

Designing a Search Mechanism for Debt Collection

Martin Del Vecchio, Shu Jin, Alana Mistretta, Hayden Rolando, and Hope Tuck

Abstract—Debt Collection firms require an efficient and accurate procedure for locating a delinquent debtor, filing a lawsuit, and collecting on a debt at minimal cost. This process is known as “skip-tracing,” with a “skip” being a delinquent debtor actively attempting to escape a debt. In this paper, we study the skip-tracing procedure for a large firm specializing in debt collection. This firm currently contracts an address search service to research and compile a list of possible addresses for each debtor and recommend one address as the most probable. We use a simple maximum likelihood estimation procedure for assessing the “false positive” and “false negative” probabilities. Having determined the accuracy of the address search service, we study the optimal decision for the firm taking into account the costs and probabilities of all the available alternatives for “skip-tracing”. Our preliminary findings suggest that at high confidence levels, the firm stands to benefit monetarily by forgoing one of the most costly steps in the verification process.

1: Introduction

Debt collection firms require an efficient, accurate procedure for locating delinquent debtors. The process of locating a delinquent debtor is known as “skip-tracing, with a “skip” being a delinquent debtor actively attempting to escape a debt. A good skip-tracing procedure needs to consider multiple factors, including the magnitude of the debt in question and the time remaining on the statute of limitations for the debt. Debtor account portfolios provide the most current address on file for a debtor, but this address is typically out-of-date. Most collection firms pay a small fee to a private agency in order to obtain a list of additional possible addresses and phone numbers. After receiving the list of potential addresses, the firm must determine which addresses constitute viable leads. Additionally, the firm must decide at some point in the skip-tracing process whether it will file a lawsuit for the debt or wait to initiate litigation, based solely on its confidence in the given information. Consequently, an effective skip-tracing procedure must be able to estimate the

probability of success associated with every course of action available to the firm during every stage of the procedure. A large firm specializing in the purchase and collection of distressed consumer assets contracted a University of Virginia Capstone Team to create a skip-tracing procedure for the firm [6]. The firm desired an automated, reproducible, skip-tracing procedure to analyze a list of addresses provided by an address search service and recommend the most probable address for a particular debtor. The procedure was implemented as a multi-layered decision process through which a given list of addresses are algorithmically clustered, or sorted, according to their common characteristics, and then subjected to one or more decision rules which recommend the most probable address.

2: Decision Analysis

2.1. Decision Trees

In order to begin analysis with a solid grasp of the firm’s current procedures, a detailed decision tree was developed to outline the firm’s current decision process. The decision tree also offered insight into which decisions might prove most profitable for the firm and which steps might be less valuable in the long-term. Specifically, the firm was interested in knowing whether it should accept the recommendation of the address search service “as is” or undertake a more costly verification process to prove the validity of an address. At the time of development, the firm was unable to offer data on past skip cases. Therefore, several decision trees were developed in order to experiment with different probability values for the branches. These trees were used to develop a “break-even” point for the probability of obtaining a true positive recommendation from the address search service. The “break-even” point corresponds to the absolute lowest probability value for a true positive, at which the firm would still benefit by foregoing the skip verification process. The decision trees are shown in Figs. 1-3.

The decision trees are exactly the same, apart from the assumptions made about the probability values for each decision. The range of probability values used in the decision tree represent educated estimates of the actual probability values at each stage. The chosen estimates were suggested by the project contact at the firm, and based on the results the firm generally obtained.

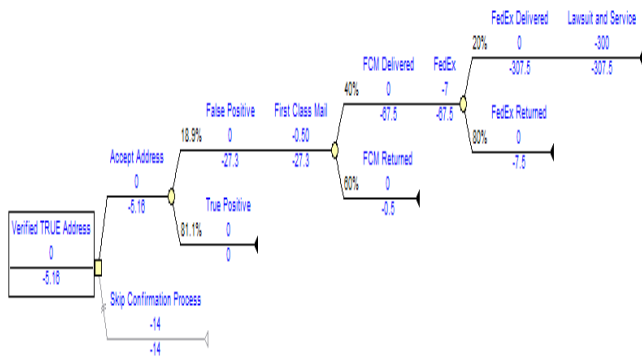


Fig. 1: Decision Tree Detailing Firm's Skip-Tracing Process at the Beginning of Analysis, Assumptions: False Positive Probability = 18.9%, True Positive Probability = 81.1%, Probability FCM Delivered = 40%, Probability FCM Returned = 60%, Probability FedEx Delivered = 20%, Probability FedEx Returned = 80%, Soutce: Author

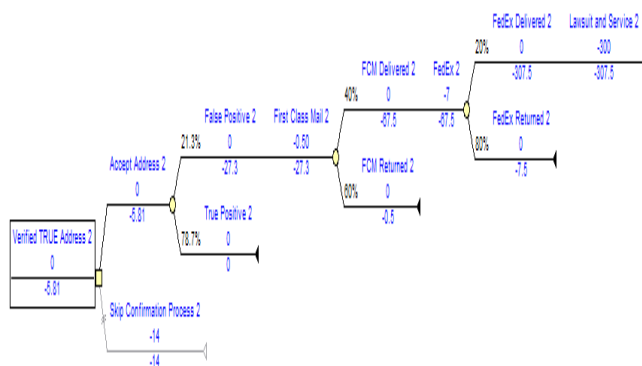


Fig. 2: Decision Tree Detailing Firm's Skip-Tracing Process at the Beginning of Analysis, Assumptions: False Positive Probability = 21.3%, True Positive Probability = 78.7%, Probability FCM Delivered = 40%, Probability FCM Returned = 60%, Probability FedEx Delivered = 20%, Probability FedEx Returned = 80%, Soutce: Author

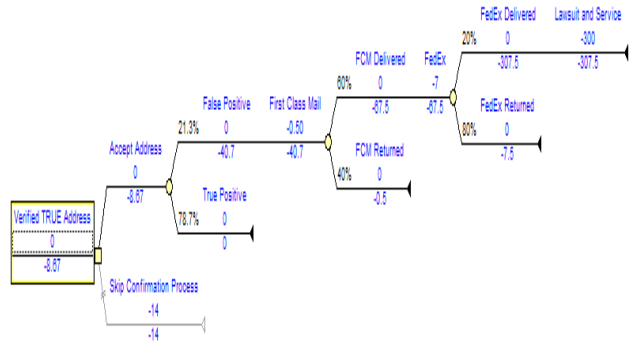


Fig. 3: Decision Tree Detailing Firm's Skip-Tracing Process at the Beginning of Analysis, Assumptions: False Positive Probability = 28.3%, True Positive Probability = 78.7%, Probability FCM Delivered = 60%, Probability FCM Returned = 40%, Probability FedEx Delivered = 20%, Probability FedEx Returned = 80%, Soutce: Author

The decision trees begin from the point of the address that the address search service recommends as the most probable. From that point, the firm chooses whether to pursue the address as correct or go through a lengthy and more costly skip verification process to verify that the address is correct. The skip verification process costs the firm \$14.00. If the firm chooses to forego this process, it proceeds to the branch "Accept Address." If the address chosen by the address search service is the correct address (true positive) the costs associated with verifying the address are essentially nil, because the firm will recover those costs in the lawsuit process. If, however, the address is a false positive, the steps that follow "Accept Address" accrue costs for the firm.

After accepting an address, the firm sends a First Class Mail to the address, indicating its intention to file suit for the debt in question. If the First Class Mail is returned, the address is assumed incorrect and the firm loses only 50 cents. If the First Class Mail is delivered, the firm sends a FedEx further declaring the firm's intention to file suit on the debt. If the FedEx is delivered, the firm assumes the address is correct and files suit. However, because FedEx will leave a package if no one is available to sign for it, FedEx letters are occasionally delivered to the wrong address. If the firm files suit based on a FedEx that was delivered to an incorrect address, it loses a total of \$307.50, including the lawsuit filing cost of \$240. If the FedEx is not delivered because the address is incorrect, the firm loses only \$7.50.

2.2. Break-Even Point

For every case tested, the expected monetary loss for accepting the address "as is" proved lower than the costs of the skip verification process. The expected monetary loss for each decision tree is shown at the bottom of the first box on the left of the tree. After further analysis, the decision trees showed the break-even probability point for a true positive to be 48.4%. The break-even point marks the probability value above which the firm would forego the skip verification process. If the firm were to eventually determine that the probability value of a true positive is lower than 48.4%, it would then benefit from the skip verification process.

3: Data Analysis

3.1. Data Set

The debt collection firm eventually provided a data set consisting of the records and corresponding results from approximately 5000 past skip cases. The data set, which came in the form of a Microsoft Excel spreadsheet, included a list of potential addresses for each debtor. The data set listed the most probable address in each case, as determined by the address search service. The data set also included fields marking the last known month showing verified activity for each address and the first known month

Table 1: Mock Excerpt from Data Set, Source: Author

RecordID	First	Mid	Last	Street Name	Street Num	Street Suffix	City	State	Zip	First Month Verified	Last Month Verified
567878977	John	Q.	Smith	Pine	134	ST	Franklin	GA	630289015	197504	197504
567878977	John	L.	Smith	Pine	134	LN	Franklin	GA	630289015	197504	197504
567878977	John		Smith	Moon	134	Way	Carson City	NV	530236621	197802	197903

showing verified activity for each address, as determined by the address search service. The data was separated into two categories, which classify the type of search procedure used by the address search service. These two categories are hereafter referred to as Service 1 and Service 2. Table 1 shows a mock excerpt from the data set. The table shows information for a single RecordID (corresponding to a single debtor) as it might appear in the data set. Worth noting are the small variations in the debtor’s name, as well as the different addresses, two of which are almost identical excepting their Street Suffix. The data set also included a separate spreadsheet showing the actual verified address for each RecordID [7]. The data as received contained one large known error. The nine-digit zip codes for many addresses were concatenated in reverse order so that the four-digit portion of the zip code preceded the five-digit portion. This error was corrected for in analysis.

3.2. Early Data Analysis

Initially, several simple decision rules were tested in order to determine the accuracy of choosing an address based on a single rule. Due to the nature of the available data (spreadsheet form), the decision rule procedures ran as Microsoft Excel Macros written in Visual Basic for Applications (VBA) programming language. The following decision rules were tested:

- 1) Choose the address recommended by the address search service.
 - a. If no address or multiple addresses are recommended by the address search service, choose the address with most recent “last verified” date.
 - i. If these dates are the same, choose the address with the longest length (if one or more addresses have the same longest length, choose arbitrarily)
- 2) Choose address with most recent “last verified” date.
 - a. If two or more addresses have the same “last verified” date, determine whether one of the addresses is recommended by the address search service. If so, choose that address.
 - i. If no or multiple addresses are recommended by the address search service,

choose the address with the longest length (if one or more addresses have the same longest length, choose arbitrarily)

- 3) (Variation on #3) Choose address with most recent “first verified” date.
 - a. If two or more addresses have the same “first verified” date, determine whether one of the addresses is recommended by the address search service. If so, choose that address.
 - i. If no or multiple addresses are recommended by the address search service, choose the address with the longest length (if one or more turn out to have the same longest length, choose arbitrarily)

Fig. 4 shows the testing result for each rule. For each data set category, the macros compared the address chosen by the decision rule to the actual address for the given record. The statistics shown in Fig. 6 correspond to the percentage of addresses chosen by each decision rule that match the actual address for the record. Matching percentages are listed in four separate statistical categories which vary in strictness of matching criteria.

The four statistics correspond to matches in the following areas:

- Stat 1: Matching State & City
- Stat 2: Matching State, City, Street Name & Street Number
- Stat 3: Matching State, City, Street Name, Street Number, & PreStreetDirection
- Stat 4: Matching State, City, Street Name, Street Number, UnitDesignation, & Designation Numbers

The matching percentage was calculated as a simple maximum likelihood estimator corresponding to the number of matching records divided by the total number of records. As seen in Fig. 4, the highest matching percentage for matching an entire address came to approximately 48%,

with a 95% confidence interval of (45.11, 51.42), shown in Table 3.

4: Clustering Algorithm

4.1. Levenshtein Distance

Because the decision rules tested returned relatively low accuracy levels, it was determined that the rules alone did not offer the required level of certainty. Further inspection of the data showed that for a single record, the address search service often returned multiple variants of a single address. The data suggested that if those variants could be grouped together, they could be considered the same address for decision purposes. The number of potential address choices for each record would decrease, thereby increasing the probability of selecting the correct address.

Automating this process required an algorithm for determining whether two or more addresses were essentially variations of the same address. The algorithm could not use the typical Boolean (true or false) comparison of two strings because such a comparison would return a false value for any two addresses that did not match exactly. After researching alternative options for comparing similarities between two strings, the Levenshtein Distance Algorithm proved the best available means for making such a comparison. The Levenshtein Distance Algorithm computes the “distance” or similarity between two textual strings. The distance represents the number of deletions or insertions required to transform one string into another [4]. A high Levenshtein Distance represents a high level of dissimilarity between two strings.

4.2. Hierarchical Clustering

The algorithm also required a procedure for determining which addresses belonged to the same group based on their Levenshtein Distance from one another. For this procedure, the algorithm used the method of hierarchical clustering. The method of hierarchical clustering is a statistical technique used in data mining to group data points together based on either their similarities to one another or their shared dissimilarities from the other data points. The method exists in three separate algorithms: single link hierarchical clustering, complete link hierarchical clustering, and average link hierarchical clustering.

4.3. Designing the Algorithm

The algorithm runs through a Microsoft Excel Macro written in Visual Basic for Applications (VBA) programming language. Because little basis existed for inferring which hierarchical clustering method would produce the best results, the algorithm specifies the clustering method according to a user-inputted variable. After the user inputs the chosen clustering method, the algorithm calculates the Levenshtein Distance between each address listed for the specified RecordID. The algorithm implements Levenshtein distance as a weighted measure, with the most weight given to similarities in city and state, and the least weight given to similarities in address locale (street name, number, prefix, suffix, etc.). The justification behind weighting the Levenshtein distance in this manner derives from the fact that if two addresses have the same city and state, they are more likely to be a variation of the same address than two addresses which differ by city and/or state.

After calculating the Levenshtein Distance, the algorithm uses the user-specified clustering method to group the addresses into clusters. The algorithm executes according to the indicated method until the minimum distance between any two clusters is less than the dynamically defined threshold. The threshold is defined as the average of the largest Levenshtein Distance and the smallest Levenshtein Distance between the addresses in the original list. After clustering the addresses in the aforementioned manner, the algorithm outputs the clustered addresses for use in the last stage of the skip-tracing procedure, where a user-specified decision rule chooses the most probable address.

4.4. Testing the Algorithm

For initial testing purposes, the algorithm was tested with a simple decision rule rather than a complicated decision procedure. The logic behind this choice derived from the fact that a more intricate decision procedure might increase the accuracy rate on its own, making the algorithm’s contribution unclear. The algorithm used a decision rule that selected the longest address from the cluster with the most addresses. This rule is based on the hypothesis that the address returned most often by the search service (in varying forms) has a higher probability of being the correct address. Likewise, among the variants, the address with the longest length will likely be the most complete.

Using the aforementioned decision rule, another Microsoft Excel Macro in Visual Basic for Applications (VBA) programming language compared the actual address listed for a given RecordID with the address selected by the decision rule. The algorithm was tested using all three clustering methods: single, complete, and average link. Table 2 shows the results of the test for data Service 1, categorized by clustering method. As seen in the table, the

complete link clustering method demonstrated the highest accuracy level at a rate of 70.46%.

5: Conclusion

5.1. Results

The results detailed in the previous chapter seem promising for the skip-tracing firm, as 70.46% represents a very high rate of accuracy. Nevertheless, the results represent an estimate garnered from a single data set. Consequently, the results may not prove entirely applicable in real-time conditions. Likewise, the actual accuracy rate for the given data set could potentially be slightly higher or lower depending on the integrity of the data. While the firm acknowledged the original error in the data set (zip code concatenation), it may have missed additional errors that remained undetected throughout the analysis and development processes.

5.2. Future Work

Plans for future work include testing the clustering algorithm in conjunction with multiple decision rules, in order to determine whether using another decision rule might increase the accuracy of the procedure. Examples of decision rules to test include choosing the most complete address from the cluster corresponding to the address recommended by the address search service or choosing the most complete address from the cluster containing the address with the most recent "last verified" date. In addition to augmenting the clustering procedure, future plans include testing the accuracy rates for various subsets of the data, such as RecordIDs with no verified activity in the past 5 years or RecordIDs with a very large or very small number of potential addresses listed.

Further plans include testing the current clustering algorithm and additional decision rules on a new data set provided by the firm. The new data set contains DMV and Voter Registration records for each RecordID, which may provide some use in future analysis. This data set may also prompt insight into new decision methods which can be incorporated into the current procedure. The clustering algorithm's current accuracy rate of 70% demonstrates promising potential, and the work to date provides a well-established, adaptable platform for future developments to the algorithm.

6. References

[1] Bogomolny, Alexander. "Distance between Strings." 02.March.2006.1996.<http://cut-the-knot.org/do_you_know/Strings.shtml>.

[2] Eppstein, David. "Computational Statistics: Hierarchical Clustering." 06 May 1999. Theory Group, Dept. Information & Computer Science, University of California at Irvine. 01 March 2006

<<http://www.ics.uci.edu/~eppstein/280/tree.html>>.

[3] Garcia, Alfredo.2005."Designing a Search Mechanism for Debt Collection." 04 September 2005.
<<http://www.sys.virginia.edu/capstone/2006/08.htm>>.

[4] Gilleland, Michael. "Levenshtein Distance in Three Flavors." Merriam Park Software. 02 March 2006. <<http://www.merriampark.com/ld.htm>>.

[5] Moore, Andrew. "K-Means and Hierarchical Clustering." 03 March 2006. School of Computer Science, Carnegie Mellon University. 2001.
<<http://www.autonlab.org/tutorials/kmeans.html>>.

[6] "Services." 2005. Credigy International. 10 October 2005.<<http://www.credigy.net/credigy/us/crdgi.asp?area=2&img=2cont=202>>.

[7] "Skip Data" [Microsoft Excel Spreadsheet]. November 2005. Credigy International.

6. Appendix

Table 2: Testing Results of Clustering Algorithm

Clustering Method	Rate of Accuracy (%)
Single	70.08%
Complete	70.46%
Average	70.27%

Table 3: 95% Confidence Intervals for Decision Rule 1, Service 2

	Percentage	Lower Bound	Upper Bound
Stat 1	64.30%	61.38%	64.61%
Stat 2	40.67%	37.68%	40.99%
Stat 3	37.18%	34.24%	37.50%
Stat 4	30.07%	27.28%	30.36%

Fig. 4: Decision Rule Testing Results, Source: Author

Results for Procedure # 1									
Service 1					Service 2				
	<i>Stat 1</i>	<i>Stat 3</i>	<i>Stat 2</i>	<i>Stat 4</i>		<i>Stat 1</i>	<i>Stat 2</i>	<i>Stat 3</i>	<i>Stat 4</i>
<i># of Addresses</i>	1182	981	905	713	<i># of Addresses</i>	1045	872	828	626
%	79.32886	65.83893	60.73826	47.85235	%	80.57055	67.23207	63.83963	48.26523

Results for Procedure # 2									
Service 1					Service 2				
	<i>Stat</i>	<i>Stat 2</i>	<i>Stat 3</i>	<i>stat 4</i>		<i>Stat 1</i>	<i>Stat 2</i>	<i>Stat 3</i>	<i>Stat 4</i>
<i># of Addresses</i>	1134	900	840	671	<i># of Addresses</i>	990	776	738	573
%	76.10738	60.40268	56.37584	45.03356	%	76.32999	59.83038	56.90054	44.17887

Results for Procedure # 3									
Service 1					Service 2				
	<i>Stat 1</i>	<i>Stat 2</i>	<i>Stat 3</i>	<i>Stat 4</i>		<i>Stat 1</i>	<i>Stat 2</i>	<i>Stat 3</i>	<i>Stat 4</i>
<i># of Addresses</i>	1113	867	799	634	<i># of Addresses</i>	1012	803	763	582
%	74.69799	58.18792	53.62416	42.55034	%	78.02621	61.9121	58.82806	44.87278

Results for Procedure # 4									
Service 1					Service 2				
	<i>Stat 1</i>	<i>Stat 2</i>	<i>Stat 3</i>	<i>Stat 4</i>		<i>Stat 1</i>	<i>Stat 2</i>	<i>Stat 3</i>	<i>Stat 4</i>
<i># of Addresses</i>	958	606	554	448	<i># of Addresses</i>	797	456	418	331
%	64.2953	40.67114	37.18121	30.06711	%	61.4495	35.15806	32.22822	25.52043