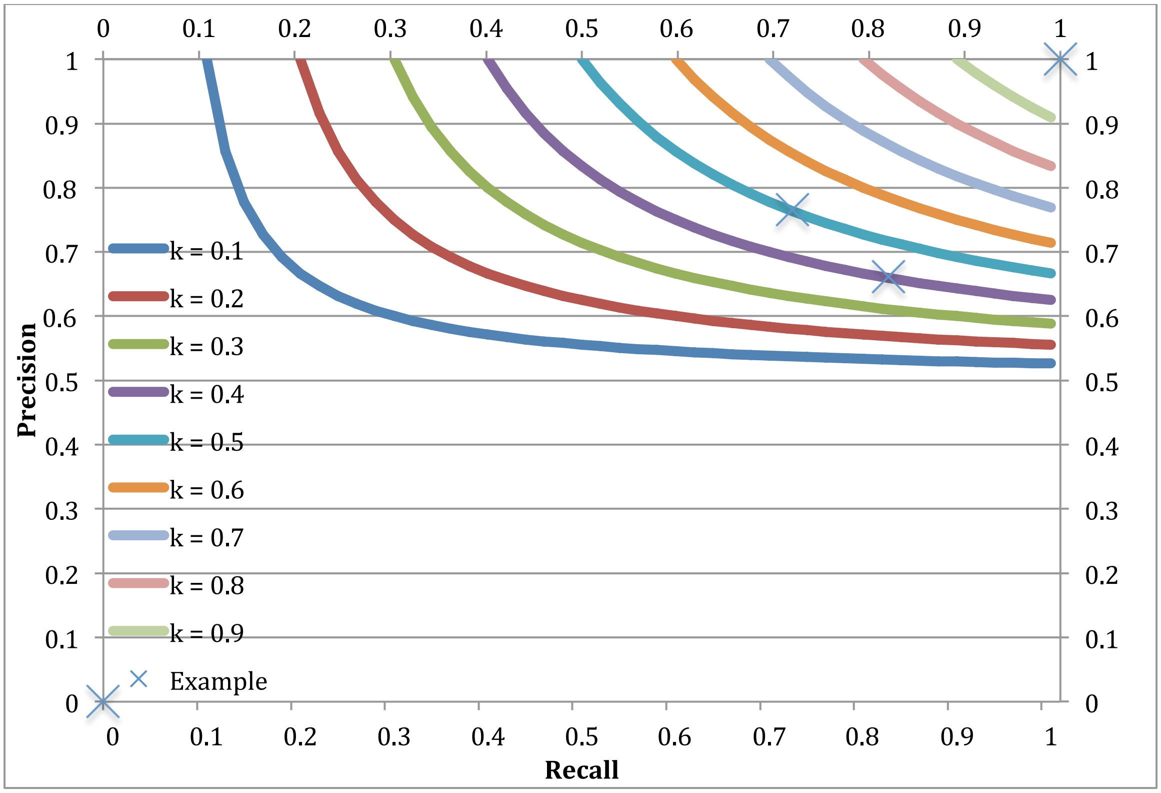


Fig. 15 – Data Points with Precision-Recall Lines at Constant Values of k

We assert that the RL system operating on the line of constant quality index k = 0.5 tends to be better than the system operating on the k = 0.4 line, particularly when precision is valued as much or more than recall.

Moving to a k-line closer to (1,1) represents an overall improvement in your RL system, whereas moving along a particular k-line represents exploring the trade-off between precision and recall at constant accuracy.

When moving along a particular k-line, you may prefer to operate at a particular point on that line, depending on the relative cost to your business process of having less recall or less precision.

Conclusions

Two approaches to testing Record Linkage systems were briefly outlined; GAMUT synthetic data for use in development and RLPDQ for use in production. These two methods may be beneficially used together to provide "cradle-to-grave" testing of your RL system.

Finally, to evaluate RL test results, and to assist in comparing and tuning RL systems, we indicate an approach using the fundamental trade-off between precision and recall as a guide.

References

1. *Handprint Recognition Test Deck*, filed 13 Apr 2012 (originally filed 2 Sep 2004), issued 30 Jul 2013 as Patent N0. 8,498,485.

2. *Method and System for Assessing Data Classification Quality*, filed 30 Jul 2010, issued 30 Jul 2013 as Patent N0. 8,498,948.

3. Paxton, K. Bradley, *Handprint Data Capture in Forms Processing – A Systems Approach*, Fossil Press, Rochester, NY, 2011

4. Paxton, K. Bradley, and Hager, Thomas, *Use of Synthetic Data in Testing Administrative Records Systems*, Proceedings, Federal Committee on Statistical Methodology (FCSM), Washington, DC, 2012.

5. Paxton, K. Bradley, *Testing Record Linkage Production Data Quality.* In JSM Proceedings, Government Statistics Section. Montreal, Canada: American Statistical Association. Pgs. 1157-1171, 2013.

6. Christen, Peter, *Data Matching – Concepts and Techniques for Record Linkage, Entity Resolution and Duplicate Detection*, Springer, 2012, p.167.

7. Alvarez, Sergio A., "An Exact Analytical Relation among Recall, Precision, and Classification Accuracy in Information Retrieval", Technical Report BC-CS-2002-01, Computer Science Department, Boston College, June 2002.