Matching Artists Names from Two Sources: A Record Linkage Application

Emily Wright and Robbie Eckels

October 18th, 2013

Abstract:

The goal of this project is to match artist names from two different online sources. Artist name is the only information available for comparing the two sets of data. Although typographical errors were not a large concern due to the quality of the data sources, small punctuation errors such as accents and periods proved to be troublesome. Overall, the Jaro-Winkler similarity metric, which accounts for these small differences, proved to perform the best when comparing the artist's records and was used to create a classification tree to predict the probability of a comparison being a match.

Introduction:

Today millions of artists share their music online in the hopes of being discovered. Not surprisingly, the company Next Big Sound believes you can predict when an artist is going to be successful by tracking their online activity. This includes the artist's activity on Facebook, Youtube, Twitter, and Soundcloud. Further, The Next Big Sounds creates a weekly chart of the top 15 artists with the fastest accelerating online activity whom they have predicted to become successful. However, "success" is not necessarily measurable. At the same time, because radio is the dominant way consumers discover new music (Nielsen, 2013) a success will be regarded as the extent of an artist's radio play for the purpose of this project. As such, the first step is to match artists who have previously been predicted to become successful with the artists who are currently aired on radio.

Therefore, the goal of our project is to successfully match the artists who have been predicted to become successful with a list of artists who are currently played on radio. In order to perform record linkage techniques we first cleaned the data. Second, we calculated several similarity scores for each comparison. Finally, we hand labeled a subset of the data as training data to create a model which can compute the probability of a comparison being a match for the entire dataset. Overall, the Jaro-Winkler similarity score which accounts for typographical errors proved to be the best metric for comparing radio artist names.

The Data:

The rising artists predicted to become successful were collected by web scraping from the website www.nextbigsound.com/charts using the R package XML. The data are from a weekly chart of artists with the fastest accelerating online activity from August 5th, 2010 up until September 26th, 2013. This resulted in 3,064 records being collected. The data was fairly simple and consistent with the 5 fields of the artist rank on the chart, the artist name, month, day, and year of the prediction.

The radio data were also collected by web scraping from the website http://www.digitalradiotracker.com/chart.html using the web query function in Excel. The data is a weekly chart of the top songs played on radio. The data is divided into several categories and genres such as the Top 200 songs played on radio, the Top 50 Independent Artist songs, the Top 50 Rock songs, the Top 50 Pop songs, the Top 50 Country songs, the Top 50 Hip-Hop and R&B songs, the Top 50 Americana songs, and the Top 50 Adult Contemporary songs. These weekly charts were collected from January 2013 through September of 2013. Each category was combined together to create one large list of radio songs. The data consisted of 7 fields which were the artist's rank on the charts, the artist's name, the song title, the number of radio plays within that week, the month, the day, and the year of the radio chart. The only concern was the inconsistency of the artist name reported. The primary artists and the featured artists were both listed under artist name as one text string. Additionally, the abbreviation for a featured artist was inconsistent and included many terms such as "feat," "feat.," "ft.," "ft," "&," or "w/". Finally, the data was reported in all capital letters. Not surprisingly, before proceeding, extensive data cleaning was required and will be further discussed in the next section.

Methods: Data Cleaning

We began by cleaning the data to remove duplicate artist names in each list. Since both The Next Big Sound and radio play lists are released on a weekly basis, they contain many of the same artists from week to week. We performed an exact matching on each list, and initially found 2,938 rising artists and 647 radio artists. Because The Next Big Sound site releases a list of individual artists each week based on social media and because we assumed a small chance of their being typographical errors, we believe that 2,938 is an accurate estimate for the unique rising artists. For the radio data there were many more duplicates since the radio data is listed by song. First, it is very likely for each song to be played on radio week after week and hence the

artists to be listed multiple times. Additionally, one artist could have multiple songs on radio throughout the year. Further, an artist could be listed in more than one of the categories. For example, you could have one song listed in the Top 200 songs as well as listed in the Top 50 Rock songs. Additionally, because the primary artist and the featured artists are listed within one string under artist name we knew that the 647 artists we identified from exact matching was not accurate because the exact matching method would fail to account for duplicates of artists who appear as a solo artists and with a featured artists within in the list.

Initially, we decided to proceed and see if similarity metrics would be able to identify matches without extracting the featured artists. As such, we converted both lists to lower-case to allow for more accurate matching across lists. We feared this may create false matches, as some artists deliberately put upper-case letters in their names but this still appeared to be the best approach. After computing the similarity metrics which will be discussed further in the next section, we found that none of the metrics performed well. As such we decided to go back and extract the featured artists from the artist names.

We explored two methods for extracting featured artists. Our first idea was to split each artist name string word-by-word into fields. We then attempted to match both within fields and across fields to obtain matches. We would code a match if every field from one string matched to at least one field in another string. For instance, if we were trying to match "meek mill" to "alley boy feat meek mill," we would code a match since the only two fields of the first string match the 4th and 5th fields of the second string, respectively. This method proved computationally infeasible since it required comparing every word in one list to every word in the other list.

Our second, more effective method for extracting featured artists was to use the "strsplit" function in R, which splits a string before and after a specified substring. We input all possible featured artist linking words into this function and were able to obtain featured artists. We ran an exact matching on the radio play data with featured artists extracted and found that there were actually 664 unique radio artists. With two lists of unique artists, we were able to better calculate representative similarity metrics.

Methods: Calculating the Similarity Metrics

Before, calculating the similarity metrics we wanted to obtain an initial estimate of the true number of matches so we could tell how well the similarity metrics were performing. To do this we used exact matching and exact substring matching algorithms. For the exact matching, we coded a match if an artist name in the Next Big Sound List was exactly the same as an artist name in the radio play data. This method produced 51 matches. Upon examining these matches we initially feared that even an exact match could be a false positive, meaning a comparison which is predicted to be a match is actually a non match. We feared this because the artist "Nelly" was listed on The Next Big Sound chart in 2010. We did not believe this was the same well known "Nelly" listed on the radio data because Nelly became popular before 2010. However, upon checking the Next Big Sound site it proved that this was indeed a true match, and that Next Big Sound had included the well-known Nelly on their chart for some unknown reason.

The exact substring matching coded a match if the Next Big Sound artist name string was contained entirely within the radio play artist name string. This method produced 121 matches, and was even more likely than exact matching to produce false positives. For example, a band called "Fun" would be matched to a band called "Fun Guys who Like to Play Music," even though it is clear that these two artists are not a match. As longs as our metrics predicted around these two estimates we knew we were on the right track.

The next step was to calculate similarity metrics for each comparison. After deduplicating each list of artists there were 1,950,832 comparisons. We calculated Jaro-Winkler similarity scores and Levenshtein similarity scores for each comparison. Below is a detailed explanation of how these two similarity scores are calculated.

First, the Jaro distance between 2 strings s1 and s2 is defined as

$$d_j = \begin{cases} 0 & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|s1|} + \frac{m}{|s2|} + \frac{m-t}{m}\right) & \text{otherwise} \end{cases}$$

Where m is the number of matching characters and t is the number of transpositions. For example, when comparing the two strings "mean" and "mena" there is one transposition switching the "n" and the "a." Further, two characters are considered matching only if they are the same and not farther than

$$\left\lfloor \frac{\max(|s_1|, |s_2|)}{2} \right\rfloor - 1.$$

From the Jaro distance, the Jaro-Winkler score is defined as

$$d_w = d_j + (\ell p(1 - d_j))$$

Where d_j is the Jaro distance between the strings, l is the length of a common prefix at the start of the string up to a maximum of 4 characters, and p is a constant scaling factor for how much the score is adjusted upwards for common prefixes. In Jaro's work, this defaults to 0.1. Overall, the scores range from 0 to 1 and a higher Jaro-Winkler scores are more likely to be a match.

For Levenshtein distance, the distance between two strings is calculated as the number of single character edits required to turn one word into another, either as insertions, deletions or substitutions, and is defined mathematically as

$$\operatorname{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \operatorname{lev}_{a,b}(i-1,j) + 1 \\ \operatorname{lev}_{a,b}(i,j-1) + 1 \\ \operatorname{lev}_{a,b}(i-1,j-1) + [a_i \neq b_j] \end{cases} & \text{otherwise.} \end{cases}$$

This distance is then transformed into a similarity score as shown below. Overall the scores range from 0 to 1 and a higher score is better.

$$lev\ Similarity = 1 - \frac{lev\ distance}{length\ of\ the\ longer\ string}$$

In addition to the Jaro-Winkler score and the Levenshtein Similarity score we tried another metric we called a "common substring ratio," which was an extension of the largest common substring matching method we used to estimate the number of matches. This was used for comparing the artist names before extracting the featured artists. The common substring ratio was defined as the length of the common substring divided by the length of the radio artist string, the equation can be seen below. Using this metric, true non matches should be coded as 0, true matches as 1, and possible matches anywhere in between 0 or 1. However, for scores between 0 and 1 a higher score does not necessarily mean there is a higher chance of the two strings being a match. Therefore, this metric did not perform well.

$$CSR = \frac{number\ of\ characters\ in\ the\ substring}{number\ of\ characters\ of\ the\ radio\ artist\ string}$$

In order to test if the metrics could accurately predict matches was had to fit the similarity scores to a hand labeled subset of the data which will be discussed further in the next section.

Methods: Training Data

After all similarity scores were calculated we hand labeled a subset of the data as training data. Our initial hand labeled data set contained 980 non-matches, 20 exact matches, and 10 ambiguous matches as determined by our common substring ratio. Labeling the exact matches allowed us to confirm that the exact matches were true matches and not artists with the same name. Since the common substring ratio was not effectively picking out matches, we hand labeled a new subset of the data based on Levenshtein similarity scores. Our new hand labeled sample contained 577 record pairs, of which 30 were exact matches with Levenshtein similarity of 1, another 200 were non matches with Levenshtein similarity of 0, and 347 had a Levenshtein similarity higher than .59 but lower than 1. This final hand labeled sample contained 35 matches and 542 non-matches, and was used to fit all subsequent models.

Predictive Modeling:

Several models were used to fit the hand labeled subset of the data. First, logistic and probit regression models were applied to the dataset. Both gave the same results but the results of the logistic regression model are easier to interpret so we will continue with the logistic regression model. When all the similarity metrics, exact, Jaro-Winkler, and Levenshtein Similarity were used as explanatory variables to predict the probability of a comparison being a match they were all insignificant possibly due to multicollinearity. As can be seen from the correlation matrix below in Table 1, Levenshtein Similarity and Jaro-Winkler are highly correlated.

Table 1: Correlation Matrix

	Exact	Levenshtein Similarity	Jaro-Winkler
Exact	1		
Levenshtein Similarity	0.3947086	1	
Jaro-Winkler	0.2580396	0.8980048	1

Accordingly, a regression was run for each individual similarity metric. As can be seen below in Table 2, the Jaro-Winkler metric performed the best with the lowest p-value. The full output of the final logistic regression model with Jaro-Winkler used as the explanatory variable is displayed in Figure 1. The only concern is that the logistic regression model reported values of either 0 or 1 which can be risky since we are never absolutely certain.

Table 2: Logistic Regression Models

B_1 p-values	Exact	Levenshtein Similarity	Jaro-Winkler
Model 1	0.999	0.345	0.235
Model 2	0.99	-	-
Model 3	-	0.0476	-
Model 4	-	-	0.0146

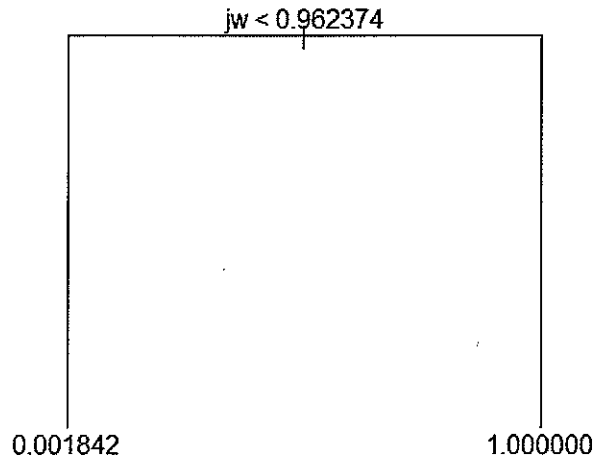
Figure 1: Logistic Regression Model

$$log(\frac{p}{1-p}) = \frac{e^{-168.62+175.63x_i}}{1+e^{-168.62+175.63x_i}}$$

```
Call:
glm(formula = id ~ jw, family = binomial(link = "logit"), data = labels)

Deviance Residuals:
Min        1Q    Median        3Q       Max
-1.25288  -0.00039   0.00000   0.00000   2.08032

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept)  -168.62      68.89  -2.448   0.0144 *
jw            175.63      71.95   2.441   0.0146 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 264.0073  on 576  degrees of freedom
Residual deviance:   8.8275  on 575  degrees of freedom
AIC: 12.828

Number of Fisher Scoring iterations: 14
```

Additionally, a classification tree and a random forest model were applied to the hand labeled subset of the data. The classification tree used the Jaro-Winkler scores of above 0.962374 to classify the matches as can be seen below in Figure 2.

Figure 2: Classifcation Tree



A high Jaro-Winkler score is not surprising given that fact that typographical errors were not a large concern given the high quality of the data from both sources. However, differences in punctuation such as accents and periods proved to be troublesome which the Jaro-Winkler score was able to pick up. A random forest model was also used to fit the subset of hand labeled data. In principle, random forest partitions the data into many random folds. Then a collection of classification trees are used to classify the data. However, random forest is often used for models with a large number of explanatory variables. Given that we only have three variables and the classification trees are only splitting on one score, the Jaro-Winkler score, random forest does not appear to be the best approach.

Results:

Overall, the performances of the logistic regression model and the classification tree were very similar. However, the classification tree performed slightly better when applied to the entire dataset. The performances were compared based on false positives and false negatives. A false positive is when the model classifies a comparison as a link but that comparison is not actually a match. A false negative is when the model fails to predict a true match. Ideally both these numbers should be small. The true number of matches was found by examining all comparisons with a Jaro-Winkler score of .6 or higher. By doing so a total of 57 matches were found. After identifying the true matches, both models were found to have zero false positives. However, the logistic regression model had one more false negative compared to the classification tree. As such, we will choose to use the classification tree for further match predictions.

Table 3: Logistic Regression Performance

	Match	Non-Matches
Linked	54	0
Unlinked	3	1950775

Table 4: Classification Tree Performance

	Match	Non-Matches
Linked	55	0
Unlinked	2	1950776

The models may have performed so well due to over-fitting. There are only 57 true matches out of about 1.9 million comparisons. In order to get significant results a large portion of the matches had to be used for the training data. Specifically, 35 matches of the total 57 were present in the training data. In order to get a true sense of the model's performance we could apply the model to a new set of data. Specifically, this model will be used to classify the radio data once it is updated with new radio data as the year goes on.

Conclusions and Discussion:

To reiterate, the Jaro-Winkler score proved to be the best similarity metric for comparing artist names. It did well matching artists names with mismatching punctuation such as accents and periods. Further, the best performing model was a classification tree which identified matches as those with a Jaro-Winkler score of over 0.962374. Our results could be improved by implementing a more effective initial deduplication process for each list of artists. After extracting the featured artists, we assumed the artist names would be consistent within in each list. However, 2 of the "true matches" were actually duplicates with different punctuation. Therefore, there are really only 55 matches in our dataset. For more accurate deduplication we could try applying the classification tree on each list of artists separately and see if it effectively deduplicates the lists before performing the cross list comparisons. However, one concern for a completely deduplicated initial list of artists is deciding which name is the accurate name for cross list comparisons. Another approach which may be more effective is to leave the duplicates as we have done already, and deduplicate after the cross comparison matches have been identified. For the time being further testing and analysis is required.

References and Acknowledgements:

"Analytics & Insights Forthe Music Industry." *Next Big Sound.* N.p., n.d. Web. 17 Oct. 2013. <http://www.nextbigsound.com/>.

Christine, Peter. "5.3 Edit String Comparison." *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection.* N.p.: Springer, 2012. N. pag. Print.

"DigitalRadioTracker.com - Your Complete Radio Airplay MonitoringSolution...."*DigitalRadioTracker.com - Your Complete Radio Airplay MonitoringSolution....*N.p., n.d. Web. 17 Oct. 2013. <http://digitalradiotracker.com/chart.html>.

Herzog, Thomas N., Fritz Scheuren, and William E. Winkler. "13 Strong Comparator Metrics for Typographical Error." *Data Quality and Record Linkage Techniques.* New York: Springer, 2007. N. pag. Print.

"Music Discovery Still Dominated by Radio, Says Nielsen Music 360 Report." *Music Discovery Still Dominated by Radio, Says Nielsen Music 360 Report.* N.p., 14 Aug. 2012. Web. 05 Apr. 2013.<http://www.nielsen.com/us/en/press-room/2012/music-discovery-still-dominated-by-radio--says-nielsen-music-360.html>.