From Jigs and Reels to Schottisar och Polskor: Generating Scandinavian-like Folk Music with Deep Recurrent Networks

Eric Hallström Simon Mossmyr Bob L. Sturm Victor Hansjons Vegeborn Jonas Wedin

KTH Royal Institute of Technology - Speech, music and hearing division

{erichal,mossmyr,bobs,victorhv,jonwed}@kth.se

ABSTRACT

The use of recurrent neural networks for modeling and generating music has been shown to be quite effective for compact, textual transcriptions of traditional music from Ireland and the UK. We explore how well these models perform for textual transcriptions of traditional music from Scandinavia. This type of music has characteristics that are similar to and different from that of Irish music, e.g., mode, rhythm, and structure. We investigate the effects of different architectures and training regimens, and evaluate the resulting models using three methods: a comparison of statistics between real and generated transcriptions, an appraisal of generated transcriptions via a semi-structured interview with an expert in Swedish folk music, and an exercise conducted with students of Scandinavian folk music. We find that some of our models can generate new transcriptions sharing characteristics with Scandinavian folk music, but which often lack the simplicity of real transcriptions. One of our models has been implemented online at http://www.folkrnn.org for anyone to try.

1. INTRODUCTION

Recent work [1] applies long short-term memory (LSTM) neural networks [2] to model and generate textual transcriptions of traditional music from Ireland and the UK. The data used in that work consists of over 23,000 tune transcriptions crowd-sourced online. [1] Each transcription is expressed using a compact textual notation called ABC. [2] The resulting transcription models have been used and evaluated in a variety of ways, from creating material for public concerts [3] and a professionally produced album [4], to numerical analyses of the millions of parameters in the network [5, 6], to an accessible online implementation. [3] The success of machine learning in reproducing idiosyncrasies of Irish traditional music transcriptions comes in large part from the expressive capacity of the LSTM network, the compact data representation designed around ABC notation, and a large amount of training data. Will such a model also perform well given another kind of traditional music expressed in a similarly compact way? What happens when the amount of training data is an order of magnitude less than for the Irish transcription models?

In this paper, we present our work applying deep recurrent modeling methods to Scandinavian folk music. We explore both LSTM and Gated Recurrent Unit (GRU) networks [7], trained with and without dropout [8]. We acquire our data from a crowd-sourced repository of Scandinavian folk music, which gives 4,083 transcriptions expressed as ABC notation. Though this data is expressed the same way as the Irish transcriptions used in [1], there are subtle differences between the styles that require a different approach, e.g., key changes in tunes. This results in a larger vocabulary for the Scandinavian transcription models, compared with the Irish ones (224 vs. 137 tokens) [1]. We also explore using pretraining with the Irish transcription dataset, with further training using only Scandinavian transcriptions. To evaluate the resulting models, we compare low-level statistics of the generated transcriptions with the training data, conduct a semi-structured interview with an expert on Swedish folk music, and perform an exercise with students of Scandinavian folk music.

We begin by briefly reviewing recurrent neural networks, including LSTM and GRU networks. We then describe the data we use, how we have process it to create training data, and how we train our models. We then present our evaluation of the models, and discuss the results and our future work.

2. RECURRENT NEURAL NETWORKS

A Recurrent Neural Network (RNN) [9] is a type of artificial neural network that uses directed cycles in its computations, inspired by the cyclical connections between neurons in the brain [10]. These recurrent connections allow the RNN to use its output in a sequence, while the internal states of the network act as memory. We test two different flavors of RNN: Long Short-Term Memory Networks (LSTM), and Gated Recurrent Units (GRU). The final layer of these networks is a softmax layer, which is produces a conditional probability distribution over a vocabulary given the previous observations. It is from this distribution one samples to generate a sequence.

[1] http://thesession.org
[2] http://abcnotation.com/wiki/abc:standard:v2.1
[3] http://www.folkrnn.org

2.1 Long Short-Term Memory (LSTM)

The LSTM is an RNN architecture designed to overcome problems in training conventional RNNs [2]. Each LSTM layer is defined by four "gates" transforming an input $\boldsymbol{x}_t$ at time step t and a previous state $\boldsymbol{h}_{t-1}$ as follows [11]:

$$\boldsymbol{i}_t = \sigma\left(\boldsymbol{W}_i \boldsymbol{x}_t + \boldsymbol{U}_i \boldsymbol{h}_{t-1} + \boldsymbol{b}_i\right) \quad (1)$$

$$\boldsymbol{f}_t = \sigma\left(\boldsymbol{W}_f \boldsymbol{x}_t + \boldsymbol{U}_f \boldsymbol{h}_{t-1} + \boldsymbol{b}_f\right) \quad (2)$$

$$\boldsymbol{o}_t = \sigma\left(\boldsymbol{W}_o \boldsymbol{x}_t + \boldsymbol{U}_o \boldsymbol{h}_{t-1} + \boldsymbol{b}_o\right) \quad (3)$$

$$\boldsymbol{c}_t = \boldsymbol{i}_t \odot \tanh\left(\boldsymbol{W}_u \boldsymbol{x}_t + \boldsymbol{U}_u \boldsymbol{h}_{t-1} + \boldsymbol{b}_u\right) + \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} \quad (4)$$

where $\sigma()$ denotes the element-wise logistic sigmoid function, and $\odot$ denotes the element-wise multiplication operator. The LSTM layer updates its hidden state by

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \tanh(\boldsymbol{c}_t). \quad (5)$$

The hidden state of an LSTM layer is the input to the next deeper layer.

2.2 Gated Recurrent Unit (GRU)

A GRU layer is similar to that of the LSTM, but each layer uses only two gates and so is much simpler to compute [7]. Each GRU layer transforms an input $\boldsymbol{x}_t$ and a previous state $\boldsymbol{h}_{t-1}$ as follows:

$$\boldsymbol{r}_t = \sigma(\boldsymbol{W}_r \boldsymbol{x}_t + \boldsymbol{U}_r \boldsymbol{h}_{t-1} + \boldsymbol{b}_r) \quad (6)$$

$$\boldsymbol{z}_t = \sigma(\boldsymbol{W}_z \boldsymbol{x}_t + \boldsymbol{U}_z \boldsymbol{h}_{t-1} + \boldsymbol{b}_z). \quad (7)$$

The GRU layer updates its state by

$$\boldsymbol{h}_t = (\boldsymbol{1} - \boldsymbol{z}_t) \odot \tanh(\boldsymbol{W}_h \boldsymbol{x}_t + \boldsymbol{U}_h(\boldsymbol{r}_t \odot \boldsymbol{h}_{t-1})) + \boldsymbol{z}_t \odot \boldsymbol{h}_{t-1}. \quad (8)$$

Compared with the LSTM, each GRU layer has fewer parameters.

3. MODELING SCANDINAVIAN FOLK MUSIC

3.1 Data

FOLKWIKI [4] is a wiki-style site dedicated to Scandinavian folk music that allows users to submit tune transcriptions to a growing database, each expressed using ABC notation. We collect transcriptions from FOLKWIKI by using a web scraper, [5] recursively gathering them using the "key" category. [6] This produces 4083 unique transcriptions. An example transcription is shown in the following:

```
%%abc-charset utf-8

X:1
T:Visa
T:ur Svenska Folkmelodier
  utgivna av C.E. Södling
B:http://www.smus.se/...(Edited by authors)
O:Småland
N:Se även +
M:3/4
L:1/8
R:Visa
Z:Nils L
K:Am
EE A2 cc | ee B2 d2 | cB (Ac) BA | ^G2 E4 ::
w:ung-er-sven med ett hur-tigt mod han
  sving-ar sig * u-ti la-get
EE A2 B2 | cd e2 d2 | cB Ac B^G | A2 A4 :|
w:fem-ton al-nar grö-na band det bär han
  u-ti sin skjort-kra-ge
```

We process these transcriptions in the following way:

1. Remove all comments and and non-musical data
2. If the tune has multiple voices, separate them as if they are individual tunes
3. Parse the head of the tune and keep the length (L:), meter (M:), and key fields (K:)
4. Parse the body of the tune
5. Clean up and substitute a few resulting tokens to keep similarity over the data set (i.e "K:DMajor" is substituted by "K:DMaj" etc.)

We keep all the following tokens in the tunes body:

- Changes in key (`K:`), meter (`M:`) or note length (`L:`)
- Any note as described in the ABC-Standard (e.g., `e`, `=a` or any valid note)
- Duplets (`2`, triplets (`3`, quadruplets (`4`, etc.
- Note length (Any integer after a note `=a 4`)
- Rest sign (`z`)
- Bars and repeat bars (`:| |:`)
- Grouping of simultaneous notes (`[` and `]`)

After processing, the transcription above appears as:

```
[L:1/8]
[M:3/4]
[K:AMin]
E E A 2 c c | e e B 2 d 2 | c B A c B A |
^G 2 E 4 :| |: E E A 2 B 2 | c d e 2 d 2 |
c B A c B ^G | A 2 A 4 :|
```

Each symbol separated by a space corresponds to one token in the model vocabulary. Notice that almost all metadata fields are removed, as well as lyrics. Reprise bars such as `::` or `:|:` have been substituted by `:| |:` to minimize the vocabulary size so the models become less complex. The output produced by our text processing is a file with all transcriptions separated by a newline. We do not keep any transcriptions with fewer than 50 tokens or more than 1000 tokens. We also do not attempt to correct human errors in transcription (e.g., miscounted bars). The resulting dataset is available in a repository. [7] The parser we created to do the above is available at the project repository. [8] The total number of unique tokens in the Folkwiki dataset is 155.

[4] http://www.folkwiki.se
[5] http://www.scrapy.org
[6] http://www.folkwiki.se/Tonarter/Tonarter
[7] https://github.com/victorwegeborn/folk-rnn/tree/master/data/9_nov
[8] http://www.github.com/ztime/polska

3.2 Pretraining models

The training of deep models typically begins with a random initialization of its weights, but it can also begin with weights found from previous training. In the latter sense, one can think of it as making the network first aware of syntactical relationships in the domain in which it is working, and then tuning the network on a subset to specialize it. We experiment with training models first using the concatenation of the FolkWiki dataset with the Irish transcription dataset that we process in the same way,[9] and then tuning the model with just the FolkWiki dataset.

3.3 Dropout

A danger with machine learning in general is the tendency to overfit to training data. One method to prevent overfitting of a network is to use a mechanism called dropout [8]. Dropout works by masking the output of a layer in the network with a random distributed binary vector during training. The dropout probability p_i is the parameter of the model that decides what output of the layer is propagated. When we use dropout, we set $p_i = 0.5$.

3.4 Model architecture and training

We use two different neural networks based on the LSTM and GRU units, with three different variations:

- LSTM with 50% dropout trained on FolkWiki (L^F_{50})
- GRU with 50% dropout trained on FolkWiki (G^F_{50})
- LSTM with 50% dropout pretrained on FolkWiki and TheSession, then only FolkWiki (L^{S+F}_{50})
- GRU with 50% dropout pretrained on FolkWiki and TheSession, then only FolkWiki (G^{S+F}_{50})
- LSTM without dropout pretrained on FolkWiki and TheSession, then only FolkWiki (L^{S+F})
- GRU without dropout pretrained on FolkWiki and TheSession, then only FolkWiki (G^{S+F})

Because of the number of unique tokens for a model depends on its training data, we adjusted the number of hidden units in layers to be about 4 times the vocabulary size [6]. We trained two models on only FolkWiki using 600 hidden nodes in each layer, whilst the models trained on both session-data and FolkWiki used 800 hidden nodes (the vocabulary size of the concatenation of the two datasets is 224). During the pretraining phase we use a batch size of 64, and for the final training we use a batch size of 32. We use a learning rate $\eta = 0.003$ with a decay of 0.97 after every 20 epochs (the same as used in [1]). All models have gradient clipping set to 5.

Figure 1 shows the mean transcription validation negative log-likelihood loss for each model. We train all models for 40 epochs on FolkWiki. The pretraining of the LSTM model on the FolkWiki and TheSession data is done for 50 epochs, but the pretraining of the GRU model on FolkWiki and TheSession data is done for 20 epochs.

[9] See footnote 7.

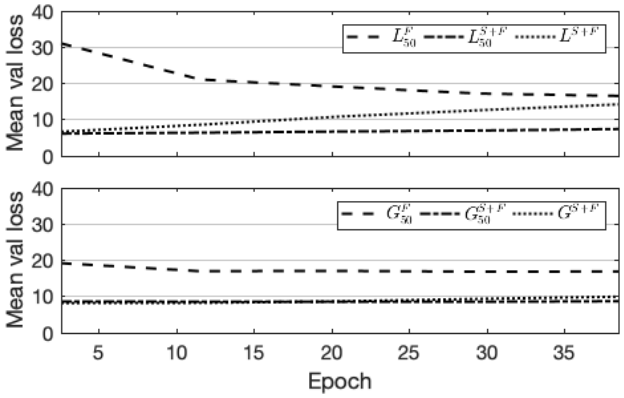

Figure 1. The mean transcription validation loss for the LSTM models (top) and the GRU models (bottom) when training on the FolkWiki dataset.

4. EVALUATION

We now evaluate the six different models we have trained. We use three different approaches to evaluation. First, we compare the descriptive statistics of the generated and real transcriptions. Second, we select output generated by the models for evaluation by an expert on Swedish traditional music. Finally, we perform an exercise with students of Scandinavian folk music.

4.1 Statistical analysis

We have each model generate 4000 transcriptions at random, and then look at how these compare with the 4083 transcriptions in the training dataset. Figure 2 compares FolkWiki with the transcriptions generated by L^F_{50} and G^F_{50} in terms of occurrences of keys, meter and number of tokens. We see a strong bias of G^F_{50} towards generating the D minor token, and away from the D major token, while L^F_{50} has a slight bias towards generating the tokens of D minor and G major, and away from D major. When it comes to the meter tokens, L^F_{50} appears to be in agreement with FolkWiki, while G^F_{50} is biased to most often produce 3/4. When it comes to the lengths of the transcriptions, L^F_{50} generates slightly shorter transcriptions than those in FolkWiki, while G^F_{50} generates transcriptions that are longer.

Figure 3 compares FolkWiki with the transcriptions generated by L^{S+F}_{50} and G^{S+F}_{50}. We see L^{S+F}_{50} is biased toward producing the D minor token and away from the D major and A minor tokens, while G^{S+F}_{50} too often generates the A major, D minor and G major tokens. As with the meters, these models either favor the 3/4 or the 4/4 tokens. In terms of number of tokens in the transcriptions, G^{S+F}_{50} generates longer ones than L^{S+F}_{50}, but both tend to produce longer transcriptions than in the FolkWiki dataset.

4.2 Semi-structured interview with Swedish expert

In order to learn about some defining characteristics of Swedish folk music, and to gauge the plausibility of material generated by our models, we interviewed Olof Misgeld, a lecturer in Music Theory and lecturer of folk violin

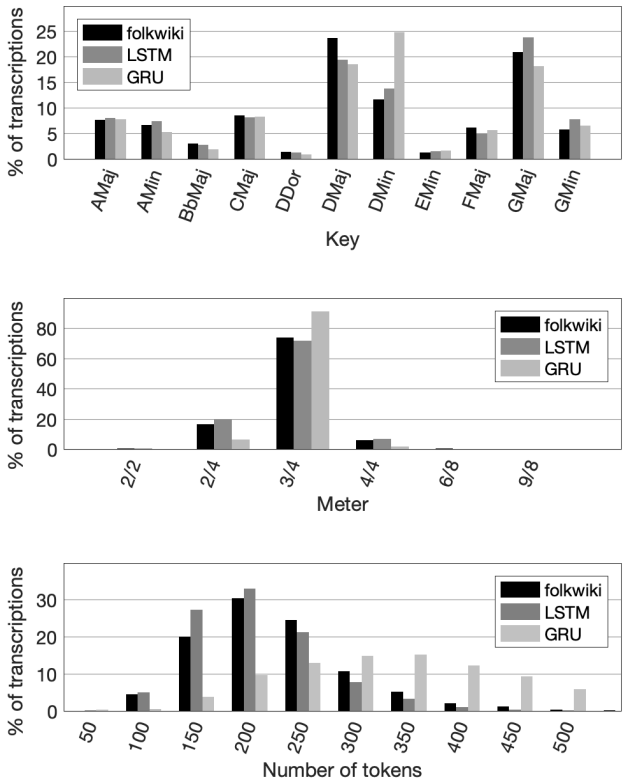

Figure 2. Comparison between FolkWiki and the L_{50}^{F} and G_{50}^{F} models. Percent of transcriptions in terms of keys (top), meters (middle), and number of tokens (bottom).

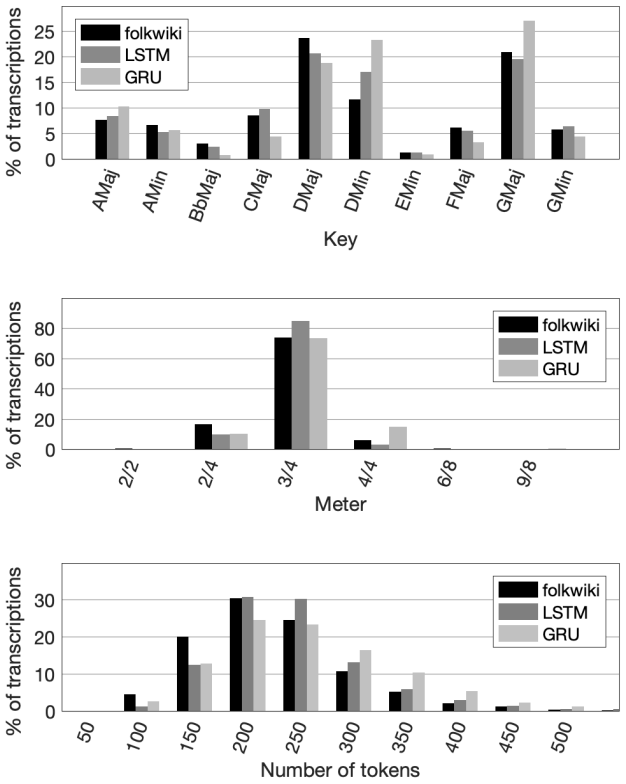

Figure 3. Comparison between FolkWiki and the L_{50}^{S+F} and G_{50}^{S+F} models. Percent of transcriptions in terms of keys (top), meters (middle), and number of tokens (bottom).

at the Royal College of Music (KMH) in Stockholm, Sweden. We told Misgeld about our research and why we were interested in interviewing him.

In preparation for the interview we created three different collections of transcriptions, each containing 500: 400 transcriptions generated by a model, and 100 real transcriptions from FolkWiki, randomly selected and ordered. The generated transcriptions of one collection come from L_{50}^{S+F} because its statistics most closely resemble the training data. In the second collection we chose to use transcriptions generated by L^{S+F} to see if not using dropout affects quality. Finally, for the third collection we chose transcriptions generated by G_{50}^{F} because its statistics look the most poor with respect to FolkWiki.

Misgeld first assessed collection L_{50}^{S+F}. We described the transcriptions "computer generated", without mentioning that some of them were from FolkWiki. We asked him to freely browse through the collection and provide observations. After a few observations we asked him to find a transcription that is really good (in your opinion) and describe why, and to find a transcription that is really poor (in your opinion) and describe why, After more observations we told him that some of the transcriptions are from the training data (real tunes), and asked if he can locate them. We performed the same procedure with the other two collections. After this, we asked Misgeld to freely compare the three collections.

Misgeld identified distinct styles in the transcriptions generated by L_{50}^{F+S}. Some comments include "maybe Slängpolska because of the 16th notes", "like triplet Polska, very occupied with G". When asked to find an example of a really good and a really bad transcription, both choices come from the generated material. Two of these are shown in Figs. 4 and 5.

For transcriptions generated by L^{F+S}, Misgeld comments that many generated tunes are "strange", in one form or another, e.g., "strange jump", "strange note", "strange [in general]", and "strange rhythm". At the same time he felt transcriptions by this model were the most convincing. Our later analysis of the transcriptions generated by this model reveals it to be plagiarizing, which could explain Misgeld's observation.

For the collection generated by G_{50}^{F}, Misgeld notes the transcriptions seem longer and more varied in structure. While he only gave specific comments on four transcriptions, two of which were generated, he notes that the generated transcriptions contain unusual chromaticism, no closure in rhythm, too many notes in a bar, and strange rhythmic patterns.

When asked for observations about all three models, and about the exercise, Misgeld said that "it's interesting, ...it makes you curious how these models work.", "The generated tunes appear to have too many ideas", "no strong motif", "not enough repetition and variation", and "funny endings". He later explained that traditional tunes often have simple and clear ideas, which assists the oral transmission of the tradition.

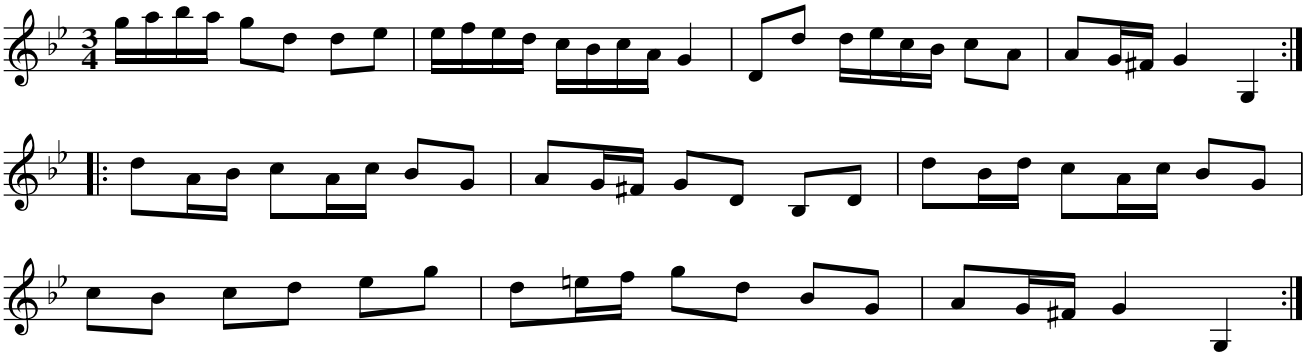

Figure 4. A tune generated by L_{50}^{S+F} for which the expert commented, "Seems real, natural ending, repeated with a 4+6 structure".

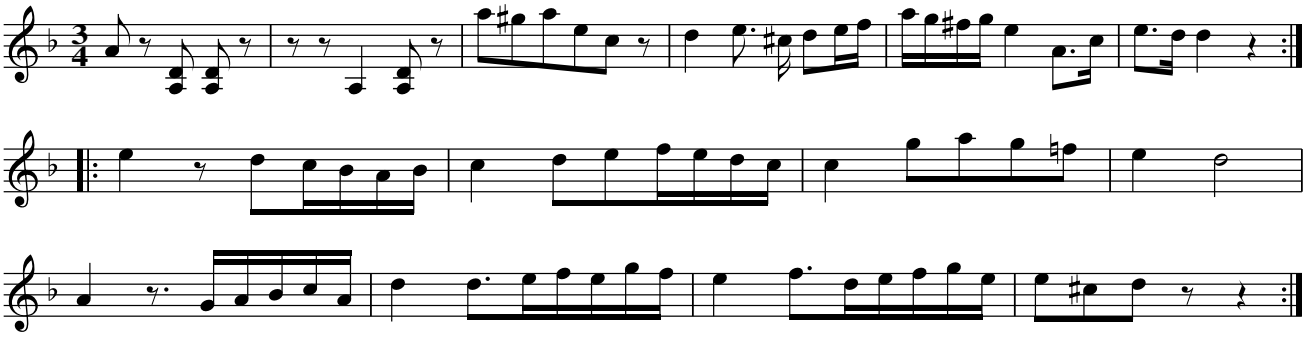

Figure 5. A tune generated by L_{50}^{S+F} for which the expert commented, "Fragmented, unexpected. Not real."

4.3 Exercise with folk music students

One of the authors (Sturm) was invited to give a workshop about AI and music to interested students of a folk music school in Bollnäs, Sweden. During one hour of the workshop, two groups of students were given different transcriptions and told to label each one as real or fake. Among the ten transcriptions given to each group, six were randomly generated by L_{50}^{S+F}, and four were randomly selected from the FolkWiki dataset. Points were awarded to each group based on the following: 2 points for each real transcription identified as real; 1 point for each fake transcription identified as fake; -1 point for each real transcription misidentified as fake; and -2 points for each fake transcription misidentified as real. The musicians were told they could also play the melodies as part of the evaluation.

One group decided to label everything as fake, and so received a total of 2 points. The other group was more deliberate, identified all real transcriptions as real, but misidentified three fake transcriptions as real, and so received a total of 5 points. Figure 6 shows the three transcriptions misidentified as real, which also illustrate some of the idiosyncrasies and weaknesses of the model. Transcription A has a pickup to the first bar which is not accounted for, but such a thing also occurs in the training data. That transcription A is so short added to the uncertainty of the students, as well as the lack of strong relationship between the two parts. Transcription B has a similar weak connection between the parts, but the first part is persuasive. The students noticed the second part becomes somewhat stuck on E minor, but they felt it could be due to someone trying to be clever. The students felt transcription C could be fake, but also felt the relationship between its two parts was good.

Figure 7 shows the three fake transcriptions the students identified correctly. The students noted nothing was especially wrong in these transcriptions. They called transcription E 'quirky', but noted the ending of both parts does not make sense. They also noted that the second part of transcription F feels stuck.

5. DISCUSSION

By comparing some of the descriptive statistics of collections of transcriptions, real and generated, we can see that the models have learned some aspects of the transcriptions of Scandinavian music. We find that the LSTM models provide a better fit to the data than GRU models, with the latter creating on average longer transcriptions. Our semi-structured interview with an expert of Swedish folk music shows many more details about the success and failure of our models. The expert identified several characteristics of the generated transcriptions, e.g., that they seem to be unfocused, to have too many ideas, but that some can be quite plausible. Transcriptions produced by the LSTM model trained without dropout were the most convincing, but this is likely due to the fact that the model was plagiarizing large amounts of the training data. The expert also noted

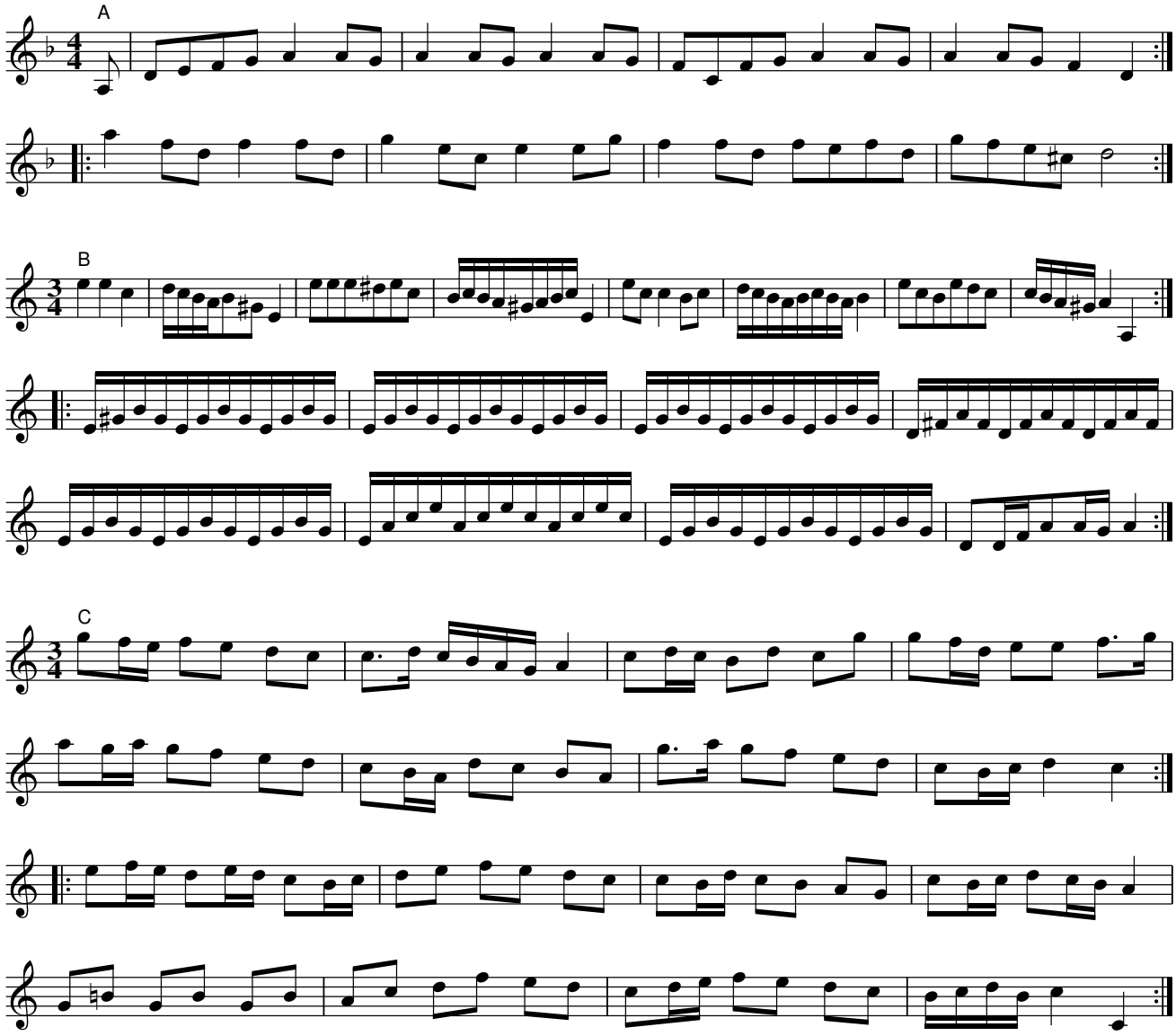

Figure 6. Three transcriptions generated by L_{50}^{S+F} that students assessed as being real.

that the transcriptions generated by the GRU model were more modern and "adventurous" than the others, but not as plausible. Conducting an exercise with students of Scandinavian folk music provided other observations about the transcriptions, and how one may think about a melody being good or not. One observation was that some of the generated transcriptions do not end on the tonic. This could be due to the fact that in creating our dataset we separated multivoice transcriptions into multiple single-voice transcriptions, In such a case, the harmonizing voice becomes a melody which often ends on the third.

6. CONCLUSIONS

We have shown that deep recurrent networks can generate music transcriptions that share characteristics with those of Scandinavian folk music. Even though our dataset is about one-sixth the size of the dataset used to train previous models of Irish traditional music [1], we have shown that the two datasets can be combined to pretrain a network, and then fine tune the network on the smaller Scandinavian music dataset to generate convincing transcriptions.

Acknowledgments

Google LLC through the *Google Cloud Education Grant*; for providing us with credits to be used on their Google Cloud platform; **Olof Misgeld**, Lecturer in Music Theory with specialization in Swedish folk music, and Director of Studies, at the Institute for Folk Music at the Royal College of Music (KMH), Stockholm, Sweden; **Bollnäs Folkhögskola** and **Esbjörn Wettermark**.

Author contribution

This work started from a data science project course at the Royal Institute of Technology in Stockholm, Sweden (Bob Sturm supervising). Apart from the joint effort in writing this paper the fine grained details of contributions from the

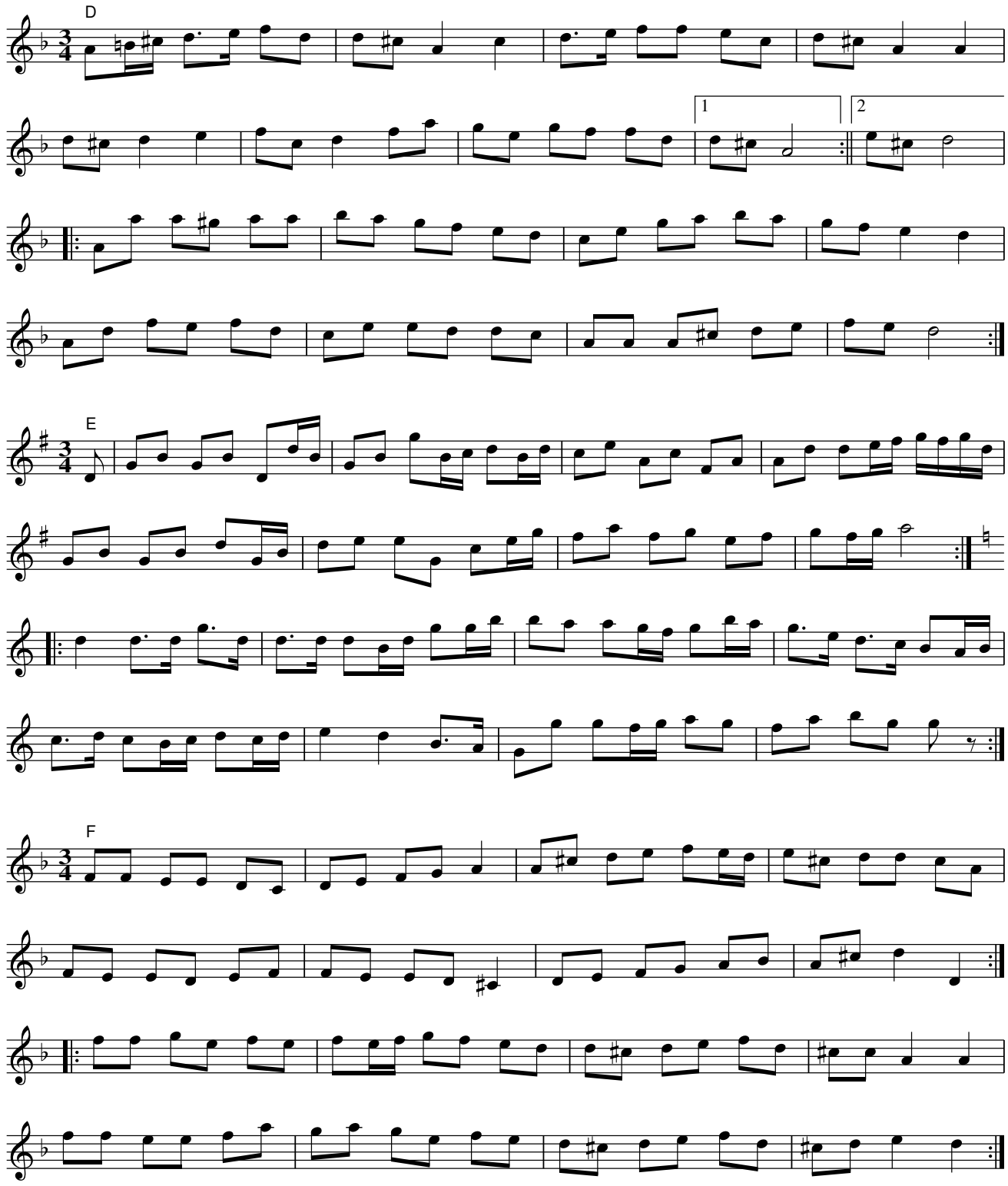

Figure 7. Three tunes generated by L_{50}^{S+F} that students assessed as being not real.

authors are the following; Eric Hallström managed the dependencies for Folk RNN, Simon Mossmyr modified the existing code base to use GRU layers instead of LSTM layers, Victor Hansjons Vegeborn assisted with the parser code and created code used for analysis and Jonas Wedin created the parser and scraper code.

7. REFERENCES

[1] B. L. Sturm, J. F. Santos, O. Ben-Tal, and I. Korshunova, "Music transcription modelling and composition using deep learning," in *Proc. Conf. Computer Simulation of Musical Creativity*, Huddersfield, UK, 2016.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] B. L. Sturm, O. Ben-Tal, U. Monaghan, N. Collins, D. Herremans, E. Chew, G. Hadjeres, E. Deruty, and F. Pachet, "Machine learning research that matters for music creation: A case study," *J. New Music Research*, vol. 48, no. 1, pp. 36–55, 2018.

[4] B. L. Sturm and O. Ben-Tal, "Let's have another Gan Ainm: An experimental album of Irish traditional music and computer-generated tunes," KTH Royal Institute of Technology, Tech. Rep., 2018.

[5] B. L. Sturm, "What do these 5,599,881 parameters mean? An analysis of a specific LSTM music transcription model, starting with the 70,281 parameters of its softmax layer," in *Proc. Music Metacreation workshop of the Int. Conf. Computational Creativity*, 2018.

[6] ——, "How stuff works: LSTM model of folk music transcriptions," in *Proc. Joint Workshop on Machine Learning for Music, ICML*, 2018.

[7] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. hi Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proc. Conf. Empirical Methods in Natural Language Processing*, 2014.

[8] N. Srivastava, G. Hinton, A. Krizhevsky, A. Krizhevsky, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Machine Learning Research*, vol. 15, pp. 1929–1958, June 2014.

[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Neurocomputing: Foundations of research," J. A. Anderson and E. Rosenfeld, Eds. Cambridge, MA, USA: MIT Press, 1988, ch. Learning Representations by Back-propagating Errors, pp. 696–699.

[10] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, 2012.

[11] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *Int. Joint Conf. Natural Language Processing*, 2015.