

¹ Nkululeko 1.0: A Python package to predict speaker characteristics with a high-level interface

³ Felix Burkhardt  ^{1,2*} and Bagus Tris Atmaja  ^{3*}

⁴ 1 audEERING GmbH, Germany 2 TU Berlin, Germany 3 Nara Institute of Science and Technology (NAIST), Japan * These authors contributed equally.

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Open Journals](#) 

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#))

⁶ Summary

⁷ Nkululeko ([Burkhardt, Wagner, et al., 2022](#)) is a Python toolkit for audio-based machine learning that uses a command-line interface and configuration files, eliminating the need for users to write code. Built on sklearn ([Pedregosa et al., 2011](#)) and PyTorch ([Chaudhary et al., 2020](#)), it enables training and evaluation of speech databases with state-of-the-art machine learning approaches and acoustic features. Key capabilities include model demonstration, database storage with predicted labels, and bias detection through correlation analysis of target labels (e.g., depression) with speaker characteristics (age, gender) or signal quality metrics.

¹⁴ Design Choices

Nkululeko targets **novice users** interested in speaker characteristics detection (emotion, age, gender) without programming expertise, focusing on **education** and **research**. Core design principles include: (1) exploring combinations of acoustic features, models, and preprocessing for optimal performance; (2) database analysis with visualizations; (3) inference on audio files or streams. Users run experiments via a single command: `nkululeko.MODULE_NAME --config CONFIG_FILE.ini`.

²¹ How Does It Work?

²² Nkululeko is a Python command-line tool that uses INI configuration files to specify experiments. Data is imported via CSV format (file path, speaker ID, gender, task labels) or audformat. The functionality is encapsulated by software modules that are called on the command line. Key modules include:

- **nkululeko**: machine learning experiments combining features and learners (e.g., opensmile with SVM);
- **explore**: data exploration and analysis with visualizations;
- **predict**: predict features like speaker diarization, signal distortion ratio, mean opinion score, age/gender with deep learning models;
- **segment**: segment database based on VAD (voice activity detection);
- **ensemble**: combine several models to improve performance;
- **demo**: demonstrate the current best model on command line or files;
- **augment**: augment training data for bias reduction;
- **optim**: search model's best hyperparameters;
- **flags**: run several experiments at once.

³⁷ Configuration files contain sections: DATA (database location, target labels), FEATS (acoustic features: opensmile ([Eyben et al., 2010](#)), wav2vec 2.0 ([Baevski et al., 2020](#))), MODEL

- ³⁹ (classifiers/regressors), and PLOT (visualization). The overall workflow is shown in [Figure 1](#).
⁴⁰ Results include images, text reports, and auto-generated LaTeX/PDF documentation.

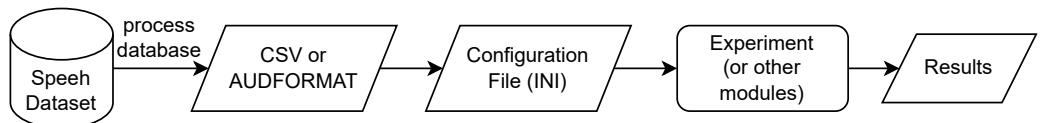


Figure 1: Nkululeko's workflow: from raw dataset to experiment results

⁴¹ Statement of Need

- ⁴² Open-source tools accelerate science through security, customizability, and transparency. While
⁴³ several open-source tools exist for audio analysis—librosa ([McFee et al., 2015](#)), TorchAudio
⁴⁴ ([Yang et al., 2021](#)), pyAudioAnalysis ([Giannakopoulos, 2015](#)), ESPNET ([Watanabe et al.,](#)
⁴⁵ [2018](#)), and SpeechBrain ([Ravanelli et al., 2021](#))—none specialize in speech analysis with
⁴⁶ high-level interfaces for novices. Nkululeko fills this gap with key principles:
- ⁴⁷ 1. minimal programming skills (CSV data preparation and command-line execution);
 - ⁴⁸ 2. standardized data formats (CSV and AUDFORMAT);
 - ⁴⁹ 3. replicability through shareable configuration files;
 - ⁵⁰ 4. high-level INI-file interface requiring no Python coding;
 - ⁵¹ 5. transparency via comprehensive debug output and automated reporting.

⁵² Nkululeko interfaces with Spotlight ([Suwelack, 2023](#)) for enhanced metadata visualization,
⁵³ combining complementary functionalities.

⁵⁴ Usage in Existing Research

- ⁵⁵ Nkululeko has been used in several research projects since 2022 ([Burkhardt, Wagner, et al.,](#)
⁵⁶ [2022](#)):
- ⁵⁷ ▪ Burkhardt, Eyben, et al. ([2022](#)) evaluated synthesized emotional speech databases;
 - ⁵⁸ ▪ Burkhardt et al. ([2024](#)) demonstrated bias detection in UACorpus and Androids datasets;
 - ⁵⁹ ▪ Atmaja et al. ([2024](#)) showcased ensemble learning with uncertainty estimation;
 - ⁶⁰ ▪ Atmaja & Sasou ([2025](#)) evaluated handcrafted acoustic features and self-supervised
⁶¹ learning for pathological voice detection with early/late fusion strategies;
 - ⁶² ▪ Atmaja et al. ([2025](#)) extended ensemble evaluations with performance weighting across
⁶³ five tasks and ten datasets.

⁶⁴ Acknowledgements

- ⁶⁵ We acknowledge support from: European SHIFT project (Grant 101060660); European EASIER
⁶⁶ project (Grant 101016982); Project JPNP20006 (NEDO, Japan); Project 24K02967 (JSPS).
⁶⁷ We thank audEERING GmbH for partial funding.

68 References

- 69 Atmaja, B. T., Burkhardt, F., & Sasou, A. (2025). Performance-weighted ensemble learning for
 70 speech classification. *2025 International Conference on Artificial Intelligence in Information*
 71 *and Communication (ICAIIIC)*. <https://doi.org/10.1109/ICAIIIC64266.2025.10920862>
- 72 Atmaja, B. T., & Sasou, A. (2025). Pathological voice detection from sustained vowels:
 73 Handcrafted vs. Self-supervised learning. *2025 IEEE International Conference on*
 74 *Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*. <https://doi.org/10.1109/ICASSPW65056.2025.11011272>
- 76 Atmaja, B. T., Sasou, A., & Burkhardt, F. (2024). Uncertainty-based ensemble learning for
 77 speech classification. *2024 27th Conference of the Oriental COCOSDA International Com-*
 78 *mittee for the Co-Ordination and Standardisation of Speech Databases and Assessment Tech-*
 79 *niques (o-COCOSDA)*, 1–6. <https://doi.org/10.1109/O-COCOSDA64382.2024.10800111>
- 80 Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for
 81 self-supervised learning of speech representations. In H. Larochelle, M. Ranzato, R. Hadsell,
 82 M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (Vol. 33,
 83 pp. 12449–12460). Curran Associates, Inc. <https://doi.org/10.48550/arXiv.2006.11477>
- 84 Burkhardt, F., Atmaja, B. T., Derington, A., & Eyben, F. (2024). Check your audio data: Nkul-
 85 uleko for bias detection. *2024 27th Conference of the Oriental COCOSDA International Com-*
 86 *mittee for the Co-Ordination and Standardisation of Speech Databases and Assessment Tech-*
 87 *niques (o-COCOSDA)*, 1–6. <https://doi.org/10.1109/O-COCOSDA64382.2024.10800580>
- 88 Burkhardt, F., Eyben, F., & Schuller, W. (2022). SyntAct : A Synthesized Database of Basic
 89 Emotions. In Jonne Sälevä & C. Lignos (Eds.), *Proc. Work. Dataset creat. Low. Lang.*
 90 *Within 13th lang. Resour. Eval. conf.* European Language Resources Association.
- 91 Burkhardt, F., Wagner, J., Wierstorf, H., Eyben, F., & Schuller, B. (2022). Nkululeko: A
 92 tool for rapid speaker characteristics detection. *2022 Language Resources and Evaluation*
 93 *Conference, LREC 2022*, 1925–1932. ISBN: 9791095546726
- 94 Chaudhary, A., Chouhan, K. S., Gajrani, J., & Sharma, B. (2020). *Deep learning with PyTorch*.
 95 <https://doi.org/10.4018/978-1-7998-3095-5.ch003>
- 96 Eyben, F., Wöllmer, M., & Schuller, B. (2010). openSMILE – the munich versatile and fast
 97 open-source audio feature extractor. *MM'10 - Proceedings of the ACM Multimedia 2010*
 98 *International Conference*, 1459–1462. <https://doi.org/10.1145/1873951.1874246>
- 99 Giannakopoulos, T. (2015). pyAudioAnalysis: An open-source Python library for audio signal
 100 analysis. *PLoS One*, 10(12), 1–17. <https://doi.org/10.1371/journal.pone.0144610>
- 101 McFee, B., Raffel, C., Liang, D., Ellis, D., McVicar, M., Battenberg, E., & Nieto, O. (2015).
 102 librosa: Audio and Music Signal Analysis in Python. *Proc. 14th Python Sci. Conf., Scipy*,
 103 18–24. <https://doi.org/10.25080/majora-7b98e3ed-003>
- 104 Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M.,
 105 Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D.,
 106 Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python.
 107 *Journal of Machine Learning Research*, 12, 2825–2830. <https://doi.org/10.5555/1953048.2078195>
- 109 Ravanelli, M., Parcollet, T., Plantinga, P., Rouhe, A., Cornell, S., Lugosch, L., Subakan, C.,
 110 Dawalatabad, N., Heba, A., Zhong, J., Chou, J.-C., Yeh, S.-L., Fu, S.-W., Liao, C.-F.,
 111 Rastorgueva, E., Grondin, F., Aris, W., Na, H., Gao, Y., ... Bengio, Y. (2021). *SpeechBrain: A general-purpose speech toolkit*. <https://doi.org/10.48550/arXiv.2106.04624>
- 113 Suwelack, S. (2023). Spotlight. In *Github repository*. <https://github.com/Renumics/spotlight/>; GitHub.

- 115 Watanabe, S., Hori, T., Karita, S., Hayashi, T., Nishitoba, J., Unno, Y., Soplin, N. E. Y.,
116 Heymann, J., Wiesner, M., Chen, N., Renduchintala, A., & Ochiai, T. (2018). ESPNet:
117 End-to-end speech processing toolkit. *Proc. Annu. Conf. Int. Speech Commun. As-*
118 *soc. INTERSPEECH, 2018-Sept*(September), 2207–2211. [https://doi.org/10.21437/](https://doi.org/10.21437/Interspeech.2018-1456)
119 [Interspeech.2018-1456](https://doi.org/10.21437/Interspeech.2018-1456)
- 120 Yang, S., Chi, P.-H., Chuang, Y.-S., Lai, C.-I. J., Lakhotia, K., Lin, Y. Y., Liu, A. T., Shi,
121 J., Chang, X., Lin, G.-T., Huang, T.-H., Tseng, W.-C., Lee, K., Liu, D.-R., Huang,
122 Z., Dong, S., Li, S.-W., Watanabe, S., Mohamed, A., & Lee, H. (2021). SUPERB:
123 Speech Processing Universal PERformance Benchmark. *Interspeech 2021*, 1194–1198.
124 <https://doi.org/10.21437/Interspeech.2021-1775>

DRAFT