A COUGH-BASED DEEP LEARNING FRAMEWORK FOR DETECTING COVID-19

Hoang Van Truong[1] *Lam Pham*[2]

[1] University of Science of HCMC, Vietnam
[2]Center For Digital Safety & Security, Austrian Institute of Technology, Austria

ABSTRACT

In this paper, we propose a deep learning-based framework for detecting COVID-19 positive subjects from their cough sounds. In particular, the proposed framework comprises two main steps. In the first step, we generate a feature representing the cough sound by combining embedding features extracted from a pre-trained model and handcrafted features, referred to as the front-end feature extraction. Then, the combined features are fed into different back-end classification models for detecting COVID-19 positive subjects. The experimental results on the Second 2021 DiCOVA Challenge - Track 2 dataset achieve the top-3 ranking with an AUC score of 81.21 on the blind Test set, improving the challenge baseline by 6.32 and showing competitive with the state-of-the-art systems.

***Index Terms*—** COVID-19, deep learning, feature extraction, embedding, handcrafted feature.

1. INTRODUCTION

The cumulative number of COVID-19 positive subjects reported globally is now over 231 million and the cumulative number of deaths by COVID-19 is more than 4.7 million [1]. Furthermore, the COVID-19 crisis now is spanning across 200 countries quickly and the number of COVID-19 infections per day is always counted in thousands without a sign of going down. It is vital that one of the effective solutions to prevent and control the current epidemic is to conduct a large number of COVID-19 testing in popularity which has been widely applied in many countries. Indeed, if COVID-19 positive subjects can be detected early, it is very useful for self-observation, isolation, and effective treatment methods. However, take a large number of rapid antigen or RT-PCR tests shows a very high cost of both time and money. As a result, the DiCOVA Challenges are designed to find scientific and engineering insights to the question - Can COVID-19 be detected from the cough, breathing, or speech sound signals of an individual? In particular, while the First 2020 DiCOVA Challenge [2] provides a dataset of cough sound, the Second 2021 DiCOVA Challenge [3] provides different sound signals of cough, speech, and breath. The audio recordings are gathered from both COVID-19 positive and non-COVID-19 individuals [1]. Given the cough, speech, and breath recordings, research community can propose systems for detecting the COVID-19, which is potentially applied on edge devices as a COVID-19 testing solution.

Focusing on cough sound, recent researchers show that it potential to detect COVID-19 through evaluating coughing. For an example, a machine learning-based framework proposed in [4], which uses handcrafted features and Support Vector Machine (SVM) model, achieved the best AUC score of 85.02 on the First 2020 DiCOVA dataset [2]. Focusing on feature extraction, Madhu et al. [5] combined the Mel-frequency cepstral coefficients (MFCC) with the delta features (i.e. The delta features are extracted from a complicated framework using Long Short-Term Memory (LSTM), Gabor filter bank, and the Teager energy operator (TEO) in the order). By using the combined feature and the LightGBM model, the authors can achieve the AUC score of 76.31 on the First 2020 DiCOVA dataset [2]. Similarly, Vincent et al. [6] conducted extensive experiments to evaluate the role of the feature extraction. In particular, they proposed to use three types or features: (1) Handcrafted features extracted by openSMILE toolkit [7], (2) the deep features extracted from different pre-trained VGGish networks which are trained with AudioSet [8], and (3) the deep features extracted from different standard pre-trained models (ResNet50, DenseNet121, MobileNetV1, etc.) trained with Imagenet dataset. They then obtained the best AUC score of 72.8 on the First 2020 DiCOVA dataset [2] by using the deep features extracted from the pre-trained VGG16 (i.e. The pre-trained VGG16 was trained with AudioSet) and the back-end LSTM-based classification. Recently, a benchmark dataset of cough sound for detecting COVID-19 [9, 10], which was recorded on mobile phone, has been published. Notably, the current achievement of 98% accuracy on this dataset shows potential to apply as an effective solution of COVID-19 testing.

In this paper, we also aim to explore cough sounds, then propose a framework for detecting COVID-19. We mainly contribute: (1) By conducting extensive experiments, we indicate that a combination of handcrafted feature and embedding-based feature is effective to representing cough sound input, and (2) we propose a robust framework which can be further developed on edge devices for an application of COVID-19 testing. Our experiments were conducted on the Second 2021 DiCOVA Challenge - Track 2 dataset (i.e. The Track 2 dataset only contains cough sounds).

The remaining of our paper is organized as follows: Section 2 presents the Second 2021 DiCOVA Challenge as well as the Track-2 dataset, evaluation setting, and metrics. Section 3 presents the proposed deep learning framework. Next, Section 4 presents and analyses the experimental results. Finally, Section 5 presents the conclusion and future work.

2. THE SECOND 2021 DICOVA CHALLENGE - TRACK 2 DATASET OF COUGH SOUNDS

2.1. The second DiCOVA Challenge

The Second 2021 DiCOVA Challenge uses a subset of the Coswara dataset [3] collected between April 2020 and July 2021 from the age group of 15 to 90. The challenge provided a dataset of different sound signals: cough, speech, and breath gathered from both COVID-19 positive and non-COVID-19 individuals as shown in Fig.

[1]https://competitions.codalab.org/competitions/34801#learn_the_details

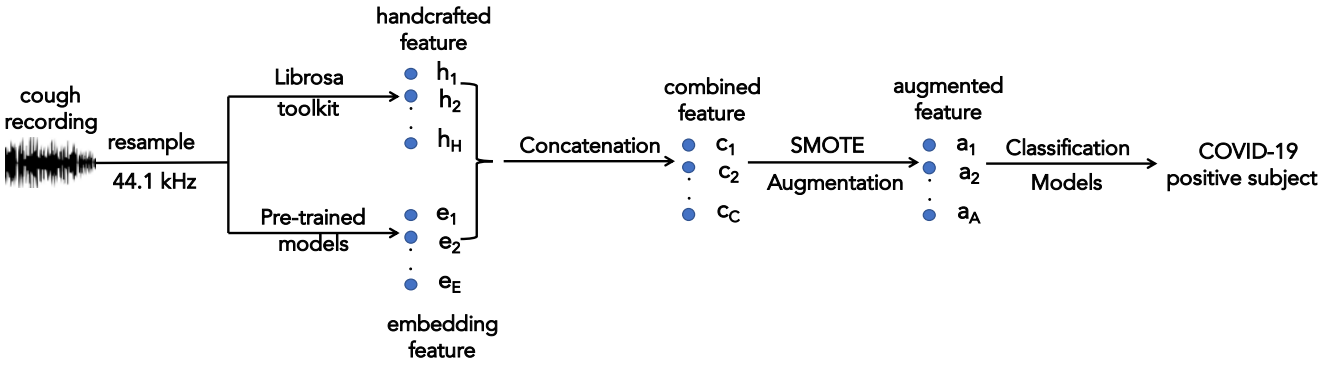

Fig. 1. The high-level architecture of deep learning framework proposed.

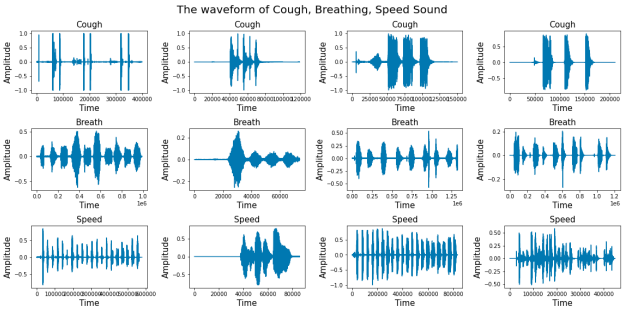

Fig. 2. The waveform of the Cough, Breathing, Speech sound from the Second 2021 DiCOVA Challenge [3].

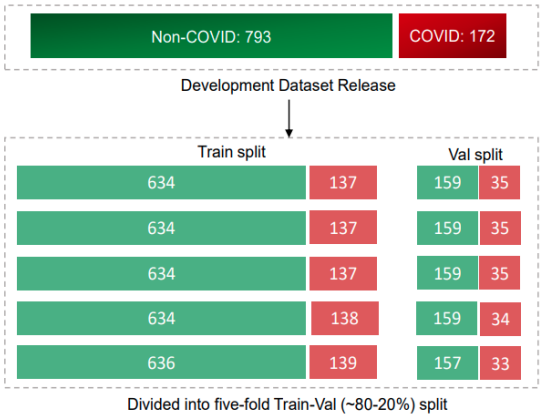

Fig. 3. The illustration of five-fold cross-validation from the Development set of the Second 2021 DiCOVA Challenge Track-2[3].

2. Given cough, speech, and breath sounds, the Second 2021 DiCOVA Challenge proposes four tracks which aim to detect COVID-19 positive subjects by exploring only breath (Track-1), only cough (Track-2), only speech (Track-3), or all sound signals (Track-4).

As we focus on cough sounds, which is also the First 2020 DICOVA Challenge [2], only Track-2 dataset is explored in this paper. The Second 2021 DiCOVA Challenge Track-2 dataset provided a Development set of 965 audio recordings and a blind Test set of 471 audio recordings. All audio recordings are not less than 500 milliseconds and recorded with different sample rates. While the Development set is used for training, and then obtaining the best model, the Blind Test set is used for evaluating and comparing the systems' performance submitted. In the Development set, there are totally 793 negative labels and 172 positive labels, which shows an unbalanced dataset [11].

2.2. The evaluation setting

To evaluate on the Development set, the challenge requires to follow five-fold cross-validation [3], each fold comprises Train and Valid subsets as shown in Fig. 3. The evaluation result on the Development set is the average of results on all five folds. To evaluate on the blind Test set, the obtained result on this set is submitted to the Second 2021 DiCOVA Challenge for evaluating, ranking, and comparing with the other submitted systems.

2.3. The evaluation metrics

The 'Area under the ROC curve' (AUC) is used as the primary evaluation metric in the Second 2021 DiCOVA Challenge. The curve is obtained by varying the decision threshold between 0 and 1 with a step size of 0.0001. Additionally, the Sensitivity (Sen.) and the Specificity (Spec.), which are computed at every threshold value, are used as the secondary evaluation metrics (Note that Spec. is required to be equal or greater than 95%). The Leaderboard evaluates the submitted systems on the blind Test set as well as the average performance on five-fold cross validation from the Development set (Avg. AUC) [3].

3. FRAMEWORK ARCHITECTURE PROPOSED

3.1. High-level framework architecture

The overall framework architecture is described as Fig. 1. As the audio recordings show different sample rates, they are firstly re-sampled to 44.1 kHz using mono channel. Then, the re-sampled recordings are fed into the front-end feature extraction where embedding-based features and handcrafted features are extracted and concatenated to obtain the combined features. To deal with the issue of unbalanced dataset mentioned in Section 2.1, SVM-based

SMOTE method [12] is applied on the combined features to make sure the equal number of positive and negative samples. Finally, the features after data augmentation are fed into different back-end classification models for detecting COVID-19 positive cases.

3.2. Front-end Feature Extraction

In this step, we propose a method to create a combined feature by combining handcrafted features and embedding features extracted from pre-trained models. Regarding handcrafted features, 64 Mel-frequency cepstral coefficients (MFCCs), 12 Chromatic (Chroma), 128 Mel Spectrogram (Mel), 1 Zero-Crossing rate, 1 Gender and 1 Duration are used in this paper. These handcrafted features are used as they are popular adoption in speech processing and show robust in the First 2020 DiCOVA Challenge [5, 6, 4]. To extract these handcrafted features, Librosa [13], a powerful library of audio signal processing, is used in this paper. As MFCC, Chromatic and Mel spectrogram are two-dimensional features, they are converted into one-dimensional shape before concatenating with the other features.

As regards the embedding features, we evaluate different embedding features which are extracted from different pre-trained models: YAMNet [14], Wave2Vec [15], TRILL [16], and the COMPARE 2016 feature sets [17] using OpenSMILE [7] toolkit. As using these pre-trained models shows effective for a wide range of classification tasks (i.e. For an example, the pre-trained TRILL model with AudioSet [8] proved robust for a wide range of classification tasks on non-semantic speech signal such as speaker identity, language, and emotional state in [16]), these embeddings are expected to work well with the 2021 DiCOVA Track-2 dataset of cough sounds. By using the pre-trained models, when we feed the cough recordings into the pre-trained models, two-dimensional embeddings are extracted. We then compute mean and standard deviation across the time dimension, concatenating mean and standard deviation to obtain one-dimensional embeddings. The embeddings are then concatenated with the handcrafted features mentioned above to create the combined features. Finally, the combined features are scaled into the range of [0:1] before doing data augmentation and then feeding into the back-end classification models.

3.3. Back-end Classification Models

In this paper, we evaluate different back-end classification models: Light Gradient Boosting Machine (LightGBM), Random Forrest (RF), Support Vector Machine (SVM), Multi-layer Perceptron (MLP), and Extra Tree Classifier (ETC). The setting of these back-end classification models are described in Table 1 and all these models are implemented by using Scikit-Learn toolkit [18]. To obtain results, each classification model is run with 10 seeds numbered from 0 to 9. The output of the cross-validation session will calculated by using soft voting [20] between seeds. The GTX 1080 Titan GPU environment is used for running classification experiments.

4. EXPERIMENTAL RESULTS AND DISCUSSION

4.1. Performance comparison across different features

To evaluate different features, we keep the back-end classification model of LightGBM unchanged while replacing different input features: handcrafted, YAMNet based embedding, COMPARE 2016 based embedding, Wave2Vec based embedding, TRILL based embedding, handcrafted & YAMNet, handcrafted & COMPARE 2016, handcrafted & Wave2Vec, and handcrafted & TRILL features. As the results are shown in Table 2, it can be seen that TRILL-based embedding outperforms the other single features, reporting an Avg. AUC score of 73.77 on the Development set. When we combine the handcrafted feature with different embedding-based features of YAMNet, COMPARE 2016, and TRILL, it is effective to improve the performance, reporting Avg. AUC scores of 77.33, 77.19, and 77.18, respectively compared with 72.62 of using handcrafted feature only. The best performance is obtained from the combination of the handcrafted feature and TRILL-based embedding feature, achieving the AUC, Sen., and Spec. scores of 81.21, 48.33, and 95.13 respectively on the blind Test set.

Table 1. Back-end classification models and setting parameters.

Models	Setting Parameters
Support Vector Machine (SVM)	C=1.0 Kernel='RBF'
Random Forest (RF)	Max Depth of Tree = 20, Number of Trees = 100
Multilayer Perceptron (MLP)	Two hidden layer (4096 nodes), Adam optimization, Max iter = 200 Learning rate = 0.001, Entropy Loss
ExtraTreesClassifier (ETC)	Max Depth of Tree = 20
LightGBM [19]	learning rate = 0.03 objective = 'binary' metric = 'auc' subsample = 0.68 colsample_bytree = 0.28 early_stopping_rounds = 100 num_iterations = 10000 subsample_freq = 1

Table 2. Performance comparison across different features with the back-end LightGBM model (the best performance results are in **bold**).

Extracted Features	AUC (blind test)	Sens. (blind test)	Spec. (blind test)	Avg. AUC (development)
Handcraft	76.36	36.66	95.13	72.62
YAMNet [14]	67.24	21.51	95.13	67.31
COMPARE 2016 [17]	63.18	15.00	95.13	71.00
Wave2Vec [15]	58.86	06.66	95.13	58.75
TRILL [16]	80.57	43.33	95.13	73.77
Handcraft + YAMNet	77.27	41.67	95.13	**77.33**
Handcraft + COMPARE 2016	69.14	25.00	95.13	77.19
Handcraft + Wave2Vec	71.00	25.00	95.13	71.47
Handcraft + TRILL	**81.21**	**48.33**	95.13	77.18

4.2. Performance comparison across different classification models

As we obtained the best handcrafted & TRILL-based embedding feature from the experiments above, we now evaluate how back-end classification models affect the performance. To this end, we keep the handcrafted & TRILL-based embedding feature unchanged while replacing the different back-end classification models: LightGBM, Support Vector Machine (SVM), Random Forest (RF), Extra Trees Classifier (ETC), and Multi-layer perceptron (MLP). As the results are shown in Table 3, the LightGBM model, which is used to evaluate different features, achieves the best scores. Meanwhile, the other models show competitive results, reporting Avg. AUC scores of 75.54, 74.04, 72.50, and 74.87 for SVM, RF, MLP, and ETC, respectively.

Table 3. Performance comparison across different back-end classification models with handcrafted and TRILL based embedding features (the best performance results are in **bold**).

Back-end Classification	AUC (blind test)	Sens. (blind test)	Spec. (blind test)	Avg. AUC (development)
SVM	76.27	36.66	95.13	75.54
RandomForest	78.72	36.66	95.13	74.04
Multi-layer Perceptron	76.34	31.66	95.13	**72.50**
ExtraTreesClassifier	77.51	38.33	95.13	74.87
LightGBM	**81.21**	**48.33**	95.13	77.18

Table 4. Performance comparison across the top-10 systems submitted and the challenge baseline (the best performance results are in **bold**).

Systems	AUC (blind test)	Sens. (blind test)	Spec. (blind test)	Avg. AUC (development)
1st system	**83.31**	43.33	95.38	72.10
2rd system	81.96	40.00	95.13	76.61
3rd (Our system)	81.21	**48.33**	95.13	77.18
4th system	80.12	35.00	95.13	**89.04**
5th system	79.06	35.00	95.13	74.13
6th system	77.85	46.67	95.13	49.31
7th system	77.60	33.33	95.13	77.49
8th system	76.98	40.00	95.13	78.60
9th system	76.36	30.00	95.13	78.12
10th system	75.95	40.00	95.13	74.58
Challenge baseline	74.89	36.67	95.13	75.21

4.3. Performance comparison across the top-10 systems submitted for the Second 2021 DiCOVA Challenge Track-2

The Table 4 presents the performance comparison across the top-10 systems submitted for the Second 2021 DiCOVA Challenge Track-2. As shown in Table 4, our best results from handcrafted & TRILL-based embedding features and LightGBM model achieve the top-3 ranking, reporting the AUC score of 81.21, the Sen. score of 48.33, the Spec. score of 95.13 on the blind Test set, and the Avg. AUC score of 77.18 on the Development set. Notably, our Sen. result on blind Test set and Avg. AUC on the Development set achieve the top-1 ranking. These results prove that our proposed system is robust, competitive, and has the potential to be further applied on edge devices for detecting COVID-19.

5. CONCLUSION AND FUTURE WORK

This paper presents a deep learning-based framework for detecting COVID-19 positive subjects by exploring their cough sounds. By conducting extensive experiments on the Second 2021 DiCOVA Challenge Track-2 dataset, we showed that our best model, which uses a combination of handcrafted & TRILL-based embedding features and LightGBM model, achieve the top-3 ranking of the challenge and are competitive to the state-of-the-art systems.

Our further research are to focus on different sound representations such as Chroma Feature, Spectral Contrast, Tonnetz, etc [21], as well as to explore breathing, speech sounds provided by the Second 2021 DiCOVA Challenge.

6. ACKNOWLEDGEMENT

I would like to express deep gratitude to the organizers and all the teams for making the Second Dicova Challenge competition.

7. REFERENCES

[1] "WHO Coronavirus Disease (COVID-19) Dashboard," https://covid19.who.int/, 2020, [Online; accessed 28-09-2021].

[2] A. Muguli and et al., "Dicova challenge: Dataset, task, and baseline system for covid-19 diagnosis using acoustics," in *Proc. INTERSPEECH*, 2021, pp. 901–905.

[3] N. K. Sharma, S. R. Chetupalli, D. Bhattacharya, D. Dutta, P. Mote, and S. Ganapathy, "The second dicova challenge: Dataset, task, and baseline system for covid-19 diagnosis using acoustics," *arXiv:2110.01177*, 2021.

[4] I. Södergren, M. P. Nodeh, P. C. Chhipa, K. Nikolaidou, and G. Kovács, "Detecting covid-19 from audio recording of coughs using random forests and support vector machines," in *Proc. INTERSPEECH*, 2021, pp. 916–920.

[5] M. R. Kamble, J. A. Gonzalez-Lopez, T. Grau, J. M. Espin, L. Cascioli, Y. Huang, A. Gomez-Alanis, J. Patino, R. Font, A. M. Peinado *et al.*, "Panacea cough sound-based diagnosis of covid-19 for the dicova 2021 challenge," in *Proc. INTERSPEECH*, 2021, pp. 906–910.

[6] V. Karas and B. W. Schuller, "Recognising covid-19 from coughing using ensembles of svms and lstms with handcrafted and deep audio features," in *Proc. INTERSPEECH*, 2021, pp. 911–915.

[7] F. Eyben, M. Wöllmer, and B. Schuller, "opensmile – the munich versatile and fast open-source audio feature extractor," in *MM'10 - Proceedings of the ACM Multimedia 2010 International Conference*, 01 2010, pp. 1459–1462.

[8] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *Proc. ICASSP*, 2017.

[9] J. Chu, "Artificial intelligence model detects asymptomatic covid-19 infections through cellphone-recorded coughs," *MIT News*, pp. 4811–4815, 10 2020. [Online]. Available: https://news.mit.edu/2020/covid-19-cough-cellphone-detection-1029

[10] J. Laguarta, F. Hueto, and B. Subirana, "Covid-19 artificial intelligence diagnosis using only cough recordings," *IEEE Open Journal of Engineering in Medicine and Biology*, vol. 1, pp. 275–281, 2020.

[11] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, "Learning from imbalanced data sets," *Springer*, vol. 10, pp. 275–281, 2018.

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority oversampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[13] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8. Citeseer, 2015, pp. 18–25.

[14] M. Plakal and D. Ellis, "Sound classification with yamnet," https://github.com/tensorflow/models/tree/master/research/audioset/yamnet, 2020, [Online; accessed 05-Oct-2021].

[15] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *CoRR*, vol. abs/2006.11477, 2020.

[16] J. Shor, A. Jansen, R. Maor, O. Lang, O. Tuval, F. de Chaumont Quitry, M. Tagliasacchi, I. Shavitt, D. Emanuel, and Y. Haviv, "Towards learning a universal non-semantic representation of speech," *ArXiv e-prints*, 2020. [Online]. Available: https://arxiv.org/abs/2002.12764

[17] B. Schuller, S. Steidl, A. Batliner, J. Hirschberg, J. K. Burgoon, A. Baird, A. C. Elkins, Y. Zhang, E. Coutinho, and K. Evanini, "The interspeech 2016 computational paralinguistics challenge: Deception, sincerity & native language," in *INTERSPEECH*, 2016.

[18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[19] G. Ke and et al., "Lightgbm: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017, pp. 3149–3157. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf

[20] R. Islam and M. Shahjalal, "Soft voting-based ensemble approach to predict early stage drc violations," *Journal of Artificial Intelligence Research*, pp. 1081–1084, 2019.

[21] H. V. Truong, N. C. Hieu, N. P. Giao, and N. X. Phong, "Unsupervised detection of anomalous sound for machine condition monitoring using fully connected u-net," *Journal of ICT Research and Applications*, vol. 15, pp. 41–55, 2021. [Online]. Available: http://journals.itb.ac.id/index.php/jictra/article/view/15353