20 YEARS OF AUTOMATIC CHORD RECOGNITION FROM AUDIO

Johan Pauwels[1] **Ken O'Hanlon**[1] **Emilia Gómez**[2] **Mark B. Sandler**[1]
[1] Centre for Digital Music, Queen Mary University of London
[2] Music Technology Group, Universitat Pompeu Fabra
{j.pauwels, k.o.ohanlon, mark.sandler}@qmul.ac.uk, emilia.gomez@upf.edu

ABSTRACT

In 1999, Fujishima published *Realtime Chord Recognition of Musical Sound: a System using Common Lisp Music*. This paper kickstarted an active research topic that has been popular in and around the ISMIR community. The field of Automatic Chord Recognition (ACR) has evolved considerably from early knowledge-based systems towards data-driven methods, with neural network approaches arguably being central to current ACR research. Nonetheless, many of its core issues were already addressed or referred to in the Fujishima paper. In this paper, we review those twenty years of ACR according to these issues. We furthermore attempt to frame current directions in the field in order to establish some perspective for future research.

1. INTRODUCTION

This year marks the twentieth anniversary of Fujishima's [17] seminal ACR system. In this work the author proposed the calculation of a 12-D chroma feature which gets compared to a dictionary of binary chord templates. The label of the most similar chord template is then considered to be the chord output. Fujishima also proposed exploiting the temporal continuity of chords by smoothing the chromas across time to produce less noisy labels, and suggested that musical information could be exploited. This system outline created the framework within which much of the early research on ACR would happen.

Perhaps the most striking evolution in ACR has been the move from knowledge-driven to data-driven systems. Initially, data-driven elements were used as one-for-one replacements, or additions, of elements to the framework set by Fujishima. Examples are more sophisticated chroma features, learnt Gaussian chord models or HMM-based temporal models. More recently ACR research has been dominated by Deep Learning (DL). In DL-ACR the relationship of tasks and elements becomes blurred as systems have become more integrated. This is perhaps most evident in the system of McFee and Bello [40], which appears superficially to be a singular unit, although closer inspection reveals convolutional filters providing short-term context, recurrent elements modelling musical language, chroma, and auxiliary targets for extra musical context. We therefore consider that the system outline provided by Fujishima is still valid in a discussion of ACR.

ACR systems have developed considerably in these twenty years and commercial products providing such functionality have recently been developed. However, there is still room for improvement. For instance, while ACR results have continually progressed, complex chords are still recognised less well than major and minor triads. Nonetheless, ACR results have improved to the point where the ambiguity in chord labels is now becoming a topic of research. Such ambiguity can be introduced by the chord labels themselves, or may result from different interpretations of chord and melody. An interesting aspect of modern ACR research derived from user research in commercial products is the prediction of user interpretations in the presence of ambiguity.

In the rest of this paper we review previous and current research in ACR. Based on such review, we propose several areas that have potential for improvement in ACR. We first consider the related features and chord models, before discussing temporal and musical context. We furthermore consider ambiguity and subjectivity in ACR, and problems associated with chord vocabularies before concluding.

2. PROBLEM 1: FINDING AN APPROPRIATE FEATURE REPRESENTATION

The chroma feature has been perhaps the most influential idea in ACR, and much early ACR research focused on producing chroma variants. Typically the chroma feature is calculated by summing the energy of elements of the same pitch class in a preliminary pitch spectrum feature. In Fujishima's paper, and subsequent others, the pitch feature was calculated by gathering the energy under windows of a spectrogram that are positioned logarithmically in frequency. Later approaches used a constant-Q transform (CQT) [4] which places windows on a multi-scale spectrogram, affording higher resolution of low frequencies, at the cost of lower temporal resolution. Such pitch-based spectra are more compact than linear spectrograms and afford simple summation in the chroma calculation, a process referred to as pitch folding. Indeed, pitch folding of logarithmic spectra into chroma features is perhaps the most important technical aspect of Fujishima's paper, as it

affords a low-dimensional semantically meaningful feature that is easily derived. However, the log-frequency spectrum may result in corruption of harmonic structure as e.g. the second overtone of a root note is located in a pitch bin labelled with the class of its perfect fifth. Such windowing also results in bins of higher pitch being of large frequency width, and possibly containing energy from several pitch class sources. However, even in the presence of such potential sources of error the pitch folded chroma feature has endured as a staple of ACR.

Many chroma variants have been designed to attempt to counter the negative effects of pitch folding. In the simplest case, a spectral weighting is employed that lowers the effect given to higher pitches, which are more likely to contain misassigned harmonics. Other researchers have applied extra weight based upon the harmonic structure in the spectrogram, for instance the harmonic product spectrum is windowed and folded in the Enhanced Pitch Class Profile [34] and in [41]. Alternatively note spectral templates are employed through convolution [18] and for spectral decomposition [38] although the log-frequency windowed spectrogram has generally prevailed. Other chroma refinements through pitch feature manipulation have been proposed. Placing less effect on the relative coefficients in the pitch feature, thereby inducing a level of timbre-invariance, is encouraged by regularisation using e.g. log compression [42] which is seen to be useful for ACR [10].

Researchers have attempted to create chroma with sharper definition. High-resolution spectral methods such as parabolic interpretation [18] and spectral reassignment [27, 46] have been used in order to sharpen a spectrogram before windowing. Likewise, sharpness of features can be diminished when a piece is not tuned to the standard 440Hz, and tuning estimation is often performed to counter this problem [18, 21]. Alternatively, higher dimensional chroma have been employed to avoid these pitfalls [18,59].

In DL-ACR systems, the network is expected to learn any necessary weightings or transforms from training on data. DNNs are often employed as direct classifiers, and for ACR can be trained with one-hot chord class vectors [3,14,15,61,68]. Using chroma feature targets has become a popular design choice in ACR [30,31]. As in more traditional ACR methods, CQT and other log-frequency spectra are seen to predominate as input features [24, 30, 40, 61]. Different inputs have occasionally been employed, e.g. the Harmonic CQT is employed in [64]. Dimensionality reduction of input data has been explored with principal component analysis (PCA) applied to a spectrogram [3] and to a CQT [68]. It is unclear whether PCA was useful other than for dimensionality reduction, and its related decrease in computational expense. Tuning is often ignored, possibly with the assumption that the data-driven systems have the capability to learn to deal with tuning errors. Some exceptions to this are the inputs used in [14,15], while training data of a CNN-based ACR system is augmented using detuning in [31].

So far there is a lack of comparative evaluation of the effects of the input feature in DL-ACR, with most papers instead focussing on the effects of later steps in the processing chain. In the particular case where chroma targets are employed in DL-ACR, the DNN can be seen simply as a replacement for e.g. tuning, compression, weighting and pitch folding. It is no surprise that fast convergence is observed in such DNNs, although rarely reported. Few alternatives to logarithmic spectra have been considered, unlike other music processing tasks where raw audio input observed has been examined [56]. We propose that the use of linear spectra for DL-ACR should be explored. Linear spectrograms do come with the caveat that invariance to pitch-shift is lost, a property that is attractive for use with CNNs. However, overtones of a given root frequency can be expected to be found approximately equidistant from each other in a linear spectrogram, a structure that should also be extractable. In the log-frequency spectrogram, such overtone structure is not so simply presented because harmonics of orders that do not possess an integer base-2 logarithm are misassigned in pitch class. One can consider this a form of information loss that may be detrimental to ACR, particularly when a large vocabulary of chords is employed and some exploration of spectrogram-based training should be undertaken. Alternatively, a multiple input DNN system could be employed using both CQT and Fourier spectrograms.

3. PROBLEM 2: DEFINING WHAT A CHORD LOOKS LIKE IN FEATURE SPACE

Chord classification requires models of chords that can be compared to a given feature such as chroma. Originally, the use of binary chroma templates for classification was proposed in tandem with the pitch-folded chroma [17]. In such a binary chroma template each pitch expected to be active in a chord is set to one. Comparison of a chroma vector with a dictionary of binary templates affords a simple classification approach to chord recognition, with each chroma vector labelled according to the most similar template. Most ACR research has similarly focussed on the comparison of chroma-based features and chord models. The most notable early alternative chord feature is the tonal centroid, or tonnetz, feature [20] which is actually a partial Fourier Transform of a chroma feature. Binary templates formed the basis of much early ACR research [21] and many chroma estimation methods [33, 38] implicitly targeted outputting chroma vectors similar to binary vectors, a goal that can still be seen in DNNs when chroma targets are employed [30, 64]. Binary templates possess the ability to model a chord when there is no representative data available, and despite their simplicity are often effective [10]. However, binary templates may be unrealistic as the effects of misasssignment of harmonics in chroma features are ignored.

An alternative perspective is to place less emphasis on manipulating chroma vectors, and create templates that are more similar to the expected data. A template can be synthesised as in [48, 50], where chord templates are formed by summing spectra of several note templates, parameterised by a number of harmonics and a roll-off factor

determining the energy in each harmonic [19]. This approach has largely been overlooked as data-driven methods that naturally learn models akin to the data became popular. Early data-driven models learnt chord models from labelled chroma features, with several different model types explored. Perhaps the most common of these have been multivariate Gaussian (mixture) models, a generative approach learnt from data, which have been applied in the context of chroma features [10, 59] and tonal centroid features [35], and also in the context of DNN features [24] Other chroma based classifiers have included support vector machines [41, 57], which have also been applied to DNN chroma outputs [68], and random forests [25].

Other research has looked beyond chroma, considering chord classification on the full pitch spectrum. A linear full spectrum classifier is applied in [8] while logistic regression is used in [30] where it is seen to improve on a similar classification applied to chroma features. Such results indicate some limitations for ACR of the chroma feature, which may lose useful information in the pitch folding process. The chroma representation is compact and octave-invariant, qualities that have allowed templates and simple Gaussians to be effective in ACR. Such approaches may be less viable in the context of extra spectral information, with data-driven methods more likely to be able to deal with the extra information coming from the increased dimensionality. The octave-invariance also prevents the chroma feature from distinguishing between different inversions of a given chord. A bass chroma calculated from a window on the lower octaves of the pitch feature was introduced in [39] in order to capture inversions. Extending this, a three windowed chroma with Gaussian models was proposed in [9] approximating full-spectrum analysis.

In DL-ACR there are different options for training and classification. Aside from chroma target outputs, DNNs can also employ more standard black-box classification vectors where each element denotes the likelihood of a different chord [14, 15, 68] or can even be trained using both chroma and label targets [3]. Furthermore, chord classification with DNNs may be performed using activations of the penultimate layer of the network as a feature, which may be passed to a subsequent network [3, 31, 61].

A new perspective is seen in the most recent DL-ACR methods, which are being trained to learn extra information encoded through the use of auxiliary targets. This can be clearly seen in the works [40] and [64] where alongside the chroma feature, one-hot feature targets representing the bass note [40, 64], root note [40], highest pitch note [64] alongside a distinctive no-chord target [40] have been introduced to network training. Such approaches, which we call *target label engineering* should be of benefit in ACR. More possibilities for auxilary targets may exist. This target label engineering displays a turnaround in ACR feature research; where traditionally the focus has been on manipulating features close to the input, it seems that more attention may now be given to the target labels, and the sorts of information that ACR researchers might like to extract.

4. PROBLEM 3: THE MISMATCH BETWEEN PROCESSING RATE AND CHORD RATE

To locate chords in time, feature representations consist of multiple time-localised frames. The simplest way is to create features at a constant, but arbitrary rate, which determines the processing rate of the entire system. This approach was taken by Fujishima, whose features had a rate of 3.906 Hz. In this case, it is important to make the rate high enough, because it effectively imposes a time grid onto which all chord labels are projected, and making this grid too coarse leads to misalignment at the chord boundaries. Therefore later approaches generally increased this feature rate to the order of tens of Hz.

The rate of chord changes is typically an order of magnitude higher than the frame rate. Although the exact number depends on the music piece, the Isophonics dataset used for MIREX has an average chord change rate of 0.46 Hz for example [52]. Especially in the traditional methods where the features are processed frame-wise, the high frame rate compared to the chord rate leads to a high risk of fragmented chord output. A number of techniques have therefore been tried to enforce temporal continuity between frames, starting with a simple smoothing filter. Fujishima himself used a mean filter, but median [47] filters have also been used. These smoothing filters can be applied either to chord probabilities [48] or to the features themselves [2] to remove noise such as percussive sounds and non-harmonic melody notes. In the latter case, a chord model then works on a smoother feature representation, which indirectly also leads to a less fragmented chord output.

The drawback of blindly applying a smoothing filter is that chord transitions may be smeared, leading to inaccuracy in boundary estimation and will cause short chords to be smoothed out even if they are very apparent in the signal. A better method is therefore to consider the chord matching outputs as the observations of an HMM and smooth them with the Viterbi algorithm [2, 5, 59]. The strength of each of the chord candidates is then taken into account and the diagonal elements of the transition matrix control the probability of a chord change [2]. The Viterbi smoothing has been shown to outperform filtering of the chroma output, and filtering of the feature representation does not bring any additional benefits [10].

A consequence of using a HMM for smoothing is that the imposed duration distribution takes the shape of a geometric distribution, meaning that the shorter a chord, the more likely it becomes. Since this obviously is not a realistic distribution, the usage of duration-explicit HMMs has been explored, which allow arbitrary duration distributions. The shape of the distribution was not found to have a major influence on the results, however [8].

An advantage of modern, recurrent neural network (RNN) based approaches is that the chord duration distribution is learnt automatically as part of the chord modelling, without further intervention. RNNs [3] or, more recently, long short-term memory (LSTM) units [16, 61, 64] are fed the feature frames one by one, but remember previ-

ous input which can be used in the prediction of the current frame. With the advent of bidirectional LSTMs [16, 64], future frames can now also be taken into account.

Feedforward networks do not learn a chord duration distribution, as they have no memory elements, but achieve temporal stability by processing multiple time frames at once instead of isolated frames [22]. The entire local environment can then be used to learn the label of the middle frame, which includes learning to integrate over time and to avoid short disturbances. A context window of 1.5 s was found to be optimal in [30]. Of course, convolutional-recurrent models [40] combine the benefits of both.

Following sporadic usage of conditional random fields (CRFs) in traditional systems [5], the combination of CRFs and deep learning is gaining popularity [31, 64]. In this case, a CRF is stacked onto a neural network as the last layer to smooth the output, where one advantage is that they can be trained together for maximum discriminability.

Another way of handling the difference between feature and chord rates is to reduce the discrepancy from the start. This can be accomplished by segmenting the time axis of the feature representation in a musically meaningful way. A first option is to use the output of a beat-tracker to resample the feature representation onto a beat-synchronous grid [2, 36, 60]. The underlying assumption is that chords only change at beat times. The features can then be smoothed over the inter-beat interval without risk of blurring the chord boundaries. It has been shown, however, that beat-synchronous processing may not be advantageous compared to smoothing the chord output with an HMM [10]. A potential reason is the reliance of this method on a correct beat estimation, which was not as common at the time these experiments were performed as it is today. A certain benefit of beat-synchronous features is that the processing rate of systems built around them is lower (0.33–1 Hz for 60–180 BPM), so they run faster.

A final possibility is to explicitly determine chord boundaries before attempting to identify the chords they delineate [13, 20]. This way, the features or chord output can be maximally smoothed without drawbacks, but determining a good chord segmentation function is hardly an easier problem. A recent deep learning approach consists of two stages [64], the first determines the chord segmentation and triad using a small vocabulary, while the second stage picks the final chord type from a larger vocabulary.

Going forwards, we note that blind feature segmentation has been amply used by neural network approaches, but musically meaningful segmentation has not. Although one of the advantages of deep learning is that a network can come up with the most optimal feature representation itself, making the input to the network more musically explicit would be worth investigating. Feeding beat-synchronous features into a network would allow it to learn chord duration distributions expressed in terms of beats instead of frames, which could be more expressive. Such an approach would accumulate the errors of the beat tracking though, so multi-task learning [65] where a single network jointly learns to predict multiple outputs might be better. Beat-tracking could be learnt together with chord recognition. In case intermediate features are desired, chroma could be calculated together with chord segmentation.

5. PROBLEM 4: ACHIEVING LONG-TERM CONSISTENCY IN CHORD SEQUENCES

Multiple chords in a sequence do not follow each other randomly, but exhibit strong temporal links. Typical chord patterns have emerged throughout history, which have been studied by scholars such that expert knowledge about them is available [58]. ACR systems have been trying to incorporate this knowledge, initially with expert-based approaches, nowadays driven by data.

Implementation-wise, chord sequence consistency is mostly dealt with together with duration modelling of a single chord, as discussed in Section 4. In HMM-based approaches, the off-diagonal elements of the transition matrix determine where a chord change will lead to, whereas the diagonal elements influence when it takes place. The sources of knowledge for chord change information are different from the ones used for duration though. The doubly nested circle of fifths has been used frequently [2, 49] (or abused, as it is a model for key similarity) as well as other expert theories [39, 54].

The chord change probabilities of those probabilistic models have also been determined through corpus analysis [43, 54], before deep learning approaches became popular [31, 40, 61, 64, 68]. Typical for deep learning is that chord changes can be handled together with chord duration and models (problems 2, 3 and 4) by a single network [40, 64] to maximally exploit their mutual information. Nonetheless, some of the proposed approaches find it beneficial to handle the problems separately [32], in a way reminiscent of earlier knowledge-based systems.

Taking into account chords beyond the directly adjacent ones remains a challenge. A standard HMM can only model preceding chords indirectly, through chaining bigrams, so increasing the Markov order to trigrams and 4-grams has been tried [26], but their overall improvement remained small. As for deep learning techniques, the typical receptive field of feedforward neural networks is too small (1.5 s for [30]) to contain multiple chord changes and learn long-term dependencies. Recurrent neural networks have the theoretical advantage that all previous and/or future frames can be remembered, but vanishing gradient problems restrict their long-term memory in practice [1]. Attention mechanisms, which are used in machine translating to remember the context of long phrases, are potential solutions to this problem. The Transformer architecture [63] is one candidate for future exploration.

Since long-term chord dependencies are so hard to take into account with the aforementioned techniques, a couple of alternatives have been proposed that rely on musical structure. Feature representations of a repeated section have been averaged in order to make them more stable [11, 37] and that average has then been used for all instances of that section. A more probabilistic version of this idea has been explored in a statistical-relational frame-

work [51]. While these techniques do not exactly use long term dependencies to improve chord recognition, at least they ensure that repeated sections are consistent.

6. PROBLEM 5: EXPLOITING RELATIONSHIPS WITH RELATED MUSICAL CONCEPTS

Chords are just one way of describing musical content. Other musical concepts such as key, genre, bass or melody describe different aspects of the same piece of music. Several research efforts have incorporated ACR in systems that recognise multiple musical concepts, jointly or in sequence, to exploit the mutual information between them. We've already discussed the use of beat information to address Problem 3, so this section focusses on other concepts.

Practically, we see that probabilistic graphical models such as HMMs [35, 54] or dynamic Bayesian networks (DBNs) [39] have been the dominant approach so far to integrate related concepts into ACR. Their inherent modularity is exploited to decompose the probabilistic relations between concepts into more comprehensive factors.

Arguably the musical concept most related to chords is the musical key, as the latter also describes the harmony in a music piece, albeit on a longer temporal scale, i.e. a key spans multiple chords in sequence. Assigning different probabilities to all combinations of keys and chords is therefore a way to exploit their relationship. These probabilities can be derived from musical knowledge [39, 54] or data-driven [26, 35, 54]. One advantage of extracting keys with chords is that the chord sequence can be expressed relative to the key, in a representation close to functional harmony analysis, which has been shown to have a higher information density [58]. The required keys to make this possible can be derived prior to [26] or jointly with chord recognition [35, 54]. However, it has not been proven that using key-independent relative chord representations lead to improvements in actual chord recognition.

Downbeat is also related to chords in the sense that chords are more likely to change on downbeats than on other beats. This link has been exploited by making chord transition probabilities depend on the metric position of a beat in a measure. One example is a joint downbeat-chord system [50] that can deal with different time signatures and added or deleted beats. Another approach included beat-dependent chord transitions as part of a larger system that involves chord inversion and key as well [39], but which is limited to the 4/4 time signature. In both cases, the probabilities were determined by expert knowledge.

Other relationships between musical concepts can be exploited in a similar way. Bass notes, for instance, can be strongly indicative of the chord being played. They have the advantage of being comparatively easy to identify in a spectrum and are used as such to inform ACR [62, 67]. A joint key-chord-structure system has been proposed [53] based on the hypothesis that certain chord sequences are more likely at the start or end of high-level structures such as chorus or verse. Finally, different genres have different idiomatic chord sequences, so genre-dependent context modelling has also been examined [34, 44].

None of the recent deep-learning approaches have involved other musical concepts so far. In theory, it would be possible to identify the key segments of a piece beforehand and then transpose all segments of all pieces into one single key, instead of augmenting the feature representation as in [22, 40]. However, just like feeding beat-synchronous representations into a network, this would suffer from errors in the preceding key recognition step. Most likely a better solution would be to create specific networks that recognise chords together with any of the discussed concepts in a multi-task learning process such as [65]. Unfortunately, data annotated with multiple concepts is even less available than data that is annotated with just chords.

7. PROBLEM 6: HANDLING AMBIGUITY AND SUBJECTIVITY

Fujishima's system was tested using a strongly controlled setup. His dataset consisted mostly of clean, well-defined chords produced by an electronic keyboard using three different presets. He furthermore tailored his settings to these specific sonorities, but noticed that these settings didn't translate well to real music. Indeed, when we want to apply chord recognition to real music, the situation is very different from a controlled environment.

An enormous variety of music exists that may not even contain chords. Music can be monophonic, from a musical tradition that doesn't know the concept, or polyphonic without containing chords (e.g. a fugue). Even if chords are present in a music signal, it can also contain many sources – such as untuned percussion – that disturb the perception of a chord in a listener, human or mechanical. Furthermore, what exactly contributes to a chord is also ill-defined. The distinction between chord sequence and melodic line is not clear cut and can be a matter of opinion. Also the granularity of a chord sequence, for instance whether fast approach chords or anacrusis (pickup) chords are transcribed, depends on the user or use-case. A final cause of ambivalence is when chords are only implied, not audible as such, as with arpeggiated chords for instance.

Because all of these reasons human annotators aren't in total agreement when it comes to transcribing chords. The reported agreement on the root of a chord lies between 76% [29] (4 annotators) and 94% [12] (2 annotators) on average, but large outliers towards the bottom can appear in individual files [23, 45]. When comparing algorithmic output to a single human reference output, it is therefore hard to tell if any disagreements are valid alternative interpretations or outright errors. Although this phenomenon has been diagnosed and quantified [23, 45] multiple times already, final solutions are yet to emerge. The availability of chord datasets with multiple annotations, such as [45] (20 songs annotated by 5 persons) and [29] (50 songs annotated by 4 persons) is certainly a first step on the way. That former dataset has been used to learn the idiosyncrasies of different annotators and create personalised algorithmic output tailored to their preferences [28]. While it does provide a means of personalisation, it requires example transcriptions for each user, which might not be avail-

able. Furthermore multiple viewpoint annotations do not answer the question of whether an algorithmic output differing from all N viewpoints is plain wrong or the $N+1$th valid viewpoint. A different way of evaluating will be necessary for this, but it will probably be hard to scale. Asking human experts whether an automatic transcription is a good chord analysis (even if not exactly their own) would work, but would also mean that every different setting of an algorithm needs to be evaluated manually. Even by crowdsourcing the process, it would be impossible to test multiple parameterisations of a system.

8. PROBLEM 7: CHORD VOCABULARY AND ASSOCIATED BALANCE PROBLEMS

Fujishima's system was able to distinguish no less than 27 different chord types, but subsequent systems quickly reduced this number to two [2] or four [21] types of triad. Apart from [39], small vocabulary sizes remained the defacto standard until the last few years, with an upward trend that seems more definitive this time [14, 40].

The necessity of imposing an algorithmic chord vocabulary stems directly from the choice of treating chord recognition as a classification problem. This design choice is extremely wide-spread, but is not inherently part of the task. Chord class-free approaches can be envisaged, such as labelling sets of active chromas, where the complexity is then shifted towards determining when a chroma is active (itself potentially a classification problem). A vocabulary-free model of chords such as proposed in [66] can be used to label the chroma sets, but only the model in isolation has been tested so far, not as part of a full ACR system.

A particularity to increasing the classes in ACR is that the distinction between them becomes smaller the more classes are added. Especially when triads and tetrads (4-chroma chords) are mixed, because the set of chromas in tetrads are a superset of the chromas in their associated triad. Commonly used flat classification approaches assume disjoint categories, and therefore aren't an optimal match for the problem [23]. A form of hierarchical or alternatively multi-stage classification [64] is one way to address this problem, but these need to be explored further in the future. Different ways of representing target labels to make classes more orthogonal [7, 40, 64] constitute another type of solution, one in which standard flat classification algorithms can continue being used. Multiple auxiliary labels are introduced in this approach, which we called *target label engineering* in Section 3. Alternatively, multiple distances between chords have been used as targets [7]. These chord label representations constitute a modern take on introducing musical knowledge into deep learning. Where feature engineering tried to shape the audio input into features that were maximally discriminative, target label engineering aims to do the same working back from the label output.

A further complication when considering chord labels as separate classes is that their frequency of occurrence is strongly unbalanced. For instance, the five most common chord types account for over 80% of popular music datasets [6, 64]. This imbalance in chord distribution affects both training data-driven ACR systems and evaluation. For training, new strategies need to be developed to avoid overemphasising the most frequent classes [16]. In evaluation, improvements on rare chords are barely visible when using a metric that reports the percentage of time the correct chord is found. Discussing performance on multiple levels, with rarer chords separated, is necessary to get more insight into algorithm performance [55].

Finally, the chord imbalance itself differs from dataset to dataset. Genre, instrumentation, cultural origin, key and chord distributions are all linked, and are manifested as a skew towards certain roots and chord types, as well as variance in timbre. Since the majority of available annotated data consists of anglophonic pop music, its representativeness is questionable. Until current algorithms are cross-checked with new datasets containing e.g. Latin, jazz and metal, their general applicability remains unproven.

9. CONCLUSIONS

In this paper, we discussed 20 years of research on automatic chord recognition, starting with Fujishima's paper [17]. Even though modern ACR systems differ strongly in their proposed technical solutions, we find that his initial system was very effective at identifying the problems that arise during the creation of ACR systems. We therefore compared past and recent solutions thematically according to these problems, with the intention of inspiring future work on this topic.

We note a tendency towards tackling the different problems in ACR as a single integrated approach, in contrast to the compartmentalised strategies of the early years. This evolution follows the move from knowledge-driven to data-driven approaches. The lack of available training data in the early years called for knowledge-based systems, which in turn required systems to be broken down into smaller components for them to remain comprehensible. Each component usually dealt with one sub-problem in isolation. Early data-driven approaches replaced these knowledge-based components with learnt ones, while retaining the modular structure. The arrival of deep learning permitted to replace all these components by a single system that is better at exploiting the interactions between the ACR problems. Nonetheless, some researchers choose to keep the modularised approach, to increase interpretability or to employ specific training data or procedures.

A consequence of moving towards integrated systems, is that the comparison between them becomes harder. No general blueprint of a DL-ACR system can be given, as many approaches compete, and therefore it is difficult to translate findings of one system to the other or to combine parts of two systems into one. Without a doubt, more innovations in deep learning will make their way to ACR, which will keep the field moving fast for the foreseeable future. It is encouraging to see that the specific characteristics of music are starting to show up in the deep learning approaches, with musical knowledge guiding the training procedure and architecture.

10. ACKNOWLEDGEMENTS

This work has been funded by the UK Engineering and Physical Sciences Research Council (EPSRC) grant EP/L019981/1.

11. REFERENCES

[1] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint*, 2018.

[2] Juan Pablo Bello and Jeremy Pickens. A robust mid-level representation for harmonic content in music signals. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR)*, pages 304–311, 2005.

[3] Nicolas Boulanger-Lewandowski, Yoshua Bengio, and Pascal Vincent. Audio chord recognition with Recurrent Neural Networks. In *Proceedings of the 14th Conference of the International Society for Music Information Retrieval (ISMIR)*, 2013.

[4] Judith C. Brown. Calculation of a constant Q spectral transform. *Journal of the Acoustical Society of America*, 89(1):425–434, January 1991.

[5] John Ashley Burgoyne, Laurent Pugin, Corey Kereliuk, and Ichiro Fujinaga. A cross-validated study of modelling strategies for automatic chord recognition in audio. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, pages 251–254, 2007.

[6] John Ashley Burgoyne, Jonathan Wild, and Ichiro Fujinaga. An expert ground-truth set for audio chord recognition and music analysis. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, pages 633–638, 2011.

[7] Tristan Carsault, Jérôme Nika, and Philippe Esling. Using musical relationships between chord labels in automatic chord extraction tasks. In *Proceedings of the 19th Conference of the International Society for Music Information Retrieval (ISMIR)*, 2018.

[8] Ruofeng Chen, Weibin Shen, Ajay Srinivasamurthy, and Parag Chordia. Chord recognition using duration-explicit hidden Markov models. In *Proceedings of the 13th International Conference on Music Information Retrieval (ISMIR)*, pages 445–450, 2012.

[9] Taemin Cho. *Improved techniques for automatic chord recognition from music audio signals*. PhD thesis, New York University, 2013.

[10] Taemin Cho and Juan P. Bello. On the relative importance of individual components of chord recognition systems. *IEEE Transactions on Audio, Speech and Language Processing*, 22(2):477–492, February 2014.

[11] Taemin Cho and Juan Pablo Bello. A feature smoothing method for chord recognition using recurrence plots. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, pages 651–656, 2011.

[12] Trevor de Clercq and David Temperley. A corpus analysis of rock harmony. *Popular Music*, 30(1):47–70, January 2011.

[13] Alessio Degani, Marco Dalai, Riccardo Leonardi, and Pierangelo Migliorati. Harmonic change detection for musical chords segmentation. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2015. conference-date: June 29 2015-July 3 2015 conference-venue: Turin, Italy.

[14] Junqi Deng and Yu-Kwong Kwok. A hybrid Gaussian-HMM-deep learning approach for automatic chord estimation with very large vocabulary. In *Proceedings of the 17th Conference of the International Society for Music Information Retrieval (ISMIR)*, pages 812–818, 2016.

[15] Junqi Deng and Yu-Kwong Kwok. Large vocabulary automatic chord estimation with an even chance training scheme. In *Proceedings of the 18th Conference of the International Society for Music Information Retrieval (ISMIR)*, 2017.

[16] Junqi Deng and Yu-Kwong Kwok. Large vocabulary automatic chord estimation using bidirectional long short-term memory recurrent neural network with even chance training. *Journal of New Music Research*, 47(1):53–67, 2018.

[17] Takuya Fujishima. Realtime chord recognition of musical sound: a system using Common Lisp Music. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 464–467, Ann Arbor, MI, USA, 1999. MPublishing, University of Michigan Library.

[18] Emilia Gómez. *Tonal description of music audio signals*. PhD thesis, Universitat Pompeu Fabra, 2006.

[19] Emilia Gómez. Tonal description of polyphonic audio for music content processing. *INFORMS Journal on Computing, Special Cluster on Computation in Music*, 18(3):294–304, Summer 2006.

[20] Christopher Harte, Mark Sandler, and Martin Gasser. Detecting harmonic change in musical audio. In *Proceedings of the 1st ACM Workshop on Audio and Music Computing Multimedia (AMCMM)*, pages 21–26, New York, NY, USA, 2006. ACM.

[21] Christopher A. Harte and Mark B. Sandler. Automatic chord identification using a quantised chromagram. In *Proceedings of the 118th Convention of the Audio Engineering Society*, Barcelona, Spain, May 28–31 2005.

[22] Eric J. Humphrey and Juan P. Bello. Rethinking automatic chord recognition with Convolutional Neural Networks. In *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 357–362, Boca Raton, FL, 12–15 December 2012.

[23] Eric J. Humphrey and Juan P. Bello. Four timely insights on automatic chord estimation. In *Proceedings of the 16th Conference of the International Society for Music Information Retrieval (ISMIR)*, pages 673–679, 2015.

[24] Eric J. Humphrey, Taemin Cho, and Juan P. Bello. Learning a robust Tonnetz-space transform for automatic chord recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 453–456. IEEE, 2012.

[25] Junyan Jiang, Wei Li, and Yiming Wu. Extended abstract for MIREX 2017 submission: Chord recognition using random forest model. In *Proceedings of the Music Information Retrieval Evaluation Exchange (MIREX)*, 2017.

[26] Maksim Khadkevich and Maurizio Omologo. Use of hidden Markov models and factored language models for automatic chord recognition. In *Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR)*, pages 561–566, 2009.

[27] Maksim Khadkevich and Maurizio Omologo. Reassigned spectrum-based feature extraction for GMM-based automatic chord recognition. *EURASIP Journal on Audio, Speech, and Music Processing*, 2013(15):1–12, June 2013.

[28] Hendrik Vincent Koops, W. Bas de Haas, Jeroen Bransen, and Anja Volk. Chord label personalization through deep learning of integrated harmonic interval-based representations. In *Proceedings of the First International Workshop on Deep Learning for Music*, pages 19–25, 2017.

[29] Hendrik Vincent Koops, W. Bas de Haas, John Ashley Burgoyne, Jeroen Bransen, Anna Kent-Muller, and Anja Volk. Annotator subjectivity in harmony annotations of popular music. *Journal of New Music Research*, pages 232–252, 2019.

[30] Filip Korzeniowski and Gerhard Widmer. Feature learning for chord recognition: The deep chroma extractor. In *Proceedings of the 17th Conference of the International Society for Music Information Retrieval (ISMIR)*, 2016.

[31] Filip Korzeniowski and Gerhard Widmer. A fully convolutional deep auditory model for musical chord recognition. In *Proceedings of the IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2016.

[32] Filip Korzeniowski and Gerhard Widmer. Improved chord recognition by combining duration and harmonic language models. In Emilia Gómez, Xiao Hu, Eric Humphrey, and Emmanouil Benetos, editors, *Proceedings of the 19th Conference of the International Society for Music Information Retrieval (ISMIR)*, pages 10–17, 2018.

[33] Kyogu Lee. Automatic chord recognition using enhanced pitch class profile. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 306–313, Ann Arbor, MI, USA, 2006. MPublishing, University of Michigan Library.

[34] Kyogu Lee. A system for automatic chord transcription using genre-specific hidden Markov models. In Nozha Boujemaa, Marcin Detyniecki, and Andreas Nürnberger, editors, *Adaptive Multimedia Retrieval: Retrieval, User, and Semantics*, volume 4918 of *Lecture Notes in Computer Science*, pages 134–146. Springer Berlin Heidelberg, 2008.

[35] Kyogu Lee and Malcolm Slaney. Acoustic chord transcription and key extraction from audio using key-dependent HMMs trained on synthesized audio. *IEEE Transactions on Audio, Speech and Language Processing*, 16(2):291–301, February 2008.

[36] Namunu C. Maddage, Changsheng Xu, Mohan S. Kankanhalli, and Xi Shao. Content-based music structure analysis with applications to music semantics understanding. In *Proceedings of the 12th ACM International Conference on Multimedia (ACM MM)*, pages 112–119. ACM, 10 October 2004.

[37] Matthias Mauch and Simon Dixon. Using musical structure to enhance automatic chord transcription. In *Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR)*, pages 231–236, 2009.

[38] Matthias Mauch and Simon Dixon. Approximate note transcription for the improved identification of difficult chords. In *Proceedings of the 11th International Conference on Music Information Retrieval (ISMIR)*, pages 135–140, 2010.

[39] Matthias Mauch and Simon Dixon. Simultaneous estimation of chords and musical context from audio. *IEEE Transactions on Audio, Speech and Language Processing*, 18(6):1280–1289, August 2010.

[40] Brian McFee and Juan Pablo Bello. Structured training for large-vocabulary chord recognition. In *Proceedings of the 18th Conference of the International Society for Music Information Retrieval (ISMIR)*, 2017.

[41] Joshua Morman and Lawrence Rabiner. A system for the automatic segmentation and classification of chord sequences. In *Proceedings of the 1st ACM Workshop on Audio and Music Computing Multimedia (AMCMM)*, pages 1–10. ACM, 27 October 2006.

[42] Meinard Müller and Sebastian Ewert. Towards timbre-invariant audio features for harmony-based music. *IEEE Transactions on Audio, Speech and Language Processing*, 18(3):649–662, March 2010. ISSN: 1558-7916.

[43] Yizhao Ni, Matt McVicar, Raúl Santos-Rodríguez, and Tijl De Bie. An end-to-end machine learning system for harmonic analysis of music. *IEEE Transactions on Audio, Speech and Language Processing*, 20(6):1771–1783, August 2012.

[44] Yizhao Ni, Matt McVicar, Raúl Santos-Rodríguez, and Tijl De Bie. Using hyper-genre training to explore genre information for automatic chord estimation. In *Proceedings of the 13th International Conference on Music Information Retrieval (ISMIR)*, pages 109–114, 2012.

[45] Yizhao Ni, Matt McVicar, Raúl Santos-Rodríguez, and Tijl De Bie. Understanding effects of subjectivity in measuring chord estimation accuracy. *IEEE Transactions on Audio, Speech and Language Processing*, 21(12):2607–2615, December 2013.

[46] Ken O'Hanlon and Mark B. Sandler. Comparing CQT and reassignment based chroma features for template-based automatic chord recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 860–864, 2019.

[47] Laurent Oudre, Cédric Févotte, and Yves Grenier. Probabilistic template-based chord recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 19(8):2249 – 2259, November 2011.

[48] Laurent Oudre, Yves Grenier, and Cédric Févotte. Chord recognition by fitting rescaled chroma vectors to chord templates. *IEEE Transactions on Audio, Speech and Language Processing*, 19(7):2222 – 2233, September 2011.

[49] Hélène Papadopoulos and Geoffroy Peeters. Large-scale study of chord estimation algorithms based on chroma representation and HMM. In *Proceedings of the International Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 53–60, Bordeaux, France, June 25–27 2007.

[50] Hélène Papadopoulos and Geoffroy Peeters. Joint estimation of chords and downbeats from an audio signal. *IEEE Transactions on Audio, Speech and Language Processing*, 19(1):138–152, January 2011.

[51] Hélène Papadopoulos and George Tzanetakis. Models for music analysis from a Markov logic networks perspective. *IEEE Transactions on Audio, Speech and Language Processing*, 25:19–34, 2017.

[52] Johan Pauwels. *Exploiting prior knowledge during automatic key and chord estimation from musical audio*. PhD thesis, Faculty of Engineering and Architecture, 2016.

[53] Johan Pauwels, Florian Kaiser, and Geoffroy Peeters. Combining harmony-based and novelty-based approaches for structural segmentation. In *Proceedings of the 14th Conference of the International Society for Music Information Retrieval (ISMIR)*, pages 138–143, Curitiba, Brazil, 2013.

[54] Johan Pauwels and Jean-Pierre Martens. Combining musicological knowledge about chords and keys in a simultaneous chord and local key estimation system. *Journal of New Music Research*, 43(3):318–330, 2014.

[55] Johan Pauwels and Geoffroy Peeters. Evaluating automatically estimated chord sequences. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 749–753, 2013.

[56] Jordi Pons, Oriol Nieto, Matthew Prockup, Erik M. Schmidt, Andreas F. Ehmann, and Xavier Serra. End-to-end learning for music audio tagging at scale. In *Proceedings of the 19th Conference of the International Society for Music Information Retrieval (ISMIR)*, pages 636–644, 2018.

[57] Zhongyang Rao, Xin Guan, and Jianfu Teng. Chord recognition based on temporal correlation support vector machine. *Applied Sciences*, 6(5):157, 2016.

[58] Ricardo Scholz, Emmanuel Vincent, and Frédéric Bimbot. Robust modeling of musical chord sequences using probabilistic n-grams. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 53–56, 2009.

[59] Alexander Sheh and Daniel P.W. Ellis. Chord segmentation and recognition using EM-trained hidden Markov models. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR)*, pages 183–189, 2003.

[60] Arun Shenoy and Ye Wang. Key, chord, and rhythm tracking of popular music recordings. *Computer Music Journal*, 29(3):75–86, 2005.

[61] Siddharth Sigtia, Nicolas Boulanger-Lewandowski, and Simon Dixon. Audio chord recognition with a hybrid recurrent neural network. In *Proceedings of the 16th Conference of the International Society for Music Information Retrieval (ISMIR)*, pages 127–133, 2015.

[62] Kouhei Sumi, Katsutoshi Itoyama, Kazuyoshi Yoshii, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno. Automatic chord recognition based on probabilistic integration of chord transition and bass pitch estimation. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, pages 39–44, 2008.

[63] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint*, 2017.

[64] Yiming Wu and Wei Li. Automatic audio chord recognition with MIDI-trained deep feature and BLSTM-CRF sequence decoding model. *IEEE Transactions on Audio, Speech and Language Processing*, 27(2):355–366, February 2019.

[65] Mu-Heng Yang, Li Su, and Yi-Hsuan Yang. Highlighting root notes in chord recognition using cepstral features and multi-task learning. In *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1–8. IEEE, 2016.

[66] Kazuyoshi Yoshii and Masataka Goto. A vocabulary-free infinity-gram model for nonparametric Bayesian chord progression analysis. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, pages 645–650, 2011.

[67] Takuya Yoshioka, Tetsuro Kitahara, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno. Automatic chord transcription with concurrent recognition of chord symbols and boundaries. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR)*, pages 100–105, 2004.

[68] Xinquan Zhou and Alexander Lerch. Chord detection using deep learning. In *Proceedings of the 16th Conference of the International Society for Music Information Retrieval (ISMIR)*, pages 52–58, 2015.