Research Proposal

Capstone Project

# 

# 

# 

# Arthur Lubambo

Nov 2023

sbs23001@student.cct.ie

# Title

**Evaluating the Potential of Similarity Estimation Using Hyperloglog for Query by Humming Problem**

# Research Objectives

The main goal of this research is to assess the potential of HyperLogLog Data Structure in Query by Humming. For that, there are the following Research Objectives:

**Develop a Hyperloglog-based Similarity Estimation Framework for Query by Humming** - It includes the adaptation of traditional string matching algorithms for similarities to use the Jaccard index, overlapping index estimation based on Hyperloglog (HLL) data structure, developing a framework for HLL parameters tuning for query by humming problem.

**Investigate the relationship between memory usage vs performance in HyperLogLog-based Similarity Estimation -** it includes running experiments to discover the impact of memory reduction and performance, establishing modelling for the relationship between the hyper-parameters and accuracy. Analysing the hyper-parameter's impact on memory usage and execution time. Finetuning it will bring the key value for the proposed method and its benefits.

**Compare and contrast Hyperloglog-based Similarity Estimation Framework with other traditional algorithms performance by metrics -** it includes experimentation and comparison about the weakness and strongness of each type of algorithm, such as dynamic time warping (DTW), Scaling and Time Warping (SWM). The key objective is to discover if the optimizations in performance (time and memory execution) will affect negatively or NOT the accuracy of the task (MRR).

# 

# 

# Literature Review

## Query By Humming

Query by Humming (QBH) is an area of Music Information Retrieval that studies the content-based search in the song database. Humming serves as a natural and efficient method for searching a musical audio database by vocalizing the melody of a song. It was originally studied by (Asif Ghias et al., 1995), using the pitch movements “Up”,” Down”, and “Same”, to encode the query, and substring searching in a database of 183 songs was conducted using parameters to accept a controlled level of mismatch. The big challenge at that time was encoding the audio signal in a string (‘U’, ‘D’, ‘S’), due to computational power and algorithms available, taking up to several minutes to process 10 seconds of audio. This Research opens the area of QBH it was quite interesting how a simple method using a UDS encoding could produce good results.

Later, other approaches were introduced to this problem such as dynamic time warping (DTW) a technique used in Times Series, but it could be modelled specifically for the audio processing context. For example (Fu et al., 2007) used the pitch level over time to model it as a times series using the dynamic programming technique DTW to calculate the matching level. Even with the optimizations, it has a quadratic cost to calculate the match between two series where n is the length of the series.

(Tripathy et al., 2009) they followed a very similar approach to the original (Asif Ghias et al., 1995) it created a layer for Wav to MIDI from the query and then used an algorithm for string matching, but they used a different method using dynamic programming to calculate the edition distance between strings.

In general, all the methods try different approaches to deal with the mismatching between what the song is, with the humming sung by a human. It can be observed that it is not only a

time alignment between the query and the song. It is also a pitch alignment problem as was highlighted by (Stasiak, 2012), who proposed a method inspired by auto-adaptive human behaviour for ignoring errors in sung melodies.

All the algorithms were focused on the accuracy of the method so far, but there is another fundamental aspect to enable the implementation of it in practical applications, scalability. It is a natural question, to understand how to apply it in a large database. (Guo et al., 2013) Introduced the application of Locality Sensitive Hashing (LSH), a technique that creates an index based on hashes operations considering the similarity between the content. They had applied this method developing other layers to tackle the problem of key transposing, it is when the music and the query are not in the same tone. So it was more scalable and improved the performance with better mean reciprocal rank (MRR).

(Alexios Kotsifakos et al., 2011) proposed a subsequence matching framework capable of dealing with gaps either in the query or the target song, performing better than the other Dynamic Programming methods, and maintaining the same time complexity.

Considering the diversity of queries, (Wang and Jyh-Shing Roger Jang, 2015) started to extract other type information from the queries that were not being used before, the lyrics. It is quite common besides humming the people to sing part of the song, or even whistle. So this study explored how the usage of lyrics in addition to the melody might impact the performance of QBH. They have combined speech recognition techniques to extract the lyrics of the humming combining it with the melody distance, reducing the error rates significantly. For obvious reasons, it would just impact music and queries with lyrics. The other challenge of this method is the language context, to make it generalizable it is necessary to have multiple languages trained, and lyrics structured for all the songs. So it implies extra effort in its implementation.

A common problem with the songs is the complex extraction of the melody in an automated way. The original studies were based on existing MIDI databases with the songs already in the symbolic format. Producing it algorithmically through raw audio format is more complex, the song usually the song has multiple instruments, with harmony and a combination of more than one melody at the same time. With this challenge in mind, (Alfaro-Paredes, Alfaro-Carrasco and Ugarte, 2021) used a voice separation to improve the melody extraction from the songs, it demonstrated better results for the encoded melody for the song, consequently improving the matching algorithms.

Most of the studies were trying to use improvements in the string match algorithm to consequently have better results with the QBH, but (Velankar and Parag, 2018) changed the paradigm, introducing the matching using n-grams and inverted index.

They combined it with the Mean Normalized Frequency (MNF) Algorithm and developed a method called “Unified Algorithm for Melodic Music Similarity”. The big advantage of this method is that it tackles the information retrieval reducing the search space for the songs where the n-Grams matches. So, the pre-computed indexes help the reduction of the query time. The N-Gram contains each segment of intervals for the song, so even with imperfections on pieces of the query encoding is unlikely to compromise the result.

As a counterpoint to (Velankar and Parag, 2018) the work from (Ulfi and Mandala, 2022) highlighted that the “Unified Algorithm” has issues with performance and works slow for big datasets. They also implement the algorithm “Query by Humming System using Frequency-Temporal Attention Network” but apply enhancements to the Partial Matching of queries on it. In the end, it concludes the method has problems with scaling for big datasets. It might indicate a gap to be explored by applying adjustments in the algorithms for scalability efficiency.

In General, all of the algorithms are based on two areas: melody extraction from the song and query; and melody matching. All of them had a different approach to encode the melody, to extract it, or to match it. Some of the methods also tackled the search problem, creating an index to retrieve the relevant candidates more efficiently. The encoding format of the melody varied between two categories: Discrete Sequence, or Continuous Sequence. Some other researchers explored multi-media format, using lyrics in addition to the typical format.

### Melody Extraction

The melody extraction or Transcription is the name of a task that transforms the musical audio signal in a temporal pitch notation with the sequence of notes. As mentioned before, it is an essential step for the QBH algorithms.

The most recent application of Machine Learning techniques on it shifts the usage from fully algorithmic solutions to data-driven models. For instance, (Yu et al., 2021) used a method inspired by the human perception of frequency, time and intensity for audio, applying Convolutional Neural Networks to extract the melody. Or (Donahue, Thickstun and Liang, 2022) use generative models to create the synthetic dataset and combine it with annotated datasets to improve the melody transcription. Or (Bittner et al., 2022) who proposed a lightweight neural network for musical transcription, it is compatible with polyphonic outputs and could be applied to a wide range of instruments and vocals.  
  
We could see some of the methods, explicitly or implicitly used source separation for better results in the QBH, It is especially important as a pre-processing layer for the melody extraction phase. (Défossez, 2022) used a combination of Frequency Domain with Time Domain, introducing the method “Hybrid Spectrogram and Waveform Source Separation”, it is based on deep neural networks architecture.

The advantage under-explored from Source Separation for QBH is the ability to encode more than one simultaneous melody. It might improve the multi-vocal, contrapoint songs. allowing the matching with the query to be on any of the melodic lines.

### Dataset

The dataset for QBH might be a challenge to produce, it needs to be done with a carefully designed procedure, to avoid bias introduction. It has been found (Salamon, Serrà and Gómez., 2012) a database with 118 recordings of sung melodies, used in multiples studies from this Literature Review. It was built from 17 subjects, keeping a good gender balance and a wide level of music knowledge level, from zero to amateur musicians. They were presented with a list of songs, and they were free to pick the ones they knew and were asked to sing any part of the melody for recording. There was no restriction on time, or what part of the melody must be sung, they were free to sing with or without lyrics. They didn’t listen to the original song before recording it. All the records were made from a simple microphone from a laptop in order to simulate a realistic scenario for QBH. This dataset is robust given all the experimentation designs and the wide usage in academic research.

## Probabilistic Algorithms and Data Structures

In QBH a big challenge is to enable more accurate results, with algorithms that scale well for big datasets. For instance, Spotify Dataset has more than 100 Million Tracks (Spotify, 2023). It is necessary to have more efficient methods for QBH. This Project will explore the potential of Probabilistic Algorithms and Data Structures in the QBH Task.

One of the most impactful data structures created was the HyperLogLog. (Flajolet et al., 2007) did enhancements in the original loglog algorithm for the cardinality estimator, making it possible to estimate the size of sets beyond 10^9 with a standard error of 2% while using a memory of only 1.5 kilobytes. This data structure creates multiple registers of hashes and computes operations of add, count and merge (two HLLs). The extremely low memory has a cost with the accuracy, but it can be tunned by a parameter that increases the number of registers. The application of this Data Structure in Big Data was disruptive, making it possible the count distinct elements estimation in a distributed system extremely quickly. (Heule, Nunkesser and Hall, 2013) made improvements to the method increasing the accuracy and reducing even more memory usage.

As mentioned previously, there were some methods that used locality-sensitive hashing LSM (Guo et al., 2013) (Matti Ryynänen and Anssi Klapuri, 2008) and MinHash to index the songs for QBH. Although there are improvements, in the MinHash algorithm called HyperMinHash (Yun William Yu and Weber, 2020), there is a good opportunity to investigate if this new implementation would impact the performance of the QBH.

Also, one of the applications of HLL could be inclusion coefficient estimation as (Nazi et al., 2018) demonstrate in an efficient method applied in database columns schema analysis. This coefficient could be defined as the fraction of the intersection of two sets relative to the smaller set. This concept can be used as an inspiration for another context. For example in the QBH, the matching between the query and the song can be modelled as the inclusion coefficient index between the melodic encodings of the query and song, using it as a set of keys. So the challenge is to transform into how to encode it with relevant information for matching it. In addition to it, the HyperLogLog methods could be used to optimize the execution, creating a scalable implementation for it.

# 

# Proposed primary research methodology

The primary research methodology employed in this study centred around experimental investigations. The main objective is to evidence the cause-and-effect relationship between the application of the proposed model and the query-by-humming performance metrics. The goal is to demonstrate that enhancements in the algorithm's execution did not affect the query-by-humming performance negatively.

In many other areas, this type of data structure is demonstrating improvements in the tasks. The Literature Review demonstrated a lack of studies with the HyperLogLog applied to Query By Humming, so this research aims to explore this approach and evaluate the results. The methodological framework was structured across 3 phases: data preparation, implementation utilizing the proposed approach with fine-tuning and experimentation, and results analysis comparing the new framework against alternative methods.

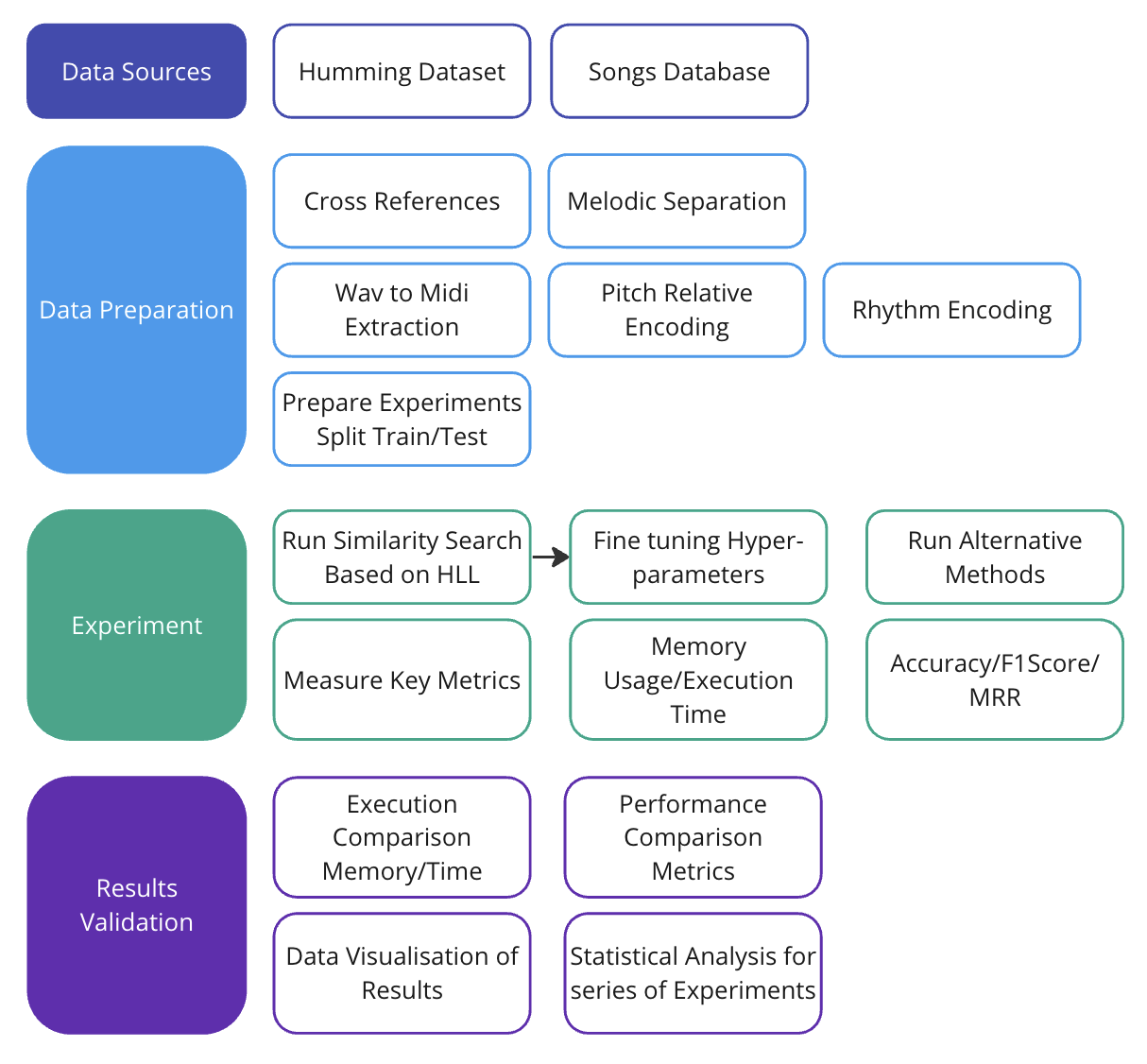


Fig 1 Diagram with Research methodology

The initial stage of the Research will be identifying suitable secondary datasets for the primary research of this capstone project. It is a key area, It is important to choose reliable sources, with representativeness, and good usage in other studies. Then the dataset will be cleaned and prepared for the next phases. The cleaning consists of standardising the song identifiers between the query dataset identification and song identification. It will enable the data to be crossed between both datasets.

Then the melodic extraction for the song dataset will be executed, It is very important because the audio tracks from the songs might include multiple instruments, and vocals, so the algorithm needs to mark from all simultaneous pitches which of them represents the main melody.. Then the Wav to MIDI algorithm will transcribe the pitches of the melody for both datasets. Alternatively, in case the challenges of finding a suitable raw WAV dataset with the songs contained, it can be used an existing symbolic song dataset with the Midi transcriptions already done. The focus of this research is not the melodic extraction phase of the algorithm, instead, it is the encoding and retrieval part of songs in the database using HLL hashes.

With the Midi transcriptions ready, two operations for Relative Pitches Sequence and Rythmical Sequences extraction need to be applied to create the Discrete Features Set for each song or query. The relative pitches are important because each person might sing in a different note the same melody, also the rhythmic encoding must be robust to different speeds of the song. Note this algorithm could be extended to any type of high-dimensional discrete feature sequence for a song.

Then the dataset needs to be prepared for the experiment, splitting into multiple folds for using a cross-validation strategy, finishing the Data Preparation Stage.

The Experimental Phase: Prepare each model candidate for the analysis, including multiple variants of the proposed method similarity estimator based on HLL, Overlapping Coefficient and other alternatives such as dynamic time warping (DTW), Scaling and Time Warping (SWM). Each model will be executed multiple times for each Fold.

Each execution will be evaluated against memory usage, execution time, accuracy, and Mean Reciprocal Rank (MRR). After collecting the results, the data visualization of the experiment results will be implemented, and the statistical analysis with the Hypotheses Test will be executed. The comparison will be done by highlighting the takeaways from the experiment results. The first key comparison will be between different hyper-parameters of the model, and the second key comparison, will be the same model but using the typical Set Cardinality estimation instead of HLL. Then at last compare with the other algorithms of the same problem.

# 

# Sampling Strategy

This project will use a Non-Probabilistic Sampling method, more specifically judgment sampling. It will assess the quality of the procedure for the data involved, the context, the representativeness of male and female voices, from different levels of musical knowledge, and also use data set with a reputation between other studies.

# Ethical and Risk Considerations

The ethical considerations for this project are based on Ethics and data protection (HAYES and KUYUMDZHIEVA, 2021). It is clear that the data involved in this research is NOT sensitive, as does NOT deal with data concerning children, or vulnerable people. The data used has a Creative Commons Attribution 4.0 International license. It is NOT invasive neither put at risk any kind of freedom of the participants. The source organization that collected the data, had the consent of the people to use the data for this type of research.

There is no sensitive Personal Identifiable Information (PII) such as Sex Orientation, Race, Religion, or similar. The data involved is already anonymised. Given the nature of the data involved, it is not necessary to implement of the DPIA process. The Data was collected in Europe and does not have any PII, so There is no Transfer of personal data to non-European countries at all.

It is important to highlight a possible risk of misusage of the algorithm developed in this research, such as plagiarism detection. It would be an unfair and unethical usage if it were implemented as an automated decision-making. Any decision made by this algorithm must be reviewed by a human. For example: In case someone uses the proposed similarity algorithm in an automated system for automated plagiarism detection, the responsible for operating it must ensure to have a human review for each positive detection, to consider this application ethical. The risk will mitigated by informing in public repository of good practices of usage of this method.

Any software, or library used in this research will be following a proper license such as MIT, Apache License, BSD license or equivalent. In the case of proprietary software or cloud platform usage, the proper permission will be in place. Regarding the Sampling Method, The intrinsic bias involved will be managed by explicitly representativeness and relevance of data selection with clear criteria and justification. For each stage of the research will be reviewed the proper usage and reduction of bias using a proper technique from the data preparation, until data visualization and Results presentation.

After taking all these measures and in case any other ethical concern was not properly detected, any external person can report it through GitHub issue, and adequate measures will be in place.

The context this project will be based on is Western music theory (12 pitches), so for example it would NOT be adequate to use it for microtonal music.

Applying the method developed on this project in real projects must be considered carefully once depending on the context it might exclude songs from other cultures to have accurate results. Due to the limitations of datasets and time constraints, it will be tackled with clear advice about the usage.

# 

# References

Alexios Kotsifakos, Panagiotis Papapetrou, Jaakko Hollmén and Dimitrios Gunopulos (2011). A Subsequence Matching with gaps-range-tolerances Framework. *Proceedings of the VLDB Endowment*, 4(11), pp.761–771. doi:https://doi.org/10.14778/3402707.3402716.

Alfaro-Paredes, E., Alfaro-Carrasco, L. and Ugarte, W. (2021). Query by Humming for Song Identification Using Voice Isolation. *Springer eBooks*, 12799(978-3-030-79463-7), pp.323–334. doi:https://doi.org/10.1007/978-3-030-79463-7\_27.

Asif Ghias, Logan, J., Chamberlin, D. and Smith, B.C. (1995). Query by Humming. *Multimedia*. doi:https://doi.org/10.1145/217279.215273.

Bittner, R., Bosch, J., Rubinstein, D., Meseguer-Brocal, G. and Ewert, S. (2022). *A LIGHTWEIGHT INSTRUMENT-AGNOSTIC MODEL FOR POLYPHONIC NOTE TRANSCRIPTION AND MULTIPITCH ESTIMATION*. [online] Available at: https://arxiv.org/pdf/2203.09893.pdf.

Défossez, A. (2022). *Hybrid Spectrogram and Waveform Source Separation*. [online] arXiv.org. doi:https://doi.org/10.48550/arXiv.2111.03600.

Donahue, C., Thickstun, J. and Liang, P. (2022). *MELODY TRANSCRIPTION VIA GENERATIVE PRE-TRAINING*. [online] Available at: https://archives.ismir.net/ismir2022/paper/000058.pdf [Accessed 11 Nov. 2023].

Flajolet, P., Fusy, É., Gandouet, O. and Meunier, F. (2007). HyperLogLog: the analysis of a near-optimal cardinality estimation algorithm. *Discrete Mathematics & Theoretical Computer Science*, DMTCS Proceedings vol. AH,...(Proceedings). doi:https://doi.org/10.46298/dmtcs.3545.

Fu, A.W.-C., Keogh, E., Lau, L.Y.H., Ratanamahatana, C.A. and Wong, R.C.-W. (2007). Scaling and time warping in time series querying. *The VLDB Journal*, 17(4), pp.899–921. doi:https://doi.org/10.1007/s00778-006-0040-z.

Guo, Z., Wang, Q., Liu, G. and Guo, J. (2013). A Query by Humming System Based on Locality Sensitive Hashing Indexes. *Signal Processing*, 93(8), pp.2229–2243. doi:https://doi.org/10.1016/j.sigpro.2012.09.006.

HAYES, B. and KUYUMDZHIEVA, A. (2021). *Ethics and Data Protection*. [online] Available at: https://ec.europa.eu/info/funding-tenders/opportunities/docs/2021-2027/horizon/guidance/ethics-and-data-protection\_he\_en.pdf.

Heule, S., Nunkesser, M. and Hall, A. (2013). HyperLogLog in Practice. *Association for Computing Machinery*, (9781450315975), pp.683–692. doi:https://doi.org/10.1145/2452376.2452456.

Matti Ryynänen and Anssi Klapuri (2008). Query by humming of midi and audio using locality sensitive hashing. doi:https://doi.org/10.1109/icassp.2008.4518093.

Nazi, A., Ding, B., Narasayya, V. and Chaudhuri, S. (2018). Efficient Estimation of Inclusion Coefficient Using Hyperloglog Sketches. *Proceedings of the VLDB Endowment*, 11(10), pp.1097–1109. doi:https://doi.org/10.14778/3231751.3231759.

Salamon, J., Serrà, J. and Gómez., E. (2012). MTG-QBH: Query By Humming dataset. *zenodo.org*, [online] 2. doi:https://doi.org/10.5281/zenodo.1290712.

Spotify. (2023). *about Spotify*. [online] Available at: https://newsroom.spotify.com/company-info/#:~:text=Today%2C%20more%20listeners%20than%20ever.

Stasiak, B. (2012). Follow That Tune - Dynamic Time Warping Refinement for Query by Humming. *Signal Processing: Algorithms, Architectures, Arrangements, and Applications*.

Tripathy, A., Chhatre, N., Surendranath, N. and Kalsi, M. (2009). Query by Humnming System. *FULL PAPER International Journal of Recent Trends in Engineering*, 2(Query by humnming system).

Ulfi, M. and Mandala, R. (2022). *Improving Query by Humming System Using Frequency-Temporal Attention Network and Partial Query Matching | IEEE Conference Publication | IEEE Xplore*. [online] ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/document/9933001 [Accessed 12 Nov. 2023].

Velankar, M. and Parag, K. (2018). Unified Algorithm for Melodic Music Similarity and Retrieval in Query by Humming. In: *Intelligent Computing and Information and Communication*. Intelligent Computing and Information and Communication. AISC, pp.373–381. doi:https://doi.org/10.1007/978-981-10-7245-1\_37.

Wang, C.-C. and Jyh-Shing Roger Jang (2015). Improving Query-by-Singing/Humming by Combining Melody and Lyric Information. *IEEE/ACM transactions on audio, speech, and language processing*, 23(4), pp.798–806. doi:https://doi.org/10.1109/taslp.2015.2409735.

Yu, S., Sun, X., Yu, Y. and Li, W. (2021). Frequency-Temporal Attention Network for Singing Melody Extraction. *arXiv (Cornell University)*. [online] doi:https://doi.org/10.1109/icassp39728.2021.9413444.

Yun William Yu and Weber, G.M. (2020). HyperMinHash: MinHash in LogLog Space. *IEEE Transactions on Knowledge and Data Engineering*, pp.1–1. doi:https://doi.org/10.1109/tkde.2020.2981311.