LeBenchmark 2.0: A standardized, replicable and enhanced framework for self-supervised representations of French speech

Titouan Parcollet, Ha Nguyen, Solène Evain, Marcely Zanon Boito, Adrien Pupier, Salima Mdhaffar, Hang Le, Sina Alisamir, Natalia Tomashenko, Marco Dinarelli, et al.

LeBenchmark 2.0: a Standardized, Replicable and Enhanced Framework for Self-supervised Representations of French Speech

Titouan Parcollet[a,b], Ha Nguyen[c], Solène Evain[d], Marcely Zanon Boito[f], Adrien Pupier[d], Salima Mdhaffar[c], Hang Le[d], Sina Alisamir[d], Natalia Tomashenko[c], Marco Dinarelli[d], Shucong Zhang[a], Alexandre Allauzen[e], Maximin Coavoux[d], Yannick Estève[c], Mickael Rouvier[c], Jerôme Goulian[d], Benjamin Lecouteux[d], François Portet[d], Solange Rossato[d], Fabien Ringeval[d], Didier Schwab[d], Laurent Besacier[f]

[a]*Samsung AI Center Cambridge, 50/60 Station Road, Cambridge, CB1 2JH, United Kingdom*
[b]*Department of Computer Science and Technology, University of Cambridge, 15 JJ Thomson Av., Cambridge, CB3 0FD, United Kingdom*
[c]*Laboratoire Informatique d'Avignon, Avignon Université, 339 Chem. des Meinajariès, Avignon, 84000, France*
[d]*Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG, 38000, Grenoble, France*
[e]*ESPCI, CNRS LAMSADE, PSL Research University, France*
[f]*NAVER LABS Europe, France*

Abstract

Self-supervised learning (SSL) is at the origin of unprecedented improvements in many different domains including computer vision and natural language processing. Speech processing drastically benefitted from SSL as most of the current domain-related tasks are now being approached with pre-trained models. This work introduces *LeBenchmark 2.0* an open-source framework for assessing and building SSL-equipped French speech technologies. It includes documented, large-scale and heterogeneous corpora with up to 14,000 hours of heterogeneous speech, ten pre-trained SSL wav2vec 2.0 models containing from 26 million to one billion learnable parameters shared with the community, and an evaluation protocol made of six downstream tasks to complement existing benchmarks. *LeBenchmark 2.0* also presents unique perspectives on pre-trained SSL models for speech with the investigation of frozen versus fine-tuned downstream models, task-agnostic versus task-specific pre-trained models as well as a discussion on the carbon footprint of large-scale model training. Overall, the newly introduced models trained on 14,000 hours of French speech outperform multilingual and previous *LeBenchmark* SSL models across the benchmark but also required up to four times more energy for pre-training.

Keywords: Self-supervised learning, speech processing, dataset, speech benchmark, French language
PACS: 89.20.Ff, 07.05.Mh
2000 MSC: 68T07, 68-04

1. Introduction

Throughout solving pretext tasks automatically extracted from massive unlabeled data, Self-Supervised Learning (SSL) powered deep learning systems deliver groundbreaking performance across a wide range of domains including audio, speech, and language processing [1, 2, 3], computer vision [4, 5], robotics [6], embedded devices and sensors [7], and medicine [8, 9]. In the specific context of speech processing, almost every sub-field has been largely impacted by newly available large pre-trained SSL models. Indeed, impressive improvements and state-of-the-art performance in competitive datasets have been reported for Automatic Speech Recognition (ASR) [10, 11, 12], Automatic Emotion Recognition (AER) [13, 14, 15], Automatic Speaker Verification (ASV) [16, 17, 15], Automatic Speech Translation (AST) [18, 19], Spoken Language Understanding (SLU) [15, 20, 21, 22], Speech Enhancement (SE) [23, 24, 25], Speech Separation (SS) [23, 26], and many others. Despite most leaderboards being conceived around the English language, SSL has also been reported to be remarkably useful for under-resourced languages as demonstrated by A. Babu et al. [19] and H. Nguyen et al. [18], drastically increasing the accessibility to cutting-edge speech technologies across many languages.

Naturally, the flourishing field of SSL for speech calls for fair comparisons and standardized evaluation protocols properly assessing the added value of each newly introduced architecture. Following other early-adopter domains

including Natural Language Processing (NLP) with, for instance, the GLUE [27] and SuperGLUE benchmarks [28], a first English-only evaluation suite appeared: SUPERB [15]. In the latter, 13 different tasks based on well-known datasets have been bundled together to benchmark novel SSL models following a common fine-tuning evaluation protocol. Nonetheless, SUPERB does not standardize the pre-training process and hyperparameters, and models trained on hundreds of thousands of hours of speech appear in the same leaderboard as those learned with a few hundred hours or even different languages. SUPERB has been extended to generative tasks such as speech enhancement following the same standardized evaluation protocol in SUPERB-SG [26] as well as multiple languages in ML-SUPERB [29] or specialised to Indian languages with Indicsuperb [30]. Others benchmarks targetting a specific task, such as SLUE for SLU [31], have also been proposed. *LeBenchmark* approached the issue of SSL benchmarking in French from a unified perspective by freezing the available pre-training data as well as the fine-tuning procedure [32, 14]. It also introduced a set of pre-trained SSL models available to the community including the largest and best-performing French SSL systems. Aside from these two attempts, most SSL models currently are being compared in an arbitrary and heterogeneous fashion across different pre-training and fine-tuning datasets, evaluation protocols, and hyperparameters tuning. The standardization and available resources revolving around SSL evaluation remain scarce, and it is crucial to the community that further efforts are put in those directions. Indeed, the scientific value of a released model may only be validated if proven against a rigorous, fair and replicable evaluation protocol.

With the first version of *LeBenchmark* [32, 14], and following the definition of D. Schlangen [33], we aimed at providing the necessary foundations for the investigation and comparison of SSL models towards French-based downstream tasks. Indeed language-specific SSL, as shown in [32, 14] and the present work, clearly outperform multilingual models depending on the downstream conditions. *LeBenchmark 2.0* builds upon the latter accomplishment to provide a standardized, fully replicable, and extended framework for the assessment and development of SSL representations of French speech. In particular, we release a well-curated pre-training set containing up to fourteen thousand hours of heterogeneous French speech (Section 3), three novel pre-trained SSL models ranging from thirty million to one billion parameters (Section 4), as well as two new evaluation tasks for ASV and syntactic analysis of spoken French (Section 5). *LeBenchmark 2.0* also widens the discussions on the topics of the energy footprint of SSL models (Section 6), the difference between language-specific and language-agnostic pre-training (Section 5). In short, *LeBenchmark 2.0* is a collective attempt at unifying the community of SSL for the French language around common models, datasets and evaluation protocols.

2. Evaluating SSL models with LeBenchmark 2.0: background and motivations

SSL for audio and speech is the process of defining and solving unsupervised proxy tasks, also referred to as pretext tasks or workers, motivated by the nature of the input signal itself. Proxy tasks define both an objective and a transformation applied to the training samples to extract the training targets. In practice, SSL models are first pre-trained following the latter strategy before turning into frozen or fine-tuned feature extractors for common supervised learning tasks. A major benefit of SSL approaches is the ability to leverage the ever-growing mass of available unlabeled data to drastically increase the performance observed on much more expensive and complex to obtain human-labeled tasks. In the context of audio and speech, SSL-based systems occupy the top ranks of most leaderboards and are widely adopted by the community with up to a tenth of recent proceedings from top-tier speech conferences (i.e. year 2023) containing at least one reference to SSL.

SSL strategies for audio and speech may be divided into four different families: generative, predictive, contrastive, and multi-task. Generative methods aim at reconstructing the input audio signal after various corruptions ranging from masking to additive noises. For instance, Mockingjay [34], Tera [20], DecoAR 2.0 [35], Speech-XLNet [36], MPC, pMPC [37] and data2vec [38] optimize their parameters towards the reconstruction or reordering of masked/shuffled input frames while Autoregressive Predictive Coding (APC) [39] reconstructs the input signal. Predictive systems, including WavLM [12], HuBERT [10], or BEST-RQ [40] aim at predicting unsupervised discrete labels (e.g. clustering) obtained from the input samples. Contrastive approaches such as wav2vec 2.0 [11] or Contrastive Predicting Coding (CPC) [41], on the other hand, optimize their latent representation to facilitate the distinction between positive and negative candidates originating from the given signal. Finally, multi-task SSL proposes to combine different objectives or modalities to build a rich feature extractor. For example, PASE+ [42] merges up to ten different workers ranging from signal reconstruction to contrastive learning during the pre-training process.

Such a rich landscape of models may be seen both as a curse and a blessing by the scientific community. It offers a wide range of possibilities and research directions but also suffers from a strong lack of evaluation standards. In fact, even the simple task of identifying the best-performing paradigm for a specific downstream task remains impossible with the current state of the art in SSL for audio and speech. Indeed, the construction and evaluation of SSL models may vary along numerous axes, hence drastically hindering the ease of comparison between novel solutions. *LeBenchmark 2.0* specifically aims at standardizing those axes to speed up, facilitate, and democratize research around SSL pre-training in French.

More precisely, the life cycle of any SSL model is comprised of three major events: pre-training data gathering, training, and downstream evaluation. Ideally, the two latter steps should be merged, enabling the evaluation and comparison of SSL models at pre-training, hence alleviating a time-consuming downstream assessment. In practice, however, this idea appears as a major scientific and technical challenge as the literature relies entirely on the above-described three-step process. Unfortunately, each step may introduce important variations leading to heterogeneous and unreplicable evaluation protocols. For instance, PASE+ was trained on 100 hours of speech while HuBERT processed 60,000 hours, making it easy to define the best-performing model but practically impossible to distinguish the best pre-training strategy. Other variations include, but are not limited to: differences in pre-training languages and data type during step one (e.g. spontaneous against read speech), compute resources at step two (e.g., a single Nvidia GTX 1080 Ti for Mockingjay against 128 Nvidia Tesla V100 for wav2vec 2.0) or the lack of standards during downstream fine-tuning at step three (e.g., fine-tuning against frozen feature extractors, pre-training dataset included or excluded from the downstream evaluation, or simply the list of downstream tasks to include). Ultimately, such requirements, and particularly the need for large compute resources limit the access to SSL pre-training research to a tiny subset of well-equipped institutions and companies, drastically limiting the exploration and emergence of novel paradigms.

Aside from pre-training efficiency, the community naturally attempted to standardize the third step while developing and comparing their models. For instance, ASR evaluation using the Librispeech dataset can be found for MockingJay, wav2vec 2.0, HuBERT, or WavLM, while speaker recognition with VoxCeleb has been reported in PASE+ and MockingJay. Nonetheless, in most cases, the employed downstream architectures, evaluation protocols, or hyper-parameters are entirely different, making it impossible to distinguish models that differ strongly in their pre-training process (e.g. PASE+ and HuBERT). This also prevents a strict comparison between close-performing models (e.g. WavLM and HuBERT).

The increasingly adopted SUPERB benchmark [15] defines a set of English downstream tasks to compare SSL models, hence facilitating step three. Despite a long list of 13 tasks, SUPERB suffers from a constrained fine-tuning procedure that forces all pre-trained SSL feature extractors to remain frozen and use a fixed decoder to solve the task of interest. Unfortunately, state-of-the-art SSL results and real-life use cases mostly, if not only, come with a joint fine-tuning of the SSL extractor and the decoder. S. Zaiem et al. [43] have also demonstrated that freezing all the downstream architectures and reducing them to a tiny subset could lead to misleading leaderboard rankings. Since the data preparation of step one is not standardized within SUPERB, it remains challenging to compare different SSL pre-training methodologies as the amount and quality of the data often vary between available SSL models. *LeBenchmark* is the first attempt at standardizing both steps one and three as well as providing replicable and competitive baselines for step two for further investigation from the community interested in the French language.

Finally, the current trend in large-scale SSL is to associate hundreds of languages [19] during pre-training without any regard to potential biases or degradation in performance induced by such a mixing. However, it remains unclear if combining unrelated and potentially distant dialects may harm the performance observed compared to a model trained on a single and well-resourced language (e.g. English). In particular, with *LeBenchmark* , we decided to benefit from the available unsupervised and heterogeneous French speech corpora available to train multiple language-specific SSL models [32, 14], and we have demonstrated that such well-designed models usually outperform massively multilingual alternatives. Interestingly enough, and despite French being the fifth most spoken language, *LeBenchmark* is the only attempt at standardizing the data collection, pre-training, and evaluation phases of French SSL models. With *LeBenchmark 2.0* we wish to further enhance our already adopted unified SSL framework for the French community, as both industry and academic institutions delivering state-of-the-art speech technologies are now building SSL-powered solutions.

More precisely, *LeBenchmark 2.0* extends [32, 14] in every aspect composing the framework and the three steps of the SSL life-cycle:

- *SSL data collection.* [32] and [14] offered carefully curated and documented corpora with 3,000 and 7,000 hours of French respectively. *LeBenchmark 2.0* extends the openly available pretraining resources to 14,000 hours thanks to a project-specific corpus: *audiocite.net* (see sec. 3.2), shared with the same quality of documentation.
- *SSL models pre-training.* [32] and [14] delivered up to seven pre-trained wav2vec 2.0 SSL models to the community based on the well-known Fairseq toolkit [44]. Following our newly introduced 14,000 hours of data, *LeBenchmark 2.0* brings three more models, of which two are the largest ones available, to the community. Pre-training and model sharing are conducted with HuggingFace and SpeechBrain [45], two frameworks renowned for their open-science-minded approach. We also propose to extend the analysis and discussion on the energy footprint of large SSL models. While we only offer pre-trained wav2vec 2.0 models due to the significant amount of compute resources and energy necessary to train them, we believe that the gathered datasets alongside the benchmarking tasks will emulate the community to extend the analysis to other SSL methods.
- *SSL Benchmarking.* [32] and [14] released four standardized tasks to evaluate and compare SSL models in French: ASR, AST, SLU, and AER. *LeBenchmark 2.0* extends this evaluation protocol to six tasks with the introduction of automatic speaker verification and syntactic analysis. We also widened the comparison with the state-of-the-art, and language-specific against language-agnostic models.

3. Gathering large collections of datasets

Up until recently, it was difficult to find publicly available large datasets of French speech (with the exception of EPAC). Recently, large multilingual corpora that include French have been made available, such as MLS [46] (1,096 h), or voxpopuli [47] (+4,500 h). However, these are restricted to either read or well-prepared speech, failing to provide diversity in the speech samples, such as accented, spontaneous and/or affective speech. In this work, we gathered a large variety of speech corpora in French that cover different accents (MLS, African Accented Speech, CaFE), acted emotions (GEMEP, CaFE, Att-Hack), telephone dialogues (PORTMEDIA), read (MLS, African Accented French, MaSS) and spontaneous sentences (CFPP2000, ESLO2, MPF, TCOF, NCCFr), broadcast speech (EPAC) and professional speech (Voxpopuli). Furthermore, to extend the amount of speech data used for pre-training by around 7k hours we also collected the audiocite.net dataset of non-professional read speech. The details of its design can be found in section 3.2. Compared to MLS and Voxpopuli, our dataset is more diverse, carefully sourced and contains detailed metadata (speech type, and speaker gender). Moreover, it has a more realistic representation of speech turns in real life, compared to MLS and VoxPopuli. Each dataset is documented and can be accessed at least for research purposes.[1] This section summarizes the datasets collected and how they were organized for the pre-training step and gives a short overview of the new *audiocite.net* dataset.

3.1. Overview of the Datasets Used for Pre-training

Table 1 summarizes the statistics of the complete list of datasets considered for the study. The datasets have been organized in five main groups.

Small dataset (≈ 1K hours) is only composed of the MLS corpus for comparison with Wav2Vec2.0 [11] which uses only read English speech. It is also gender-balanced.

Medium-clean dataset (≈ 2.7K hours) contains MLS and EPAC only to enable further investigation on the impact of spontaneous speech on SSL representations. EPAC is a corpus of conversational speech in broadcast news.

Medium dataset (≈ 3K hours) includes 2,933 h of speech, from which 1,115 h is read speech, 1,626 h broadcast speech, 123 h spontaneous speech, 38 h acted telephone dialogues, and 29 h acted emotional speech. Regarding gender, we collected 1,824 h from male speakers, 1,034 h from female speakers, and 74 h from unknown gender.

[1] Some of them being released by ELRA, they are available for a small fee.

Table 1: Statistics for the speech corpora used to train SSL models according to gender information (male / female / unknown). The small dataset is from MLS only. Every dataset is composed of the previous one + additional data; duration: hour(s):minute(s).

Corpus	License	# Utterances	Duration	# Speakers	Mean Utt. Duration	Speech type
		Small dataset – 1K				
MLS French [46]	CC BY 4.0	**263,055** 124,590 / 138,465 / –	**1,096:43** 520:13 / 576:29 / –	**178** 80 / 98 / –	**15 s** 15 s / 15 s / –	Read
		Medium-clean dataset – 2.7K				
EPAC** [48]	ELRA NC	**623,250** 465,859 / 157,391 / –	**1,626:02** 1,240:10 / 385:52 / –	**Unk** – / – / –	**9 s** – / – / –	Radio Broadcasts
2.7k dataset total		**886,305** 590,449 / 295,856 / -	**2,722:45** 1,760:23 / 962:21 / -	-	-	-
		Medium dataset – 3K				
African Accented French [49]	Apache 2.0	**16,402** 373 / 102 / 15,927	**18:56** – / – / 18:56	**232** 48 / 36 / 148	**4 s** – / – / –	Read
Att-Hack [50]	CC BY-NC-ND	**36,339** 16,564 / 19,775 / –	**27:02** 12:07 / 14:54 / –	**20** 9 / 11 / –	**2.7 s** 2.6 s / 2.7 s / –	Acted Emotional
CaFE [51]	CC NC	**936** 468 / 468 / –	**1:09** 0:32 / 0:36 / –	**12** 6 / 6 / –	**4.4 s** 4.2 s / 4.7 s / –	Acted Emotional
CFPP2000* [52]	CC BY-NC-SA	**9853** 166 / 1,184 / 8,503	**16:26** 0:14 / 1:56 / 14:16	**49** 2 / 4 / 43	**6 s** 5 s / 5 s / 6 s	Spontaneous
ESLO2 [53]	CC BY-NC-SA	**62,918** 30,440 / 32,147 / 331	**34:12** 17:06 / 16:57 / 0:09	**190** 68 / 120 / 2	**1.9 s** 2 s / 1.9 s / 1.7 s	Spontaneous
GEMEP [54]	User agreement	**1,236** 616 / 620 / –	**0:50** 0:24 / 0:26 / –	**10** 5 / 5 / –	**2.5 s** 2.4 s / 2.5 s / –	Acted Emotional
MPF [55], [56]	CC BY-NC-SA 4.0	**19,527** 5,326 / 4,649 / 9,552	**19:06** 5:26 / 4:36 / 9:03	**114** 36 / 29 / 49	**3.5 s** 3.7 s / 3.6 s / 3.4 s	Spontaneous
PORTMEDIA (French) [57]	ELRA NC	**19,627** 9,294 / 10,333 / –	**38:59** 19:08 / 19:50 / –	**193** 84 / 109 / –	**7.1 s** 7.4 s / 6.9 s / –	Acted telephone dialogue
TCOF (Adults) [58]	CC BY-NC-SA	**58,722** 10,377 / 14,763 / 33,582	**53:59** 9:33 / 12:39 / 31:46	**749** 119 / 162 / 468	**3.3 s** 3.3 s / 3.1 s / 3.4 s	Spontaneous
Medium dataset total		**1,111,865** 664,073 / 379,897 / 67,895	**2,933:24** 1,824:53 / 1,034:15 / 74:10	-	-	-
		Large dataset – 7K				
MaSS [59]	MIT	**8,219** 8,219 / – / –	**19:40** 19:40 / – / –	**Unk** – / – / –	**8.6 s** 8.6 s / – / –	Read
NCCFr [60]	User agreement	**29,421** 14,570 / 13,922 / 929	**26:35** 12:44 / 12:59 / 00:50	**46** 24 / 21 / 1	**3 s** 3 s / 3 s / 3 s	Spontaneous
Voxpopuli [47] *Unlabeled*	CC0	**568,338** – / – / –	**4,532:17** – / – / 4,532:17	**Unk** – / – / –	**29 s** – / – / –	Professional speech
Voxpopuli [47] *transcribed*	CC0	**76.281** – / – / –	**211:57** – / – / 211:57	**327** – / – / –	**10 s** – / – / –	Professional speech
Large dataset total* **		**1,814,242** 682,322 / 388,217 / 99,084	**7,739:22** 1,853:02 / 1,041:07 / 4,845:07	-	-	-
		Extra Large dataset – 14K				
Audiocite.net (SLR139) [61]	CC BY + ND/NC/SA	**817 295** 425 033 / 159 691 / 232 571	**6698:35** 3477:24 / 1309:49 / 1911:21	**130** 35 / 32 / 63	**29 s** 29 s / 29 s / 29 s	Read
Niger-Mali Audio collection [62] [63]	CC BY-NC-ND	**38 332** 18 546 / 19 786 / –	**111:01** 52:15 / 58:46 / –	**357** 192 / 165 / –	**10 s** 10 s / 10 s / –	Radio broadcasts
Extra Large dataset total		**2 669 869** 1 125 901 / 567 694 / 331 655	**14 548:58** 5 382:41 / 2 409:42 / 6 756:28	-	-	-

*Composed of audio files not included in the CEFC corpus v2.1, 02/2021; **speakers are not uniquely identified.; ***Stats of CFPP2000, MPF and TCOF have changed a bit due to a change in data extraction; License: CC=Creative Commons; NC=non-commercial; BY= Attribution; SA= Share Alike; ND = No Derivative works; CC0 = No Rights Reserved; User agreement = open for research and NC.

Large dataset (≈ 7.7K hours) has 4 additional corpora: MaSS, NCCFr and Voxpopuli (unlabeled + transcribed). It includes 7,739 h of speech, from which 1,135 h is read speech, 1,626 h broadcast speech, 165 h spontaneous speech, 38 h acted telephone dialogues, 29 h acted emotional speech, and 4744 h professional speech. Except for NCCFr, no information about gender is given in these datasets.

New Extra large dataset (≈ 14K hours) has two additional corpora: audiocite.net and Niger-Mali Audio Collection. Audiocite.net includes freely shareable audiobooks of more than 6 600 hours. We created this dataset specifically for the project and section 3.2 gives details about how it has been acquired. The Niger-Mali Audio Collection is data web-crawled from Studio Kalangou and Studio Tamani websites, with the authorization of Fondation Hirondelle. The gender labels were automatically produced by the `LIUM_SpkDiarization` tool [64]. With these two added datasets, the Extra-large dataset is then composed of read speech (7,834 hours), broadcast speech (1,737 h), spontaneous speech (165 h), acted telephone dialogues (38 h), acted emotional speech (29 h), and professional speech (4,744 h).

New Gender-specific datasets (≈ 1k hours) are built using all datasets present in the Large dataset that contain gender information: MLS, Att-Hack, CaFE, CFPP2000, ESLO2, EPAC, GEMEP, PORTMEDIA, TCOF, NCCFr. For EPAC, we keep the totality of female speech (385 h), and downsample the male speech to a comparable amount (413 h). This results in 1,041 h of female speech, and 1,006 h of male speech in the final gender-specific datasets.

Pre-processing for SSL training: Recordings were segmented using time stamps from transcriptions, or cut every 30 seconds when there was no transcription (VoxPopuli *unlabeled*, audiocite.net). When available, we retrieved speaker labels and gender information. Following [11], we removed utterances shorter than 1 s, and longer than 30 s. When possible, overlapping speech sentences were also removed. When necessary, audio segments were converted to mono PCM 16 bits, 16 kHz.

3.2. Audiocite.net Dataset

Audiocite.net is a corpus of read French speech scrapped from the www.audiocite.net website in November 2021 thanks to the kind authorization of the website administrator. The website is composed of voluntary work of speakers who explicitly uploaded their content under a CC licence[2] The audiobooks are available online for free and are classified into 15 categories: tales, world news, short stories, poetry, erotic stories, documentaries, science fiction, novels, animals, audiocite-juniors, religions, kitchen, philosophies, history and theatre. All the original texts are either in the public domain or under an open license.

The Audiocite.net is composed of more than 6600 hours or recordings from 130 speakers and can be found distributed on OpenSLR (www.openslr.org/139/) with the same license as the original work. The recordings were labeled within 15 categories from junior books to philosophy with the novel category being the majority (about 50%). All the recordings are distributed in their raw format as we downloaded them from audiocite.net (with background music, noise, unexpected speakers, mp3 format, mono or stereo). No pre-processing was applied to the files nor ASR performed on them. We, however, added information of the gender in a 'best effort' manner by guessing the gender from the name and checking the voice in case of uncertainty. This information must not be considered as ground truth and is only intended to be used for a rough statistical estimate. This estimate indicates that female voice represents 34% of the speech duration. No attempt to remove speech that could be seen as offensive or sensitive was made. Although the dataset is provided with training, validation and testing partitions, the whole corpus was used for LeBenchmark 14K model training. Further details about the corpus can be found on the OpenSLR website.

4. Building an Open Collection of Pre-trained French SSL Models

LeBenchmark 2.0 introduces three novels pre-trained French wav2vec 2.0 to the community based on the Extra Large dataset (i.e. 14,000 hours of speech): 14K-light, 14K-large and 14K-xlarge. More precisely, *LeBenchmark 2.0* is an open collection of 14 pre-trained SSL models made entirely available on the HuggingFace platform[3]. It is worth noticing that the number of released SSL models has doubled from *LeBenchmark* to *LeBenchmark 2.0* as four others have been added for preliminary gender analyses from M. Z. Boito et al. [65]. The latter four models are not depicted in Table 2, as they were introduced in [65]. In practice, the three new models cover different use cases as the 14K-large and 14K-xlarge are expected to deliver top-notch performance in unconstrained and centralized environments while our 14K-light will bring SSL features to more resource-constrained devices. We decided to replace the standard base model with an even smaller version (14K-light) as previous works [14] have shown minimal performance improvement for base models when the amount of pre-training data was increasing. The evaluation of small models pre-trained from scratch, however, was a fairly open question. As of now, these additions represent both the most powerful and parameters-efficient SSL-powered models for the French language.

[2] Some of them have supplementary conditions as SA (Share Alike), ND (No Modification) or NC (No commercial Use) while others are in the public domain (CC-0).

[3] `https://huggingface.co/LeBenchmark`

Table 2: Summary of the pre-trained wav2vec 2.0 models delivered with LeBenchmark and LeBenchmark 2.0. Newly released models are denoted in bold. "GPU Hours" refer to the total training time cumulated over "GPU Count" to reach the number of "Updates".

Model	Pre-training data	Parameters Count	Output Dimension	Updates	GPU Count	GPU Hours
1K-*base*	1,096 h	90M	768	200K	4	1,000
1K-*large*	1,096 h	330M	1024	200K	32	3,700
2.7K-*base*	2,773 h	90M	768	500K	32	4,100
3K-*base*	2,933 h	90M	768	500K	32	4,100
3K-*large*	2,933 h	330M	1024	500K	32	10,900
7K-*base*	7,739 h	90M	768	500K	64	7,900
7K-*large*	7,739 h	330M	1,024	500K	64	13,500
LeBenchmark 2.0						
14K-*light*	14,000 h	26M	512	500K	32	5,000
14K-*large*	14,000 h	330M	1,024	1M	64	28,800
14K-*xlarge*	14,000 h	965M	1,280	1M	104	54,600

4.1. On the Choice of Wav2vec 2.0

At the time of *LeBenchmark*, wav2vec 2.0 was the only state-of-the-art open-source and available SSL pre-training strategy. Other alternatives such as MockingJay [34] or CPC-based methods [66] were already invented, but could not guarantee, at least from the original articles, state-of-the-art performance once scaled to hundreds of millions of parameters and many thousands of hours of pre-training data. Wav2vec 2.0 naturally fitted our requirements as it was also achieving state-of-the-art performance. According to the SUPERB benchmark [26], the three best-performing pre-training strategies to date are WavLM, HuBERT, and wav2vec 2.0. However, no implementation of WavLM may be found to replicate the pre-training process and the reported results. HuBERT, on the other hand, suffers from a much more complex training process due to the iterative refining of the discrete targets obtained with k-means. At the time of writing, a new faster implementation of HuBERT is now available [67], but this was not the case when *LeBenchmark* started. Furthermore, and as depicted in [26], the downstream performance of *BASE* and *LARGE* models for HuBERT and wav2vec 2.0 are similar despite a slight advantage for HuBERT, potentially originating from an extensive hyperparameters tuning. Indeed, from our experience, the SSL pre-training behavior varies significantly following hyperparameter changes. In summary, the wav2vec 2.0 architecture enables *LeBenchmark 2.0* to compare fairly with previously introduced French models while retaining state-of-the-art performance compared to existing alternatives.

Finally, and despite a clear interest, it quickly appeared intractable to consider pre-training from scratch multiple SSL methods. Indeed, and as shown in Table 13, the total energy cost as well as necessary compute resources would have exceeded by far the available limits. Hence, we decided to focus on a single architecture while giving to the community both the full pre-training dataset as well as downstream tasks to further push the study with better or newly introduced SSL algorithms.

4.2. Pre-training Hardware and Sofware Environments

Large-scale SSL pre-training was mostly conducted within the Jean Zay French supercomputer converged platform made available to researchers by GENCI[4]. As of January 2023, Jean Zay is offering access to 2,696 Nvidia Tesla V100 (16GB and 32GB) split between four and height GPU per node with dual Intel CPU as well as 416 Nvidia Tesla A100 80GB with dual AMD CPU. Jean Zay is mostly powered by nuclear energy, hence facilitating a low carbon emission rate per FLOPS. The reported Power Usage Effectiveness (PUE) ratios are 1.21 and 0.65 depending on the scenario (i.e. considering the heat transfer to nearby infrastructures), putting Jean Zay in the list of the most efficient supercomputers worldwide[5]. Most models except 14K-xlarge have been trained on nodes equipped with four 32GB

[4]GENCI Jean Zay official presentation: `http://www.idris.fr/eng/jean-zay/jean-zay-presentation-eng.html`

[5]`https://systematic-paris-region.org/wp-content/uploads/2022/06/slideshow-Hub-Day-HPC-Hybride.pdf`

Nivida Tesla V100 hence triggering multi-node training to reach the desired 32 or 64 GPU. 14K-xlarge was trained with 80GB Nvidia Tesla A100 nodes equipped with height GPU each. Data read and write operations were made throughout a fast Nested File System (NFS) without any streaming library. A total of 2.9 TB of storage was necessary to hold the entire 14,000 hours dataset. In practice, wav2vec 2.0 models could be trained with much fewer and less powerful GPU, but at the cost of significantly longer training time (i.e. mostly intractable) due to the gradient accumulation needed to reach the large batch size required by the contrastive learning nature of wav2vec 2.0.

The speech and audio processing toolkits landscape has significantly expanded in the last decade, however, only two tools support the full wav2vec 2.0 pre-training: Fairseq [44] and SpeechBrain [45]. In practice, all models trained with the Large dataset (7,000 hours) or smaller sets have been produced with Fairseq, while the others have been trained with SpeechBrain by adapting the wav2vec 2.0 CommonVoice recipe to our data. The change to SpeechBrain was motivated by a simpler and faster pre-training. In practice, we also re-trained a 7k-large model with SpeechBrain to make sure that the performance was identical to the Fairseq model. The Python environments are corresponding to those detailed in the toolkit installation scripts attached to each commit.

4.3. Wav2vec 2.0 Hyperparameters

The wav2vec 2.0 architecture can be summarized in four distinct blocks: an acoustic feature extractor made of a convolutional neural network, a latent or contextual extractor composed of a Transformer network, a quantization module, and the final contrastive block. The entire detailed list of architectural hyperparameters (i.e. more than 70 parameters) for each model can be found in the corresponding HuggingFace repository[6]. In short, all models share the same CNN encoder architecture and mostly differ in the hidden dimension size and depth of the Transformer and quantizer. For instance, the sizes of the intermediate and hidden layers are $[2048, 3072, 4096, 5120]$ and $[512, 768, 1024, 1280]$ for the *light*, *base*, *large*, and *xlarge* models respectively. The number of blocks in the Transformer also increases following the model size with $[6, 12, 24, 48]$. In practice, *LeBenchmark 2.0* follows the configurations initially reported by A. Baevki et al. [11], A. Babu et al. [19] and T. Ashihara et al. [68] as extensive hyperparameter and architecture searches are intractable even with Jean Zay resources.

4.4. Pre-training Hyperparameters

The extensive list of pre-training hyperparameters is reported in the *LeBenchmark* HuggingFace repository while the most impactful ones are given in the following. The duration of each model pre-training is measured in *"steps"* or *"updates"* referring to an effective update of the neural network weights or a call to the *".backward()"* function in PyTorch. This quantity varies with the amount of available data as well as the number of neural parameters to optimize. For instance, the 14K-xlarge made one million updates compared to 200,000 for the 1K-large. Increasing the number of updates ultimately leads to better downstream performance. Nevertheless, the latter behavior must be contrasted with the high cost associated with longer training times as many dozens of GPU are being used at once. Again, we fixed the number of steps according to the original wav2vec 2.0 and XLS-R. All models are trained with the Adam optimizer and decoupled weight decay (i.e. AdamW) [69] following a two-step scheduler made of around 8% of warmup steps and a polynomial decay to zero.

Each training step is then associated with an effective batch size measured, for convenience, in seconds. All models from *LeBenchmark 2.0* have been trained with an effective batch size of between two and three hours. For instance, the 14K-large model used 40 GPU that could fit 118 seconds of speech each per batch alongside a gradient accumulation factor of two, resulting in a total effective batch size of $(40 \times 118 \times 2)/3600 = 2.63$ hours of speech signal per step.

Ensuring a constant effective batch size necessitates a constant amount of signal per GPU. To this extent, both Fairseq and SpeechBrain toolkits implement a dynamic batching mechanism that automatically bundles samples together depending on their size to match the desired maximum boundary. The latter boundary depends on the VRAM capacity of the GPU and varies with the size of the model. For instance, the 14K-large model was trained with a boundary of 118 seconds on 32GB GPU while the 14K-xlarge model stayed at 60 seconds with 80GB GPU.

To limit the VRAM consumption due to the quadratic increase in complexity of the Transformer self-attention following the sequence length, all samples are cropped at 20 seconds. The remaining signal is simply used as a

[6] `https://huggingface.co/LeBenchmark`

different example. Similarly, and to avoid masking the entirety of the sample for contrastive loss, segments shorter than 2 seconds are removed from Fairseq models while they are concatenated with SpeechBrain. For the largest models, i.e. 14K-xlarge, audio samples are cropped at 15 seconds as no real degradation from slightly shorter sequences is to be expected, as demonstrated by Y. Gao et al [70].

Finally, masks applied to the output of the feature extractor (i.e. CNN) are of 10 consecutive frames for all models. Masking probabilities, however, change with the model size. Indeed, 50% of the frames are masked for *base* models compared to 75% for the *large* and *xlarge* models as reported by A. Baevki et al. [11], A. Babu et al. [19].

4.5. Wav2vec 2.0 Pre-training: tips and tricks

Due to the very high entry ticket to pre-training large SSL models, almost no resources exist to help the community with this process. In particular, we found both Fairseq and SpeechBrain wav2vec 2.0 pre-training recipes do not simply transfer seamlessly to the *LeBenchmark 2.0* datasets. Such difficulties in training certainly originate from the high complexity of the pipeline: hundreds of millions of neural parameters on thousands of hours of speech with dozens of distributed GPU. In the following, we propose a few tips and tricks to avoid the two most common issues encountered while pre-training *LeBenchmark 2.0* models: exploding losses and collapsing representations.

Exploding losses (i.e. NaN values) were the most common issue faced when training for longer periods of time. Indeed, all models trained for more than 500M steps experienced infinite losses at some point whether it was with Fairseq or SpeechBrain. As expected, simply changing the learning rate did not help as both toolkits already carefully designed the scheduler as well as gradient clipping strategies to avoid any perturbation in the gradient flow. In fact, the mixed precision training was most of the time responsible for this issue. Hence, upon explosion, a simple resuming of the training with full precision (i.e. fp32) was sufficient to finish the training. The resulting VRAM increase was low enough to enable training with the same batch size. This may be explained by extremely small weights or values appearing after extended training with AdamW due to the weight decay, hence reaching the rounding-to-zero floor of 16-bit floats. It is worth noticing that switching to bfloat16 (i.e. fp32 values range) instead of float16 solved entirely the issue with SpeechBrain while preserving the training speed.

Collapsing representations may happen when the contrastive task is too hard or too simple. It may easily be spotted at training time as the accuracy, defined as the cosine similarity between projected and quantized latent representations over the initially masked frames, quickly jumps to values close to 100%. In practice, this can easily be avoided by increasing the diversity loss and the mask length or probability. Indeed, a very high accuracy may simply mean that only very few entries of the quantization codebook are used, hence easy to predict. The latter phenomenon may especially arise with a dataset composed of short audio segments.

5. Standardized and Replicable Benchmark of French Tasks for SSL Models

LeBenchmark 2.0 introduces two new candidates to *LeBenchmark* for a total of six tasks: automatic speech recognition (ASR), spoken language understanding (SLU), automatic speech translation (AST), automatic emotion recognition (AER), syntactic analysis (SA) and automatic speaker verification (ASV). We designed our benchmark to best cover the different criteria that the community may expect from pre-trained extractors. In particular, SSL models must integrate information related to transcription (ASR), semantics (SLU, SA), translation (AST), and paralinguistics (AER, ASV). To validate the robustness of SSL to varying data volumes, we also selected corpora matching all potential use cases: high (ASR, AST), medium (SLU, AST, ASV), and low (ASR, AER, SA) resource.

SSL Models configurations. In all the following evaluations, SSL models may be described as *"fine-tuned"* or *"frozen"* and *"task-agnostic"* or *"task-specific"*. Indeed, and contrary to the SUPERB benchmark, we are investigating different scenarios corresponding to various real-life applications. In SUPERB, all models are frozen, meaning that the neural parameters of the SSL models are not updated with the downstream task of interest. Having a heterogeneous set of downstream decoders is of crucial importance to avoid variations in the final ranking as demonstrated by S. Zaiem et al. [43]. In *LeBenchmark 2.0* , we also investigate the results obtained with fine-tuned models, where both the SSL model and the downstream decoder are trained to solve a given task, hence reaching much higher levels of performance. The latter is done at the cost of a more compute resource-intensive fine-tuning stage. Task-agnosticism or specificity simply defines the standard pre-trained SSL models or their already finetuned equivalent. For instance,

one may wish to first fine-tune a wav2vec 2.0 architecture on speech recognition before evaluating it on SLU. The latter ASR-specific model is referred to as being "task-specific".

Considered SSL baselines. Among the different evaluations reported below, *LeBenchmark 2.0* aims at providing two different levels of study: (a) evaluate the relative performance improvement between the different provided French SSL models, and (b) evaluate the relevance of language-specific SSL models against multilingual or out-of-domain large-scale models. First, and as *LeBenchmark 2.0* is the only resource available for French SSL, the former comparison will be only conducted with our own models. Second, *LeBenchmark 2.0* wav2vec will be compared to XLSR-53 [71] and XLS-R-1B [19], which for consistency we refer as *XLS-R-xlarge*, for the multilingual aspect. Whisper [72] will not be considered as two major drawbacks prevent its use in a fair comparison: (a) the training data is virtually unknown and may already contain the train or test sets of our downstream tasks, and (b) it is based on weakly supervised training, not SSL.

5.1. Automatic Speech Recognition

Automatic speech recognition is a common downstream task to evaluate SSL models. We investigated the behavior of the different *LeBenchmark 2.0* models with two scenarios: a challenging low-resource scenario with only 22 hours of training data on TV and radio shows, and a high-resource scenario with 428 hours of read speech.

Downstream datasets. Two different French datasets were used. The first one is the ETAPE corpus. This is the official dataset released during the French ETAPE evaluation campaign in 2011 [73]. It is composed of diverse French shows: TV news, TV debates, TV amusement, and radio shows. These shows are split into three subcorpora – training: 22 h, validation: 7 h, and testing: 7 h. ETAPE is distributed in the ELRA catalogue.[7] It is free for academic research. The second dataset is the French part, version 6.1, of the well-known CommonVoice project [74]. This project started in July 2017 and employs crowdsourcing for both speech data collection and human validation for a large number of languages. The speech data is made of read sentences extracted from Wikipedia. It contains 428 h, 24 h, and 25 h of speech data for the training, validation, and testing sets respectively.

Downstream models and hyperparameters. To conduct our experiments, we employed the SpeechBrain toolkit [45], which is built on PyTorch and designed specifically for speech-processing tasks. Additionally, we utilized the Hugging Face version of the *LeBenchmark* models even though fairseq checkpoints are also available. The SpeechBrain toolkit offers a diverse array of recipes, and to ensure the reproducibility of our experiments, we followed the SpeechBrain recipe specific to the CommonVoice ASR task. In both of the aforementioned scenarios, we initiated with a pre-trained *LeBenchmark* model and added three additional dense hidden layers of size 1,024 on top with random initialization. Each of these layers was associated with the LeakyReLU activation function. Subsequently, we performed fine-tuning for speech recognition on the training data by applying the SpecAugment data augmentation technique. For optimization, we employed the Connection Temporal Classification (CTC) loss function [75]. The overall model's output consists of 78 tokens, encompassing both characters, sub-word units, and the CTC blank character. This number is higher than in English due to the presence of numerous accented letters in French. For example, variations derived from the letter e include é, è, ê, and ë. Other letters like ç or œ have also to be taken into account. Following the SpeechBrain recipe for CommonVoice, we optimized the model using two optimizers. The Adam optimizer was dedicated to the parameters derived from the *LeBenchmark* wav2vec2.0 model, while the AdaDelta optimizer was used for all the parameters on top of it. We applied a dropout technique to train the three top dense hidden layers, with a probability of 0.15. For each experiment, the training was made on 80 epochs, keeping the model that reached the best results on the development data.

LeBenchmark 1.0. In a prior study [14], we demonstrated that *LeBenchmark* SSL models pre-trained solely on French data outperformed comparable models pre-trained on English (e.g. wav2vec 2.0 on Librispeech) or multilingual data (e.g. XLSR-53), which also included French. Our new findings focus on assessing the impact of additional data on the pretraining of the French dataset used for SSL. Moreover, we exclude our previous frozen SSL setting as it

[7] `https://catalogue.elra.info/en-us/repository/browse/ELRA-E0046/`

Table 3: ASR results in terms of word error rate (WER%, lower is better) on Common Voice and ETAPE corpora, with pre-trained wav2vec 2.0 models fine-tuned on labeled ASR data. Gray numbers denote the standard deviation.

Corpus	**CommonVoice**		**ETAPE**	
Representation	**Dev**	**Test**	**Dev**	**Test**
1K-*large*	9.49±0.20	11.21±0.23	28.57±0.79	30.58±0.88
3K-*large*	**8.00**±0.19	**9.27**±0.20	22.26±0.76	24.21±0.85
7K-*large*	8.02±0.18	9.39±0.21	**21.34**±0.74	**23.46**±0.83
14K-*light*	19.86±0.28	22.81±0.34	58.30±0.66	59.82±0.7
14K-*large*	8.39±0.19	9.83±0.21	23.67±0.81	26.03±0.89
14K-*xlarge*	8.26±0.19	9.83±0.21	22.38±0.95	24.67±0.83

is consistently inferior to end-to-end fine-tuning, and we find that it is also a setup that is becoming less and less relevant to the ASR community. Our previous investigations also showed that end-to-end ASR fine-tuning is consistently more effective using large models as initialization, and thus our *LeBenchmark* benchmark for ASR present results only for large versions of already evaluated models (1K, 3K, 7K), and for all new models (14K) from Table 2. Finally, for baselines, we consider XLS-R-xlarge only, as XLSR-53 was previously evaluated, and it presented inferior results to all *LeBenchmark* models (base or large).

Results analysis and discussions. As depicted in Table 3, and consistent with the results described in [14], the 1K-*large* model exhibits a higher word error rate compared to the 3K-*large* model. Section 3.1 provides detailed information about the content of the pretraining datasets used for the different models. Incorporating 2,000 hours of primarily broadcast news speech with the initial 1,000 hours of read speech used to pretrain the 1K-large model significantly improves the performance of the 3K-large model for speech recognition. However, further augmentation with 7,000 hours of formal read or prepared speech (parliamentary events) does not yield substantial improvement for the 7K-large model. Nevertheless, the performance of the 7K-large model is still significantly better than the 3K-large on broadcast news (ETAPE) and comparable to read speech (CommonVoice). This phenomenon is worsened by the introduction of the 14K hours dataset, as 14K-large and 14K-light are not able to outperform even the 3K-large. This can be explained by the nature of the added data, i.e. read speech, that may simply not help at reaching better performance above a certain threshold. In most cases, the 14K-light model exhibits degraded performance for automatic speech recognition even though speech recognition rates remain better than those of XLSR-53 reported in [14].

5.2. Spoken Language Understanding

Spoken Language Understanding (SLU) aims at extracting semantic representations from speech signals containing sentences in natural language [76]. Classical approaches to SLU used a cascade model made of an ASR system feeding a Natural Language Understanding (NLU) module [77, 78, 79, 80, 81, 82, 83]. Neural networks led to large advances for *end-to-end* SLU systems [84, 85, 86, 87, 88, 89, 90, 91], which are preferred to cascade systems, in particular for their ability to reduce error propagation effects and to exploit acoustic components to deduct semantic information [92].

Downstream datasets. For French SLU benchmarking we used the well-known MEDIA corpus [93], used also in [14] and allowing thus for direct comparison. The MEDIA corpus focuses on the domain of hotel information and reservations in France. It is made of 1,250 human-machine dialogues transcribed and annotated with 76 semantic concepts. The complete MEDIA task requires models to extract both concepts and their normalized values. For instance from the (chunk of speech whose transcription is the) phrase *"two double rooms"* the model must predict the following semantic annotation: `room-number`[2] `room-type`[double]. All models applied to this task in the past were trained to extract concepts only (`room-number` and `room-type`), while normalized values are extracted in a post-processing step using rule-based modules. The latter is necessary because of the presence of open-domain values, like dates or amounts (e.g. *"from October 26th ..."* must be annotated as `start-date[26/10/2024]`), which are difficult to normalize for models trained from scratch, while that is an easy task for rule systems. Like in previous work thus [32, 14], we train models to predict tokens, concept boundaries and concepts directly from speech. This

means models must learn to transcribe, segment and annotate speech signals, which is a much more difficult task than predicting domain, intent and slot-value pairs like in more traditional SLU tasks such as FSC. A comparison of performances on MEDIA and FSC tasks can be appreciated in our previous work [94], although results on FSC are always given in terms of accuracy, even an end-to-end model with basic features reaches an accuracy over 90%.

The MEDIA corpus is split into 12,908 utterances (41.5 hours of speech) for training, 1,259 for development (3.5 hours), and 3,005 for test (11.3 hours).

Downstream models and hyperparameters. SLU models used in this paper are the same as in [14], few modifications have been introduced which will be described along this section. Such models have a *sequence-to-sequence* architecture based on LSTMs and attention mechanisms [95, 96]. The encoder has a similar pyramidal structure as the one proposed in [97], the decoder uses two attention mechanisms, one for attending the encoder's hidden states, and one for attending the decoder's previous predictions, like the self-attention module of Transformers [98]. One difference with respect to LeBenchmark models proposed in [14] is that we added a layer normalization after each decoder layer, which made learning more stable when using features extracted with SSL models as input, and improved the model's generalization. All models were trained to minimize the CTC loss [75]. We note that our models can be trained also with a *cross-entropy* loss, which is often used with sequence-to-sequence architectures predicting discrete symbols. However in preliminary experiments we found that CTC loss leads always to significantly better performances. In all our experiments we use SSL models as feature extractors. Features were given as input to SLU models as an alternative to traditional features (e.g. MFCC). Following [14], we used both task-agnostic and task-specific SSL models. In the task-specific case, SSL models were fine-tuned for the ASR output like in [14], and we use XLSR-53-large and XLS-R-xlarge as baselines.

Models described in [14] were learned with three training steps. Each training step uses the model learned in the previous step for initializing the current model's parameters. While this strategy is the most effective, it implies a relatively high training cost. In this work we instead use full end-to-end training such as in [94]. Hence, models are learned from scratch in a single training procedure. In addition, in this work, we tested a multi-task learning setting where the encoder and the decoder are learned with two different CTC losses: with respect to the ASR output for the encoder; with respect to the SLU output for the decoder following a standardized output format [14]. This multi-task learning setting will be indicated with *mt* in Table 4. In order to study the impact of the SLU model size on results, especially when using features from SSL models, we tested hidden layers of size 256 and 300. Since these gave similar results in most cases, we did not further optimize the model size. We also found it benef7icial, when using fine-tuned SSL model's features, to increase the temperature at the output softmax layer to 2 (indicated as *t2* in the table). This strategy has been used successfully for model distillation [99], and intuitively has a similar effect as using smoothed label representations as targets [100]. Beyond these differences, all models use the same hyperparameters as those used in [14].

LeBenchmark 1.0. Similarly to ASR, in the previous version of *LeBenchmark* we also observed consistent inferior performance of features extracted from base models, compared to their large counterparts. As we do not expect our differences in training regime and model architecture to impact this conclusion, in this work we present complementary results only for the 7K-large, the best of the *LeBenchmark-1.0* models on SLU, and for all new models (14K) from Table 2. Results with the spectrograms (our basic features), 7K-large, XLSR-53-large features are comparable to [14]. The other results are contributions of this work. Results with 7K-large allow thus to distinguish improvements due to architecture modifications (see paragraph *Downstream models and hyperparameters*) from improvements coming from new SSL models.

Results analysis and discussion. All results on the SLU task are depicted in table 4. We report Concept Error Rate (CER, the lower the better) on development and test data (respectively columns **Dev** and **Test** in the table), as well as the raw error rate (column **RawER**) on the development data, which is the error rate computed on the raw output of the system. The raw output of the system includes words, concept boundaries and concepts, please refer to the appendix of https://aclanthology.org/2020.wmt-1.71/[14] for details. CER is computed on concepts only. For each experiment, we report the best results with respect to the hidden layer size on the development data, and its corresponding multi-task learning setting (*mt*). We also specify *t2* in the table if the best results were obtained with a temperature of 2 at the output softmax.

Table 4: End2End SLU results in concept error rate (CER %, lower is better ↓) on the MEDIA corpus. "h256" and "h300" refer to a hidden size of 256 and 300 and neurons respectively, while "mt" is a multi-task ASR-SLU training. "t2" means that a Softmax temperature of value 2 was applied.

Corpus: MEDIA, Metric: Concept Error Rate (CER %) ↓					
Representation	**Model**	**Params (SSL/SLU)**	**RawER**	**Dev**	**Test**
Task-agnostic models					
spectrogram	h300	-/13.18	59.36	64.96	58.12
spectrogram	h300 mt	-/13.18	**30.44**	30.24	**30.39**
7K-*large*	h256 [14]	330/12.18	-	19.68	18.77
7K-*large*	h300	330/15.45	17.26	18.57	16.99
7K-*large*	h300 mt	330/15.45	**15.36**	**16.62**	**15.47**
14K-*light*	h256	26/12.18	22.71	22.62	20.92
14K-*light*	h256 mt	26/12.18	**19.29**	**19.41**	**18.67**
14K-*large*	h300	330/15.45	18.26	18.63	17.35
14K-*large*	h300 mt	330/15.45	**15.91**	**16.62**	**14.43**
14K-*xlarge*	h256	965/12.70	21.44	17.52	16.24
14K-*xlarge*	h256 mt	965/12.70	**14.73**	**15.66**	**14.43**
XLSR-53-*large*	h256 [14]	330/12.18	-	**18.45**	18.78
XLSR-53-*large*	h256	330/12.18	18.38	18.99	18.68
XLSR-53-*large*	h256 mt	330/12.18	**17.74**	18.84	**17.16**
XLS-R-*xlarge*	h300	965/16.07	18.02	20.75	30.08
XLS-R-*xlarge*	h300 mt	965/16.07	18.53	**18.57**	**29.07**
Task-specific models (ASR fine-tuning)					
7K-*large*	h256 [14]	330/12.18	-	14.58	13.78
7K-*large*	h300 t2	330/15.45	10.82	**13.53**	12.80
7K-*large*	h300 mt t2	330/15.45	10.83	13.95	**12.71**
14K-*light*	h300	26/15.45	15.07	17.03	20.02
14K-*light*	h300 mt	26/15.45	**13.66**	**15.48**	**18.22**
14K-*large*	h256 t2	330/12.18	**10.43**	**12.96**	13.78
14K-*large*	h256 mt t2	330/12.18	13.76	14.43	**12.85**
14K-*xlarge*	h256 t2	965/12.70	**11.19**	**13.74**	**13.88**
14K-*xlarge*	h256 mt t2	965/12.70	11.35	14.58	14.53
XLSR-53-*large*	h300	330/15.45	18.45	17.42	**15.01**
XLSR-53-*large*	h300 mt	330/15.45	**13.09**	**15.22**	16.60
XLS-R-*xlarge*	h300 t2	965/16.07	14.87	17.22	24.15
XLS-R-*xlarge*	h300 mt t2	965/16.07	**13.39**	**15.42**	**23.30**

Our best result on validation data with spectrogram features is 30.44, which is only slightly worse than the 29.07 obtained in [14], with the advantage that in this work, the model is trained end-to-end in one step with the multi-task setting. Additionally, the increased generalization power of the model allows us to reach an error rate of 30.39 on test data, which is slightly better than the 31.10 reported in [14].

Using input features from *LeBenchmark* models (7K-large; 14K-light, large, and xlarge) improvements on SLU results are impressive, with the best CER respectively on validation and test data of 15.66 and 14.43 obtained with the French wav2vec2 14K-xlarge model's features. It is interesting to see, yet expected, that the more data is used to train the SSL model, and the bigger the SSL model in terms of parameters, the better the SLU results are. This trend is not completely respected with task-specific fine-tuned models, in particular with the 14K-large model. Most intuitively, this is because of the small amount of data available for the downstream task, which does not allow for an optimal fine-tuning of so many parameters. The fact that SSL models are tuned for ASR output may also play a role since SLU output already contains tokens instantiating concepts and this may lead the model toward more accurate token predictions which do not always correspond to tokens triggering concepts. This last fact is supported by raw error rates (**RawER** column), accounting for tokens, concept boundaries, and concepts, where not always the best result corresponds to the best CER on validation or test data. On an absolute scale nevertheless, results obtained with fine-tuned French wav2vec2 models are the best. The concept error rate of 12.71 on test data, obtained with the *7K-large* model, is the new state-of-the-art on this task with an end-to-end model. When using fine-tuned model features, the best results on the test data do not always correspond to the best results on validation data, underlying that probably a more accurate hyper-parameter tuning is needed. An interesting outcome of SSL fine-tuned models is that the best results are almost always obtained with an increased softmax temperature (*t2*). In particular, we observed erratic

training behaviors with the default temperature, leading often to gradient explosions. Using an increased temperature not only allows us to obtain good results but also stabilizes model training.

For comparison, we also experimented with some multi-lingual models from the literature, namely XLSR-53-large and XLS-R-xlarge [19]. While the XLSR-53 large model, both without and with fine-tuning, provides interesting results considering that it is not a model specialized for French, the extra-large model provides clearly inferior performances on the test data compared to French models. For the XLS-R extra-large model we hypothesize that the larger size of the model together with its multilingualism does not allow a good generalization on a specific French task like SLU.

5.3. Automatic Speech Translation

Automatic speech-to-text translation (AST) consists of translating a speech utterance in a source language to a text in a target language. In this work, we are interested in translating directly from French speech into text in another language, without the use of transcriptions. We investigate two downstream applications for *LeBenchmark* models: *hybrid* models and *end-to-end* fine-tuning. For the former, the pre-trained model is leveraged as a feature extractor i.e. frozen. For the latter, a decoder is appended to the pre-trained model, and the whole architecture is fine-tuned on the target dataset. Training an end-to-end AST model from a pre-trained speech encoder was first proposed in [101].

Downstream datasets. For both AST downstream strategies, we use the multilingual TEDx dataset [102]. It covers translation directions from French to three target languages: English (en), Spanish (es), and Portuguese (pt), with following training sizes: 50 h (en), 38 h (es), and 25 h (pt). For end-to-end fine-tuning, we also present results for the CoVoST dataset V2 [103] containing 302 hours of French speech from CommonVoice version 4.0 translated to English.

Hybrid downstream models and hyperparameters. In this set of experiments, we focus on leveraging the pre-trained models as feature extractors, using their output speech representation as input for an end-to-end AST model which is trained from randomly initialized parameters. Inspired by [18, 32], this AST model is an encoder-decoder architecture which takes SSL features as input, passing them through a block of Linear-ReLU followed by 2 convolutional layers with strides of $[2, 2]$ and kernel sizes of $[5, 5]$. These 1D-convolutional layers reduce the sequence length by 4 which is then sent to a Transformer [98] model having 6 layers of encoder, 3 layers of decoder, and hidden dimension $D = 256$. This is inspired by the `s2t_transformer_xs` recipe from the fairseq s2t toolkit [104]. For each language pair, we train in total 13 end-to-end models which take as input features extracted from different SSL pre-trained models shown in Table 5. We normalize the punctuation of the text before building a $1K$ unigram vocabulary using `Sentencepiece` [105] without pre-tokenization. For GPU efficiency, utterances having more than $3,000$ frames are filtered out. Each of these AST models is trained for 500 epochs. For all our experiments, we exploit the Adam optimizer [106] whose initial learning rate is set to $2e-3$. This learning rate is linearly increased for the first $10K$ warm-up steps then decreased proportionally to the inverse square root of the step counter. The last 10 checkpoints are averaged and used for decoding with a beam size of 5. Table 5 reports the detokenized case-sensitive BLEU computed using `sacreBLEU` [107].

End-to-end downstream models and hyperparameters. End-to-end AST models are trained on SpeechBrain [45] using the HuggingFace Transformers [108] wav2vec 2.0 interface with spectogram augmentation enabled (same setting than Section 5.1). The encoder stack is made of a wav2vec 2.0 model, followed by a linear projection of output dimension 512. The decoder stack is an 8-heads, 6-layers Transformer with feed forward projections of 2,048 neurons and an embedding size of 512. The weights for the wav2vec 2.0 model are initialized from one of the models in Table 2, and the model is trained with NLL loss. As for end-to-end ASR models (Section 5.1), two different instances of the Adam optimizer manage the weight updates: one dedicated to the wav2vec 2.0 module, the other one to the following layers. The learning rates are respectively $1e-5$ and $1e-4$. The models are trained on a single A100 80GB Nvidia GPU, for 50 (CoVoST), or 100 epochs (mTEDx). In all cases, sentences longer than 35 s are removed for GPU efficiency. For models trained on the mTEDx dataset, we found that it was beneficial for performance to remove layer-dropout and dropout within the wav2vec 2.0 stack during training. We hypothesize that this is due to the limited amount of data available for fine-tuning, as large architectures seemed to benefit the most from this modification. The total number of trainable parameters depends on the wav2vec 2.0 model used: it varies between 121.3M (base),

Table 5: AST BLEU results (higher is better) of the feature extraction experiments (Hybrid with frozen SSL encoders) on the mTEDx dataset. The best results are in **bold**. Gray numbers denote the standard deviation computed using bootstrap re-sampling [110].

	fr-en		fr-es		fr-pt	
Representation	**Dev**	**Test**	**Dev**	**Test**	**Dev**	**Test**
1K-*base* [14]	9.18±0.36	8.98±0.36	5.09±0.27	5.64±0.30	0.39±0.05	0.49±0.08
1K-*large* [14]	15.31±0.46	14.46±0.46	13.74±0.43	14.77±0.46	8.29±0.34	9.37±0.38
2.7K-*base* [14]	15.09±0.49	14.69±0.48	13.27±0.43	14.04±0.43	4.72±0.27	5.51±0.28
3K-*base* [14]	15.05±0.49	14.80±0.47	13.19±0.44	14.27±0.44	4.44±0.29	4.72±0.25
3K-*large* [14]	17.94±0.51	18.00±0.51	16.40±0.49	18.12±0.48	8.64±0.34	9.55±0.36
7K-*base* [14]	15.13±0.45	14.50±0.45	12.78±0.40	13.61±0.44	2.65±0.20	2.66±0.23
7K-*large* [14]	**19.23**±0.54	**19.04**±0.53	**17.59**±0.49	**18.24**±0.49	**9.68**±0.37	**10.98**±0.41
14K-*light*	10.31±0.38	10.92±0.43	9.83±0.33	10.52±0.42	4.96±0.31	5.79±0.33
14K-*large*	**18.93**±0.40	**18.97**±0.47	**17.22**±0.41	**18.12**±0.42	9.03±0.35	10.11±0.39
14K-*xlarge*	18.14±0.42	18.35±0.48	15.90±0.39	17.19±0.43	5.46±0.29	6.59±0.35
XLSR-53-*large* [14]	7.81±0.33	6.75±0.29	0.49±0.13	0.52±0.08	0.43±0.07	0.36±0.05
XLSR-R-*xlarge*	13.80±0.37	13.88±0.38	11.45±0.37	12.56±0.40	1.59±0.29	1.77±0.31

342.5M (large), and 989.7M (xlarge). Pre-tokenization strategy is the same as the Hybrid AST setup. Lastly, we do not use pre-trained weights to initialize the AST decoder, and we do not explore partially freezing the wav2vec 2.0 Transformer encoder layers (i.e. training only a subset of the layers) as in Boito et al. [109].

LeBenchmark 1.0. During the first version of this benchmark, only the hybrid setup (SSL as feature extractor) was explored. In contrast to that, we now also include end-to-end AST fine-tuning, which has since become popular in the speech community. In order to provide the reader with the full context, and allow easy comparison between hybrid and end-to-end approaches, we present results for all models, previously evaluated and new.

Hybrid results analysis and discussion. Results are presented in Table 5 and analyzed with the following aspects:

- **Monolingual versus multilingual**. Comparing SSL models of the same size (large models), training on monolingual data (*LeBenchmark* models) seems to be beneficial in comparison with training on multilingual data (XLSR-53 and XLS-R models). From $3K$ hours of French data, all *LeBenchmark* large models outperform both XLSR-53 and XLS-R models.
- **Pre-training data**. Concerning the amount of monolingual data, we observe that with the same model size (base or large), SSL models tend to improve when the amount of pre-training data increases, except for the 14K-large model whose performance is on par with that of the 7K-large model. We suspect that adding too much read speech data ($6,600h$) might lead to stagnation in terms of BLEU scores when jumping from $7K$ to $14K$ hours of training data on the mTEDx domain.
- **Model size**. Table 5 illustrates that with the same amount of pre-training data, larger models tend to be better than smaller ones for producing speech features. Surprisingly, xlarge models underperform large models, observed with both *LeBenchmark* 14K and XLS-R. Lastly, we observe that the 14K-light model significantly underperforms its base and large counterparts, hinting that it insufficiently represents speech signals due to its limited capacity.

End-to-End results analysis and discussions. Table 6 and 7 present BLEU scores for the mTEDx and CoVoST datasets respectively.

- **Monolingual versus multilingual.** Overall, we notice that the importance of the backbone model (monolingual or multilingual) is less important in end-to-end mode compared to hybrid mode (the XLS-R model is not too far

Table 6: AST end-to-end BLEU results (higher is better) for the mTEDx dataset. The best results are in **bold**. Gray numbers denote the standard deviation computed using bootstrap re-sampling [110].

	fr-en		fr-es		fr-pt	
Representation	**Dev**	**Test**	**Dev**	**Test**	**Dev**	**Test**
1K-*base*	15.2±0.48	14.0±0.54	13.0±0.42	13.2±0.40	8.2±0.33	8.6±0.34
1K-*large*	16.7±0.49	16.6±0.46	15.3±0.46	16.1±0.45	9.4±0.33	10.7±0.38
2.7K-*base*	18.9±0.52	18.7±0.52	17.9±0.50	17.8±0.49	11.7±0.39	12.3±0.40
3K-*base*	17.9±0.48	17.9±0.51	16.8±0.49	17.1±0.46	11.3±0.42	12.4±0.42
3K-*large*	17.6±0.51	16.9±0.47	15.1±0.45	15.6±0.46	8.6±0.34	9.7±0.37
7K-*base*	18.8±0.51	18.2±0.50	18.4±0.52	18.2±0.68	12.6±0.41	13.4±0.44
7K-*large*	20.1±0.52	19.0±0.57	17.4±0.52	18.8±0.49	10.7±0.37	12.0±0.41
14K-*light*	6.5±0.27	5.9±0.28	5.7±0.27	5.7±0.26	3.0±0.21	2.9±0.17
14K-*large*	23.6±0.59	23.1±0.55	23.3±0.58	24.2±0.62	18.7±0.54	21.8±0.58
14K-*xlarge*	**25.1**±0.59	**24.4**±0.60	**23.7**±0.56	**25.5**±0.59	**20.7**±0.58	**23.7**±0.62
XLSR-53-*large*	15.6±0.49	12.5±0.47	15.6±0.45	15.8±0.44	8.4±0.31	9.1±0.36
XLS-R-*xlarge*	23.4±0.58	22.7±0.55	**23.3**±0.61	**25.0**±0.60	19.3±0.54	21.3±0.56

behind the best-performing monolingual *LeBenchmark* model). Nevertheless, between the two best-performing models for both CoVoST and mTEDx, *LeBenchmark* 14K outperforms XLS-R in all settings. It should however be highlighted that XLS-R is a model covering 128 languages, with the potential to reach similarly good results for at least a small portion of its covered languages.

- **Pre-training data.** Focusing on monolingual models, and looking at results for large architectures only, we do not see any hints of saturation: *LeBenchmark* models pre-trained with more speech data tend to provide better initializations for the AST task (Tables 6 and 7). Looking at base architectures, the exception for this seems to be the 2.7K-base model, which performs on par with 3K and 7K base and large models. This model differs from 3K by not including spontaneous speech. We hypothesize this pre-training setting could provide a better initialization for mTEDx and CoVoST datasets, which are made of prepared and read speech respectively.

- **Model size.** Looking at mTEDx (Table 6) and CoVoST results (Table 7), we reach the overall conclusion that using larger pre-trained models as initialization for end-to-end AST tend to result in better AST performance. The only exceptions to these findings seem to be the 3K-large model applied to the mTEDx dataset, which results in AST models that are consistently worse than using 3K-base. Moreover, we also observe that for the mTEDx fr-pt setting, the 7K-large model is inferior to its base version. We believe these findings could hint that these two models (3K and 7K-large) are providing a suboptimal initialization for the AST task, which is more apparent in the mTEDx experiments, as in this setting we have considerably less trainable fine-tuning examples compared to CoVoST, and in particular for the languages where we observe the largest gaps (es and pt). Finally, it seems to always be beneficial for end-to-end AST fine-tuning to have a xlarge wav2vec 2.0 model, compared to large, but this marginal difference in performance adds a considerable overhead in the number of trainable parameters (647.1M extra trainable parameters). Lastly, we observe that the 14K-light model is a poor initialization choice for end-to-end AST. We believe this highlights how the capacity of the model is related to the encoding of high-abstraction level speech features: smaller Transformer stacks results in poor speech features (Table 5) and encoders (Tables 6 and 7). Indeed, Pasad et al. [111, 112] argue that the wa2vec 2.0 pretext task forces a drop in abstraction-level at the last layers. Due to this, the middle of the Transformer stack is where most of high-level (phonemic and word-level) information is encoded.

5.4. Automatic Emotion Recognition

Recent psychological studies suggest that emotion is needed and used in every aspect of our lives, from filtering the sensory information, our perception of an event, reasoning, and thus the decisions we make [113, 114]. The auto-

Table 7: ST end-to-end BLEU results (higher is better) for the CoVoST dataset. The best results are in **bold**. Gray numbers denote the standard deviation computed using bootstrap re-sampling [110].

Representation	Dev	Test
1K-*base*	28.5±0.21	27.9±0.20
1K-*large*	30.1±0.21	30.0±0.21
2.7K-*base*	30.8±0.21	30.2±0.21
3K-*base*	29.8±0.21	29.4±0.21
3K-*large*	29.4±0.21	29.0±0.21
7K-*base*	30.1±0.21	29.7±0.20
7K-*large*	32.7±0.22	32.5±0.21
14K-*light*	20.5±0.18	20.0±0.18
14K-*large*	32.1±0.21	31.7±0.21
14K-*xlarge*	**33.9**±0.22	**33.7**±0.21
XLSR-53-*large*	30.4±0.21	29.6±0.20
XLS-R-*xlarge*	32.9±0.21	32.5±0.21

matic recognition of human emotions from audio recordings is therefore a technology that can influence many areas such as education, healthcare, and entertainment. Although much progress has been made in emotion recognition in recent years, there are still challenges including different emotional expressions by different speakers, or different microphones, making AER not quite ready for everyday use [115]. SSL models being trained on large amounts of data, have been shown to be exceptionally good in addressing generalisation issues [14]. *LeBenchmark 2.0* further evaluates such methods for French speech, with new trained SSL models.

Downstream datasets. Following *LeBenchmark* , we used the RECOLA [116] and the AlloSat [117], which contain continuous conversations, and the THERADIA corpus, which contains utterance-based conversations. Both the RECOLA and Allosat datasets contain spontaneous French speech. The RECOLA recordings are emotionally induced conversations recorded in a laboratory environment, whereas the AlloSat recordings are telephonic conversations. The annotations for both datasets are time-continuous dimensional. For Allosat, a frustration-to-satisfaction dimension is used with a sampling rate of 4 Hz. And for RECOLA, the emotion dimensions are based on arousal (from passive to active), and valence (from negative to positive), sampled at 25 Hz rate. Moreover, the AlloSat dataset contains a total of 29,704 utterances (21 h), divided into 20,785 utterances (15 h) as training set, 4272 utterances (3 h) for development and 4643 utterances (3 h) as test partition. On the other hand, the RECOLA dataset is much smaller, with 9 files of 5 minutes each for the training, development, and test sets. Since the continuous conversations used in RECOLA and AlloSat datasets differ from utterance-based training of the used SSL models, we also used the THERADIA corpus, which contains segments divided by utterances divided by pauses for breath.

The THERADIA corpus contains 61 senior participants, nine of whom had Mild Cognitive Impairments (MCIs). The participants performed digital cognitively stimulating exercises while interacting with a virtual assistant in a natural way. The THERADIA corpus contains emotion labels that are annotated based on the perceived intensity of the label on a scale from zero (not existent), to 100. We report results on the prediction of the ten most common core set labels in the THERADIA corpus: relaxed, interested, frustrated, confident, satisfied, happy, annoyed, surprised, desperate, and anxious. The THERADIA corpus contains 2,735 utterances (6 h) in total which are divided into 1,110 utterances as training partition, 851 utterances for validation, and 774 utterances for testing.

Downstream models and hyperparameters. The experiments are conducted using a one-to-one sequence model with either SSL representations or Mel Filter Bank features. The experiments for the RECOLA and AlloSat datasets consist of time-linear sequence-to-sequence prediction of continuous dimensions of emotion. A GRU with one layer and 32 hidden units was trained with CCC as the loss function, similar to [14]. Since both the AlloSat and RECOLA dataset contains continuous long audio files, we were not able to fine-tune them. On the other hand, for the experiments

Table 8: The AER results are expressed in terms of concordance correlation coefficient (CCC, higher is better). The left table describes the continuous prediction of frustration-satisfaction, arousal and valence dimensions for the AlloSat and RECOLA corpora (for concatenated outputs), with frozen representations (feature extraction setting). The results on the right table describe the average (gray numbers describe standard deviation) emotion prediction across the core set emotion labels of the THERADIA corpus in the frozen (feature extraction) and fine-tuned (end-to-end).

Corpora: AlloSat & RECOLA
Metric: Concordance Correlation Coefficient (CCC) ↑

Representation	**Satisfaction**	**Arousal**	**Valence**
Mel Filter Bank	.413	.313	.258
1K-*base*	.487	.427	.055
1K-*large*	.021	.018	.001
2.7K-*base*	.596	.629	.455
3K-*base*	.602	.358	.007
3K-*large*	.040	.097	.000
7K-*base*	.470	.335	.116
7K-*large*	.050	.009	.037
14K-*light*	.518	.614	.348
14K-*large*	.462	**.664**	**.466**
14K-*xlarge*	**.657**	.649	.437
XLSR-53-*large*	.264	.149	.146
XLS-R-x*large*	.415	.311	.229

Corpus: THERADIA
Metric: Concordance Correlation Coefficient (CCC) ↑

Representation	**Frozen**	**Fine-tuned**
Mel Filter Bank	.075±.120	-
1K-*base*	.151±.103	.246±.161
1K-*large*	.001±.002	**.319±.172**
2.7K-*base*	.083±.094	.013±.015
3K-*base*	.061±.069	.019±.016
3K-*large*	.002±.004	.224±.151
7K-*base*	.106±.082	.000±.000
7K-*large*	.000±.002	.230±.143
14K-*light*	**.241±.133**	.283±.129
14K-*large*	.190±.127	.229±.151
14K-*xlarge*	.237±.131	.226±.145
XLSR-53-*large*	.004±.006	.232±.144
XLS-R-*xlarge*	.072±.059	.205±.132

for the THERADIA corpus, we used one linear layer trained with mean squared error as the loss function. We also tried using CCC as the loss function and GRU as the emotion prediction model for THERADIA, but did not find any significant improvement in the results. Also, since THERADIA corpus contains separated utterances, we also investigated the effect of fine-tuning the SSL models for emotion recognition.

Furthermore, for all the experiments, the training was done with the Adam optimizer with a learning rate of 0.001 for frozen models, and 0.0001 for fine-tuning, and trained for 200 epochs with early stopping. It should be noted that that the sampling rate of the dimensional annotations differs from the Mel features, which are sampled at a rate of 100 Hz, and the wav2vec representations, which are sampled at a rate of 50 Hz. Thus, during training for RECOLA and AlloSat datasets, the targets are resampled to match the sampling rate of the representations with linear interpolation to keep the graph active for backpropagation, while during testing, the outputs of the model are mapped to the sampling rate of the targets to keep the targets untouched for testing. On the other hand, for the THERADIA corpus, the outputs of the model are averaged over the sequence, because each emotion label is defined per sequence and not continuously over the sequence.

LeBenchmark 1.0. In the previous version of the AER benchmark, we have investigated the use of SSL models only for acoustic feature extraction using the AlloSat and RECOLA datasets. It should also be noted that the difference between the results of the previous work and this paper is due to the change of the architecture and framework used for training the SSL models. This paper also presents additional results for the THERADIA corpus for both the use of SSL models for feature extraction (referred to as "frozen") and end-to-end fine-tuning (referred to as "fine-tuned").

Results analysis and discussion. The results are shown in Table 8. Overall, the results across different 1K, 3K, and 7K models seem to vary a lot, while the results for 14K models seem to be consistently good for different emotion recognition tasks. For example, it can be observed that the 1K-large, 3K-large and 7K-large models perform poorly when frozen. This result is not limited to the AER tasks in this section, but also extends to Section 5.5 (see WER in Table 9). This may indicate that these models were trained in such a way that does not allow them to generalise to the AER task in an off-the-shelf setting. Further investigation is needed to determine the reason for this.

Regarding 14K models, results tend to vary less across different experiments, and are consistently better than traditional features. This suggests that the pre-training stage is not suffering of saturation, and that adding more data at that stage results in a more robust off-the-shelf model, which can achieve consistently better results for AER than

traditional features. We can also observe that all the 14K models trained on French speech, when frozen, perform better than the multilingual SSL models (XLSR-53-large and XLSR-R-xlarge).

Focusing on AlloSat and RECOLA results, we notice that for the prediction of emotion dimensions, the 14K-large and 14K-xlarge models achieved the best scores. While these two models achieved similar results for the prediction of arousal and valence dimensions of emotion on the RECOLA corpus, their results are not similar for the frustration-satisfaction dimension of the AlloSat corpus. This may be because RECOLA contains clean speech, and thus the smaller parameter size of 14K-large, compared to 14K-xlarge, may be sufficient to extract useful features for continuous emotion recognition on this dataset. However, the better results of the 14K-xlarge model compared to 14K-large for the AlloSat corpus, might suggest that for the real-life telephonic conversations, the use of a larger model is beneficial.

Similarly, the prediction of emotion labels for the THERADIA corpus with the frozen SSL models shows that the 14K models perform best. However, when fine-tuning the wav2vec 2.0 models, we observe that the base models (except for 1K-base) perform worse than their frozen counter parts. On the other hand, for the large models (except for 14K-xlarge), the performance is improved when fine-tuning the SSL models. The better performance of the fine-tuned wav2vec 2.0 models is consistent with the literature. Indeed, fine-tuning specializes the representations, which results in better AER performance for a particular data distribution. However, we may lose the ability to generalize to other data distributions [118]. Moreover, the fact that 1K-large model performs best when fine-tuned, may suggest that pre-training the wav2vec 2.0 models with more data or more parameters does not necessarily have a one-to-one correlation with better prediction performance of emotion labels. This, in turn, may mean that predicting THERADIA's emotion labels without regard to generalisation does not necessarily require complex architectures or large amounts of data to train and then fine-tune a base model.

5.5. Syntactic Analysis

The syntactic analysis task (also known as parsing) is a staple task in NLP, and historically one of the first. Syntactic parsing consists of assigning a syntactic structure to a given sentence. Thus, it is a structured prediction task. Corpora annotated with syntactic trees are key to data-driven linguistic studies, syntactic trees may also provide useful features for downstream tasks. We focus on evaluating the pre-trained SSL models trained on the task of joint ASR and syntactic analysis. The traditional technique to obtain the syntactic structure from speech would be to use a pipeline approach. Using an ASR model to get the transcription and then using a pre-trained model such as BERT to predict the syntactic structure. However, this method removes important features contained in the signal for the syntactic predictions, such as the prosody. Moreover, it has been shown that end-to-end speech parsing models perform better, despite having much fewer parameters than pipeline-based parsers [119].

Downstream datasets. We use the CEFC-ORFEO [120] dataset. This corpus is a collection of multiple subcorpora[121, 122, 123, 124, 125, 126, 127, 128] annotated in syntactic dependencies in the *conll* format. This corpus contains a wide variety of speech situations, such as store owner/customer interactions in a cheese shop, or informal conversations between friends. We removed the TCOF sub-corpus from the dataset, as it was included in the pretraining data for the *LeBenchmark 2.0* models. The Orféo treebank contains both human-annotated syntactic trees and automatically annotated syntactic trees [129], henceforth referred to as silver data. Partitions are built such that the dev and test set only contain gold trees, i.e. all silver trees are in the training set.

Downstream models and hyperparameters. The downstream model is wav2tree [119]. This model is composed of an encoder module, an ASR module, and a syntax module. It performs joint syntactic analysis and automatic speech recognition in a multitasking manner. The encoder is composed of three fully connected layers of size 1024 in the fine-tuning setting. In the frozen setting, the encoder is a two-layer bi-LSTM with a hidden size 1024 and a dropout of 0.4 between the two layers. In wav2tree, the speech recognition module has two purposes, the first one is the normal speech recognition task, acquiring the transcriptions. The second one is the segmentation of the representation learned by wa2vec 2.0. The CTC scheme labels each representation with either a letter, a blank, or a whitespace. The wav2tree model uses the whitespace information to segment the representation. The syntax module is composed of two elements. The first one uses the segmentation from the ASR module to create word-level representations from the encoder representation via a 2-layer LSTM with a hidden size of 500. These word-level representations are then used by a two-layer bi-LSTM with a hidden size of 800 and then are classified into three tasks in parallel with a simple

Table 9: End-to-end results for the SA task in terms of part of speech tagging f1-score, unlabeled attachment score (UAS), and labeled attachment score (LAS) metrics (higher is better). Results are correlated to the speech recognition results expressed in CER and WER. Syntactic analysis training started only for model with WER below the threshold of 50.

Representation	Frozen	WER %	CER %	POS	UAS	LAS
1K-*base*	No	51.26	31.32	—	—	—
	Yes	80.35	48.67	—	—	—
1K-*large*	No	45.99	28.60	65.18	59.98	53.48
	Yes	99.82	93.80	—	—	—
2.7K-*base*	No	100	88.40	—	—	—
	Yes	93.97	57.95	—	—	—
3K-*base*	No	100	88.40	—	—	—
	Yes	81.54	48.16	—	—	—
3K-*large*	No	44.81	27.55	65.81	61.54	55.11
	Yes	99.94	88.33	—	—	—
7K-*base*	No	100	88.40	—	—	—
	Yes	70.70	39.97	—	—	—
7K-*large*	**No**	**42.39**	**26.59**	**67.15**	**63.34**	**56.94**
	Yes	99.81	78.59	—	—	—
14K-*light*	No	71.08	40.51	—	—	—
	Yes	76.56	46.28	—	—	—
14K-*large*	No	43.04	27.12	66.71	62.72	56.45
	Yes	52.16	30.51	—	—	—
14K-*xlarge*	No	44.82	28.00	65.09	61.17	54.61
	Yes	45.43	26.51	66.60	60.63	53.94
XLSR-53-*large*	No	44.57	26.77	66.40	61.20	54.69
	Yes	97.51	77.81	—	—	—
XLS-R-*xlarge*	No	54.88	32.50	—	—	—
	Yes	99.16	76.08	—	—	—

linear layer. The first one predicts the part of speech (POS) of the word, the second one predicts the head of the current word (UAS) and the last one predicts the syntactic function (e.g. subject, dependent, etc), i.e. the relationship between the head and the dependent. Each model is trained with a batch size of 8, except for the fine-tuning of the xlarge which is trained with a batch size of 2 and a gradient accumulation of 4, in order to maintain the comparability of results. The ASR module is trained with a CTC loss and the classification tasks are trained with a negative likelihood loss. The optimizer is AdaDelta [130] with a learning rate of 1 and ρ of 0.95. Each model is optimized on the word error rate and once it decreases below a threshold of 50, the training activates the syntax learning and is then optimized on the LAS metrics.

Parsing evaluation metrics. The three metrics used for the parsing task are the f-measure for part of speech tagging labeled POS, the unlabeled attachment score UAS which is the accuracy of the syntactic link between the word, i.e. is the word correctly attached to its head. The last one is the labeled attachment score LAS extending the UAS metrics by also taking into account the nature of the link (root, subject, etc). Since the ASR transcript differ from the gold transcript, an alignment between the two must be done in order to compute these different metrics. We use the same methods as [119]. By using the WER alignment computed during the evaluation of the ASR module, we can realign the two transcript and compute these metrics.

Results analysis and discussion. All results for the syntactic analysis task are depicted in Table 9. The parsing result is heavily correlated to the ASR metrics. This is an expected behavior, since a correct tree cannot be produced if some words are missing. The WER and CER clearly reflect the challenge of the target dataset, as with a similar architecture,

a score of 10 % of WER is obtained on CommonVoice 6.1 [131]. The difficulty of the dataset implies that none of the base models can get good enough results to start learning the parsing task. All the models also need to be fine-tuned on the dataset with the notable exception of the 14K-xlarge suggesting that the pre-trained representation of this model is general enough to fit more out-of-domain data like the CEFC-Orfeo dataset.

We observe that the quantity of data used to pre-train the model is important and seems to follow the classic machine learning paradigm that more data and scale are better. However, the best model for this task is the 7K-large and not the 14k-large or xlarge. Our hypothesis is that this model is trained on a more balanced distribution of type of speech (read, prepared and spontaneous), thus being more suited to learning good representations for spontaneous speech. Another interesting fact is that the 14k-large outperforms the 14k-xlarge. This may be simply because the bigger model needs more data, whereas the smaller one is more easily tunable to the downstream dataset. One of the most surprising results is the one on the 3K-base and 7k-base, where the models perform better without fine-tuning. We also compare to the multilingual XLSR-53 model. The multilingual model has slightly worse performance compared to most of our models, but exhibits similar properties, such as the need to finetune it on the downstream corpus to reach good performance. The smaller multilingual model achieves better performance than the bigger one, XLSR-R-xlarge. This may be a combination of the same phenomenon as the one observed on the 14k-large and 14k-xlarge and the difference of the two dataset used to train these model. XLS-R-xlarge is trained on 128 languages, whereas XLSR-53 on 53. The added languages may negatively impact the performance on spontaneous speech since spontaneous speech corpora are less common (even unannotated), thus unbalancing the distribution between the different type of speech.

5.6. Automatic Speaker Verification

Automatic Speaker Verification (ASV) refers to the task of verifying the identity claimed by a speaker from that person's voice [132]. In ASV, deep neural networks have brought about significant advancements in voice representations, outperforming the previous state-of-the-art *i*-vector framework [133]. One of these DNN approaches seeks to extract a high-level speaker representation, known as *speaker embedding* directly from acoustics excerpts. To achieve this, DNN models are trained through an ASV task, where speech segments are classified into distinct speaker identities. Each layer of the DNN is trained to extract relevant information for discriminating between different speakers, and one of the hidden layers is used as the speaker embedding. One of the main advantages of this approach is that speaker embeddings produced by the DNN can generalize well to speakers beyond those present in the training set. The benefits of speaker embedding, in terms of speaker detection accuracy, have been demonstrated during the last evaluation campaigns: NIST SRE [134, 135, 136] and VoxCeleb 2020 [137, 138, 139].

Recently, much progress has been achieved in ASV through the utilization of SSL models. In [16], the authors proposed a novel approach: instead of exclusively using the representations from the final layer of the SSL model, they employ a weighted average of the representations from all hidden layers of the SSL model. This approach allows for harnessing speaker-related information embedded throughout the entire SSL model. More recently, the authors in [140] investigated methods to fine-tune a pre-trained wav2vec2 model specifically for ASV. They proposed a fine-tuning approach that introduces a CLS token at the beginning of the process for utterance-pair classification.

Downstream datasets. For evaluation, we used the Fabiole dataset [141]. Fabiole is a french speaker verification dataset that has been collected for use to highlight the importance of the "speaker factor" in forensic voice comparison. It contains 7,372 segments from 130 male native French speakers. Due to the absence of an established evaluation protocol, we took the initiative to create one ourselves. We removed all the segments with less than 2 seconds of voice and those that exceed 12 seconds. Then, we randomly selected 300,000 target pairs (i.e. enrollment and test segments that target same speakers) and 300,000 non-target pairs (i.e. enrollment and test segment that non-target same speaker). We trained the systems using ESTER-1 [142], ESTER-2 [143], ETAPE [73] and REPERE [144] training datasets (that correspond to 2.911 speakers and more than 250 hours of data). Voice Activity Detection (VAD) processing was not applied on the training datasets. Additionnaly, we applied data augmentation into the training process by incorporating noise from the MUSAN dataset and reverberation using the RIR dataset [145]. Equal Error Rate (EER) and the Detection Cost Function (DCF) are used as the performance criterion of ASV. EER is the threshold value such that the false acceptance rate and miss rate are equal. Whereas DCF is a weighted sum:

$$C_{det} = C_{Miss} \times P_{Miss|Target} \times P_{Target} + C_{FalseAlarm} \times P_{FalseAlarm|NonTarget} \times P_{NonTarget}, \quad (1)$$

Table 10: Results for the downstream task of ASV. Performance are expressed in terms of Equal Error Rate (EER, lower is better) and Minimum of the Detection Cost Function (minDCF, lower is better).

Corpora: Fabiole
Metric: EER and minDCF ↓

Representation	**EER**	**minDCF^{-10}**	**minDCF^{-100}**
1K-*base*	8.27	0.556	0.722
1K-*large*	6.75	0.508	0.705
3K-*base*	4.82	0.374	0.567
3K-*large*	5.06	0.374	0.521
7K-*base*	4.73	0.364	0.538
7K-*large*	5.23	0.383	0.575
14K-*light*	7.39	0.508	0.711
14K-*large*	3.54	0.297	0.480
14K-*xlarge*	**2.90**	**0.241**	**0.416**
XLSR-53-*large*	6.68	0.492	0.677
XLS-R-*xlarge*	6.67	0.457	0.633

with the prior probabilities P_{Target} and $P_{NonTarget} = 1 - P_{Target}$ of target and impostor speakers, respectively. The relative costs of detection errors in this function are the costs of miss C_{Miss} and false alarm errors $C_{FalseAlarm}$. These parameters were set as follows: $P_{Target} = 0.01$ (or $P_{Target} = 0.001$), $C_{Miss} = 1$ and $C_{FalseAlarm} = 1$.

Downstream models and hyperparameters. We use the ECAPA-TDNN classifier [146]. This classifier, when combined with SSL models, has demonstrated impressive performance [147] in ASV. Our ECAPA-TDNN has the following parameters: the number of SE-Res2Net Blocks is set to 3 with dilation values 2, 3 and 4, the number of filters in the convolutional frame layers is set to 512 and embedding layer size is set to 256. The training lasted for 8 epochs with the *AdamW* optimizer. We trained all the models with Circle Margin Loss and set the margin to 0.35. During the training process, we randomly sampled 3s segments from each utterance to construct a training batch. We remind that the ECAPA-TDNN takes as input a weighted average of the representations from all hidden layers of the SSL model.

Results analysis and discussions. All results for the ASV task are depicted in Table 10. XLSR-53-*large* and XLS-R-*xlarge* were used as a baseline. The findings can be summarized as follows:

- **Monolingual versus Multilingual.** We observed that except for 1K models, systems trained on monolingual models (i.e. *LeBenchmark*) achieved better performance than the multilingual model (XLSR-53-*large* and XLS-R-*xlarge*). The best monolingual model (*LeBenchmark* -14K-*xlarge*) obtained 2.90% of EER while the multilingual model (XLS-R-*xlarge*) obtained 6.67% EER.
- **Pre-training data.** Focusing on monolingual models, we observed a link between performance and the quantity of pre-training data. LeBenchmark models pre-trained with larger speech datasets tend to provide better performance. Indeed, the LeBenchmark model trained on 1,000 hours of data (LeBenchmark-1k-large) obtained 6.75% of EER, while the LeBenchmark model trained on 14.000 hours of data (LeBenchmark-14k-xlarge) obtained 2.90% of EER.
- **Model size.** Always focusing on monolingual models, *large* or *xlarge* models obtain better performance than *basic* models, except for 3K models and 7K models. Figure 1 shows the contribution from each layer of various SSL models. We remind that LeBenchmark base models contain 12 layers, large models contain 24 layers and

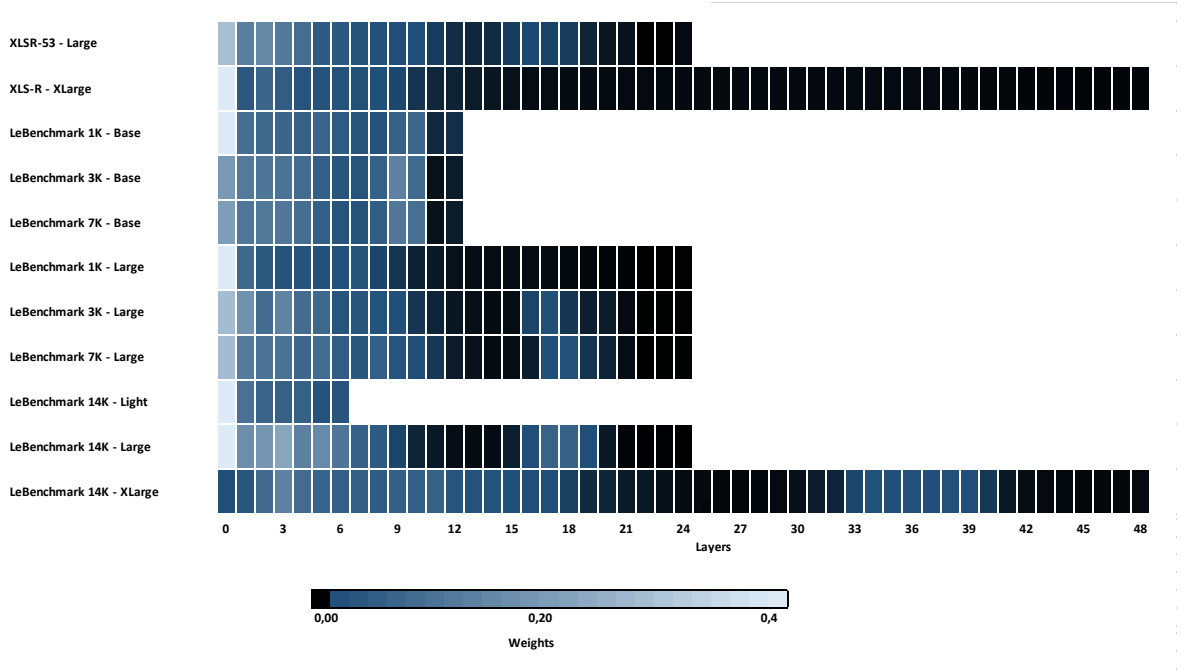

Figure 1: The visualization of the normalized weight values in the proposed architecture. Each weight can be interpreted as the relevance of a given layer for the ASV task. Earlier layers are particularly relevant for ASV.

xlarge models contain 48 layers. In general, we observe that speaker-related information is most pronounced in the first layers (lower layers) of the SSL model. Even if the speaker-related informations is most pronounced in the first layer, we notice that for LeBenchmark base models, all layers contribute to the construction of the representations. In contrast, for LeBenchmark large or xlarge models, the higher layers contribute less compared to the lower layers.

5.7. Inference cost considerations

Once an SSL model has been pre-trained and fine-tuned, it may be deployed in many scenarios with many different hardware environments. Reporting exhaustively the cost of inferring over our SSL models is intractable. A few preliminary works tried to approach this problem by estimating or analyzing the cost of inferring with pre-trained models. Still, it always remains tied to a specific use-case of deployment [148]. This section provides a basic inference cost analysis based on CPU and GPU inference over the four different sizes of model provided by *LeBenchmark 2.0* .

Task definition and measurements. Among the SSL models of *LeBenchmark 2.0* , the only factor affecting the inference cost is the size. Hence, the analysis is conducted by reporting different metrics obtained over the processing of one hundred sequences of length 15s by the following models: 14K-*light*, 1K-*base*, 14K-*large* and 14K-*xlarge*. The real-time factor (RTF) was obtained by taking the average time necessary to infer over 100 sentences divided by the length of one sequence. Hence, lower is better. The peak VRAM is also included. RTFs are reported both for CPU and GPU inference. Measurements are conducted on an isolated compute node with a single Nvidia RTX 3090 and 8 cores of an Intel Xeon 5218. Despite a higher number of cores being available, we fixed it to 8 to remain close to a relatively low-resource hardware setup. No optimization of the pre-trained model is performed. The checkpoint are simply loaded in SpeechBrain and used for inference.

Results analysis and discussion. Table 11 reports the obtained RTFs and the peak VRAM for every model. As one may expect, the bigger the model, the higher the RTF, and the slower the inference. This must be put into perspective

Table 11: Real-time factor (RTF) and peak VRAM observed for the inference of 100 sentences of length 15s on a single RTX 3090 and 8 cores of an Intel Xeon CPU. Lower is better.

Model	**CPU RTF ↓**	**GPU RTF ↓**	**Peak VRAM ↓**
14K-*light*	0.028	0.001	1.2 GB
1K-*base*	0.038	0.001	2.2 GB
14K-*large*	0.096	0.002	5.1 GB
14K-*xlarge*	0.229	0.003	11.5 GB

with the obtained downstream performance of the previous sections. Indeed, the very small accuracy gains obtained with the 14K-*xlarge* may not be sufficient to overcome the massive increase in inference cost, especially on CPU. For instance, 14K-*large* is more than twice as fast as 14K-*xlarge* and its downstream performances remain remarkable. As expected, GPU inference does not suffer from this distinction from the model size as the RTF remains extremely small in all scenarios. The peak VRAM, however, increases drastically as a GPU with under 8GB of VRAM would not be able to handle the 14K-*xlarge* natively.

5.8. Summary of results

In the preceding sections, we presented multiple configurations along with their respective outcomes for the six downstream tasks introduced in *LeBenchmark 2.0*. Unlike SUPERB, which primarily focused on evaluating SSL models with shallow probes, our approach aimed to assess our SSL models using diverse strategies applicable to downstream tasks. In this section, we summarize the key discoveries and insights derived from our exploration.

SSL as feature extractor (SLU, AST, AER, ASV). In this setting, downstream tasks exploited the representations produced by SSL models without adapting them during downstream training. However, for all mentioned tasks except ASV, this setup does not align with the SUPERB settings: representations are extracted from the penultimate layer instead of being weighted across the output generated by different layers. Overall, we find that in this setting, models trained with more hours of speech tend to yield the best results. For SLU, AST, AER, and ASV, the best SSL model for feature extraction was a 14K model, but the preferred feature dimension varied: SLU, AER and ASV achieved their best results with 14K-xlarge, while AST achieved them using 14K-large.

SSL as speech encoder (ASR, AST, AER, SA). In this setting, the SSL models were updated (fine-tuned) during downstream training, serving as speech encoders. The trend of results in this setting is not consistent. For ASR (CommonVoice and ETAPE datasets), we observe that 14K models are unable to outperform results obtained by models trained with less data: for both datasets, 3K and 7K-large models achieve the best results. Similarly, SA reached their best scores with the 7K-large model. For the AST task (mTEDx and CoVoST), the best results are achieved by using the 14K-xlarge model. Surprisingly, for the AER task (THERADIA), the best-performing model was the 1K-large. We believe that this inconsistency in model ranking illustrates that probing rankings should not be used for selecting a model for end-to-end fine-tuning. Indeed, in Zaiem et al. [43], the authors observed that model ranking quickly changes when downstream settings vary. We also believe that aspects such as the mismatch of pre-training and downstream datasets, the target task, the chosen architecture, and downstream optimization may all play key roles in the final performance of models.

SSL as feature extractor vs SSL as speech encoder (AST, AER). In cases where both feature extraction and end-to-end fine-tuning approaches are employed for downstream tasks, we observe that feature extraction performance tends to lag behind end-to-end fine-tuning, as seen in AST and AER. This outcome is anticipated, given that the latter benefits from adapting the entire SSL model representation to the target data. However, this advantage comes at a higher training cost, also requiring more fine-tuning examples. Moreover, it appears that our multilingual baselines (XLSR-53-large and XLS-R-xlarge) particularly benefit from fine-tuning, as it may enable them to specialize.

Table 12: Summary of the best SSL models for each downstream task.

Task	Best Feature Extractor	Best Speech Encoder
ASR	-	{3K-7K}-*large*
SLU	14K-*xlarge*	-
AST	7K-*large*	14K-*xlarge*
AER	14K-{*light*, *large*, *xlarge*}	1K-*large*
SA	-	7K-*large*
ASV	14K-*xlarge*	-

Overall simplified SSL model ranking (task-wise). Table 12 summarizes the best models for each evaluated task and setup based on our experiments. Overall, the 7K and newly introduced 14K models outperform others across the benchmark. In all instances, larger models achieve superior performance. The smaller models, base and light, exhibit relatively lower performance, yet still within an acceptable range. In all scenarios, *LeBenchmark 2.0* models demonstrate a higher level of performance than the multilingual baselines (XLSR-53-base and XLS-R-xlarge). This highlights that language-specific SSL models are still a better choice for application in downstream tasks, compared to general purpose multilingual models.

6. Carbon Footprint

This section gives an overview of the carbon footprint of the SSL pre-training. The fine-tuning footprint is omitted as it was performed in many different and heterogeneous platforms making it impossible to compute proper estimates. The carbon footprint of each model may be estimated following the protocol defined by T. Parcollet et al. [149]. In practice, it is a function of the PUE of the compute infrastructure, the total energy consumed, and the carbon emission rate of the energy grid. The Jean-Zay supercomputer is energy efficient with a PUE of 0.65 (including heat transfer to nearby infrastructures). We only consider the energy consumed by the GPU and CPU. Power draw measurements are taken every five seconds and then averaged. France's carbon rate is 52 gCO2/kWh [150].

Table 13: Estimates of the energy in kilowatt hour (kwH) and CO_2 equivalent in kilogram produced by the training of the *LeBenchmark 2.0* models.

Model	Training time *(hours)*	GPUs	Total GPU hours *(hours)*	Energy *(kWh)*	CO_2 *(kg)*
1K-*base*	250	4 Tesla V100	1,000	195.0	10.5
1K-*large*	925	4 Tesla V100	3,700	721.5	37.5
2.7K-*base*	128	32 Tesla V100	4,096	682.2	35.4
3K-*base*	128	32 Tesla V100	4,096	682.2	35.4
3K-*large*	341	32 Tesla V100	10,912	1,817.5	94.5
7K-*base*	123	64 Tesla V100	7,872	1,535.0	79.8
7K-*large*	211	64 Tesla V100	13,504	4,501.0	234
14K-*light*	156	32 Tesla V100	4,992	1,497.6	77.8
14K-*large*	436	64 Tesla V100	27,904	8,371.2	435
14K-*xlarge*	525	104 Tesla A100	54,600	16,511.2	859

Table 13 reports the total energy consumed as well as the estimated carbon emissions of all *LeBenchmark 2.0* models. First, it is quite clear that the carbon footprint of training large SSL models is far from being negligible, even

in countries with a relatively clean energy mix. As supported by previous research, the location of the training must be considered as a critical design choice when starting such a project as the total CO_2 emissions can easily be multiplied by four to height if gas and oil are integrated in the mix. Finally, one may wonder if the extra kWhs thrown at the model are worth it given the relatively small downstream performance improvement between the 3K-large, 7K-large, and 7K-xlarge, in contrast to the energy consumption being multiplied by a factor 3.5.

7. Conclusion

LeBenchmark 2.0 establishes new foundations for the development of French SSL-equipped speech technologies. Following the three steps of the lifecycle of any SSL model, we gathered and documented the largest available collection of unlabeled French speech for SSL pre-training, we trained three new pre-trained SSL models for a total of 10 available checkpoints, and we evaluated them on two new tasks increasing the total number of tasks to six. *LeBenchmark 2.0* models are shared with the community via the HuggingFace Hub.

8. Acknowledgements

This work was performed using HPC resources from GENCI–IDRIS (Grants 2022-A0131013821, 2023-A0151014633, AD011013257R2) and was also partially supported by MIAI@Grenoble-Alpes (ANR-19-P3IA-0003) and E-SSL project (ANR-22-CE23-0013). This paper was also partially funded by the European Commission through the SELMA project under grant number 957017, and UTTER project under grant number 101070631. We would like to thank William N. Havard for the original idea of scraping the Audiocite website.

References

[1] A. Mohamed, H.-y. Lee, L. Borgholt, J. D. Havtorn, J. Edin, C. Igel, K. Kirchhoff, S.-W. Li, K. Livescu, L. Maaløe, et al., Self-supervised speech representation learning: A review, arXiv preprint arXiv:2205.10643 (2022).

[2] S. Liu, A. Mallol-Ragolta, E. Parada-Cabaleiro, K. Qian, X. Jing, A. Kathan, B. Hu, B. W. Schuller, Audio self-supervised learning: A survey, Patterns 3 (2022) 100616.

[3] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, J. Tang, Self-supervised learning: Generative or contrastive, IEEE Transactions on Knowledge and Data Engineering (2021).

[4] L. Jing, Y. Tian, Self-supervised visual feature learning with deep neural networks: A survey, IEEE transactions on pattern analysis and machine intelligence 43 (2020) 4037–4058.

[5] C. J. Reed, X. Yue, A. Nrusimha, S. Ebrahimi, V. Vijaykumar, R. Mao, B. Li, S. Zhang, D. Guillory, S. Metzger, et al., Self-supervised pretraining improves self-supervised pretraining, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 2584–2594.

[6] S. Sinha, A. Mandlekar, A. Garg, S4rl: Surprisingly simple self-supervision for offline reinforcement learning in robotics, in: Conference on Robot Learning, PMLR, 2022, pp. 907–917.

[7] Y. Wang, C. Albrecht, N. A. A. Braham, L. Mou, X. Zhu, Self-supervised learning in remote sensing: A review, IEEE Geoscience and Remote Sensing Magazine (2022).

[8] R. Krishnan, P. Rajpurkar, E. J. Topol, Self-supervised learning in medicine and healthcare, Nature Biomedical Engineering (2022) 1–7.

[9] S. Shurrab, R. Duwairi, Self-supervised learning methods and applications in medical imaging analysis: A survey, PeerJ Computer Science 8 (2022) e1045.

[10] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, A. Mohamed, Hubert: Self-supervised speech representation learning by masked prediction of hidden units, IEEE/ACM Transactions on Audio, Speech, and Language Processing 29 (2021) 3451–3460.

[11] A. Baevski, Y. Zhou, A. Mohamed, M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations, Advances in Neural Information Processing Systems 33 (2020) 12449–12460.

[12] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, et al., Wavlm: Large-scale self-supervised pre-training for full stack speech processing, arXiv preprint arXiv:2110.13900 (2021).

[13] P. Sarkar, A. Etemad, Self-supervised ecg representation learning for emotion recognition, IEEE Transactions on Affective Computing (2020).

[14] S. Evain, H. Nguyen, H. Le, M. Z. Boito, S. Mdhaffar, S. Alisamir, Z. Tong, N. Tomashenko, M. Dinarelli, T. Parcollet, A. Allauzen, Y. Estève, B. Lecouteux, F. Portet, S. Rossato, F. Ringeval, D. Schwab, laurent besacier, Task agnostic and task specific self-supervised learning from speech with lebenchmark, in: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2), 2021. URL: `https://openreview.net/forum?id=TSvj5dmuSd`.

[15] S. wen Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, T.-H. Huang, W.-C. Tseng, K. tik Lee, D.-R. Liu, Z. Huang, S. Dong, S.-W. Li, S. Watanabe, A. Mohamed, H. yi Lee, SUPERB: Speech Processing Universal PERformance Benchmark, in: Proc. Interspeech 2021, 2021, pp. 1194–1198. doi:`10.21437/Interspeech.2021-1775`.

[16] Z. Chen, S. Chen, Y. Wu, Y. Qian, C. Wang, S. Liu, Y. Qian, M. Zeng, Large-scale self-supervised speech representation learning for automatic speaker verification, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 6147–6151.
[17] H. Zhang, Y. Zou, H. Wang, Contrastive self-supervised learning for text-independent speaker verification, in: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2021, pp. 6713–6717.
[18] H. Nguyen, F. Bougares, N. Tomashenko, Y. Estève, L. Besacier, Investigating self-supervised pre-training for end-to-end speech translation, in: Proc. Interspeech 2020, 2020, pp. 1466–1470. doi:10.21437/Interspeech.2020-1835.
[19] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino, et al., Xls-r: Self-supervised cross-lingual speech representation learning at scale, arXiv preprint arXiv:2111.09296 (2021).
[20] A. T. Liu, S.-W. Li, H.-y. Lee, Tera: Self-supervised learning of transformer encoder representation for speech, IEEE/ACM Transactions on Audio, Speech, and Language Processing 29 (2021) 2351–2366.
[21] M. Dinarelli, M. Naguib, F. Portet, Toward Low-Cost End-to-End Spoken Language Understanding, in: Proc. Interspeech 2022, 2022, pp. 2728–2732. doi:10.21437/Interspeech.2022-10702.
[22] G. Laperriere, V. Pelloin, M. Rouvier, T. Stafylakis, Y. Esteve, On the use of semantically-aligned speech representations for spoken language understanding, arXiv preprint arXiv:2210.05291 (2022).
[23] Z. Huang, S. Watanabe, S.-w. Yang, P. García, S. Khudanpur, Investigating self-supervised learning for speech enhancement and separation, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 6837–6841.
[24] A. Sivaraman, M. Kim, Efficient personalized speech enhancement through self-supervised learning, IEEE Journal of Selected Topics in Signal Processing 16 (2022) 1342–1356.
[25] Y.-C. Wang, S. Venkataramani, P. Smaragdis, Self-supervised learning for speech enhancement, arXiv preprint arXiv:2006.10388 (2020).
[26] H.-S. Tsai, H.-J. Chang, W.-C. Huang, Z. Huang, K. Lakhotia, S.-w. Yang, S. Dong, A. Liu, C.-I. Lai, J. Shi, et al., Superb-sg: Enhanced speech processing universal performance benchmark for semantic and generative capabilities, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 8479–8492.
[27] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. Bowman, Glue: A multi-task benchmark and analysis platform for natural language understanding, in: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, 2018, pp. 353–355.
[28] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, S. Bowman, Superglue: A stickier benchmark for general-purpose language understanding systems, Advances in neural information processing systems 32 (2019).
[29] J. Shi, D. Berrebbi, W. Chen, E.-P. Hu, W.-P. Huang, H.-L. Chung, X. Chang, S.-W. Li, A. Mohamed, H. yi Lee, S. Watanabe, ML-SUPERB: Multilingual Speech Universal PERformance Benchmark, in: Proc. INTERSPEECH 2023, 2023, pp. 884–888. doi:10.21437/Interspeech.2023-1316.
[30] T. Javed, K. Bhogale, A. Raman, P. Kumar, A. Kunchukuttan, M. M. Khapra, Indicsuperb: A speech processing universal performance benchmark for indian languages, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 37, 2023, pp. 12942–12950.
[31] S. Shon, A. Pasad, F. Wu, P. Brusco, Y. Artzi, K. Livescu, K. J. Han, Slue: New benchmark tasks for spoken language understanding evaluation on natural speech, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 7927–7931.
[32] S. Evain, H. Nguyen, H. Le, M. Z. Boito, S. Mdhaffar, S. Alisamir, Z. Tong, N. Tomashenko, M. Dinarelli, T. Parcollet, A. Allauzen, Y. Estève, B. Lecouteux, F. Portet, S. Rossato, F. Ringeval, D. Schwab, L. Besacier, LeBenchmark: A Reproducible Framework for Assessing Self-Supervised Representation Learning from Speech, in: Proc. Interspeech 2021, 2021, pp. 1439–1443. doi:10.21437/Interspeech.2021-556.
[33] D. Schlangen, Targeting the benchmark: On methodology in current natural language processing research, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), 2021, pp. 670–674.
[34] A. T. Liu, S.-w. Yang, P.-H. Chi, P.-c. Hsu, H.-y. Lee, Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 6419–6423.
[35] S. Ling, Y. Liu, Decoar 2.0: Deep contextualized acoustic representations with vector quantization, arXiv preprint arXiv:2012.06659 (2020).
[36] X. Song, G. Wang, Y. Huang, Z. Wu, D. Su, H. Meng, Speech-xlnet: Unsupervised acoustic model pretraining for self-attention networks (2020).
[37] X. Yue, H. Li, Phonetically motivated self-supervised speech representation learning., in: Interspeech, 2021, pp. 746–750.
[38] A. Baevski, A. Babu, W.-N. Hsu, M. Auli, Efficient self-supervised learning with contextualized target representations for vision, speech and language, arXiv preprint arXiv:2212.07525 (2022).
[39] G.-P. Yang, S.-L. Yeh, Y.-A. Chung, J. Glass, H. Tang, Autoregressive predictive coding: A comprehensive study, IEEE Journal of Selected Topics in Signal Processing 16 (2022) 1380–1390.
[40] C.-C. Chiu, J. Qin, Y. Zhang, J. Yu, Y. Wu, Self-supervised learning with random-projection quantizer for speech recognition, in: International Conference on Machine Learning, PMLR, 2022, pp. 3915–3924.
[41] A. v. d. Oord, Y. Li, O. Vinyals, Representation learning with contrastive predictive coding, arXiv preprint arXiv:1807.03748 (2018).
[42] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, Y. Bengio, Multi-task self-supervised learning for robust speech recognition, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 6989–6993.
[43] S. Zaiem, Y. Kemiche, T. Parcollet, S. Essid, M. Ravanelli, Speech self-supervised representation benchmarking: Are we doing it right?, in: ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2023.
[44] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, M. Auli, fairseq: A fast, extensible toolkit for sequence modeling, arXiv preprint arXiv:1904.01038 (2019).
[45] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, et al., Speech-

brain: A general-purpose speech toolkit, arXiv preprint arXiv:2106.04624 (2021).
[46] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, R. Collobert, Mls: A large-scale multilingual dataset for speech research, in: INTERSPEECH, Shanghai, China, 2020.
[47] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, E. Dupoux, VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online, 2021.
[48] Y. Estève, T. Bazillon, J.-Y. Antoine, F. Béchet, J. Farinas, The EPAC Corpus: Manual and Automatic Annotations of Conversational Speech in French Broadcast News, in: Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), European Language Resources Association (ELRA), Valletta, Malta, 2010. URL: http://www.lrec-conf.org/proceedings/lrec2010/pdf/650_Paper.pdf.
[49] African accented french, slr57, 2003. Type: dataset, https://www.openslr.org/57/.
[50] C. Le Moine, N. Obin, Att-HACK: An Expressive Speech Database with Social Attitudes, in: Speech Prosody, 2020.
[51] P. Gournay, O. Lahaie, R. Lefebvre, A Canadian French emotional speech dataset, in: MMSys, 2018.
[52] S. Branca-Rosoff, S. Fleury, F. Lefeuvre, M. Pires, Discours sur la ville. Présentation du Corpus de Français parlé Parisien des années 2000 (CFPP2000), 2012. Http://cfpp2000.univ-paris3.fr/CFPP2000.pdf.
[53] I. Eshkol-Taravella, O. Baude, D. Maurel, L. Hriba, C. Dugua, I. Tellier, Un grand corpus oral "disponible" : le corpus d'Orléans 1968-2012, Ressources Linguistiques Libres - Traitement Automatique des Langues 53 (2011) 17–46. URL: https://www.atala.org/content/un-grand-corpus-oral-%C2%AB-disponible-%C2%BB-le-corpus-d%E2%80%99orl%C3%A9ans-1968-2012.
[54] T. Bänziger, M. Mortillaro, K. Scherer, Introducing the Geneva Multimodal Expression Corpus for Experimental Research on Emotion Perception, Emotion (Washington, D.C.) 12 (2012) 1161–79. URL: https://www.researchgate.net/publication/51796867_Introducing_the_Geneva_Multimodal_Expression_Corpus_for_Experimental_Research_on_Emotion_Perception. doi:10.1037/a0025827.
[55] G. Françoise, Les parlers jeunes dans l'île-de-France multiculturelle, Paris and Gap, Ophrys (2017).
[56] Mpf, 2019. Https://hdl.handle.net/11403/mpf/v3, ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr.
[57] F. Lefèvre, D. Mostefa, L. Besacier, Y. Estève, M. Quignard, N. Camelin, B. Favre, B. Jabaian, L. Rojas-Barahona, Robustesse et portabilités multilingue et multi-domaines des systèmes de compréhension de la parole : le projet PortMedia, in: Actes de la conférence conjointe JEP-TALN-RECITAL 2012, volume 1:JEP, Grenoble, France, 2012, pp. 779–786. URL: https://www.researchgate.net/publication/225285476_Robustesse_et_portabilites_multilingue_et_multi-domaines_des_systemes_de_comprehension_de_la_parole_le_projet_PortMedia.
[58] ATILF, TCOF : Traitement de corpus oraux en français, 2020. Https://hdl.handle.net/11403/tcof/v2.1, ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr.
[59] M. Z. Boito, W. Havard, M. Garnerin, É. Le Ferrand, L. Besacier, MaSS: A large and clean multilingual corpus of sentence-aligned spoken utterances extracted from the Bible, in: Proceedings of the 12th Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2020.
[60] F. Torreira, M. Adda-Decker, M. Ernestus, The Nijmegen Corpus of Casual French, Speech Communication 52 (2010) 201. URL: https://hal.archives-ouvertes.fr/hal-00608402. doi:10.1016/j.specom.2009.10.004, publisher: Elsevier : North-Holland.
[61] S. Felice, S. Evain, F. Portet, Audiocité, 2022. URL: https://www.audiocite.net/.
[62] M. Zanon Boito, F. Bougares, F. Barbier, S. Gahbiche, L. Barrault, M. Rouvier, Y. Estève, Niger-mali audio collection, 2022. URL: https://demo-lia.univ-avignon.fr/studios-tamani-kalangou/.
[63] M. Zanon Boito, F. Bougares, F. Barbier, S. Gahbiche, L. Barrault, M. Rouvier, Y. Estève, Speech resources in the tamasheq language, in: Proceedings of LREC 2022, 2022. URL: https://arxiv.org/pdf/2201.05051.pdf.
[64] S. Meignier, T. Merlin, Lium spkdiarization: an open source toolkit for diarization, in: CMU SPUD Workshop, 2010.
[65] M. Z. Boito, L. Besacier, N. Tomashenko, Y. Estève, A study of gender impact in self-supervised models for speech-to-text systems, arXiv preprint arXiv:2204.01397 (2022).
[66] Y.-A. Chung, W.-N. Hsu, H. Tang, J. Glass, An Unsupervised Autoregressive Model for Speech Representation Learning, in: Proc. Interspeech 2019, 2019, pp. 146–150. doi:10.21437/Interspeech.2019-1473.
[67] W. Chen, X. Chang, Y. Peng, Z. Ni, S. Maiti, S. Watanabe, Reducing barriers to self-supervised learning: Hubert pre-training with academic compute, arXiv preprint arXiv:2306.06672 (2023).
[68] T. Ashihara, T. Moriya, K. Matsuura, T. Tanaka, Deep versus Wide: An Analysis of Student Architectures for Task-Agnostic Knowledge Distillation of Self-Supervised Speech Models, in: Proc. Interspeech 2022, 2022, pp. 411–415. doi:10.21437/Interspeech.2022-11313.
[69] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, in: International Conference on Learning Representations, 2018.
[70] Y. Gao, J. Fernandez-Marques, T. Parcollet, P. P. de Gusmao, N. D. Lane, Match to win: Analysing sequences lengths for efficient self-supervised learning in speech and audio, IEEE Spoken Language Technology Workshop (2022).
[71] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, M. Auli, Unsupervised Cross-Lingual Representation Learning for Speech Recognition, in: Proc. Interspeech 2021, 2021, pp. 2426–2430. doi:10.21437/Interspeech.2021-329.
[72] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, I. Sutskever, Robust speech recognition via large-scale weak supervision, arXiv preprint arXiv:2212.04356 (2022).
[73] G. Gravier, G. Adda, N. Paulsson, M. Carré, A. Giraudel, O. Galibert, The ETAPE corpus for the evaluation of speech-based TV content processing in the French language, in: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), 2012, pp. 114–118.
[74] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, G. Weber, Common voice: A massively-multilingual speech corpus, in: Proceedings of the Twelfth Language Resources and Evaluation Conference, 2020, pp. 4218–4222.
[75] A. Graves, S. Fernández, F. Gomez, J. Schmidhuber, Connectionist temporal classification: Labelling unsegmented sequence data with

recurrent neural networks, in: Proceedings of ICML, ACM, 2006, pp. 369–376. doi:10.1145/1143844.1143891.
[76] R. De Mori, Spoken Dialogues with Computers, Academic Press, Inc., Orlando, FL, USA, 1997.
[77] C. Raymond, F. Béchet, R. De Mori, G. Damnati, On the use of finite state transducers for semantic interpretation, Speech Communication 48 (2006) 288–304. doi:10.1016/j.specom.2005.06.012.
[78] M. Dinarelli, A. Moschitti, G. Riccardi, Re-ranking models based-on small training data for spoken language understanding, in: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2009, pp. 1076–1085. URL: https://aclanthology.org/D09-1112.
[79] M. Dinarelli, A. Moschitti, G. Riccardi, Concept segmentation and labeling for conversational speech, in: Interspeech, Brighton, U.K., 2009.
[80] S. Quarteroni, G. Riccardi, M. Dinarelli, What's in an ontology for spoken language understanding, in: Interspeech, Brighton, U.K., 2009.
[81] S. Hahn, M. Dinarelli, et al., Comparing stochastic approaches to spoken language understanding in multiple languages, IEEE TASLP 99 (2010).
[82] A. Caubrière, S. Ghannay, et al., Error analysis applied to end-to end spoken language understanding, in: In ICASSP, Barcelona, Spain, 2020.
[83] S. Ghannay, A. Caubrière, et al., Where are we in semantic concept extraction for Spoken Language Understanding? ⋆, in: In SPECOM 2021, Saint Petersburg, Russia, 2021.
[84] Y. Dupont, M. Dinarelli, I. Tellier, Label-dependencies aware recurrent neural networks, in: A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing, Springer International Publishing, Cham, 2018, pp. 44–66.
[85] D. Serdyuk, Y. Wang, et al., Towards end-to-end spoken language understanding, CoRR abs/1802.08395 (2018). arXiv:1802.08395.
[86] M. Dinarelli, V. Vukotic, C. Raymond, Label-dependency coding in Simple Recurrent Networks for Spoken Language Understanding, in: Interspeech, Stockholm, Sweden, 2017. URL: https://hal.inria.fr/hal-01553830.
[87] T. Desot, F. Portet, M. Vacher, SLU FOR VOICE COMMAND IN SMART HOME: COMPARISON OF PIPELINE AND END-TO-END APPROACHES, in: ASRU Workshop, Sentosa, Singapore, Singapore, 2019.
[88] L. Lugosch, M. Ravanelli, et al., Speech model pre-training for end-to-end spoken language understanding, 2019. arXiv:1904.03670.
[89] A. Caubrière, N. A. Tomashenko, et al., Curriculum-based transfer learning for an effective end-to-end spoken language understanding and domain portability, CoRR abs/1906.07601 (2019). arXiv:1906.07601.
[90] M. Dinarelli, N. Kapoor, B. Jabaian, L. Besacier, A data efficient end-to-end spoken language understanding architecture, 2020. arXiv:2002.05955.
[91] V. Pelloin, N. Camelin, et al., End2End Acoustic to Semantic Transduction, in: In ICASSP, Toronto, ON, Canada, 2021. doi:10.1109/ICASSP39728.2021.9413581.
[92] T. Desot, F. Portet, M. Vacher, End-to-end spoken language understanding: Performance analyses of a voice command task in a low resource setting, Computer Speech & Language 75 (2022).
[93] H. Bonneau-Maynard, C. Ayache, et al., Results of the French evalda-media evaluation campaign for literal understanding, in: In LREC, European Language Resources Association (ELRA), Genoa, Italy, 2006.
[94] M. Dinarelli, M. Naguib, F. Portet, Toward Low-Cost End-to-End Spoken Language Understanding, in: Interspeech 2022, ISCA, Incheon, South Korea, 2022, pp. 2728–2732. URL: https://hal.science/hal-03872546. doi:10.21437/Interspeech.2022-10702.
[95] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (1997).
[96] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: Y. Bengio, Y. LeCun (Eds.), In ICLR, 2015.
[97] W. Chan, N. Jaitly, Q. V. Le, O. Vinyals, Listen, attend and spell, CoRR abs/1508.01211 (2015). arXiv:1508.01211.
[98] A. Vaswani, N. Shazeer, et al., Attention is all you need, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), In NIPS, volume 30, Curran Associates, Inc., 2017.
[99] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter, CoRR abs/1910.01108 (2019). URL: http://arxiv.org/abs/1910.01108. arXiv:1910.01108.
[100] R. Müller, S. Kornblith, G. Hinton, When Does Label Smoothing Help?, Curran Associates Inc., Red Hook, NY, USA, 2019.
[101] X. Li, C. Wang, Y. Tang, C. Tran, Y. Tang, J. Pino, A. Baevski, A. Conneau, M. Auli, Multilingual speech translation from efficient finetuning of pretrained models, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online, 2021, pp. 827–838. URL: https://aclanthology.org/2021.acl-long.68. doi:10.18653/v1/2021.acl-long.68.
[102] S. Elizabeth, W. Matthew, B. Jacob, R. Cattoni, M. Negri, M. Turchi, D. W. Oard, P. Matt, The multilingual tedx corpus for speech recognition and translation, in: Proceedings of Interspeech 2021, 2021, pp. 3655–3659.
[103] C. Wang, A. Wu, J. Gu, J. Pino, CoVoST 2 and Massively Multilingual Speech Translation, in: Proc. Interspeech 2021, 2021, pp. 2247–2251. doi:10.21437/Interspeech.2021-2027.
[104] C. Wang, Y. Tang, X. Ma, A. Wu, D. Okhonko, J. M. Pino, fairseq S2T: fast speech-to-text modeling with fairseq, CoRR abs/2010.05171 (2020). URL: https://arxiv.org/abs/2010.05171. arXiv:2010.05171.
[105] T. Kudo, J. Richardson, SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 66–71. URL: https://aclanthology.org/D18-2012. doi:10.18653/v1/D18-2012.
[106] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015. URL: http://arxiv.org/abs/1412.6980.
[107] M. Post, A call for clarity in reporting BLEU scores, in: Proceedings of the Third Conference on Machine Translation: Research Papers, Association for Computational Linguistics, Belgium, Brussels, 2018, pp. 186–191. URL: https://www.aclweb.org/anthology/W18-6319.

[108] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. Rush, Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 38–45. URL: https://aclanthology.org/2020.emnlp-demos.6. doi:10.18653/v1/2020.emnlp-demos.6.

[109] M. Z. Boito, J. Ortega, H. Riguidel, A. Laurent, L. Barrault, F. Bougares, F. Chaabani, H. Nguyen, F. Barbier, S. Gahbiche, Y. Estève, ON-TRAC consortium systems for the IWSLT 2022 dialect and low-resource speech translation tasks, in: E. Salesky, M. Federico, M. Costa-jussà (Eds.), Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022), Association for Computational Linguistics, Dublin, Ireland (in-person and online), 2022, pp. 308–318. URL: https://aclanthology.org/2022.iwslt-1.28. doi:10.18653/v1/2022.iwslt-1.28.

[110] P. Koehn, Statistical significance tests for machine translation evaluation, in: EMNLP, ACL, 2004, pp. 388–395.

[111] A. Pasad, J.-C. Chou, K. Livescu, Layer-wise analysis of a self-supervised speech representation model, in: 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), IEEE, 2021, pp. 914–921.

[112] A. Pasad, B. Shi, K. Livescu, Comparative layer-wise analysis of self-supervised speech models, in: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2023, pp. 1–5.

[113] A. Moors, K. R. Scherer, The role of appraisal in emotion, Handbook of cognition and emotion (2013) 135–155.

[114] T. Brosch, K. Scherer, D. Grandjean, D. Sander, The impact of emotion on perception, attention, memory, and decision-making, Swiss medical weekly 143 (2013) w13786–w13786.

[115] S. Alisamir, F. Ringeval, F. Portet, Multi-corpus affect recognition with emotion embeddings and self-supervised representations of speech, in: 2022 10th International Conference on Affective Computing and Intelligent Interaction (ACII), IEEE, 2022.

[116] F. Ringeval, A. Sonderegger, J. Sauer, D. Lalanne, Introducing the recola multimodal corpus of remote collaborative and affective interactions, in: 2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG), IEEE, 2013, pp. 1–8.

[117] M. Macary, M. Tahon, Y. Estève, A. Rousseau, Allosat: A new call center french corpus for satisfaction and frustration analysis, in: Proceedings of the 12th Language Resources and Evaluation Conference, 2020, pp. 1590–1597.

[118] S. Alisamir, F. Ringeval, F. Portet, Multi-corpus affect recognition with emotion embeddings and self-supervised representations of speech, in: 2022 10th International Conference on Affective Computing and Intelligent Interaction (ACII), IEEE, 2022, pp. 1–8.

[119] A. Pupier, M. Coavoux, B. Lecouteux, J. Goulian, End-to-End Dependency Parsing of Spoken French, in: Proc. Interspeech 2022, 2022, pp. 1816–1820. doi:10.21437/Interspeech.2022-381.

[120] C. Benzitoun, J.-M. Debaisieux, H.-J. Deulofeu, Le projet orféo: un corpus d'étude pour le français contemporain, Corpus (2016).

[121] CLESTHIA, Cfpp2000, 2018. URL: https://hdl.handle.net/11403/cfpp2000/v1, ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr.

[122] ICAR, Clapi, 2017. URL: https://hdl.handle.net/11403/clapi/v1, ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr.

[123] ATILF, Tcof : Traitement de corpus oraux en français, 2020. URL: https://hdl.handle.net/11403/tcof/v2.1, ORTOLANG (Open Resources and TOols for LANGuage) –www.ortolang.fr.

[124] A. Mathieu, B. Marie-José, C. Gilles, D. Federica, J. L. Anne, Corpus ofrom – corpus oral de français de suisse romande, (2012-2020). URL: www.unine.ch/ofrom, université de Neuchâtel.

[125] V. André, Fleuron: Français langue Étrangère universitaire–ressources et outils numériques, 2016. URL: https://fleuron.atilf.fr/index.php?lg=fr.

[126] J. Carruthers, French oral narrative corpus, 2013. Commissioning Body / Publisher: Oxford Text Archive.

[127] E. Cresti, F. B. do Nascimento, A. M. Sandoval, J. Veronis, P. Martin, K. Choukri, The c-oral-rom corpus. a multilingual resource of spontaneous speech for romance languages, 2004, pp. 26–28.

[128] E. DELIC, S. Teston-Bonnard, J. Véronis, Présentation du corpus de référence du français parlé, Recherches sur le français parlé 18 (2004) 11–42. URL: https://halshs.archives-ouvertes.fr/halshs-01388193, equipe DELIC.

[129] N. Alexis, D. Franck, B. Frederic, F. Benoit, Annotation syntaxique automatique de la partie orale du orfÉo, in: Langages, 2020. doi:https://doi.org/10.3917/lang.219.0087.

[130] M. D. Zeiler, ADADELTA: an adaptive learning rate method, CoRR abs/1212.5701 (2012). URL: http://arxiv.org/abs/1212.5701. arXiv:1212.5701.

[131] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, G. Weber, Common voice: A massively-multilingual speech corpus, in: Proceedings of the Twelfth Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2020, pp. 4218–4222. URL: https://aclanthology.org/2020.lrec-1.520.

[132] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-García, D. Petrovska-Delacrétaz, D. A. Reynolds, A tutorial on text-independent speaker verification, EURASIP Journal on Advances in Signal Processing 2004 (2004) 1–22.

[133] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, P. Ouellet, Front-end factor analysis for speaker verification, IEEE Transactions on Audio, Speech, and Language Processing 19 (2010) 788–798.

[134] J. Villalba, N. Chen, D. Snyder, D. Garcia-Romero, A. McCree, G. Sell, J. Borgstrom, F. Richardson, S. Shon, F. Grondin, et al., The jhu-mit system description for nist sre18 (2018).

[135] K. A. Lee, H. Yamamoto, K. Okabe, Q. Wang, L. Guo, T. Koshinaka, J. Zhang, K. Shinoda, The nec-tt 2018 speaker verification system., in: Proc. Interspeech, 2019, pp. 4355–4359.

[136] P.-M. B. Mickael Rouvier, The lia system description for nist sre 2019 (2019).

[137] J. Thienpondt, B. Desplanques, K. Demuynck, The idlab voxceleb speaker recognition challenge 2020 system description (2020).

[138] N. Brummer, L. Burget, O. Glembek, P. Matejka, L. Mošner, O. Novotnỳ, O. Plchot, J. Rohdin, A. Silnova, T. Stafylakis, et al., But+ omilia system description voxceleb speaker recognition challenge 2020 (????).

[139] N. Torgashov, Id r&d system description to voxceleb speaker recognition challenge 2020, 2020.
[140] N. Vaessen, D. A. Van Leeuwen, Fine-tuning wav2vec2 for speaker recognition, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 7967–7971.
[141] M. Ajili, J.-F. Bonastre, J. Kahn, S. Rossato, G. Bernard, Fabiole, a speech database for forensic speaker comparison, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), 2016, pp. 726–733.
[142] S. Galliano, E. Geoffrois, G. Gravier, J.-F. Bonastre, D. Mostefa, K. Choukri, Corpus description of the ESTER evaluation campaign for the rich transcription of french broadcast news., in: LREC, Citeseer, 2006, pp. 139–142.
[143] S. Galliano, G. Gravier, L. Chaubard, The ESTER 2 evaluation campaign for the rich transcription of French radio broadcasts, in: Tenth Annual Conference of the International Speech Communication Association, 2009.
[144] A. Giraudel, M. Carré, V. Mapelli, J. Kahn, O. Galibert, L. Quintard, The REPERE corpus: a multimodal corpus for person recognition., in: LREC, 2012, pp. 1102–1107.
[145] D. Snyder, G. Chen, D. Povey, Musan: A music, speech, and noise corpus, 2015. arXiv:1510.08484.
[146] B. Desplanques, J. Thienpondt, K. Demuynck, ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN based speaker verification, in: Interspeech, 2020.
[147] J. Thienpondt, B. Desplanques, K. Demuynck, The IDLab VoxSRC-20 submission: Large margin fine-tuning and quality-aware score calibration in DNN based speaker verification, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021.
[148] A. S. Luccioni, Y. Jernite, E. Strubell, Power hungry processing: Watts driving the cost of ai deployment?, arXiv preprint arXiv:2311.16863 (2023).
[149] T. Parcollet, M. Ravanelli, The Energy and Carbon Footprint of Training End-to-End Speech Recognizers, in: Proc. Interspeech 2021, 2021, pp. 4583–4587. doi:10.21437/Interspeech.2021-456.
[150] E. E. Agency, Greenhouse gas emission intensity of electricity generation, European Environment Agency, 2020.