A Case for Intrinsic Evaluation of Optical Music Recognition

Jan Hajič jr.
Institute of Formal and Applied Linguistics
Charles University
Email: hajicj@ufal.mff.cuni.cz

***Abstract*—Evaluating Optical Music Recognition (OMR) has long been an acknowledged sore spot of the field. This short position paper attempts to bring some clarity to what are actually open problems in OMR evaluation: a closer look reveals that the main problem is finding an edit distance between some practical representations of music scores. While estimating these editing costs in the transcription use-case of OMR is difficult, I argue that the problems with modeling the subsequent editing workflow can be de-coupled from general OMR system development using an intrinsic evaluation approach, and sketch out how to do this.**

I. We Need a Music Score Edit Distance

Optical Music Recognition (OMR) has a known problem with evaluation [1]–[3]. We can approach OMR evaluation from two angles: extrinsic and intrinsic. By *extrinsic*, we mean evaluation in application contexts: how well does an OMR system address a specific need (such as retrieval, transcription, playback, ...)? *Intrinsic* evaluation asks a different question: how much of the information encoded by the music score has a given OMR system recovered? An example of extrinsic OMR evaluation can be found, e.g., in [4], where OMR is evaluated in the context of a cross-modal retrieval system; (partial) intrinsic evaluation is done i.a. in [5], where pitches and durations of recognized notes are counted against ground truth data. In this short position paper, I assess what the outstanding problems in evaluating OMR are, and propose intrinsic evaluation as a sensible way forward for OMR research.

The major problem in OMR evaluation is that given a ground truth encoding of a score and the output of a recognition system, there is no automatic method capable of reliably computing how well the recognition system performs that would (1) be rigorously described and evaluated, (2) have a public implementation, (3) give meaningful results. Other applications such as retrieval or extracing MIDI can be evaluated using more general methodologies. E.g., when using OMR to retrieve music scores, there is little domain-specific to defining success compared to retrieving other documents; any time MIDI output is required, metrics used to evaluate multi-f0 estimation can be adapted; score following has well-defined evaluation metrics at different levels of granularity as well. Within the traditional OMR pipeline [6], the partial steps (such as smbol detection) also can use more general evaluation metrics. However, when OMR is applied to retypesetting music (which is arguably its original motivation), no evaluation metric is available.

In fact, computing an "edit distance" between a ground truth representation of a full music score and OMR output may be the only evaluation scenario where satisfactory measures are not available. The notion of "edit cost" [7] or "recognition gain" [8] that defines success in terms of how much time a human editor saves by using an OMR system is yet more problematic, as it depends on the specific toolchain used.

What can be done? One can try and implement such a metric. However, because cost-to-correct depends on the toolchains music editors use to work with OMR outputs, developing extrinsic evaluation metrics of OMR for transcription would require user studies at a scale which is not feasible for the few active OMR researchers. For these reasons, we argue it would be helpful for OMR development to have an *intrinsic* evaluation metric. After all, why address individual concerns that OMR users may have when full-pipeline OMR does have the potential to address *all* the application scenarios of OMR, as it attempts to extract *all* the information available from a music score?

II. Music Notation Formats are Problematic

A part of the edit distance problem lies in the ways music notation is stored digitally. MusicXML or MEI, which represent current best practices in open-source formats of digital representation of music scores, have some properties that make it difficult to compute a useful edit distance between two such files (useful in the sense that it would meausre either the amount of errors that an OMR system made, or the actual difficulty of changing one score to the other). Furthermore, the formats can encode the same score in multiple ways – e.g., MusicXML stores scores either measure-wise, or voice-wise.

Next, both formats are designed top-down, as trees that represent in their nodes both abstract concepts like a voice or note and graphical entities such as stems or beams. This implies that they cannot represent partial recognition results, and cannot encode syntactically incorrect notation. Furthermore, while the hierarchical structure mostly reflects the abstract structures of music such as voices and measures, it does not reflect the structure of music *notation*: local changes in the score can lead to several changes in the encoding that occur far apart, and vice versa. This is an inherent limitation of their tree structure.

The LilyPond format is impractical for anything but attempts at end-to-end OMR, as it hides much of the graphical representation in its engraving engine, and has so many ways of representing the same music that it is hard to meaningfully compare LilyPond files. The MuNG format [3] does to some extent overcome this locality problem by assuming a directed acyclic graph instead of a tree stucture, but it is limited to OMR ground truth and lacks conversions to other formats than MIDI.

The lesson here is that one should not bind intrinsic OMR evaluation to specific notation formats. After all, these formats change much faster than music notation itself. Rather, an evaluation metric should focus on inherent properties of music notation.

III. Arguing for Intrinsic Evaluation

Intrinsic evaluation of OMR systems means to answer the question *"How good is this system?"* without having to add, *"for this specific purpose?"* – thus de-coupling research of OMR methods from their individual use-cases, including the problematic score transcription. After all, music notation is the same regardless of whether it is being recognized for the purpose of searching a database or for producing a digital edition of the score.

There is no reason why this should not be possible: there is a finite amount of information that a music document carries, which can be exhaustively enumerated. It follows that we should be able to measure what proportion of this information our systems recover correctly. The benefit of intrinsic evaluation would be shedding the burden of accounting for score editing toolchains, independence on problematic music notation formats used in broad practice, and a clearly interpretable automatic metric for guiding OMR development (and potentially usable as a differentiable loss function for training full-pipeline end-to-end machine learning-based systems).

IV. A Roadmap

What would such an intrinsic evaluation metric measure? At the fullest, we expect two classes of outputs from an OMR system. First, a digital re-encoding of the score itself — creating a digital document that would convey exactly the same to a reader as the original. Second, recovering the semantic musical information: primarily the pitches, durations and onsets of notes (the minimum to build a MIDI representation of the given composition).

A thorough definition of error types in OMR was done by Bellini et al. [8]. They ask human evaluators to count errors for individual symbol types, and what they call "high-level" mistakes: pitch and duration attributes of note symbols. This seems like a good starting point from which to develop an automated intrinsic OMR evaluation metric.

The reason why [8] do not automate error-counting was a (then) lack of ground truth data. This has now been alleviated by the DeepScores dataset [9] at the low level, and MUSCIMA++ dataset [3] at both levels. The other step to automating the metric of [8] is aligning the recognition output and the ground truth score. At the graphical level, where the outputs are in principle symbol and their relationships, success can be measured using some graph similarity metric. At the semantic level, distance on lists of `(onset, duration, pitch)` triplets would be conditioned on some optimal alignment; DTW seems like a possible starting point for tractably finding this alignment, as it harshly penalizes ordering errors, which are rather critical due to the sequential nature of music. Given that noteheads can be thought of as carriers of the semantic information within the graphical level, the graph alignment function can also be used to directly find corresponding semantic triplets.

V. Finally

I hope this short paper will inspire discussion on the merits of intrinsic evaluation of OMR (I am especially keen to find out how I am wrong!), and perhaps nudge along the musical score edit distance problem that has been a thorn in the side of OMR research for the duration of its existence.

Acknowledgments

This work is supported by the Czech Science Foundation, grant P103/12/G084, the Charles University Grant Agency, grants 1444217 and 170217, and by SVV project 260 453.

References

[1] M. Szwoch, "Using MusicXML to Evaluate Accuracy of OMR Systems," *Proceedings of the 5th International Conference on Diagrammatic Representation and Inference*, pp. 419–422, 2008. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-87730-1_53

[2] Donald Byrd and Jakob Grue Simonsen, "Towards a Standard Testbed for Optical Music Recognition: Definitions, Metrics, and Page Images," *Journal of New Music Research*, vol. 44, no. 3, pp. 169–195, 2015. [Online]. Available: http://dx.doi.org/10.1080/09298215.2015.1045424

[3] J. Hajič jr. and P. Pecina, "The MUSCIMA++ Dataset for Handwritten Optical Music Recognition," in *14th International Conference on Document Analysis and Recognition, ICDAR 2017, Kyoto, Japan, November 13 - 15, 2017*, Dept. of Computer Science and Intelligent Systems, Graduate School of Engineering, Osaka Prefecture University. New York, USA: IEEE Computer Society, 2017, pp. 39–46.

[4] S. Balke, S. P. Achankunju, and M. Müller, "Matching Musical Themes based on noisy OCR and OMR input," pp. 703–707, 2015.

[5] Victor Padilla, Alan Marsden, Alex McLean, and Kia Ng, "Improving OMR for Digital Music Libraries with Multiple Recognisers and Multiple Sources," *Proceedings of the 1st International Workshop on Digital Libraries for Musicology - DLfM '14*, pp. 1–8, 2014. [Online]. Available: http://dx.doi.org/10.1145/2660168.2660175

[6] Ana Rebelo, Ichiro Fujinaga, Filipe Paszkiewicz, Andre R. S. Marcal, Carlos Guedes, and Jaime S. Cardoso, "Optical Music Recognition: State-of-the-Art and Open Issues," *Int J Multimed Info Retr*, vol. 1, no. 3, pp. 173–190, Mar 2012. [Online]. Available: http://dx.doi.org/10.1007/s13735-012-0004-6

[7] J. Hajič jr., J. Novotný, P. Pecina, and J. Pokorný, "Further Steps towards a Standard Testbed for Optical Music Recognition," in *Proceedings of the 17th International Society for Music Information Retrieval Conference*, M. Mandel, J. Devaney, D. Turnbull, and G. Tzanetakis, Eds., New York University. New York, USA: New York University, 2016, pp. 157–163. [Online]. Available: https://18798-presscdn-pagely.netdna-ssl.com/ismir2016/wp-content/uploads/sites/2294/2016/07/289_Paper.pdf

[8] Pierfrancesco Bellini, Ivan Bruno, and Paolo Nesi, "Assessing Optical Music Recognition Tools," *Computer Music Journal*, vol. 31, no. 1, pp. 68–93, Mar 2007. [Online]. Available: http://dx.doi.org/10.1162/comj.2007.31.1.68

[9] Lukas Tuggener, Ismail Elezi, Jürgen Schmidhuber, Marcello Pelillo, and Thilo Stadelmann, "DeepScores - A Dataset for Segmentation, Detection and Classification of Tiny Objects," *CoRR*, vol. abs/1804.00525, 2018. [Online]. Available: http://arxiv.org/abs/1804.00525