RANKING LYRICS FOR ONLINE SEARCH

Robert Macrae
Centre for Digital Music
Queen Mary University of London
robert.macrae@eecs.qmul.ac.uk

Simon Dixon
Centre for Digital Music
Queen Mary University of London
simon.dixon@eecs.qmul.ac.uk

ABSTRACT

When someone wishes to find the lyrics for a song they typically go online and use a search engine. There are a large number of lyrics available on the internet as the effort required to transcribe and post lyrics is minimal. These lyrics are promptly returned to the user with customary search engine page ranking formula deciding the ordering of these results based on links, views, clicks, etc. However the content, and specifically, the accuracy of the lyrics in question are not analysed or used in any way to determine the rank of the lyrics, despite this being of concern to the searcher. In this work, we show that online lyrics are often inaccurate and the ranking methods used by search engines do not distinguish the more accurate annotations. We present an alternative method for ranking lyrics based purely on the collection of lyrics themselves using the Lyrics Concurrence.

1. INTRODUCTION

Multiple copies of song lyrics are available on the internet for almost any song. Due to this free availability, search engines have become the common tool for finding lyrics. As lyrics are relatively easy to mine from the web, and given that the words to a song contain rich semantic information, lyrics are also used for information retrieval such as for karaoke data production, song-browsing, and thumbnailing [2, 3, 12, 15, 16]. The content of a song's lyrics can indicate the topic of the song [4], which genre it belongs to [13], or be used for music indexing and artist similarity [8]. Another example of lyric based information retrieval uses natural language processing to extract language, structure, categorisation, and similarity from lyrics [11].

A contributing factor to the abundance of lyrics and a potential problem for research in this area is the lack of requirements, such as training or language knowledge, that are typically necessary for professionally annotating lyrics. Due to these issues, there is a high potential for song lyrics to contain errors. This can lead to inaccurate lyrics being presented to those using search engines to find lyrics as well as music information retrieval researchers who wish to mine the rich semantic content within lyrics.

In this paper we are concerned with ranking web based song lyrics. Whilst previous work has focused on using multiple sequence alignment to determine the single most accurate lyrics for a song [5, 6], ours is concerned with ranking lyrics, so that users can apply their own selection should the first result not be appropriate. To the best of our knowledge, lyrics ranking has only previously been attempted as part of more generalised web resource ranking methods [14]. In order to evaluate song lyrics ranking we first describe a test data set for this purpose and we then proceed to mine the web for lyrics of the songs in this dataset. We then formulate a metric to compare each lyric to the ground truth, as an accuracy measurement, and to other versions to calculate the Lyrics Concurrence, an adaptation of the Chords Concurrence and Structure Concurrence used to rank guitar tablature [10]. We then adapt the ranking methods outlined previously to evaluate these methods by measuring their correlation with the lyrics' accuracy.

2. TEST DATA: THE MUSIXMATCH DATASET

The Million Song Dataset (MSD) [1] is a collection of metadata for a million popular music tracks [1] produced by LabROSA in collaboration with The Echo Nest. A subset of this data, called the musicXmatch Dataset (MXMD), [2] consists of 237,662 lyrics to songs within the MSD provided in a Bag-of-words format with the 5000 most common (stemmed) words.

2.1 Bag-of-words Format

The Bag-of-words format (BOW) is primarily a means of summarising text by listing the unique words with the number of occurrences of each word in the text, with all punctuation removed. These word and count pairs are ordered by their count with the most common coming first. For example:

"On mules we find two legs behind and two we find before. We stand behind before we find what those behind be for."

can be represented in BOW format as:

"we:4, find:3, behind:3, two:2, before:2, on:1, mules:1, legs:1, and:1, stand:1, what:1, those:1, be:1, for:1"

[1] http://labrosa.ee.columbia.edu/millionsong/
[2] http://labrosa.ee.columbia.edu/millionsong/musixmatch

Additionally the words are stemmed [9] so that words with different endings are reduced to their root form, reducing the number of unique words. Using this BOW format avoids copyright issues with sharing lyrics for the purposes of research.

3. LYRICS MINING

For each of the 237,662 tracks in the MXMD we searched DogPile [3] for lyrics using the following terms: "<artist><song>lyrics -video". DogPile was chosen as it returns results from all the popular search engines and yet is more easy to data mine. Previous web mining approaches have used the Google Web API in a similar fashion [5, 6], however we required a search engine with an unrestricted number of searches. From the list of URLs returned by this search we selected only those that contained the song title in the URL. This set of URLs provides a similar representation of the URLs a user might select when manually searching for lyrics. 888,745 URLs were found using this method for the 237,662 tracks. In order to extract the lyrics from the URLs we separated and analysed each line to determine whether it contained lyrics-like text and then selected the longest sequence of lyrics-like text lines in the page. Any lyrics that were less than three lines or over 200 lines long were discarded. As we are interested in comparing with Concurrence, we discarded songs and their lyrics if they had less than three lyrics associated with the song. The lyrics extraction process is demonstrated in Figure 1.

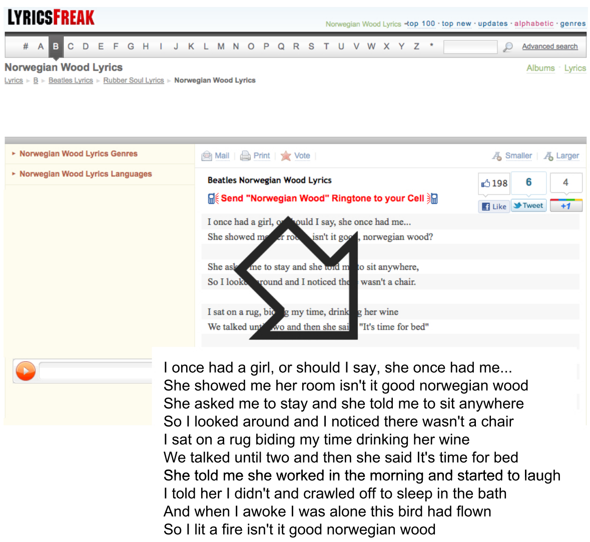

Figure 1. An example lyrics web page and the lyrics extracted from it.

4. LYRICS EVALUATION METRICS

In this section we give an overview of the metrics used in judging the accuracy and similarity of lyrics. The first method, Levenshtein Edit Distance, is a well known Dynamic Programming method for comparing strings. We use the Levenshtein Edit Distance to judge the similarity of lyrics and this is used later in the Lyrics Concurrence ranking method.

[3] http://www.dogpile.com/

4.1 Levenshtein Edit Distance

The Levenshtein Edit Distance (LED) [7] counts the number of "edits" required to transform one string into another. An edit is classed as an insertion, deletion, or substitution of a single character. LED uses a cost of 0 for matches and 1 for any edit (insertion, deletion or alteration). As such the LED of "sun" and "sing" is 2 (substitution of the letter 'u' for 'i' and insertion of the letter 'g'). The LED cost is found by calculating a path $P(U,V) = (p_1, p_2, ..., p_W)$ through a matrix of costs between strings $U = (u_1, u_2, ..., u_M)$ and $V = (v_1, v_2, ..., v_N)$. This cost matrix is described as $d_{U,V}(m,n)$ where $m \in [1:M]$ and $n \in [1:N]$ where each position in the path is designated as $p_k = (m_k, n_k)$. A simple bottom-up algorithm for calculating the LED in $O(N^2)$ time and space is shown in Algorithm 1. In this example a matrix of edit costs is calculated between two strings, so that the cell in the final row and column would contain the total number of required edits. Additionally, an example of the "cost matrix" and the solution this algorithm produces can be seen in Table 1.

Input: String A, String B
Output: Levenshtein Edit Distance LED
Matrix m; $m[0,0] := (A[0] == B[0]?\ 0 : 1)$;
for $a \in [1..A.length]$ **do**
  $m[a,0] := (A[a] == B[0]?\ 0 : 1) + m[a-1,0]$;
end
for $b \in [1..B.length]$ **do**
  $m[0,b] := (B[b] == A[0]?\ 0 : 1) + m[0,b-1]$;
end
for $a \in [1..A.length]$ **do**
  for $b \in [1..B.length]$ **do**
    $m[a,b] := (A[a] == B[b]?\ m[a-1,b-1] : 1 + min(m[a-1,b], m[a-1,b-1], m[a,b-1]))$;
  end
end
return $LED := m[A.length, B.length]$;

Algorithm 1: The Levenshtein Edit Distance.

4.2 Lyric Accuracy (LA)

In order to calculate the accuracy of the lyrics we first convert the lyrics to the BOW format with the 5000 most common stemmed words (as designated by the MXMD set) using the same stemming code the MXMD set used. We describe the ground truth MXMD BOW $G = (g_1, g_2, ..., g_M)$ and the lyrics BOW $L = (l_1, l_2, ..., l_N)$ as sets of word (w_i) and count (x_i) pairs where $g_i = (w_i, x_i)$. Each word in the ground truth BOW G is looked for in the lyrics BOW L so that if a match is found *i.e.* $g_m(w) = l_n(w)$. Therefore each ground truth word yields an expected word count $g_m(x)$ and a found word count of $l_k(x)$ if the word was present in the lyrics BOW and 0 if not. If the found word

String A: all the other kids
String B: with their pumped up kicks

	a	l	l	t	h	e	o	t	h	e	r	k	i	d	s
w	**1**	2	3	4	5	6	7	8	9	10	11	12	13	14	15
i	2	**2**	3	4	5	6	7	8	9	10	11	12	12	13	14
t	3	**3**	3	3	4	5	6	7	8	9	10	11	12	13	14
h	4	4	**4**	4	3	4	5	6	7	8	9	10	11	12	13
t	5	5	5	**4**	4	4	5	5	6	7	8	9	10	11	12
h	6	6	6	5	**4**	5	5	6	5	6	7	8	9	10	11
e	7	7	7	6	5	**4**	5	6	6	5	6	7	8	9	10
p	8	8	8	7	6	**5**	5	6	7	6	6	7	8	9	10
u	9	9	9	8	7	6	**6**	6	7	7	7	7	8	9	10
m	10	10	10	9	8	7	7	**7**	7	8	8	8	8	9	10
p	11	11	11	10	9	8	8	8	**8**	8	9	9	9	9	10
e	12	12	12	11	10	9	9	9	9	**8**	9	10	10	10	10
d	13	13	13	12	11	10	10	10	10	**9**	9	10	11	10	11
u	14	14	14	13	12	11	11	11	11	**10**	10	10	11	11	11
p	15	15	15	14	13	12	12	12	12	11	**11**	11	11	12	12
k	16	16	16	15	14	13	13	13	13	12	12	**11**	12	12	13
i	17	17	17	16	15	14	14	14	14	13	13	12	**11**	12	13
c	18	18	18	17	16	15	15	15	15	14	14	13	**12**	12	13
k	19	19	19	18	17	16	16	16	16	15	15	14	13	**13**	13
s	20	20	20	19	18	17	17	17	17	16	16	15	14	14	**13**

Table 1. An example of a Levenshtein Edit Distance (LED) requiring 13 edits (with spaces removed).

count is greater than the expected word count, the found count is replaced as the expected count minus the difference or 0 if this difference is greater than the expected count. The LA is calculated as the sum of the found word counts divided by the sum of the expected word counts multiplied by 100 and divided by the sum of the ground truth counts *expected*, so as to be expressed as a percentage. Equation 1 shows this calculation and Table 2 shows an example of the LA measurement.

$$\text{LA}(G,L) = \frac{\sum \max(g_m(x) - |g_m(x) - l_k(x)|, 0)}{\sum g_m(x)} \times 100 \tag{1}$$

Ground Truth: "Are we human or are we dancer? My sign is vital, my hands are cold"
Lyrics: "Are we human or are we dancers? My signs are vital, my hands are cold"
Lyrics Accuracy (LA): $(12/14) \times 100$ = 85.7%
(wrong count for "is" and wrong count for "are")

Table 2. Lyrics Accuracy (LA) example.

4.3 Lyrics Similarity (LS)

The Lyrics Similarity is a measure of how similar two lyrics, L_1 and L_2 are. We use the LED of the entire sequence of characters in both lyrics, not stemmed and with all the punctuation included. We convert the LED to a similarity score by normalising to the perfect score, then inverting and multiplying by 100 to give a value from 0 to 100:

$$\text{LS}(L_1, L_2) = \left(1 - \frac{\text{LED}(L_1, L_2)}{\max(L_1, L_2)}\right) \times 100 \tag{2}$$

For the Lyrics Ranking experiments we additionally tried a variation of the LS called LS_{ns} where spaces are removed from the input lyrics L_1 and L_2. The incentive for removing spaces is that, as the average english word length is 5 characters, spaces make up roughly $\frac{1}{6}$ of the text and possibly contain less relevant information than the rest of the text. As the LED has quadratic costs, reducing the input sequences by $\frac{1}{6}$ reduces the processing time and memory requirements of this method by 31%.

Lyrics 1: "On mules we find two legs behind and two we find before."	
Lyrics 2: "We stand behind before we find what those behind be for."	
Lyrics Similarity (LS):	43.8%
Lyrics Similarity no spaces (LS_{ns}):	45.7%
Lyrics 1: "Are we human or are we dancer? My sign is vital, my hands are cold' '	
Lyrics 2: "Are we human or are we dancers? My signs are vital, my hands are cold"	
Lyrics Similarity (LS):	92.9%
Lyrics Similarity no spaces (LS_{ns}):	90.9%
Lyrics 1: "Scaramouche, Scaramouche, will you do the Fandango"	
Lyrics 2: "Scallaboosh, Scallaboosh, will you to the banned tango"	
Lyrics Similarity (LS):	69.1%
Lyrics Similarity no spaces (LS_{ns}):	66.0%
Lyrics 1: Radiohead - High and Dry (azlyrics.com/lyrics/radiohead/highdry.html)	
Lyrics 2: Jamie Cullum - High and Dry (azlyrics.com/lyrics/jamiecullum/highanddry.html)	
Lyrics Similarity (LS):	86.6%
Lyrics Similarity no spaces (LS_{ns}):	86.0%

Table 3. Lyrics Similarity (LS) examples.

5. LYRICS STATISTICS

The final list of lyrics included 358,535 lyrics for 67,156 songs with an average Lyrics Accuracy of 38.6%. The distribution of the lyrics over these songs can be seen in Figure 2. This lyrics distribution shows a quick drop off in the number of lyrics per song after the songs with less than three lyrics were removed. The range of lyrics accuracy results can be seen in the histogram in Figure 3. The large number of low accuracy lyrics and the low average Lyrics Accuracy suggest the lyrics mining procedure failed to filter out all the non-text lyrics, however, this is not a trivial task for users browsing the web either and so we allow these non lyrics to be considered within the ranking method experiments as one purpose of these methods is to differentiate between lyrics and non-lyrics. In Section 7.1 we examine the possibility of removing these non-lyrics to judge their effect on the ranking experiments. Table 4 shows the top twenty lyrics domains based on their average Lyrics Accuracy. The increase in Lyrics Accuracy of these domains over the average suggests that a simple filter restricting the results to known accurate lyrics domains would remove most of the non-lyrics.

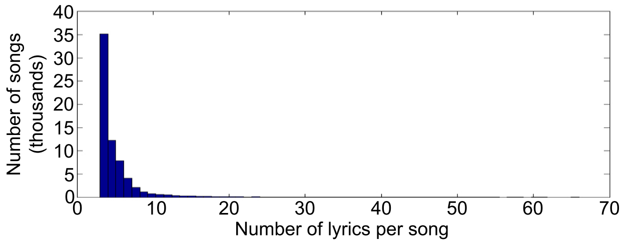

Figure 2. A histogram showing the distribution of lyrics for the 61,755 songs.

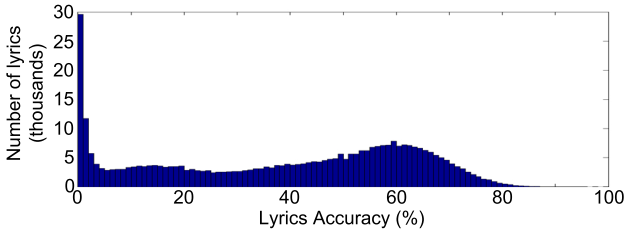

Figure 3. A histogram showing the distribution of the lyrics accuracies.

LA	Domain	Lyrics
55.82%	www.alivelyrics.com	123
52.75%	www.sing365.com	15798
52.53%	www.popular-lyrics.com	142
52.43%	www.plyrics.com	127
52.34%	www.musicsonglyrics.com	3307
52.33%	www.lyricspond.com	535
52.25%	www.songteksten.nl	1178
51.97%	www.lyricsdepot.com	3301
51.93%	www.azlyrics.com	7006
51.30%	www.1songlyrics.com	253
51.11%	www.absolutelyrics.com	1360
51.02%	www.lyricsondemand.com	2909
50.85%	www.sarkisozum.gen.tr	138
50.72%	www.christian-lyrics.net	167
50.62%	www.lyricsdomain.com	925
50.57%	www.lyricstop.com	235
50.084%	www.cowboylyrics.com	1656
49.26%	www.lyriczz.com	682
49.08%	www.lyricsreg.com	1877
49.01%	www.lyricmania.com	155

Table 4. Average accuracy rates for different lyrics domains.

6. LYRICS RANKING METHODS

The following methods describe how we apply the ranking methods to the lyrics.

6.1 Search Engine Results Page Rank

The lyric's Search Engine Results Page Rank (SERP Rank) corresponds to where the URL of the lyric is found in the ordered list of DogPile's ranked search results. Values range from 1 (best) to 100 (worst known), as our mining was restricted to the top 100 results (see Section 3). All the lyrics were mined using DogPile and as such had an associated SERP Rank.

6.2 Date Modified

The Date Modified value is expressed as the number of milliseconds since 00:00:00 January 1, 1970 GMT. 137,875 of the 358,535 lyrics had an associated last date modified that was greater than 0. Any value of 0 is ignored as it was presumed that such a date was unknown.

6.3 Lyrics Concurrence

To determine the extent to which lyrics of songs agree with a set of lyrics, we measure the Lyrics Concurrence as the average of the Lyrics Similarities between a lyric L_k and the other lyrics of the same song $L_i (i \neq k)$.

$$\mathrm{LC}(L_k) = \sum_{i=1, i \neq k}^{n} \mathrm{LS}(L_k, L_i)/(n-1) \qquad (3)$$

6.4 Lyrics Concurrence NS (LC$_{ns}$)

Additionally, we measure the Lyrics Concurrence No Spaces as the average of the LS$_{ns}$ between a lyrics' L_k and the other Lyrics of the same song $L_i (i \neq k)$.

$$\mathrm{LC}_{ns}(L_k) = \sum_{i=1, i \neq k}^{n} \mathrm{LS}_{ns}(L_k, L_i)/(n-1) \qquad (4)$$

7. LYRICS RANKING EVALUATION

In order to measure correlation we use two alternative measurements, the Pearson Product-Moment Correlation Coefficient (PCC), and the Spearman's Rank Correlation Coefficient (SCC). Table 5 shows the correlations found between the lyrics LA and the 4 ranking methods described above. Figure 4 shows scatter graphs of the accuracy and rank of the lyrics using two of the methods: SERP Rank and Lyrics Concurrence. The correlations show the Lyrics Concurrence having the strongest correlation, the SERP Rank having a weak correlation (the negative correlation is expected as lower values indicate a better SERP Rank) and the Date Modified having a very low correlation. Comparing LC and LC$_{ns}$ we find that discarding spaces improves the correlation slightly therefore LC$_{ns}$ improves performance in both accuracy and efficiency. The results of this experiment show that analysing the content of the metadata in comparison to the other metadata available leads to

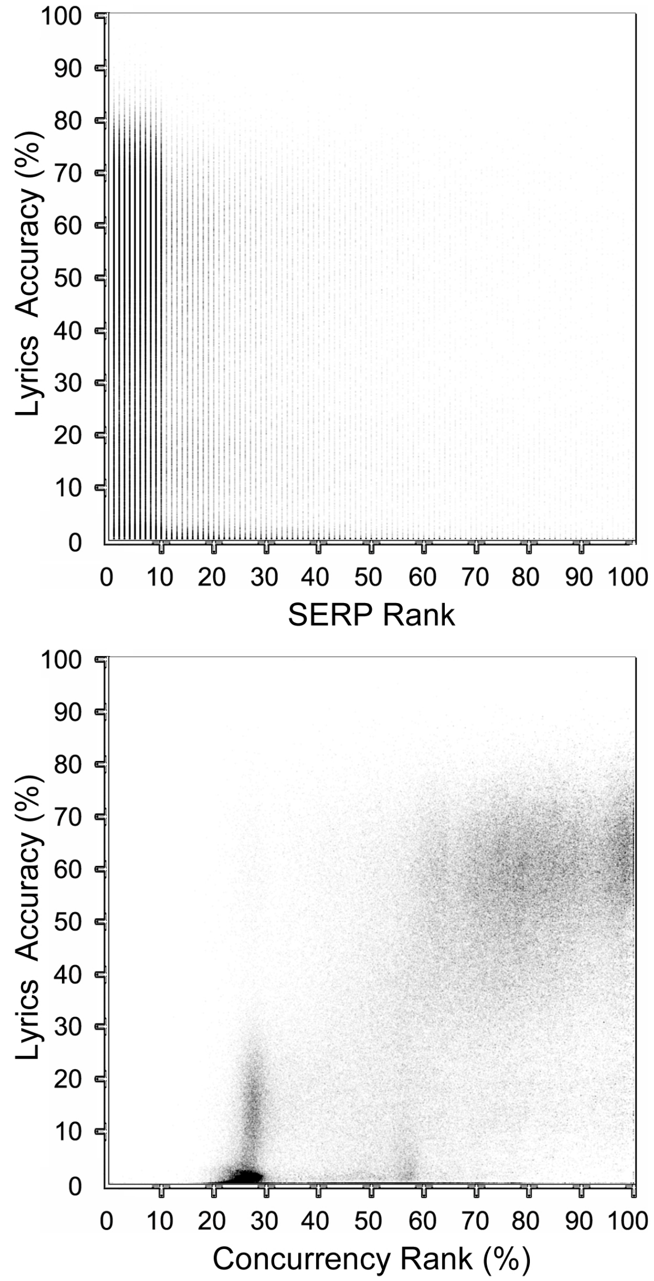

Figure 4. Scatter graphs showing the trends between LA and respectively the SERP Rank (above) and Lyrics Concurrence (below) on 358,535 lyrics.

a better ranking system than methods based on user statistics and link analysis or the date modified.

Ranking Method	PCC (r)	SCC (ρ)	Samples
	LA		
Lyrics Concurrence	0.654	0.607	358535
Lyrics Concurrence NS	0.657	0.609	358535
SERP Rank	-0.0806	-0.0890	358535
Date Modified	0.016	0.012	137875

Table 5. Number of samples and correlation values between various ranking methods and the Lyrics Accuracy (LA).

7.1 To What Extent do Non-Lyrics Affect Ranking Correlations?

As mentioned previously, the Lyrics data contains many files that are not lyrics at all (as is evident from the dark cluster of low accuracy results in Figure 4) and this may affect the correlations. We therefore repeat the ranking methods experiment excluding the files that have a Lyrics Accuracy of less than 10%, the results of which are shown in Table 6. The ranking methods all see a reduction in the correlation between rank and Lyrics Accuracy. However, this difference also suggests that the methods could be used to help distinguish lyrics from non-lyrics.

Ranking Method	PCC (r)	SCC (ρ)	Samples
	LA		
Lyrics Concurrence	0.477	0.477	289346
Lyrics Concurrence NS	0.484	0.484	289346
SERP Rank	-0.0891	-0.0891	289346
Date Modified	0.009	0.033	107661

Table 6. A modified version of Table 5 showing correlation values between various ranking methods and the Lyrics Accuracy (LA) without the lyrics with an LA of less than 10%.

7.2 Is Lyrics Concurrence Dependent on Sample Size?

To see if the number of lyrics available for a particular song effects the correlation of Lyrics Concurrence with lyrics Accuracy, we calculate the correlation between N (the number of lyrics for a particular song) and C (correlation between LA and LC) for each of the 61,755 songs. The result, 0.074, is not statistically significant for the sample size, suggesting that Lyrics Concurrence, like Chord Concurrence is a relevant indicator of accuracy providing the sample size is at least 3 as is the case in these tests.

7.3 Song Lyrics Detection

We also attempt to use the lyrics ranking methods as lyrics detection systems by taking the highest ranking lyrics for each of the 61,755 songs. Table 7 shows the average accuracy of the ranking methods. Of the ranking methods, the Lyrics Concurrence is the most successful feature for selecting the most accurate lyrics to use.

Detection Method	Lyrics Accuracy
Lyrics Concurrence	47.3%
Date Modified	43.6%
SERP Rank	42.5%
Randomly Selected	38.6%

Table 7. The average Lyrics Accuracy of the top ranked lyrics over 61,755 tracks (41,614 tracks in the case of Date Modified as 38.5% of the lyrics don't have an associated date). The final row shows the average as if the lyrics were randomly selected.

8. DISCUSSION

In this paper we have examined the need for greater ranking of online music metadata and proposed a solution to this problem. The Lyrics Concurrence is a method for ranking music lyrics based on the similarity of its lyrical content to other lyrics of the same song. The rationale of the Concurrence factor is that the correctness of metadata, is determined by agreement of expert human annotators. We have shown that Lyrics Concurrence is a reliable indicator of accuracy, providing a greater correlation with the accuracy of the lyrics than the date modified or SERP Rank. During the time of this experiment there were no ratings available for the lyrics, however, some lyrics websites have started to incorporate this feature. User ratings can act as an additional ranking method and future work could compare this method with those evaluated here, however, a similar study found user ratings to be a poor ranking method for guitar tablature [10].

It is hoped that the Concurrence ranking method can be utilised in search engines to ensure that accurate annotations are ranked more favourably, although the computational costs involved in comparing hundreds of lyrics with each other may limit the usage of such a technique to offline cases. Future ranking methods might focus on combining Concurrence with SERP Rank, User Rating, or linking lyrics with other sources of metadata such as chords, in order to improve the correlation of the ranking with the accuracy. Such an approach may allow a more complete annotation of a different type to fill out any missing or abbreviated segments by repeating the aligned section.

9. REFERENCES

[1] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The Million Song Dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, pages 591–596, 2011.

[2] Hiromasa Fujihara, Masataka Goto, Jun Ogata, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno. Automatic Synchronization between Lyrics and Music CD Recordings Based on Viterbi Alignment of Segregated Vocal Signals. In *Proceedings of the 8th IEEE International Symposium on Multimedia (ISM)*, pages 257–264, Washington, DC, USA, 2006.

[3] Min-Yen Kan, Ye Wang, Denny Iskandar, Tin Lay Nwe, and Arun Shenoy. LyricAlly: Automatic Synchronization of Textual Lyrics to Acoustic Music Signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):338–349, 2008.

[4] Florian Kleedorfer, Peter Knees, and Tim Pohle. Oh Oh Oh Whoah! Towards Automatic Topic Detection In Song Lyrics. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, pages 287–292, September 2008.

[5] Peter Knees, Markus Schedl, and Gerhard Widmer. Multiple Lyrics Alignment: Automatic Retrieval of Song Lyrics. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR)*, pages 564–569, London, UK, September 2005.

[6] Jan Korst and Gijs Geleijnse. Efficient Lyrics Retrieval and Alignment. In *Proceedings of the 3rd Philips Symposium on Intelligent Algorithms (SOIA)*, Eindhoven, the Netherlands, December 2006.

[7] V. I. Levenshtein. Binary Codes Capable of Correcting Deletions, Insertions, and Reversals. *Soviet Physics*, 10(8):707–710, February 1966.

[8] Beth Logan, Andrew Kositsky, and Pedro Moreno. Semantic Analysis of Song Lyrics. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, volume 2, pages 827–830, Baltimore,Maryland, USA, 2004.

[9] Julie Beth Lovins. Development of a Stemming Algorithm. *Mechanical Translation and Computational Linguistics*, 11:22–31, 1968.

[10] Robert Macrae and Simon Dixon. Guitar Tab Mining, Analysis and Ranking. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, pages 453–458, 2011.

[11] Jose P. G. Mahedero, Álvaro MartÍnez, Pedro Cano, Markus Koppenberger, and Fabien Gouyon. Natural Language Processing of Lyrics. In *Proceedings of the 13th Annual ACM International Conference on Multimedia (ACM-MM)*, pages 475–478, New York, NY, USA, 2005. ACM.

[12] Matthias Mauch, Hiromasa Fujihara, and Masataka Goto. Lyrics-to-Audio Alignment and Phrase-Level Segmentation Using Incomplete Internet-Style Chord Annotations. In *Proceedings of the 7th Sound and Music Computing Conference (SMC)*, 2010.

[13] Rudolf Mayer, Robert Neumayer, and Andreas Rauber. Rhyme and Style Features for Musical Genre Classification by Song Lyrics. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, 2008.

[14] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Stanford InfoLab, 1999.

[15] Ye Wang, Min-Yen Kan, Tin Lay Nwe, Arun Shenoy, and Jun Yin. LyricAlly: Automatic Synchronization of Acoustic Musical Signals and Textual Lyrics. In *Proceedings of the 12th Annual ACM International Conference on Multimedia (ACM-MM)*, pages 212–219, New York, NY, USA, 2004.

[16] Chi Wong, Wai Szeto, and Kin Wong. Automatic Lyrics Alignment for Cantonese Popular Music. *Multimedia Systems*, 12:307–323, 2007.