Actes de la 29e Conférence sur le Traitement Automatique des Langues Naturelles
Avignon, France, 27 juin au 1er juillet 2022

Vers la compréhension automatique de la parole bout-en-bout à moindre effort

Marco Naguib François Portet Marco Dinarelli
Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG, 38000 Grenoble, France
marco.naguib@hotmail.com,
(francois.portet|marco.dinarelli)@univ-grenoble-alpes.fr

RÉSUMÉ

Les approches de compréhension automatique de la parole ont récemment bénéficié de l'apport de modèles préappris par autosupervision sur de gros corpus de parole. Pour le français, le projet *LeBenchmark* a rendu disponibles de tels modèles et a permis des évolutions impressionnantes sur plusieurs tâches dont la compréhension automatique de la parole. Ces avancées ont un coût non négligeable en ce qui concerne le temps de calcul et la consommation énergétique. Dans cet article, nous comparons plusieurs stratégies d'apprentissage visant à réduire le coût énergétique tout en conservant des performances compétitives. Les expériences sont effectuées sur le corpus MEDIA, et montrent qu'il est possible de réduire significativement le coût d'apprentissage tout en conservant des performances à l'état de l'art.

ABSTRACT

Towards automatic end-to-end speech understanding with less effort

Recent advances in spoken language understanding benefited from Self-Supervised models trained on large speech corpora. For French, the LeBenchmark project has made such models available and has led to impressive progress on several tasks including spoken language understanding. These advances have a non-negligible cost in terms of computation time and energy consumption. In this paper, we compare several learning strategies aiming at reducing such cost while keeping competitive performances. The experiments are performed on the MEDIA corpus, and show that it is possible to reduce the learning cost while maintaining state-of-the-art performances.

MOTS-CLÉS : compréhension de la parole, apprentissage autosupervisé, apprentissage par transfert.

KEYWORDS: Spoken Language Understanding, Self-Supervised Learning, Transfer Learning.

1 Introduction

La compréhension automatique de la parole (SLU de *Spoken Language Understanding*) vise à extraire une représentation sémantique à partir d'un signal audio contenant un énoncé en langage naturel (Mori, 1997). Les approches classiques utilisées pour extraire la sémantique de la parole ont consisté à mettre en cascade un module de reconnaissance automatique de la parole avec un système de compréhension du langage naturel (Raymond *et al.*, 2006; Dinarelli *et al.*, 2009b,a; Hahn *et al.*, 2010; Dinarelli, 2010; Caubrière *et al.*, 2020; Ghannay *et al.*, 2021). Les réseaux de neurones ont permis l'avancement des systèmes bout-en-bout pour la SLU (Serdyuk *et al.*, 2018a; Desot *et al.*, 2019; Lugosch *et al.*, 2019; Caubrière *et al.*, 2019; Dinarelli *et al.*, 2020; Pelloin *et al.*, 2021), qui sont

Volume 1 : conférence principale, pages 257–268.

préférés aux systèmes en cascade, notamment pour leur capacité à réduire l'effet d'erreur en cascade et à exploiter des composantes acoustiques pour déduire certaines informations sémantiques (Desot *et al.*, 2019).
Bien que des approches ont proposé un apprentissage de bout en bout du modèle (Qian *et al.*, 2017; Desot *et al.*, 2019; Palogiannidi *et al.*, 2020), de nombreux travaux ont appliqué un apprentissage graduel du modèle sur des tâches de plus en plus spécifiques. Partant de l'hypothèse qu'un modèle de SLU doit nécessairement apprendre une représentation de la parole, (Lugosch *et al.*, 2019) et (Dinarelli *et al.*, 2020) proposent une approche où le modèle est progressivement entraîné à reconnaître la transcription puis à en extraire la sémantique. On peut également citer (Serdyuk *et al.*, 2018b) et (Radfar *et al.*, 2020) qui apprennent, en première étape, un classificateur de domaine auquel se rapporte l'intention, avant d'optimiser le modèle pour classifier les intentions et les attributs sémantiques d'un énoncé. Un changement récent dans les approches de SLU de bout en bout est l'utilisation de modèles appris par *auto-supervision* (SSL de *Self-Supervised Learning*) sur de très gros corpus de parole, tels que *wav2vec* ou *HuBERT* (Schneider *et al.*, 2019; Baevski *et al.*, 2020; Hsu *et al.*, 2021). Dans (Lai *et al.*, 2020), un modèle wav2vec préappris est utilisé comme encodeur de la parole tandis que le benchmark SUPERB (Yang *et al.*, 2021) propose une tâche de *slot-filling* et de classification d'intention parmi les tâches d'évaluation des modèles de la parole préappris. En 2021, de tels modèles ont été mis à disposition de la communauté française (Evain *et al.*, 2021a,b), permettant une amélioration impressionnante des performances sur des tâches telles que la SLU.
Si l'on peut se féliciter que les avancés de la recherche ont permis d'améliorer les performances obtenues sur les tâches visées, ces avancées ont un coût non négligeable en ce qui concerne le temps de calcul et la consommation énergétique (Parcollet & Ravanelli, 2021). Même des modèles SSL monolingues (Evain *et al.*, 2021a,b) demandent près de deux semaines d'apprentissage sur 64 GPUs. On peut arguer que ce coût reste contenu par le fait que ces modèles sont entraînés une fois et utilisés ensuite pour beaucoup d'applications différentes. Cependant, ces modèles ne constituent souvent qu'un encodeur et doivent être adaptés sur les tâches en aval pour améliorer les performances des systèmes. Cette pratique multiplie davantage les phases d'apprentissage et conduit par conséquent à des consommations de ressources importantes.
Dans cet article nous nous intéressons à réduire le coût nécessaire pour obtenir des performances compétitives sur des tâches de SLU en utilisant les modèles SSL déployés par (Evain *et al.*, 2021a,b). Dans cet article, nous proposons une étude visant à trouver un meilleur compromis entre les performances et le coût énergétique. Pour cela, nous étudions des stratégies d'apprentissage différentes de celles utilisées par (Evain *et al.*, 2021a,b) que nous couplons à une stratégie d'apprentissage par transfert avec des modèles appris pour d'autres tâches (Lefèvre *et al.*, 2012), et à une phase d'affinage d'un modèle SSL français directement sur la tâche SLU, au lieu de la tâche ASR comme proposé dans (Evain *et al.*, 2021b). Bien que cette dernière soit relativement coûteuse par rapport à l'apprentissage des modèles en aval pour la SLU, elle reste moins lourde que les approches similaires proposées récemment pour la même tâche (Pelloin *et al.*, 2021; Ghannay *et al.*, 2021), tout en permettant d'obtenir des performances comparables.

2 Compréhension automatique de la parole et modèles SSL

Dans cet article, nous exploitons des modèles préappris de manière autosupervisée et cherchons à tirer parti de ceux-ci de la manière la plus économe pour une tâche de SLU. Nous utiliserons les modèles *LeBenchmark* (Evain *et al.*, 2021b). Ces modèles ont été évalués sur quatre tâches dont la

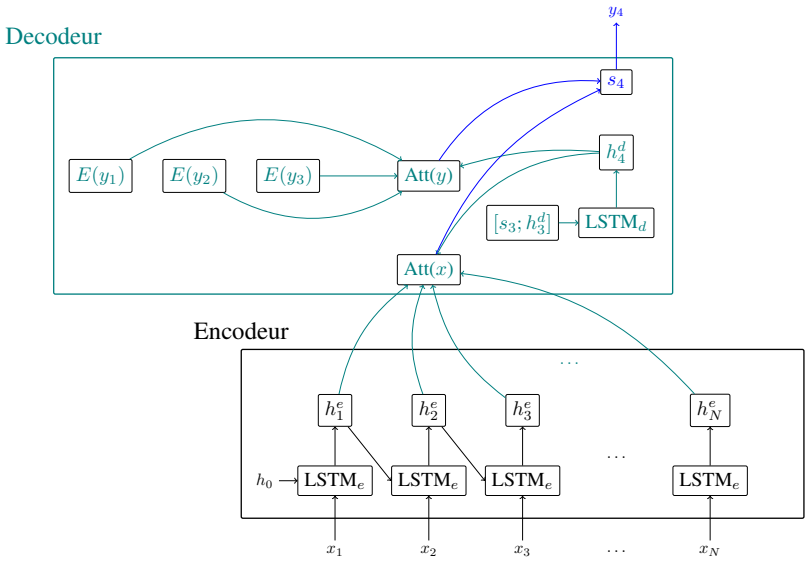

FIGURE 1 – *Schéma de notre architecture neuronale pour la SLU*

SLU. Sur cette dernière, le modèle *w2v2-fr-7k* a montré les meilleurs résultats, c'est pourquoi nous l'utilisons dans nos expériences.

Les modèles pour la SLU que nous utilisons sont les mêmes que ceux utilisés dans (Evain *et al.*, 2021b) [1]. Il s'agit de modèles *encodeur-décodeur* basés sur les LSTM et le mécanisme d'attention (Hochreiter & Schmidhuber, 1997; Bahdanau *et al.*, 2015). L'encodeur a une structure pyramidale similaire à (Chan *et al.*, 2015) tandis que le décodeur intègre deux mécanismes d'attention, un pour atteindre les états cachés de l'encodeur, l'autre pour atteindre toutes les prédictions précédentes, comme le module de *self-attention* des Transformers (Vaswani *et al.*, 2017). Un schéma de cette architecture neuronale est montrée dans la figure 1, où nous différencions les éléments de l'encodeur et du décodeur avec les exposants ^{e} et ^{d} respectivement. $E(y_i)$ indique le plongement de l'étiquette y_i, les deux mécanismes d'attention sont indiqués respectivement avec $Att(x)$ et $Att(y)$. Les modèles sont appris en minimisant la fonction de coût CTC (Graves *et al.*, 2006). Le choix du meilleur modèle est fait sur la base du taux d'erreur sur les données de développement de la tâche visée, en considérant uniquement les tours de parole utilisateur (cf section 3.2). Dans ce travail, nous utilisons les modèles SSL pour extraire les *features* qui alimentent les modèles SLU comme alternative aux paramètres classiques (MFCC, spectrogrammes, etc.).

Les modèles décrits dans (Evain *et al.*, 2021b) demandent au total trois étapes d'apprentissage, stratégie indiquée avec *3 steps* dans les tableaux, chacune utilisant le modèle de l'étape précédente pour l'initialisation des paramètres du modèle : **1)** apprentissage de l'encodeur sur la transcription ; **2)** apprentissage de l'encodeur sur la SLU ; **3)** apprentissage du modèle final, encodeur et décodeur, sur la SLU. Bien que cette stratégie soit la plus efficace, elle demande un coût d'apprentissage important, d'autant plus que les modèles les plus performants dans (Evain *et al.*, 2021b) demandent une étape additionnelle et coûteuse d'adaptation du modèle SSL sur la tâche finale.

Les performances des modèles SLU avec entrée produite par un modèle SSL sont très élevées, intuitivement donc des performances similaires, ou légèrement inférieures, peuvent être obtenues avec un coût d'apprentissage moindre, permettant d'économiser des ressources. Nous souhaitons avec nos analyses trouver un meilleur compromis entre effort d'apprentissage et performance finale du

1. Nous avons téléchargé les modèles disponibles sur https://huggingface.co/LeBenchmark et les systèmes disponibles sur https://github.com/LeBenchmark/NeurIPS2021

modèle.
Afin de valider cette hypothèse nous essayons alors deux stratégies d'apprentissage différentes par rapport à (Evain *et al.*, 2021b) (**Stg appr.** dans les tableaux) : *2 steps*, nous effectuons uniquement les étapes 2 et 3 de la stratégie *3 steps* ; *1 step*, nous entraînons directement le modèle final sur la SLU. Nous notons que cette dernière stratégie correspond à un apprentissage *réellement* de bout en bout d'un modèle SLU.

3 Évaluation

3.1 Évaluation du coût d'apprentissage

Les résultats quantitatifs de la SLU sont évalués avec le taux d'erreur sur les concepts (CER – *Concept Error Rate*). Nous évaluons le coût computationnel de nos modèles en mesurant : le temps d'apprentissage, la consommation électrique en *kWh* et sa conversion en grammes de CO_2 (gCO_2 dans les tableaux). Ces 2 dernières valeurs sont obtenues grâce au logiciel *codecarbon* [2]. Puisque celui-ci surestime le coefficient de conversion de kWh vers grammes de CO_2, comme dans (Parcollet & Ravanelli, 2021) nous utilisons le coefficient officiel de 51 grammes/kWh. [3] Afin d'avoir un cadre plus complet nous montrons également le coût en *kWh* par point de CER gagné (indiqué avec **kWh/p** dans les tableaux). Pour obtenir cette valeur, nous présumons qu'un modèle étalon $\mathcal{M}_e$ moins coûteux obtient des résultats inférieur à un modèle comparé $\mathcal{M}_c$. En indiquant alors avec $\text{kWh}(\mathcal{M}_i)$ et $\text{CER}(\mathcal{M}_i)$ respectivement la consommation énergétique et le CER du modèle $\mathcal{M}_i$, la valeur *kWh/p* est obtenue par $\frac{\text{kWh}(\mathcal{M}_c)-\text{kWh}(\mathcal{M}_e)}{\text{CER}(\mathcal{M}_e)-\text{CER}(\mathcal{M}_c)}$, avec la contrainte $\text{kWh}(\mathcal{M}_c) \geq \text{kWh}(\mathcal{M}_e)$. Puisque $\mathcal{M}_e$ est moins coûteux, le numérateur est positif. En présumant $\mathcal{M}_e$ moins performant, le dénominateur est aussi positif puisque le CER est plus faible pour un modèle plus performant. Dans les tableaux suivants, le *kWh/p* est donné par rapport au modèle le moins coûteux entraîné avec les mêmes *features* en entrée. Ce modèle est indiqué avec $\mathcal{M}_e$ dans les tableaux. Quand un modèle $\mathcal{M}_c$ est plus coûteux et moins performant que le modèle étalon $\mathcal{M}_e$, nous utilisons par convention la valeur ∞, indiquant qu'une stratégie d'apprentissage plus coûteuse ne donnerait aucun gain en performances.
L'interprétation de la valeur *kWh/p* doit être faite gardant en tête l'hypothèse selon laquelle un modèle moins coûteux est aussi moins performant. Alors, pour des features en entrée données, *kWh/p* mesure le coût additionnel en kWh pour améliorer hypothétiquement les résultats d'un point. Ce coût pourrait être dû à l'utilisation d'un modèle plus grand et/ou plus de données d'apprentissage.

3.2 Données

Si le nombre de corpus pour la SLU est important en langue anglaise, il existe un nombre relativement restreint de ce type de corpus pour le français. On peut citer MEDIA (Bonneau-Maynard *et al.*, 2006), PORTMEDIA (Lefèvre *et al.*, 2012), le HIS (Fleury *et al.*, 2013), Sweet-Home (Vacher *et al.*, 2014) Vocadom (Portet *et al.*, 2019), ou encore Voice-Home-2 (Bertin *et al.*, 2019). Nous utiliserons principalement le corpus MEDIA (Bonneau-Maynard *et al.*, 2006), dans le domaine de la réser-

2. https://codecarbon.io
3. Disponible sur https://www.eea.europa.eu/data-and-maps/indicators/overview-of-the-electricity-production-3/assessment.

	Train		Dev		Test	
Durée totale audio	27.92h		6.27h		5.79h	
dont utilisateur	8.49h		2.02h		1.79h	
# phrases	13 452		3 067		2 886	
dont utilisateur	6 495		1 485		1 391	
	Mots	Étiquettes	Mots	Étiquettes	Mots	Étiquettes
# tokens	212 301	24 065	47 101	5 410	44 850	4 956
dont utilisateur	45 710	17 064	10 465	3 814	9 898	3 451
dictionnaire	2 292	37	1 339	33	1 168	29
OOV%	-	-	0.30	0.0	0.34	0.0

TABLE 1 – Statistiques du corpus PortMEDIA

	Train		Dev		Test	
Durée totale audio	41.27h		3.60h		11.28h	
dont utilisateur	16.58h		1.63h		4.59h	
# phrases	26 966		2 662		6 789	
dont utilisateur	12 887		1 259		3 005	
	Mots	Étiquettes	Mots	Étiquettes	Mots	Étiquettes
# tokens	286 327	57 915	28 213	6 219	76 591	15 418
dont utilisateur	95 881	43 832	11 049	4 816	25 921	11 632
dictionnaire	2 785	71	1 032	59	1 310	67
OOV%	-	-	0.0	1,69	0.0	2.99

TABLE 2 – Statistiques du corpus MEDIA

vation hôtelière, que nous avons largement utilisé dans le passé (Quarteroni *et al.*, 2009; Dinarelli *et al.*, 2010; Dinarelli & Tellier, 2016; Dupont *et al.*, 2017; Dinarelli *et al.*, 2017). Les statistiques pour les partitions de données d'apprentissage (Train), de développement (Dev) et de test (Test) sont montrées dans le tableau 2. Ce corpus est constitué de 1 250 dialogues humain-machine acquis avec une approche par Magicien d'Oz, où 250 utilisateurs ont suivi 5 scénarios de réservation. Les signaux de parole ont été transcrits et annotés avec 76 concepts sémantiques. Le corpus est composé de 12 908 énoncés ($41,5\,h$) pour l'entraînement, 1 259 énoncés ($3,5\,h$) pour le développement et 3 005 énoncés ($11,3\,h$) pour le test. Les sessions de dialogue mettant en scène un utilisateur et un magicien, seuls les tours de parole des utilisateurs ont été annotés avec des concepts et peuvent être utilisés pour entraîner les modèles SLU. Dans nos expériences, nous avons constaté cependant qu'en utilisant aussi les tours de parole du magicien d'Oz pour la SLU les résultats s'améliorent. Pour ce faire, nous avons construit automatiquement le format d'annotation SLU pour les tours de parole du magicien en leur associant le concept conventionnel *MachineSemantic*. Pour que le modèle puisse alors distinguer entre les mots qui instancient de vrais concepts dans les tours utilisateur et les mêmes mots dans les tours du magicien, un marqueur d'*orateur* est ajouté dans les signaux audio en entrée. [4] Par contre, l'ensemble des tours de parole (magicien et utilisateur) ont été transcrits manuellement et peuvent donc être utilisés pour entraîner un modèle ASR. Dans le tableau 2, nous montrons ainsi à la fois les statistiques sur l'ensemble des données ainsi que les statistiques pour les tours de parole des utilisateurs seulement (indiqué avec *dont utilisateur* dans le tableau). Dans cet article, la tâche MEDIA est considérée comme tâche cible, c'est-à-dire la tâche sur laquelle nous souhaitons obtenir les meilleurs résultats possibles à moindre coût, en partant éventuellement de ressources déjà disponibles telles que des modèles SSL et des modèles SLU preappris sur d'autres tâches.
Afin de tester l'intérêt d'un transfert d'apprentissage et suivant (Caubrière *et al.*, 2019) nous avons également considéré le corpus PORTMEDIA (Lefèvre *et al.*, 2012) dédié à la réservation de billets pour le Festival d'Avignon 2010. Le corpus a été acquis et annoté en suivant le même paradigme

4. Ce marqueur est constitué d'un tenseur de taille 3 ajouté en tête et à la fin du signal original. Le tenseur contient uniquement des valeurs $+5.0$ pour l'utilisateur et uniquement des valeurs -5.0 pour le magicien.

que MEDIA afin de minimiser les différences entre les 2 corpus (hormis le domaine). Il est également divisé en trois parties : un ensemble d'entraînement contenant 5 900 énoncés, une partie développement contenant 1 400 énoncés, et un ensemble de test contenant 2 800 énoncés. Le corpus PORTMEDIA a été annoté manuellement avec 36 concepts sémantiques proches de l'ensemble de concepts MEDIA : PORTMEDIA et MEDIA partagent 26 concepts sémantiques communs. Les statistiques sur les partitions de données d'apprentissage (*Train*), de développement (*Dev*) et de test (*Test*) pour les corpus PortMEDIA sont montrées dans le tableau 1. Les considérations faites sur le corpus MEDIA concernant la répartition des données en tours de parole des magiciens d'Oz et des utilisateurs valent également pour PortMEDIA.

3.3 Résultats

Corpus : PortMEDIA, Métrique : taux d'erreur (CER)						
Stg appr.	**Entrée**	**KWh** (gCO2)	**kWh/p**	**T appr.**	**DEV**	**TEST**
Features de base						
3 steps	spectro	4,473 (228)	0,099	36h14'	35.91	40.57
2 steps	spectro	2,989 (152)	∞	24h14'	65.80	87.32
1 step	spectro	1,708 (87)	$\mathcal{M}_e$	15h52'	59.22	68.50
3 steps	w2v2-fr	3,983 (203)	2,235	36h22'	22.17	22.51
2 steps	w2v2-fr	2,707 (138)	1,939	24h27'	21.86	23.02
1 step	w2v2-fr	1,815 (93)	$\mathcal{M}_e$	18h08'	25.53	23.48
Features affinés (+100h x4 GPU)						
1 step +1	w2v2-fr slu	1,214 (62)	-	11h34'	21.50	22.13

TABLE 3 – Résultats sur le corpus PortMEDIA, pour tous les détails voir dans le texte.

3.3.1 Expériences préliminaires sur *PortMEDIA*

Afin obtenir des conditions favorables à l'entretient de modèles à moindre coût sur notre tâche cible (MEDIA), nous avons entraîné, en plus des modèles SSL pour le français (Evain *et al.*, 2021b), des modèles SLU sur la tâche PortMEDIA. Ces modèles seront ensuite utilisés pour préinitialiser les modèles pour la tâche MEDIA. Les résultats sont montrés dans le tableau 3. Pour avoir une vue globale, nous avons entraîné des modèles à la fois avec les *features* de base (spectrogrammes, indiqué avec *spectro*) et avec les features produits par le modèle SSL pour le français *w2v2-fr 7k* (Evain *et al.*, 2021b). Comme on peut le voir, les meilleurs résultats sont toujours obtenus avec la stratégie d'apprentissage la plus coûteuse *3 steps*. Cependant, en utilisant les features du modèle *w2v2-fr* la différence entre la stratégie *3 steps* et la stratégie *1 step* est de moins d'un point de CER, alors que cette dernière stratégie est bien moins coûteuse, à la fois en termes de temps d'apprentissage que de consommation énergétique. Des modèles encore plus performants en termes de CER peuvent être obtenus, et ce à un coût encore inférieur, avec des features extraits à partir du modèle *w2v2-fr* affiné sur la tâche SLU MEDIA, montrés dans la dernière ligne du tableau 3 (features *w2v2-fr slu*). Ces résultats sont accompagnés avec *1 step +1* pour prendre en compte l'affinage du modèle sur la tâche MEDIA, ce qui demande 100 heures d'apprentissage sur 4 GPUs (**+100h x4 GPU** dans les tableaux). Le coût de cet apprentissage domine le coût d'apprentissage du modèle SLU. Comme nous l'avions mentionné cependant, cet affinage est effectué sur la tâche MEDIA et une fois pour toutes. Nous avons choisi d'optimiser sur la tâche MEDIA seulement, à la fois pour réduire le coût

d'apprentissage total, et parce que le modèle SLU entraîné sur la tâche PortMEDIA est utilisé par la suite pour préinitialiser le modèle pour la tâche MEDIA. Comme nous l'avions anticipé donc, grâce à des modèles SSL pour le français, il est possible d'obtenir des performances compétitives sur la tâche SLU visée à un coût bien moindre (stratégie *3 steps* vs stratégie *1 step*). Logiquement les résultats sont encore meilleurs si on dispose d'un modèle SSL affiné.
Grâce à ces premières expériences, nous disposons de modèles SLU, en plus des modèles SSL, utilisables pour effectuer un apprentissage par transfert sur la tâche MEDIA. Le transfert est effectué en préinitialisant les modèles appris sur MEDIA avec un modèle appris sur PortMEDIA. Dans un contexte réel, il est souhaitable que ces ressources existent à l'avance, et qu'elles soient re-utilisées en exploitant le même système pour apprendre les modèles SLU pour la tâche visée, ce que nous faisons dans ce travail.

Corpus : MEDIA, Métrique : taux d'erreur (CER)						
Stg appr.	**Entrée**	**KWh** (gCO_2)	**kWh/p**	**T appr.**	**DEV**	**TEST**
Features de base						
(Evain *et al.*, 2021b) 3 steps	spectro	-	-	*57h*	**29.07**	**31.10**
3 steps	spectro	6,651 (314)	0,273	56h55'	**28.35**	**28.95**
2 steps	spectro	4,417 (225)	0.173	40h52'	32.04	32.85
1 step	spectro	2,407 (123)	$\mathcal{M}_e$	22h16'	46.57	44.50
(Evain *et al.*, 2021b) 3 steps	w2v2-fr	-	-	*36h*	**17.25**	**16.25**
3 steps	w2v2-fr	3.597 (183)	0,550	36h01'	**18.69**	**16.14**
2 steps	w2v2-fr	2.445 (125)	0,116	24h29'	18.24	16.23
1 step	w2v2-fr	2.150 (110)	$\mathcal{M}_e$	21h32'	19.68	18.77
Features affinés (+100h x4 GPU)						
2 steps +1	w2v2-fr slu	2.569 (131)	∞	27h28'	14.25	13.78
1 step +1	w2v2-fr slu	2.529 (129)	∞	27h02'	**14.16**	**13.26**
(Evain *et al.*, 2021b)$^{(*)}$ 3 steps +1	w2v2-fr asr	-	-	*36h*	**14.58**	**13.78**
Transfert						
1 step +PM	w2v2-fr	2.420 (123)	0,125	25h04'	18.27	16.61
Transfert + features affinés (+100h x4 GPU)						
1 step +1 +PM	w2v2-fr slu	2.026 (103)	$\mathcal{M}_e$	19h23'	**13.59**	**13.21**
État de l'art						
(Pelloin *et al.*, 2021)	MFCC	-	-	-	**16.1**	**13.6**
(Ghannay *et al.*, 2021)	w2v2-fr slu$^{(**)}$	-	-	-	-	**11.2**

TABLE 4 – Résultats sur le corpus MEDIA, pour tous les détails voir dans le texte. $^{(*)}$ le fine-tuning pour ces résultats était effectué pour l'ASR et non pas pour la SLU comme dans notre travail. $^{(**)}$ Features affinés en plusieurs étapes (ASR, SLU, modèle de langue) sur la tâche MEDIA.

3.3.2 Experiences à moindre coût sur *MEDIA*

Les résultats sur la tâche MEDIA sont montrés dans le tableau 4. Dans le bloc **Features de base** sont montrés les résultats obtenus dans les mêmes conditions expérimentales que celles utilisées pour la tâche PortMEDIA. Ces résultats confirment qu'un modèle SLU compétitif peut être obtenu à un moindre coût (*3 steps* vs *1 step* avec features *w2v2-fr*). Le bloc **Features affinés** montre les résultats obtenus avec des features extraits à partir du modèle SSL w2v2-fr affiné sur la tâche SLU de MEDIA (w2v2-fr slu). Il est intéressant de noter que le modèle appris complètement de bout en bout (*1 step +1*) obtient de meilleurs résultats que le modèle appris en 2 étapes (*2 steps +1*). Ceci grâce au fait que le modèle appris de bout en bout peut bénéficier d'un apprentissage plus *agressif*, notamment une régularisation plus faible. Ces réglages ne sont pas efficaces avec une stratégie en 2 étapes, intuitivement parce qu'ils vont "effacer" l'information fournie par le modèle utilisé pour la préinitialisation, éloignant le modèle de l'optimum. Puisque le modèle SSL est affiné sur la même tâche que le modèle SLU final, il n'est pas étonnant que ce dernier obtienne des résultats très com-

pétitifs avec un coût d'apprentissage moindre. En effet, comparés aux derniers modèles à l'état de l'art sur la tâche MEDIA (bloc **État de l'art** dans le tableau), nos résultats sont meilleurs que ceux de (Pelloin *et al.*, 2021), et assez proche de (Ghannay *et al.*, 2021), alors qu'ils sont obtenus avec un coût bien moindre par rapport à ces travaux. (Pelloin *et al.*, 2021) et (Ghannay *et al.*, 2021) ne mentionne pas le coût computationnel de leur modèle, mais de ce qui est reporté dans leurs travaux il est possible d'estimer un coût supérieur à celui de l'affinage du modèle SSL que nous avons effectué ((Ghannay *et al.*, 2021) notamment effectue plusieurs de ces affinages).
Puisque les résultats commentés jusque là confirment que les modèles SSL permettent d'atteindre des performances très compétitives même avec un apprentissage de bout en bout (*1 step*), pour les expériences suivantes nous avons utilisé uniquement cette stratégie d'apprentissage moins coûteuse.
Dans les blocs **Transfert** et **Transfert + features affinés** du tableau 4 nous montrons respectivement les résultats obtenus avec apprentissage par transfert de la tâche PortMEDIA, et apprentissage par transfert de la même tâche en utilisant des features extraits avec le modèle SSL w2v2-fr affiné. Avec apprentissage par transfert seul (**Transfert**), utilisant un modèle entraîné sur PortMEDIA comme point de départ pour le modèle MEDIA (**+PM**), les résultats s'améliorent remarquablement sur les données de test (18.77 vs 16.61) sans aucun coût additionnel (en présumant qu'un modèle pour la tâche PortMEDIA soit disponible). La valeur *kWh/p* des modèles utilisant les features *w2v2-fr* est calculée par rapport à ce modèle (le plus économe avec cette entrée).
Avec apprentissage par transfert et features affinés pour la SLU (**Transfert + features affinés**), alors qu'il y a une amélioration sur les données de développement (14.16 vs 13.59), il n'y en a pratiquement pas sur les données de test (13.26 vs 13.21). Nous considérons que cela est dû au fait que le modèle SSL étant déjà affiné sur la tâche SLU, le petit apport des données PortMEDIA (ce corpus est même plus petit que MEDIA) à travers l'utilisation d'un modèle appris sur cette tâche n'ajoute pas plus d'information que celle déjà fournie par les features w2v2-fr affinées. Ce modèle a tout de même l'avantage d'être plus économe (19h23' vs 27h02'), toujours dans la perspective d'avoir un modèle SLU PortMEDIA déjà disponible en avance. La valeur *kWh/p* pour les modèles utilisant des features *w2v2-fr slu* est calculée par rapport à ce dernier modèle.

4 Conclusions

Dans cet article, nous avons analysé des stratégies d'apprentissage pour des modèles SLU sur la tâche MEDIA, visant à diminuer le coût computationnel de l'apprentissage des modèles tout en gardant des performances compétitives. Nos résultats montrent que, en passant par l'utilisation de modèles SSL pour le français, il est possible d'atteindre ces objectifs. Avec le coût additionnel de l'affinage d'un modèle SSL sur la tâche SLU, nous obtenons le deuxième meilleur résultat de la littérature sur MEDIA, et ce avec un entraînement complètement de bout en bout du modèle SLU. Bien que l'affinage soit relativement coûteux, notre modèle est bien moins gourmand en ressources que les meilleurs modèles de l'état de l'art. Afin d'avoir une vision plus complète de l'impact énergique des modèles utilisés dans le TALN sur une tâche donnée, il serait souhaitable que la communauté adopte un standard d'évaluation des modèles d'un point de vue de leur consommation énergétique, notamment vis-à-vis de l'utilisation de plus en plus fréquente de modèles de plus en plus massifs et énergivores.

Remerciements

Ce travail a été effectué en utilisant les ressources de calcul HPC de GENCI - IDRIS, numéro de contrat AD011011615R1.
Ce travail a été supporté partiellement par le projet JCJC CREMA (*Coreference REsolution into MAchine translation*) financé par l'Agence Nationale de la Recherche (ANR), numéro de contrat ANR-21-CE23-0021-01.

Références

BAEVSKI A., ZHOU Y., MOHAMED A. & AULI M. (2020). wav2vec 2.0 : A framework for self-supervised learning of speech representations. In H. LAROCHELLE, M. RANZATO, R. HADSELL, M. F. BALCAN & H. LIN, Éds., *Advances in Neural Information Processing Systems*, volume 33, p. 12449–12460 : Curran Associates, Inc.

BAHDANAU D., CHO K. & BENGIO Y. (2015). Neural machine translation by jointly learning to align and translate. In Y. BENGIO & Y. LECUN, Éds., *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

BERTIN N., CAMBERLEIN E., LEBARBENCHON R., VINCENT E., SIVASANKARAN S., ILLINA I. & BIMBOT F. (2019). Voicehome-2, an extended corpus for multichannel speech processing in real homes. *Speech Commun.*, **106**, 68–78.

BONNEAU-MAYNARD H., AYACHE C., BECHET F., DENIS A., KUHN A., LEFEVRE F., MOSTEFA D., QUIGNARD M., ROSSET S., SERVAN C. & VILLANEAU J. (2006). Results of the French evalda-media evaluation campaign for literal understanding. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy : European Language Resources Association (ELRA).

CAUBRIÈRE A., GHANNAY S., TOMASHENKO N., DE MORI R., LAURENT A., MORIN E. & ESTÈVE Y. (2020). Error analysis applied to end-to end spoken language understanding. In *45th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Barcelona, Spain. HAL : hal-02465899.

CAUBRIÈRE A., TOMASHENKO N., LAURENT A., MORIN E., CAMELIN N. & ESTÈVE Y. (2019). Curriculum-based transfer learning for an effective end-to-end spoken language understanding and domain portability. In *20th Annual Conference of the International Speech Communication Association (InterSpeech)*, p. 1198–1202, Graz, Austria. DOI : 10.21437/interspeech.2019-1832, HAL : hal-02304597.

CAUBRIÈRE A., TOMASHENKO N. A., LAURENT A., MORIN E., CAMELIN N. & ESTÈVE Y. (2019). Curriculum-based transfer learning for an effective end-to-end spoken language understanding and domain portability. *CoRR*, **abs/1906.07601**.

CHAN W., JAITLY N., LE Q. V. & VINYALS O. (2015). Listen, attend and spell. *CoRR*, **abs/1508.01211**.

DESOT T., PORTET F. & VACHER M. (2019). SLU FOR VOICE COMMAND IN SMART HOME : COMPARISON OF PIPELINE AND END-TO-END APPROACHES. In *IEEE Automatic Speech Recognition and Understanding Workshop*, Sentosa, Singapore, Singapore. HAL : hal-02464393.

DINARELLI M. (2010). *Spoken Language Understanding : from Spoken Utterances to Semantic Structures*. Thèse de doctorat, International Doctoral School in Information and Communication Technology, Dipartimento di Ingegneria e Scienza dell' Informazione, via Sommarive 14, 38100 Povo di Trento (TN), Italy.

DINARELLI M., KAPOOR N., JABAIAN B. & BESACIER L. (2020). A data efficient end-to-end spoken language understanding architecture.

DINARELLI M., MOSCHITTI A. & RICCARDI G. (2009a). Concept segmentation and labeling for conversational speech. In *Interspeech*, Brighton, U.K.

DINARELLI M., MOSCHITTI A. & RICCARDI G. (2009b). Re-ranking models based on small training data for spoken language understanding. In *Conference of Empirical Methods for Natural Language Processing*, p. 1076–1085, Singapore,Singapore.

DINARELLI M., STEPANOV E., VARGES S. & RICCARDI G. (2010). The luna spoken dialog system : Beyond utterance classification. In *International Conference on Acoustic, Speech and Signal Processing*, Dallas, Texeas, U.S.A.

DINARELLI M. & TELLIER I. (2016). Improving recurrent neural networks for sequence labelling. *CoRR*, **abs/1606.02555**.

DINARELLI M., VUKOTIC V. & RAYMOND C. (2017). Label-dependency coding in Simple Recurrent Networks for Spoken Language Understanding. In *Interspeech*, Stockholm, Sweden. HAL : hal-01553830.

DUPONT Y., DINARELLI M. & TELLIER I. (2017). Label-dependencies aware recurrent neural networks. In *Proceedings of the 18th International Conference on Computational Linguistics and Intelligent Text Processing*, Budapest, Hungary : Lecture Notes in Computer Science (Springer).

EVAIN S., NGUYEN H., LE H., ZANON BOITO M., MDHAFFAR S., ALISAMIR S., TONG Z., TOMASHENKO N., DINARELLI M., PARCOLLET T., ALLAUZEN A., ESTÈVE Y., LECOUTEUX B., PORTET F., ROSSATO S., RINGEVAL F., SCHWAB D. & BESACIER L. (2021a). LeBenchmark : A Reproducible Framework for Assessing Self-Supervised Representation Learning from Speech. In *INTERSPEECH 2021 : Conference of the International Speech Communication Association*, Brno, Czech Republic. HAL : hal-03317730.

EVAIN S., NGUYEN M. H., LE H., ZANON BOITO M., MDHAFFAR S., ALISAMIR S., TONG Z., TOMASHENKO N., DINARELLI M., PARCOLLET T., ALLAUZEN A., ESTÈVE Y., LECOUTEUX B., PORTET F., ROSSATO S., RINGEVAL F., SCHWAB D. & BESACIER L. (2021b). Task Agnostic and Task Specific Self-Supervised Learning from Speech with LeBenchmark. In *Thirty-fifth Conference on Neural Information Processing Systems (NeurIPS 2021)*, NeurIPS 2021 Datasets and Benchmarks Track, on-line, United States. HAL : hal-03407172.

FLEURY A., VACHER M., PORTET F., CHAHUARA P. & NOURY N. (2013). A french corpus of audio and multimodal interactions in a health smart home. *Journal on Multimodal User Interfaces*, **7**(1), 93–109.

GHANNAY S., CAUBRIÈRE A., MDHAFFAR S., LAPERRIÈRE G., JABAIAN B. & ESTÈVE Y. (2021). Where are we in semantic concept extraction for Spoken Language Understanding ? ⋆. In *SPECOM 2021 23rd International Conference on Speech and Computer*, Saint Petersburg, Russia. HAL : hal-03372494.

GRAVES A., FERNÁNDEZ S., GOMEZ F. & SCHMIDHUBER J. (2006). Connectionist temporal classification : Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of ICML*, p. 369–376 : ACM. DOI : 10.1145/1143844.1143891.

HAHN S., DINARELLI M., RAYMOND C., LEFÈVRE F., LEHEN P., DE MORI R., MOSCHITTI A., NEY H. & RICCARDI G. (2010). Comparing stochastic approaches to spoken language understanding in multiple languages. *IEEE TASLP*, **99**.

HOCHREITER S. & SCHMIDHUBER J. (1997). Long short-term memory. *Neural Comput.*, **9**(8).

HSU W., BOLTE B., TSAI Y. H., LAKHOTIA K., SALAKHUTDINOV R. & MOHAMED A. (2021). Hubert : Self-supervised speech representation learning by masked prediction of hidden units. *CoRR*, **abs/2106.07447**.

LAI C., CHUANG Y., LEE H., LI S. & GLASS J. R. (2020). Semi-supervised spoken language understanding via self-supervised speech and language model pretraining. *CoRR*, **abs/2010.13826**.

LEFÈVRE F., MOSTEFA D., BESACIER L., ESTÈVE Y., QUIGNARD M., CAMELIN N., FAVRE B., JABAIAN B. & ROJAS-BARAHONA L. M. (2012). Leveraging study of robustness and portability of spoken language understanding systems across languages and domains : the PORTMEDIA corpora. In *LREC*, p. 1436–1442.

LUGOSCH L., RAVANELLI M., IGNOTO P., TOMAR V. S. & BENGIO Y. (2019). Speech model pre-training for end-to-end spoken language understanding.

MORI R. D. (1997). *Spoken Dialogues with Computers*. Orlando, FL, USA : Academic Press, Inc.

PALOGIANNIDI E., GKINIS I., MASTRAPAS G., MIZERA P. & STAFYLAKIS T. (2020). End-to-end architectures for asr-free spoken language understanding.

PARCOLLET T. & RAVANELLI M. (2021). The Energy and Carbon Footprint of Training End-to-End Speech Recognizers. working paper or preprint.

PELLOIN V., CAMELIN N., LAURENT A., DE MORI R., CAUBRIÈRE A., ESTÈVE Y. & MEIGNIER S. (2021). End2End Acoustic to Semantic Transduction. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, ON, Canada. DOI : 10.1109/ICASSP39728.2021.9413581, HAL : hal-03128163.

PORTET F., CAFFIAU S., RINGEVAL F., VACHER M., BONNEFOND N., ROSSATO S., LECOUTEUX B. & DESOT T. (2019). Context-Aware Voice-based Interaction in Smart Home - VocADom@A4H Corpus Collection and Empirical Assessment of its Usefulness. In *PICom 2019 - 17th IEEE International Conference on Pervasive Intelligence and Computing*, p. 811–818, Fukuoka, Japan.

QIAN Y., UBALE R., RAMANARYANAN V., LANGE P., SUENDERMANN-OEFT D., EVANINI K. & TSUPRUN E. (2017). Exploring asr-free end-to-end modeling to improve spoken language understanding in a cloud-based dialog system. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, p. 569–576. DOI : 10.1109/ASRU.2017.8268987.

QUARTERONI S., RICCARDI G. & DINARELLI M. (2009). What's in an ontology for spoken language understanding. In *Interspeech*, Brighton, U.K.

RADFAR M., MOUCHTARIS A. & KUNZMANN S. (2020). End-to-end neural transformer based spoken language understanding.

RAYMOND C., BÉCHET F., DE MORI R. & DAMNATI G. (2006). On the use of finite state transducers for semantic interpretation. *Speech Communication*, **48**(3-4), 288–304. DOI : 10.1016/j.specom.2005.06.012.

SCHNEIDER S., BAEVSKI A., COLLOBERT R. & AULI M. (2019). wav2vec : Unsupervised pre-training for speech recognition. *CoRR*, **abs/1904.05862**.

SERDYUK D., WANG Y., FUEGEN C., KUMAR A., LIU B. & BENGIO Y. (2018a). Towards end-to-end spoken language understanding. *CoRR*, **abs/1802.08395**.

SERDYUK D., WANG Y., FUEGEN C., KUMAR A., LIU B. & BENGIO Y. (2018b). Towards end-to-end spoken language understanding.

VACHER M., LECOUTEUX B., CHAHUARA P., PORTET F., MEILLON B. & BONNEFOND N. (2014). The Sweet-Home speech and multimodal corpus for home automation interaction. In *The 9th edition of the Language Resources and Evaluation Conference (LREC)*, p. 4499–4506, Reykjavik, Iceland.

VASWANI A., SHAZEER N., PARMAR N., USZKOREIT J., JONES L., GOMEZ A. N., KAISER L. U. & POLOSUKHIN I. (2017). Attention is all you need. In I. GUYON, U. V. LUXBURG, S. BENGIO, H. WALLACH, R. FERGUS, S. VISHWANATHAN & R. GARNETT, Éds., *Advances in Neural Information Processing Systems*, volume 30 : Curran Associates, Inc.

YANG S., CHI P., CHUANG Y., LAI C. J., LAKHOTIA K., LIN Y. Y., LIU A. T., SHI J., CHANG X., LIN G., HUANG T., TSENG W., LEE K., LIU D., HUANG Z., DONG S., LI S., WATANABE S., MOHAMED A. & LEE H. (2021). SUPERB : speech processing universal performance benchmark. *CoRR*, **abs/2105.01051**.