Mise en place d'un modèle compact à architecture Transformer pour la détection jointe des intentions et des concepts dans le cadre d'un système interactif de questions-réponses

Nadège Alavoine[1, 2] Arthur Babin[1, 3]
(1) Université Paris-Saclay, LISN, Campus Universitaire bâtiment 507, Rue du Belvédère, 91400 Orsay, France
(2) École 42, 96 boulevard Bessières, 75017 Paris, France
(3) ENSIIE, 1 square de la Résistance, 91000 Évry-Courcouronnes
prénom.nom@lisn.fr, prénom.nom@ensiie.fr

RÉSUMÉ

Les tâches de détection d'intention et d'identification des concepts sont toutes deux des éléments importants de la compréhension du langage naturel. Elles sont souvent réalisées par deux modules différents au sein d'un *pipeline*. L'apparition de modèles réalisant conjointement ces deux tâches a permis d'exploiter les dépendances entre elles et d'améliorer les performances obtenues. Plus récemment, des modèles de détection jointe reposant sur des architectures *Transformer* ont été décrits dans la littérature. Par ailleurs, avec la popularité et taille croissante des modèles *Transformer* ainsi que les inquiétudes ergonomiques et écologiques grandissantes, des modèles compacts ont été proposés. Dans cet article, nous présentons la mise en place et l'évaluation d'un modèle compact pour la détection jointe de l'intention et des concepts. Notre contexte applicatif est celui d'un système interactif de questions-réponses français.

ABSTRACT

Implementation of a light model with Transformer architecture for joint intent classification and slot filling aimed at an interactive question-answers system

Intent classification and slot filling are important tasks of Natural Language Understanding. They're usually performed by two distinct modules inserted in one pipeline. Models conducting both tasks emerged in literature and improved previous performances by exploiting the dependencies between them. More recently, models performing intent detection and slot filling based on *Transformer*'s architecture were described. On another note, with the growing popularity and size of *Transformer*'s models as well as increasing ecological and ergonomic concerns, light versions were proposed. This article presents the implementation and evaluation of a joint detection light model for a French interactive question answering system.

MOTS-CLÉS : Détection jointe, Détection d'intention, Identification de concepts, Transformer, BERT joint, Classifieur DIET, CamemBERT, FrALBERT.

KEYWORDS: Joint detection, Intent classification, Slot filling, Transformer, Joint BERT, DIET classifier, CamemBERT, FrALBERT.

1 Introduction

La compréhension du langage naturel ou NLU (*natural language understanding)* est un domaine classique du traitement de la parole transcrite ainsi qu'un élément essentiel aux systèmes de dialogues. Son but est d'extraire les concepts sémantiques du discours. Elle consiste notamment en la détection de l'intention et l'identification des concepts d'une phrase (ou tâche de *slot filling*). Ces deux tâches sont illustrées en Tableau 1 avec l'exemple *"Trouve les horaires de la bibliothèque municipale demain"*. Dans cet exemple, l'intention est d'obtenir les horaires d'ouverture d'un lieu précis à une date précise : deux concepts devant être identifiés pour formuler la réponse attendue.

Mots	trouve	les	horaires	de	la	bibliothèque	municipale	demain
C	O	O	O	O	O	B-lieu	I-lieu	B-date
I	obtenir_les_horaires							

FIGURE 1 – Exemple de discours avec étiquettes d'intention (**I**) et de concepts (**C**). Les concepts sont étiquetés selon la norme BIO.

Ces deux tâches sont fréquemment réalisées par des modules indépendants insérés dans un même *pipeline* et ne partageant pas directement d'informations entre-eux (Hakkani-Tür *et al.*, 2016; Goo *et al.*, 2018). En conséquence, le *pipeline* peut souffrir de propagation et d'accumulation d'erreurs. Depuis une quinzaine d'années (Weld *et al.*, 2022), des modèles réalisant conjointement ces tâches ont été proposés. Ils reposent sur différentes stratégies impliquant notamment des champs aléatoires conditionnels (Jeong & Lee, 2008), des réseaux neuronaux convolutifs (Xu & Sarikaya, 2013), des réseaux neuronaux récurrents (Guo *et al.*, 2014; Hakkani-Tür *et al.*, 2016; Liu & Lane, 2016), des modèles avec *slot-gate* (Goo *et al.*, 2018) ou des mécanismes d'attention (Chen *et al.*, 2016; Liu & Lane, 2016). Cette détection jointe permet d'exploiter les dépendances entre les deux tâches et d'améliorer les performances obtenues. Certains de ces travaux, réalisés principalement sur des données en langue anglaise, ont pu être adaptés pour d'autres langues en conservant les mêmes architectures (Weld *et al.*, 2022).

Plus récemment, l'apparition des modèles *Transformer* (Vaswani *et al.*, 2017) utilisant des mécanismes d'attention a permis de réaliser de nombreux progrès dans le domaine du Traitement Automatique des Langues (TAL). Des modèles pré-entraînés reposant sur des variations de cette architecture, tel que le modèle BERT (Devlin *et al.*, 2019), ont permis d'atteindre de nouveaux états de l'art pour de multiples tâches du TAL. Des modèles s'appuyant sur ces architectures *Transformer* pour réaliser une détection jointe de l'intention et des concepts ont été proposés dans la littérature (Chen *et al.*, 2019; Castellucci *et al.*, 2019; Bunk *et al.*, 2020). Ces modèles permettent d'obtenir de meilleures performances que les précédents modèles présentés pour la détection jointe.

Parallèlement, une tendance à la création de modèles de langues de tailles croissantes, en termes de données et de nombre de paramètres, est constatée. Si cette augmentation permet d'obtenir des performances grandissantes, elle s'associe à un coût financier ainsi qu'écologique (Strubell *et al.*, 2019; Moosavi *et al.*, 2020; Bender *et al.*, 2021). De plus, de trop grands modèles ne sont pas toujours utilisables en fonction de limites matérielles. Afin de répondre à ces problématiques, des modèles de langues compacts aux performances similaires à ceux de plus grandes tailles sont proposés (Cattan *et al.*, 2022). C'est notamment le cas d'ALBERT (Lan *et al.*, 2020), version allégée de BERT, et de son équivalent pré-entraîné sur un corpus français FrALBERT (Cattan *et al.*, 2021). Tous deux sont

optimisés grâce à des méthodes de réduction et de partage des poids.

Nos contributions principales sont l'élaboration d'un corpus en langue française de questions destinées à un système de dialogue fournissant des renseignements généraux sur des bibliothèques universitaires, son annotation en intentions et concepts pour des tâches de NLU, la mise en place d'un modèle compact réalisant une détection jointe des intentions et concepts évalué sur ce corpus et sa comparaison à un modèle de plus grande taille. Le contexte applicatif est décrit en section 2. Les modèles présentés reposent sur des architectures existantes, présentées en section 3, avec l'utilisation de modèles pré-entraînés sur des corpus français. Les données utilisées, les optimisations des modèles et les protocoles expérimentaux, ainsi que les résultats et leurs analyses sont exposés en section 4.

2 Contexte

Notre problématique initiale concerne la création d'un système de compréhension du langage lorsque aucune donnée n'est disponible. Notre contexte applicatif est le projet DIBISO. Ce projet est issu d'une collaboration entre la DIBISO (DIRECTION DES BIBLIOTHÈQUES, DE L'INFORMATION ET DE LA SCIENCE OUVERTE) et le LISN (LABORATOIRE INTERDISCIPLINAIRE DES SCIENCES DU NUMÉRIQUE). Il consiste en l'élaboration d'un système interactif de questions-réponses (SQR) destiné à fournir des renseignements relatifs aux BIBLIOTHÈQUES UNIVERSITAIRES PARIS-SACLAY. Le LISN et la DIBISO étant engagés sur le plan écologique et l'impact de l'intelligence artificielle sur l'environnement n'étant pas négligeable (Strubell *et al.*, 2019; Bender *et al.*, 2021), il fut décidé que les solutions adoptées pour ce projet s'inscriraient dans une utilisation raisonnée des ressources.

Une première étude de faisabilité révéla l'absence de corpus disponible répondant précisément à ce besoin. Nous avons réalisé une campagne de collecte de données annotées. Dans ce but, un prototype fut créé et mis à disposition d'une vingtaine d'annotateurs sélectionnés par la DIBISO. Les données utilisées pour la conception de ce prototype ont été fournies par la plateforme UBIB [1], permettant de mettre en relation les usagers de certaines bibliothèques universitaires avec des bibliothécaires à travers une interface de messagerie instantanée. Afin d'en permettre un déploiement rapide, la partie agent de dialogue de ce prototype fut construite avec la librairie RASA [2] (Bocklisch *et al.*, 2017) destinée à la mise en place aisée de systèmes conversationnels (*chatbot*). Les données initiales furent découpées en corpus d'entraînement et d'évaluation puis augmentées par des données générées. Techniquement, le prototype réalise d'abord une identification jointe de l'intention et des concepts des questions des utilisateurs à travers un premier modèle reposant sur un *Transformer*. Ce modèle repose sur une architecture DIET (Bunk *et al.*, 2020). Selon l'intention détectée, la réponse fournie est soit générique, soit générée en fonction des concepts identifiés ou fait appel à d'autres modèles pour identifier le passage le plus pertinent dans une base de données issues des pages internet des bibliothèques universitaires. Grâce à ce prototype, 471 questions ont été récupérées. L'analyse des erreurs obtenues nous laisse suggérer qu'une amélioration du système d'identification jointe de l'intention et des concepts est nécessaire pour permettre au SQR de fournir des réponses plus pertinentes. Par ailleurs, bien que modulable, la librairie RASA présente un effet « boîte noire » limitant les manipulations possibles.

Dans l'optique de construire un nouveau SQR plus manipulable et permettant toujours une détection jointe de l'intention et des concepts d'une question, un modèle reposant sur une architecture jointe BERT (Chen *et al.*, 2019) est envisagé.

1. https://ubib.fr
2. https://rasa.com

3 Établissement de modèles compactes pour la détection jointe

Dans cette section, nous aborderons l'architecture et les modalités de mise en place du modèle utilisé dans le prototype, ainsi que du modèle visant à le remplacer.

3.1 Préliminaire : Détection jointe à l'aide du classifieur DIET

Le classifieur DIET (pour *Dual Intent and Entity Transformer*) est une architecture *Transformer* légère adaptée à la compréhension du langage. Ce classifieur, présenté par l'équipe de recherche de la plateforme RASA (Bunk *et al.*, 2020) et intégré à leur librairie, réalise la détection jointe de l'intention et des concepts d'une phrase. Son architecture, telle que décrite dans l'article de Bunk *et al.* (2020), est présentée ci-dessous.

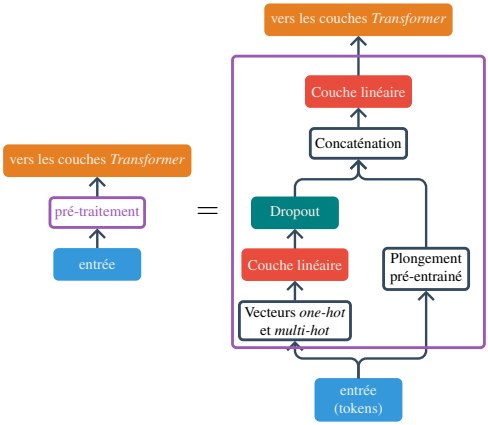


FIGURE 2 – Description du pré-traitement des tokens dans le modèle DIET

Les données sont d'abord étiquetées selon le format BILOU (Ramshaw & Marcus, 1995). Ainsi, le premier mot constitutif d'un concept est identifié par B (pour *Beginning*), les suivants par I (pour *Inside*) et le dernier par L (pour *Last*), suivis de l'étiquette associée. Les mots non constitutifs d'un concept sont identifiés par O (pour *Outside*). Les concepts d'une seule unité sont aussi identifiés, par la lettre U (pour *Unit-length*). Cet étiquetage est plus complet que BIO, plus fréquemment utilisé, où la première lettre d'un concept sont identifiées par B et les suivantes par I. Un pré-traitement des données, représenté en Figure 2, est ensuite réalisé : les phrases présentées en entrée au modèle sont transformées en représentations denses et éparses. Les premières sont obtenues à partir de plongements lexicaux pré-entraînés tels que ceux issus de la dernière couche de BERT (Devlin *et al.*, 2019). Les secondes sont obtenues par un processus de tokenisation suivi d'un encodage vectoriel *one-hot* et *multi-hot* de n-grammes de caractères (avec $n \leq 5$). Les informations des représentations éparses pouvant être redondantes, un *dropout* leur est appliqué pour éviter un phénomène de sur-apprentissage. Les représentations éparses sont alignées aux dimensions des denses par une couche entièrement connectée. Un token **[CLS]** représentant la classification d'une phrase (Devlin *et al.*, 2019) est ajouté à la fin de chaque entrée. Les différentes représentations sont ensuite concaténées. Ces représentations peuvent être affinées par apprentissage du modèle, ou être gelées. Si dans l'article de Bunk *et al.* (2020), un système de masquage des tokens associé à sa fonction de coût est décrit, les résultats de leur étude d'ablation montrent que ce système peut légèrement diminuer les performances des modèles. Le calcul de ce coût est donc inactivé par défaut dans l'outil proposé par RASA.

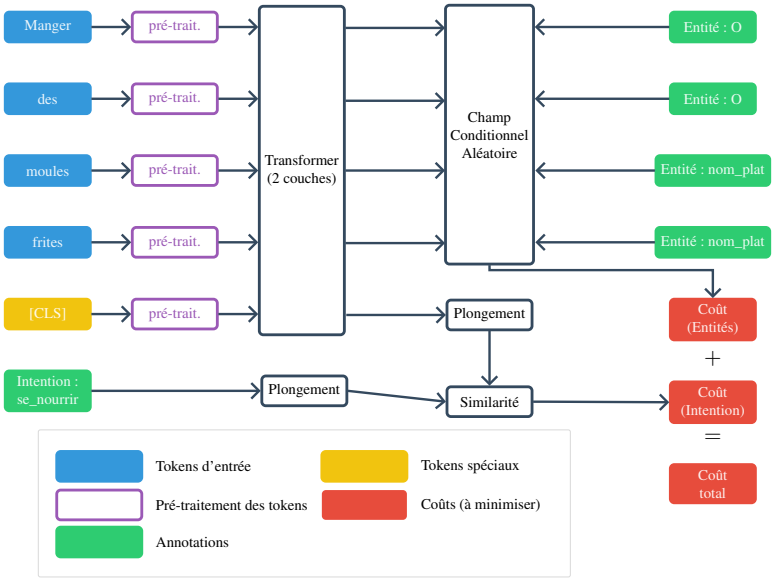


FIGURE 3 – Schéma de l'architecture du classifieur DIET d'après Bunk *et al.* (2020). La version présentée ne comporte pas le masquage des mots, inactive par défaut dans la librairie RASA.

Par la suite, et comme représenté en Figure 3, ces représentations sont présentées à deux couches successives d'un *Transformer* (Vaswani *et al.*, 2017) avec un système d'attention aux positions relatives (Shaw *et al.*, 2018). Afin que les dimensions des représentations soient adaptées à celle du *Transformer*, un passage par une couche entièrement connectée est réalisé au préalable. La prédiction des concepts de la phrase est réalisée à partir des sorties de la seconde couche *Transformer*, passées à une couche d'étiquetage à champ aléatoire conditionnel (Lafferty *et al.*, 2001). Un coût lié aux entités est calculé par log-vraisemblance négative entre la séquence prédite et celle attendue (Lample *et al.*, 2016). La sortie de la seconde couche *Transformer* pour le token **[CLS]** et les étiquettes d'intentions sont intégrées à un unique espace vectoriel sémantique. Un coût par produit scalaire de similarité est calculé (Wu *et al.*, 2018; Henderson *et al.*, 2019; Valsov *et al.*, 2019). Le calcul de ce coût vise à maximiser les similarités entre les prédictions issues de l'espace vectoriel sémantique pour la totalité des tokens **[CLS]** et leur étiquette positive (cible), ainsi qu'à minimiser celles avec les étiquettes négatives (non-cibles). Pour l'inférence, un calcul de similarité par produit scalaire permet d'ordonner les différentes intentions possibles. Lors de l'entraînement d'un classifieur DIET, c'est le coût total qui est minimisé. Ce coût est la somme de ceux calculés pour l'identification des concepts et pour la détection d'intention. Afin de réduire l'effet de déséquilibre des classes (Japkowicz & Stephen, 2002), les lots d'entraînement de données sont arrangés selon une stratégie d'équilibrage (Valsov *et al.*, 2019). La taille de ces lots augmente avec les itérations de l'entraînement dans une optique de régularisation (Smith *et al.*, 2018).

Lors de la conception du prototype, le choix s'est porté sur ce classifieur en raison de son intégration à la librairie RASA. Cette librairie avait été choisie en fonction de contraintes techniques. L'aspect modulaire de RASA a permis d'y intégrer le modèle FrALBERT$_{base}$ (Cattan *et al.*, 2021), non proposé initialement par la librairie. FrALBERT est utilisé dans ce classifieur pour l'obtention des plongements lexicaux et en tant que couches *Transformer*. Seules ces représentations par plongements lexicaux sont utilisées dans notre configuration du classifieur DIET, sans concaténation avec des représentations éparses. Le modèle résultant sera désigné comme DIET FrALBERT.

Concernant FrALBERT, il s'agit d'un modèle compact de langage basé sur l'architecture *Transformer*, pré-entraîné sur un corpus français. Comme sa version anglaise, ALBERT (Lan *et al.*, 2020), FrALBERT utilise des méthodes de partage et de réduction des paramètres permettant de réduire sa complexité et d'accélérer ses phases d'entraînement comme d'inférence. FrALBERT_{base} a été entraîné sur la version française du corpus de l'encyclopédie Wikipédia comprenant 4 gigabytes de texte (abrégé *wiki-4GB*).

RASA offre la possibilité d'utiliser un autre modèle pré-entraîné sur un corpus français et basé sur une architecture *Transformer* : le modèle CamemBERT (Martin *et al.*, 2020) dont l'architecture se rapproche de RoBERTa (Liu *et al.*, 2020). Le modèle proposé par défaut dans la librairie est celui pré-entraîné sur le corpus français OSCAR (pour *Open Super-large Crawled ALMAnaCH coRpus*) (Ortiz Suárez *et al.*, 2020). La mise en place d'un classifieur DIET utilisant $\text{CamemBERT}_{\text{base,wiki}-4\text{GB}}$ avait été réalisée pour comparer les performances de DIET FrALBERT à celle d'un modèle de plus grande taille et entraîné sur le même corpus. Malheureusement, le format **.h5** attendu par la librairie RASA étant indisponible pour cette version de CamemBERT, les résultats du modèle DIET $\text{CamemBERT}_{\text{base,wiki}-4\text{GB}}$ obtenus étaient aberrants. Plus précisement, les résultats étaient inférieurs de plus d'une vingtaine de points à ceux obtenus avec FrALBERT sur les différentes métriques suivies et dans les mêmes conditions d'entraînement.

Le Tableau 1 résume les principales caractéristiques de taille des modèles FrALBERT_{base} et CamemBERT_{base}. Rappelons que le choix de l'utilisation de FrALBERT est motivé par des raisons écologiques et énergétiques. Son coût de calcul (en temps, énergie et CO_2 produit) étant moins élevé que celui de CamemBERT de par sa taille réduite et ses méthodes d'optimisation (Cattan *et al.*, 2022).

Modèle	Nombre de paramètres	Taille du modèle
$\textbf{FrALBERT}_{\text{base}}$	12 millions	50 MB
$\textbf{CamemBERT}_{\text{base}}$	110 millions	445 MB

TABLE 1 – Caractéristiques techniques relatives à la taille des modèles utilisés. Les tailles sont exprimées en mégabytes (MB).

Concernant l'usage des données, l'outil RASA ne nécessite qu'un lot d'entraînement, qu'il sépare en lot d'entraînement et de validation, pour entraîner le modèle. Deux lots, d'entraînement et d'évaluation, avaient donc été constitués depuis les données issues de la plateforme UBIB. La librairie RASA étant avant tout destinée à des fins de production, des « gardes-fou » sont présents et limitent son utilisation. À titre d'exemple, il est impossible de réaliser de l'apprentissage par transfert depuis un modèle préalablement entraîné avec RASA. Ces limites nous ont poussées à chercher un nouveau système permettant toujours cette détection jointe de l'intention et des concepts dans un contexte plus expérimental. Le modèle DIET FrALBERT servira de *baseline* à nos expériences.

3.2 Mise en place d'un modèle à architecture jointe BERT

L'architecture jointe BERT pour la classification d'intention et la détection des concepts (désigné BERT joint) (Chen *et al.*, 2019) est une version modifiée du modèle BERT (Devlin *et al.*, 2019).

Le modèle BERT est un encodeur de type *Transformer* (Vaswani *et al.*, 2017), multicouche et bidirectionnel. Il existe plusieurs variations du modèle BERT selon le corpus de pré-entraînement, le nombre de couches de blocs d'encodeur de modèle *Transformer*, les dimensions des sorties entre ces couches ou encore le nombre de modules d'attention. Les données utilisées pour son pré-entraînement sont d'abord représentées sous forme de plongements lexicaux fondés sur WordPiece (Wu *et al.*,

2016), après ajout d'un token spécial de classification **[CLS]** en début de phrase. Si plusieurs phrases sont assemblées en paire dans une même donnée, des tokens de séparation **[SEP]** sont insérés. Des plongements positionnels, représentant la position des mots dans une phrase, et segmentaires, représentant la position d'une phrase dans une paire de phrases, sont aussi utilisés. Ces différentes représentations sont ensuite concaténées avant d'être présentée au premier bloc d'encodeur de modèle *Transformer*. BERT est pré-entraîné sur deux tâches : Le masquage de mots (*Masked Language Model*) et la prédiction de la prochaine phrase (*Next Sentence Prediction*). Il peut ensuite être affiné par *fine-tuning* sur une variété d'autres tâches dont la détection d'intention ou la détection des concepts.

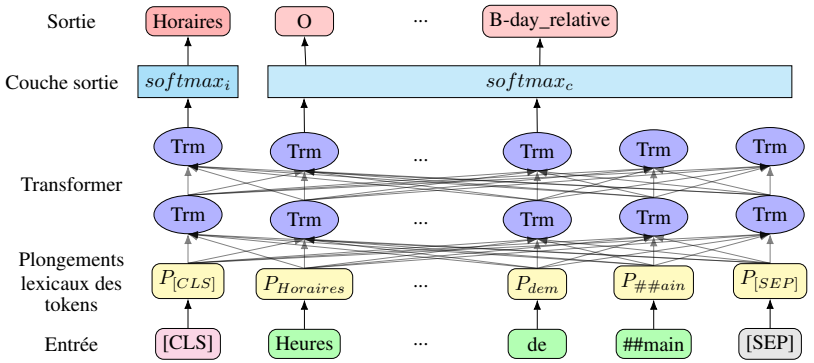


FIGURE 4 – Schéma simplifié de l'architecture du modèle BERT joint d'après Chen *et al.* (2019). La requête en entrée est "Heures d'ouverture de la bibliothèque demain".

Le modèle BERT joint (Chen *et al.*, 2019) propose d'utiliser les états cachés finaux du modèle BERT pour prédire l'intention et les concepts de la phrase passée en entrée. Son architecture est représentée en Figure 4. Dans ce modèle BERT joint, l'état caché final du token **[CLS]** permet de déduire l'intention à l'aide d'une couche de sortie $\mathrm{softmax_i}$. Sa sortie est désignée par la prédiction y^i. Pour les concepts, c'est l'état caché final du premier sous-token qui correspond au début de chacun des n mots de la phrase qui est présenté à une couche de sortie $\mathrm{softmax_c}$. Sa sortie est désignée par la prédiction y^c. L'objectif joint de ce modèle consiste alors au produit des probabilités conditionnelles des concepts et de l'intention :

$$p(y^i, y^c|x) = p(y^i|x) \prod_{n=1}^{N} p(y_n^c|x). \quad (1)$$

Le but est ensuite de maximiser cette probabilité conditionnelle par minimisation d'une fonction de coût par entropie croisée. L'ajout d'une couche à champ aléatoire conditionnel (Lafferty *et al.*, 2001), remplaçant la couche de sortie *softmax*, pour la détection des concepts a été testé par les auteurs Chen *et al.* (2019) mais n'apporte pas toujours une amélioration. L'hypothèse avancée étant que le système d'attention des blocs *Transformer* suffit pour modéliser la structure des différents concepts.

L'architecture du modèle BERT joint peut facilement être ré-implémentée afin de remplacer BERT par $\mathrm{FrALBERT_{base,wiki-4GB}}$ et $\mathrm{CamemBERT_{base,wiki-4GB}}$. Les modèles correspondants seront respectivement désignés par FrALBERT joint et CamemBERT joint.

Précisons que dans l'article sur le classifieur DIET (Bunk *et al.*, 2020), une comparaison avec BERT joint avait été réalisé par les auteurs sur les sets de données ATIS (Hemphill *et al.*, 1990)

et SNIPS (Coucke *et al.*, 2018). Les performances du classifieur DIET étant en deçà de celles du BERT joint de 1 à 2 points pour l'exactitude des intentions et la F-mesure des concepts en utilisant uniquement des représentations éparses. Les auteurs précisaient que ces performances pouvaient être liées à leur méthode d'étiquetage utilisée pour les paramètres, BILOU (Ramshaw & Marcus, 1995), plus complexe que la notation BIO classiquement utilisée. Il est donc probable que FrALBERT joint obtienne de meilleures performances que DIET FrALBERT. Nous utiliserons pour nos modèles à architecture BERT joint la méthode d'étiquetage BIO utilisée dans l'article de Chen *et al.* (2019).

4 Protocoles expérimentaux et résultats

Le but de notre travail est de remplacer le classifieur DIET avec modèle $\mathrm{FrALBERT}_{\mathrm{base,wiki-4GB}}$ préalablement utilisé. Pour cela, nous comparerons ses performances à celles d'un modèle FrALBERT joint. Par ailleurs, les performances de notre modèle compact FrALBERT joint seront comparées à celles d'un modèle CamemBERT joint de plus grande taille. Ceci permettra d'observer les conséquences de la taille du modèle *Transformer* utilisé dans une architecture BERT joint.

4.1 Jeux de données

Comme cela avait été exposé en section 2, nous disposons de deux jeux de données pour lesquels des étiquettes d'intentions et de concepts avaient été décidés préalablement à la création du prototype. Ces deux jeux de données ont été corrigés à l'issue de la campagne d'annotation par un unique annotateur.

— Le corpus UBIB. Il s'agit de questions d'utilisateurs de certaines bibliothèques universitaires métropolitaines françaises destinées à des bibliothécaires, annotées manuellement. Les questions portent généralement sur des informations pratiques, des recherches documentaires ou des problèmes d'accès. Elles ont préalablement été augmentées par génération de données selon la méthode de remplissage de patrons.

— Le corpus DIBISO. Il s'agit des questions obtenues suite à la campagne d'annotation, annotées par le prototype DIET FrALBERT avant correction manuelle. Elles ont été posées par des annotateurs sélectionnés par la DIBISO sans consignes sur les mots clés à utiliser. Ce sont des questions sur des renseignements généraux à propos des BIBLIOTHÈQUES UNIVERSITAIRES PARIS-SACLAY, la plupart ciblées sur les intentions prises en compte par le SQR.

Corpus	DIBISO	$\mathrm{UBIB}_{\mathrm{train}}$	$\mathrm{UBIB}_{\mathrm{test}}$
Nombre de questions	471	12402	997
Taille du vocabulaire	3071	15710	15601
Nombre d'intentions	7	7	7
Nombre de concepts	4	9	9

TABLE 2 – Caractéristiques des différents corpus utilisés. Les lots d'entraînement (*train*) et d'évaluation (*test*) du corpus UBIB sont présentés séparément.

Les caractéristiques de ces jeux de données sont résumées dans la Tableau 2.

Les corpus ont été corrigés avec les 7 intentions utilisées lors de l'élaboration du prototype, présentées dans le Tableau 3. La majorité de ces intentions concernent des catégories de renseignements généraux sur les BIBLIOTHÈQUES UNIVERSITAIRES PARIS-SACLAY : heures d'ouverture, adresse, domaines scientifiques des livres d'une bibliothèque ou bibliothèques comportant des livres d'un domaine scientifique particulier. D'autres sont plus spécifiques au système de SQR : salutations, mise en

question de l'identité du SQR. Une dernière catégorie d'intention comprend toutes les questions n'appartenant pas aux précédentes. Le Tableau 3 révèle aussi la présence d'un déséquilibre entre les proportions des différentes intentions dans le corpus DIBISO comme dans le corpus UBIB pour ses lots d'entraînement (*train*) et d'évaluation (*test*). Concernant les concepts, ils sont étiquetés avec 9 catégories différentes. Ces étiquettes correspondent aux noms des bibliothèques, aux domaines scientifiques, à une date (jour, semaine, mois, période de l'année, etc.) ou des termes relatifs à une date. Les différentes étiquettes utilisées et leurs nombres sont présentés en Tableau 4. Comme les intentions, un fort déséquilibre de répartition des concepts est présent entre les différents corpus.

Intention	DIBISO	$\text{UBIB}_{\text{train}}$	$\text{UBIB}_{\text{test}}$
bot challenge	26	62	9
get timetable of library	7	2812	162
out of scope	346	1433	356
search library fields	14	4925	141
greet	26	11	3
search library address	29	2637	256
search libraries from field	23	522	70

TABLE 3 – Répartition des intentions dans les différents corpus. Les lots d'entraînement (*train*) et d'évaluation (*test*) du corpus UBIB sont présentés séparément.

Concepts	DIBISO	$\text{UBIB}_{\text{train}}$	$\text{UBIB}_{\text{test}}$
library	136	11920	609
field	66	8353	515
month	0	67	30
period_relative	1	545	76
day_of_month	0	1079	9
day_relative	3	287	54
week_relative	0	61	9
day_of_the_week	0	576	43
date_relative	0	43	13

TABLE 4 – Répartition des intentions dans les différents corpus. Les lots d'entraînement (*train*) et d'évaluation (*test*) du corpus UBIB sont présentés séparément.

4.2 Métriques d'évaluation

Nous avons d'abord évalué chacune des tâches de manière indépendante. Pour la détection d'intention, seule l'exactitude sera utilisée. Pour la détection des concepts, la précision, le rappel et la F-mesure seront utilisés. Une autre métrique permet de calculer le résultat conjoint des deux tâches : L'exactitude du cadre sémantique à l'échelle des phrases. Le cadre sémantique d'une phrase est considéré comme exact lorsque son intention et ses concepts ont tous été parfaitement prédits.

4.3 Protocole d'entraînement

Deux séries d'expériences sont réalisées sur les modèles FrALBERT joint et CamemBERT joint afin d'étudier la qualité du corpus DIBISO. La question sous-jacente étant : est-ce que ces données DIBISO vont être suffisantes pour obtenir un modèle avec de bonnes capacités de prédiction ? Pour cela, des modèles joints seront entraînés et testés exclusivement sur le corpus DIBISO. D'autres modèles joints

seront entraînés sur le lot d'entraînement du corpus UBIB associé à une partie du corpus DIBISO. Dans ce second cas, le lot d'évaluation du corpus UBIB servira alors de lot de validation. Les performances des modèles seront exclusivement évaluées sur une sous-partie des données DIBISO.

En raison de la petite quantité de données du corpus DIBISO, une validation croisée à *k*-blocs (Kohavi, 1995) est réalisée : Le corpus est mélangé, découpé en cinq blocs de tailles proches, puis quatre de ces blocs sont assemblés pour former le lot d'entraînement tandis que le dernier bloc constitue celui d'évaluation. Cinq différentes répartitions de lots d'entraînements et d'évaluations sont ainsi obtenues. La moyenne et l'écart-type sur ces différentes répartitions seront calculés.

Des modèles FrALBERT joint et CamemBERT joint sont mis en place et entraînés sur chacune des différentes répartitions. Concernant le classifieur DIET FrALBERT, celui-ci est entraîné pendant 300 itérations totales du corpus d'entraînement (*epoch*) UBIB ré-annoté, pour reproduire des conditions d'entraînement similaires à celles du prototype déployé lors de la campagne d'annotation. Si ce protocole expérimental ne permet pas une véritable comparaison entre classifieur DIET et modèle joint, le but est avant tout de s'assurer que le nouveau modèle mis en place soit plus performant que la *baseline* représentée par le précèdent.

Concernant les optimisations, l'optimiseur AdamW (Loshchilov & Hutter, 2019) est utilisé pour les modèles joints. Des optimisations intégrées à la librairie RASA, difficiles à identifier en raison d'un effet « boîte noire », sont appliquées au classifieur DIET. Avec RASA, les résultats d'un entraînement à un autre sont peu variables. Un seul *run* sera donc réalisé pour le classifieur DIET.

Modèle joint	**Exactitude**$_i$	**Précision**$_c$	**Rappel**$_c$	**F-mesure**$_c$	**Exactitude**$_{cs}$
FrALBERT$_{base,wiki-4GB}$	92,49±0,91	75,87±2,12	78,27±1,97	77,05±1,98	49,11±3,36
CamemBERT$_{base,wiki-4GB}$	**94,67**±0,78	**81,51**±0,62	**85,19**±0,27	**83,31**±0,39	**60,52**±0,49

TABLE 5 – Performances des modèles joints après entraînement de 10 *epoch* sur le corpus ATIS-FR pour les intentions (i), les concepts (c) et le cadre sémantique au niveau des phrases (cs). Les résultats présentés sont les moyennes et écarts-types sur 10 *runs* sur le lot d'évaluation d'ATIS-FR.

Puisque le corpus DIBISO est de faible taille et afin de spécialiser le modèle *Transformer* utilisé sur le système de détection jointe, un pré-entraînement de 10 *epoch* est réalisé sur le corpus français ATIS-FR issu du corpus MultiATIS+++ (Xu *et al.*, 2020) pour les modèles joints. Ce pré-entraînement n'est pas possible pour le classifieur DIET inclus dans la librairie RASA car la librairie ne permet pas l'entraînement continu et l'apprentissage par transfert. À l'issue de ce pré-entraînement simple, dont les résultats sont présentés en Tableau 5, on constate une différence non-négligeable entre les deux modèles joints. Ces différences sont en faveur du modèle CamemBERT joint avec des écarts de 2,18 points pour l'exactitude des intentions, et de 6,26 points pour la F-mesure des concepts. Les auteurs Cattan *et al.* (2022) ont pourtant démontré que les modèles monolingues FrALBERT$_{base,wiki-4GB}$ et CamemBERT$_{base,wiki-4GB}$ ont des performances similaires sur ce corpus ATIS-FR. Dans leurs travaux, les résultats pour la F-mesure des concepts étaient de 92,8 pour FrALBERT$_{base,wiki-4GB}$ et de 92,5 pour CamemBERT$_{base,wiki-4GB}$. L'écart que nous observons dans le Tableau 5 est plus important que celui de 0,3 point qu'ils avaient constaté.

Plusieurs hypothèses peuvent expliquer ces différences, notamment le fait que l'objectif à minimiser dans un modèle joint diffère de celui d'un problème de classification en classes multiples plus classique. Par ailleurs, nous utilisons des hyperparamètres fixes lors de ce pré-entraînement alors qu'une optimisation par *population based training* (Jaderberg *et al.*, 2017) était utilisée dans les

travaux de Cattan *et al.* (2022). Nous choisissons donc d'utiliser le *population based training*, illustré en Figure 5, dans nos expérimentations sur les modèles joints. Nous explorerons un taux d'apprentissage entre 1 et 5, une taille de *batch* d'entraînement entre 8 et 32 et un nombre d'*epoch* d'apprentissage entre 8 et 14. Le nombre de versions parallèles entraînées (épreuves) sera fixé à 8 et la meilleure épreuve pour chaque *run* sera sélectionnée en fonction du plus petit résultat de coût obtenu sur le lot de validation. Le *population based training* ne sera pas utilisé lors du pré-entraînement, notre but n'étant pas de spécialiser nos modèles sur le corpus ATIS-FR. Dans le cas où seul le corpus DIBISO est utilisé, un nouveau découpage de 10% du lot d'entraînement est réalisé pour constituer un lot de validation nécessaire au *population based training*. L'utilisation du même lot de validation que celui des modèles entraînés en partie sur le corpus UBIB n'est pas envisageable : ce lot est de trop grande taille et contient des concepts que ne présente pas le corpus DIBISO. Ce nouveau découpage, ainsi que les différences de distribution des lots de validation (en quantité et origine des données), peuvent constituer des biais statistiques à prendre en compte lors de l'analyse de nos résultats.

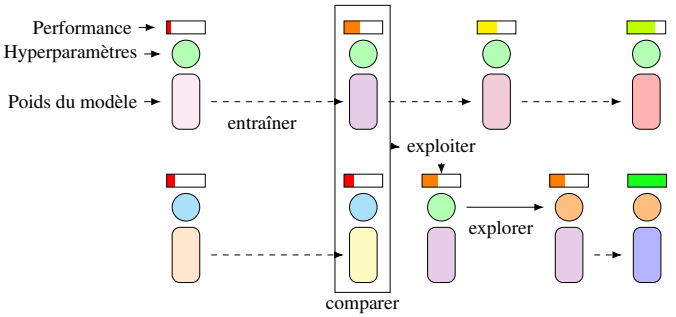


FIGURE 5 – Représentation schématique du *population based training* d'après Jaderberg *et al.* (2017). Plusieurs versions (épreuves) d'un même modèle sont initialisées aléatoirement au niveau de leurs hyperparamètres et de leurs poids. Elle sont entraînées et évaluées parallèlement. Les différentes configurations de poids et d'hyperparamètres sont représentées ici par des couleurs différentes. À intervalles réguliers et après comparaison statistique des différentes versions, les poids et hyperparamètres d'une version peu performante sont remplacés par ceux d'une meilleure version. De nouveaux hyperparamètres sont alors explorés sur cette copie en appliquant un facteur de perturbation ou en les réinitialisant de manière aléatoire.

Les résultats des modèles joints pré-entraînés sur ATIS-FR seront confrontés à ceux du classifieur DIET FrALBERT entraîné sur le corpus UBIB uniquement.

4.4 Résultats et analyse

Les performances de nos modèles sont présentées en Tableau 6 pour les différentes métriques.
Une analyse des erreurs des meilleurs modèles FrALBERT joint et CamemBERT joint est aussi réalisée : Pour chaque expérience, les modèles avec les plus petites valeurs de coûts sur leurs lots d'évaluation sont sélectionnés parmi les différents *runs*. Pour permettre cette comparaison, ces modèles sont sélectionnés sur la même répartition de la validation croisée à k-blocs. Ensuite, des prédictions sur l'intégralité du corpus DIBISO sont réalisées. Les résultats de cette analyse sont représentés par des matrices de confusion en Figures 6 et 7. Des tests de McNemar sont réalisés avec une valeur seuil de 0,05 entre ces meilleurs modèles pour déterminer si leurs différences sont significatives sur les tâches de détection d'intention et d'identification des concepts.

Modèle	Données	Exactitude
DIET FrALBERT$_{\text{base,wiki-4GB}}$	U	72,59±3,45
FrALBERT$_{\text{base,wiki-4GB}}$ joint	D	87,15±5,43
CamemBERT$_{\text{base,wiki-4GB}}$ joint	D	86,62±4,80
FrALBERT$_{\text{base,wiki-4GB}}$ joint	U + D	**87,55**±4,76
CamemBERT$_{\text{base,wiki-4GB}}$ joint	U + D	87,52±4,97

(a) Performances pour les intentions

Modèle	Données	Précision	Rappel	F-mesure
DIET FrALBERT$_{\text{base,wiki-4GB}}$	U	85,29±7,49	88,79±2,08	**92,87**±4,21
FrALBERT$_{\text{base,wiki-4GB}}$ joint	D	81,50±10,60	75,71±11,65	78,14±9,92
CamemBERT$_{\text{base,wiki-4GB}}$ joint	D	81,33±7,52	81,30±8,15	81,00±6,23
FrALBERT$_{\text{base,wiki-4GB}}$ joint	U + D	**90,97**±4,46	**94,04**±4,03	92,38±3,03
CamemBERT$_{\text{base,wiki-4GB}}$ joint	U + D	85,00±6,13	90,95±5,37	87,73±4,63

(b) Performances pour les concepts

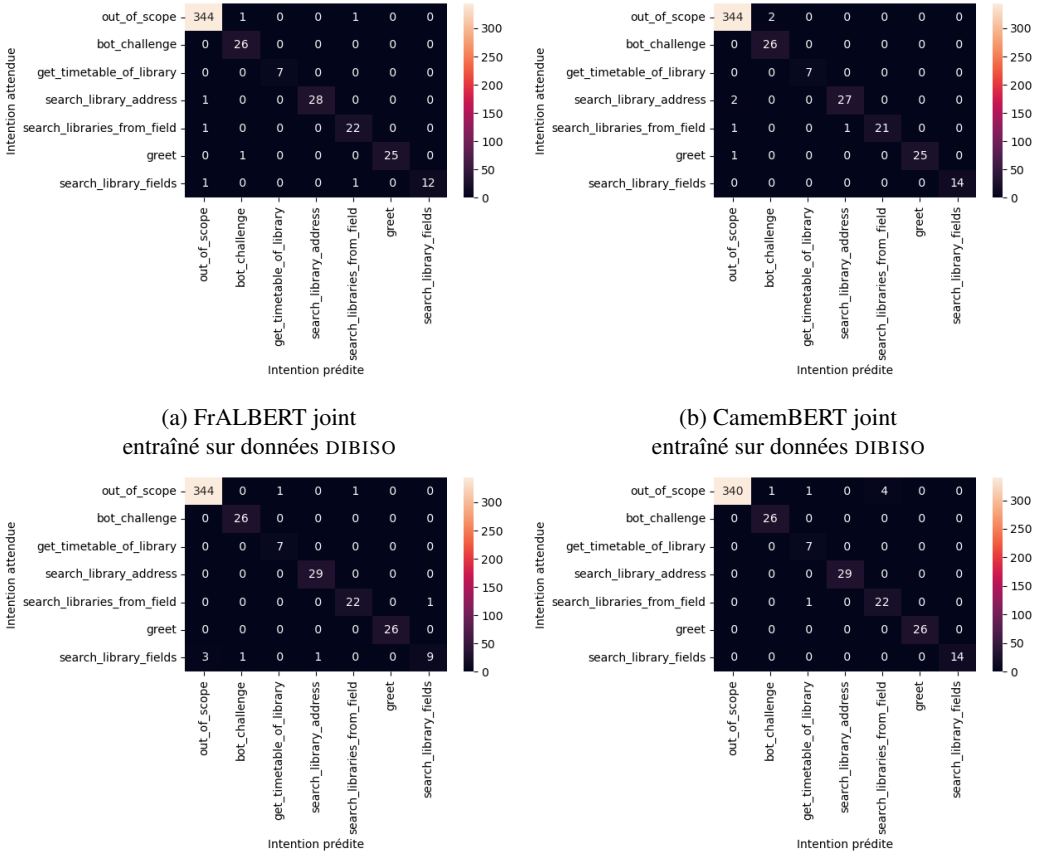
Modèle	Données	Exactitude
DIET FrALBERT$_{\text{base,wiki-4GB}}$	U	67,44±4,99
FrALBERT$_{\text{base,wiki-4GB}}$ joint	D	78,38±6,09
CamemBERT$_{\text{base,wiki-4GB}}$ joint	D	77,28±5,03
FrALBERT$_{\text{base,wiki-4GB}}$ joint	U + D	**83,51**±5,31
CamemBERT$_{\text{base,wiki-4GB}}$ joint	U + D	81,42±5,45

(c) Performances pour le cadre sémantique au niveau des phrases

TABLE 6 – Performances des différents modèles. Les modèles sont entraînés sur les données d'entraînement UBIB (U) et/ou DIBISO (D), comme indiqué en colonne "Données", et évalués sur les différents lots d'évaluation issus du corpus DIBISO pour chaque découpage de la validation croisée. Les résultats présentés pour les modèles joints sont les moyennes et écarts-types sur 6 *runs*, après calcul des valeurs sur les 5 blocs de validation croisée pour chaque *run*. Les résultats présentés pour le modèle DIET sont issus d'un unique *run* pour lequel la moyenne et l'écart-type sur les 5 blocs d'évaluation de la validation croisée des modèles joints ont été calculés.

4.4.1 Comparaison des modèles à architecture jointe sur la tâche de détection d'intention

Pour la tâche de détection d'intention dont les résultats sont présentés en Tableau 6a, les modèles FrALBERT joint et CamemBERT joint entraînés uniquement sur le corpus DIBISO ont obtenu respectivement 87,15 et 86,62 d'exactitude, soit seulement 0,53 point d'écart en faveur de FrALBERT joint. Les modèles joints entraînés sur les deux corpus ont des performances très similaires, avec 87,55 pour FrALBERT joint et 87,52 pour CamemBERT joint, soit 0,03 point d'écart. Ces faibles écarts de moyennes entre modèles entraînés sur les mêmes ensembles de données, associés à de forts écarts-types (entre 4,76 et 5,43), montrent que dans nos conditions d'expérimentations FrALBERT joint parvient à égaler CamemBERT joint pour la tâche de détection d'intention, malgré son plus faible nombre de paramètres. Par ailleurs, les performances similaires entre modèles entraînés uniquement sur le corpus DIBISO ou associé au corpus UBIB indiquent que malgré sa faible quantité de données, le contenu du corpus DIBISO est suffisamment complet pour permettre aux modèles de distinguer les différentes intentions.



(a) FrALBERT joint entraîné sur données DIBISO

(b) CamemBERT joint entraîné sur données DIBISO

(c) FrALBERT joint entraîné sur données DIBISO et UBIBs

(d) CamemBERT joint entraîné sur données DIBISO et UBIBs

FIGURE 6 – Matrices de confusion des intentions sur le corpus DIBISO.

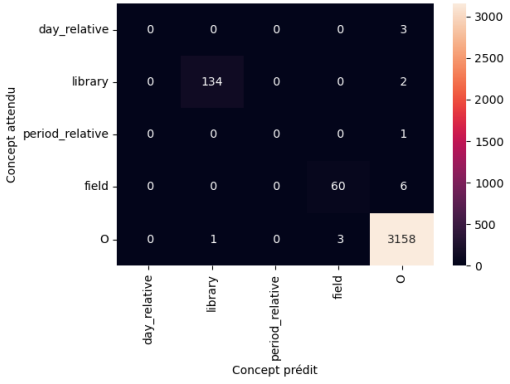
Une analyse des erreurs des prédictions de ces intentions pour les modèles joints, représentée par la Figure 6, confirme cette bonne correspondance des intentions attendues et prédites. De plus, les valeurs-p ne permettent pas d'établir de différences significatives pour la tâche de détection d'intention entre les meilleurs modèles utilisant le même *Transformer* (0,7600 pour les modèles FrALBERT, 1 pour les modèles CamemBERT) ou entre les meilleurs modèles entraînés sur les mêmes corpus (1 pour les modèles entraînés sur le corpus DIBISO uniquement, 0.7600 pour ceux entraînés sur les deux corpus). Pour près de la moitié des erreurs commises par les modèles entraînés uniquement sur le corpus DIBISO, c'est l'intention majoritaire (*out_of_scope*) du corpus qui a été prédite. Dans le cas des modèles entraînés sur les deux corpus, le meilleur modèle FrALBERT joint rencontre plus de difficulté à identifier correctement l'intention *search_library_fields* avec un tiers d'erreurs pour les phrases y correspondant. Il s'agit pourtant de l'intention la plus présente du corpus UBIB utilisé pour l'entraînement. Le meilleur modèle CamemBERT joint identifie certaines questions comme étant étiquetées *search_libraries_from_field* parmi les *out_of_scope* (4 des 7 erreurs commises par ce modèle). En regardant les questions concernées par ces erreurs, certaines formulations peuvent

prêter à confusion entre les deux intentions. Par exemple, la question *"je cherche des livres de droit"* est considérée comme *out_of_scope* car il s'agit d'une recherche de livre, mais est identifiée comme *search_libraries_from_field* par le modèle. La question équivalente *search_libraries_from_field* serait plutôt *"dans quelle bibliothèque trouver des livres de droit"* ou *"où trouver des livres de droit"*.

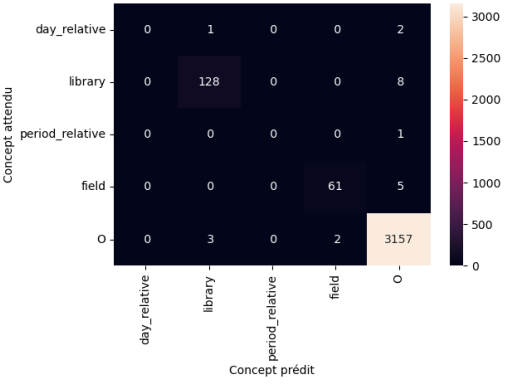
4.4.2 Comparaison des modèles à architecture jointe sur la tâche d'identification des concepts

Concernant l'identification des concepts en Tableau 6b, on observe un écart important entre les modèles selon les données d'entraînement utilisées : Le modèle FrALBERT joint gagne 14,24 points de F-mesure s'il est entraîné sur les deux corpus (92,38) plutôt que sur le corpus DIBISO seul (78,14). L'écart entre les modèles CamemBERT est de 6,73 points pour la F-mesure avec 81,00 s'il est entraîné sur le corpus DIBISO uniquement contre 87,73 s'il est entraîné sur les deux corpus. Les différences sont significatives pour cette tâche entre les meilleurs modèles FrALBERT entraînés sur corpus DIBISO seul par rapport à celui entraîné sur les deux corpus (valeur-p de 0,0028). Ces différences sont aussi significatives entre les meilleurs modèles CamemBERT selon les données d'entraînement (valeur-p de 0,0201). Ainsi, il semblerait que les modèles joints ne parviennent pas à extraire suffisamment d'informations uniquement du corpus DIBISO pour cette tâche. La faible présence de certains concepts dans ce corpus (*day_relative* et *period_relative*) peut empêcher leur identification correcte et expliquer en partie ces scores des modèles entraînés uniquement sur le corpus DIBISO. Par ailleurs, de par les découpages en lots d'évaluation et de validation du corpus DIBISO, certains de ces modèles peuvent ne pas avoir été entraînés sur des questions présentant ces concepts. Les matrices de confusion correspondant aux meilleurs modèles entraînés uniquement sur le corpus DIBISO, en Figure 7a et 7b, confirment que les étiquettes présentes en fortes quantité dans ce corpus (*library* et *field*) sont plutôt bien identifiées, avec la présence de quelques faux-négatifs et faux-positifs. Les deux autres étiquettes ne sont pas du tout détectées.

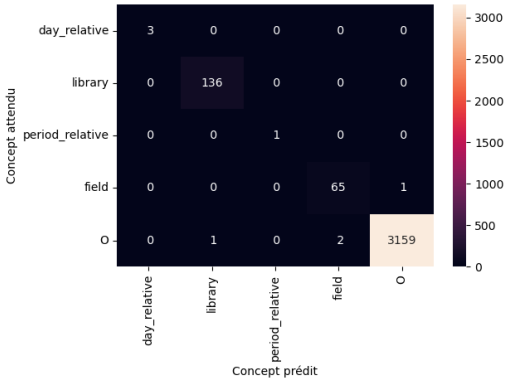
Au sujet des différences entre *Transformer* utilisés, elles sont non négligeables au niveau des moyennes. Lors de l'entraînement sur le corpus DIBISO uniquement, le score de rappel est en faveur de CamemBERT joint avec une valeur de 81,20 contre 5,49 points de moins pour FrALBERT joint. L'écart entre les F-mesures est de 2,86 points en faveur de CamemBERT joint et l'écart entre les scores de précision de 0,17 point en faveur de FrALBERT joint (81,50 contre 81,33 pour CamemBERT joint). Cependant, pour ces modèles, FrALBERT présente de plus importants écart-types (entre 9,92 et 11,65) que CamemBERT (entre 6,23 et 8,15). Lors de l'entraînement sur les deux corpus, les résultats sont en faveur de FrALBERT joint uniquement : Son score de précision est de 90,97 soit 5,97 points de plus que CamemBERT joint, son score de rappel avec une valeur de 94,04 est 3,09 points plus élevé que CamemBERT joint et enfin, sa F-mesure est en avance de 4,65 points. Le peu d'erreurs présentes dans la matrice de confusion du meilleur modèle FrALBERT joint entraîné sur les deux corpus, en Figure 7c, corrobore ces résultats. Ces différences de score en faveur de FrALBERT joint sont surprenantes en considérant que ce modèle est de plus petite taille que CamemBERT. Une hypothèse pouvant être formulée est que les conditions de l'optimisation par *population based training* telle que la fourchette du nombre d'*epoch* explorée, ou le petit nombre de *runs* réalisé ont grandement influencés les résultats obtenus. Si les scores de précision et de rappel de CamemBERT joint entraîné sur les deux corpus montrent une faible tendance à la détection de faux-positifs, les prédictions du meilleur modèle en Figure 7d montrent plutôt quelques faux-négatifs. La valeur-p entre les meilleurs modèles confirment que pour les modèles entraînés sur le corpus DIBISO seul, les différences sont significatives avec 0,027. Par contre, la valeur-p de 0,24 entre les meilleurs modèles FrALBERT et CamemBERT entraînés sur l'ensemble des corpus ne permet pas d'établir de différence significative sur cette tâche avec le test de McNemar.



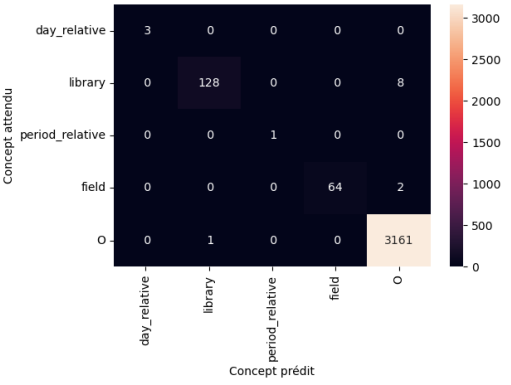
(a) FrALBERT joint
entraîné sur données DIBISO



(b) CamemBERT joint
entraîné sur données DIBISO



(c) FrALBERT joint
entraîné sur données DIBISO et UBIBs



(d) CamemBERT joint
entraîné sur données DIBISO et UBIBs

FIGURE 7 – Matrices de confusion des concepts sur le corpus DIBISO.

4.4.3 Comparaison des modèles à architecture jointe sur le cadre sémantique

Les résultats de l'exactitude du cadre sémantique présentés en Tableau 6c suivent ceux observés pour les concepts : Le meilleur modèle sur cette métrique est FrALBERT joint entraîné sur l'ensemble des deux corpus avec 83,51 d'exactitude, contre 81,42 pour le modèle CamemBERT joint équivalent. Les modèles joints entraînés sur association des deux corpus obtiennent de meilleures performances que ceux entraînés sur le corpus DIBISO seul avec un écart de 5,13 points pour les modèles FrALBERT joint et de 4,14 points pour les modèles CamemBERT joint.

4.4.4 Comparaison du classifieur DIET aux modèles à architecture jointe

DIET FrALBERT présente des résultats inférieurs à l'ensemble des différents modèles joints pour la tâche de détection d'intention avec 72,59, et en conséquence pour l'exactitude du cadre sémantique (67,44). Concernant la tâche d'identification des concepts, ses résultats sont meilleurs que les modèles joints entraînés sur le corpus DIBISO uniquement notamment avec une F-mesure de 92,87. Pour les

modèles entraînés sur plus de données, FrALBERT joint obtient globalement de meilleurs résultats que DIET FrALBERT. Nos conditions d'expérimentations ne nous permettent pas d'estimer si ces différences de performances entre classifieur DIET et modèles joints sont liées aux architectures, aux optimisations, au pré-entraînement des modèles joints sur ATIS-FR ou à l'influence des données du corpus DIBISO.

4.4.5 Réflexion sur les écarts-types

On observe sur l'ensemble des résultats la présence d'un fort écart-type pour chacune des métriques. Il s'explique avant tout par l'utilisation d'une validation croisée engendrant des lots potentiellement très déséquilibrés. Une autre explication, plus difficile à prouver, est l'effet en « dent de scie » entre les deux tâches qui peut être observé en fin d'entraînement chez les modèles réalisant une détection jointe (Hui *et al.*, 2021) : l'augmentation de l'exactitude d'une tâche peut déclencher la diminution de l'exactitude de l'autre tâche. Ainsi, pour des calculs de coûts équivalents, deux modèles peuvent avoir des performances relativement différentes pour les tâches de détection d'intention et d'identification des concepts.

5 Conclusion

Dans cet article, nous avons détaillé la mise en place d'un modèle réalisant une détection jointe de l'intention et des concepts, dont l'architecture repose sur un modèle *Transformer* compact. Notre contexte applicatif, un système interactif de questions-réponses (SQR), ne disposait initialement d'aucun corpus répondant précisément à notre besoin. Un premier prototype utilisant un corpus relativement similaire (corpus UBIB), et reposant sur une architecture de classifieur DIET (Bunk *et al.*, 2020) a permis de collecter un corpus de 471 questions. À partir de ces nouvelles données, des modèles reposant sur une architecture BERT joint (Chen *et al.*, 2019) ont été entraînés. Le modèle compact de *Transformer* FrALBERT$_{base,wiki-4GB}$ a été utilisé pour l'élaboration du classifieur DIET et des modèles joints. Dans le cas des modèles joints, FrALBERT a été comparé avec le *Transformer* CamemBERT$_{base,wiki-4GB}$. Une de nos finalités étant d'évaluer si le modèle compact FrALBERT a des performances comparables à un modèle de plus grande taille pour cette détection jointe et dans notre contexte applicatif.

Nos résultats montrent que les nouvelles données obtenues, bien que correspondant mieux à notre contexte, sont insuffisantes à elles seules pour permettre de meilleures performances que le prototype sur la tâche d'identification des concepts. Elles permettent par contre aux modèles joints d'obtenir une meilleure exactitude pour la tâche de détection d'intention. Un entraînement à la fois sur les données UBIB et les données récoltées permet d'obtenir des modèles joints plus performants que le prototype DIET FrALBERT sur les deux tâches. Bien que notre étude ne permette pas une véritable comparaison entre les architectures classifieur DIET et BERT joint, nous pouvons donc envisager remplacer le classificateur DIET actuel de notre SQR par un modèle joint entraîné selon notre protocole expérimental. Par ailleurs, le modèle joint FrALBERT entraîné sur les deux corpus présente des performances similaires à son équivalent CamemBERT joint. L'usage de FrALBERT dans notre SQR permettra donc l'obtention de bons résultats en réduisant notre consommation des ressources. Enfin, nous prévoyons d'enrichir notre corpus DIBISO, notamment pour lui ajouter les concepts qui lui manquent. Pour cela, l'utilisation de méthodes de génération ou d'augmentation de données, comme les méthodes de patron (Boulanger *et al.*, 2022) ou *Tri-training* (Boulanger, 2020), seront envisagées.

Remerciements

Nous tenons à remercier la Direction des Bibliothèques, de l'Information et de la Science Ouverte de l'Université Paris-Saclay pour la proposition de ce projet, ainsi que leur collaboration et leur aide lors de sa mise en place. Nous adressons nos sincères remerciements à la plateforme Ubib et son intermédiaire, Natacha LECLERC, pour nous avoir fourni des conversations issues de leur service. Nous remercions Mathilde VERON pour son travail et son rapport sur l'étude de faisabilité du projet. Tous nos remerciements vont également aux différents acteurs du projet Humane.AI pour leur accueil et échanges stimulants lors de nos réunions hebdomadaires. Enfin, nous tenons à exprimer toute notre gratitude à notre équipe encadrante du LISN pour leur bienveillance, leurs conseils avisés et l'opportunité de rédiger cet article : Sophie ROSSET, Christophe SERVAN, Laure SOULIER et Sahar GHANNAY.

Références

BENDER E. M., GEBRU T., MCMILLAN-MAJOR A. & SHMITCHELL S. (2021). On the dangers of stochastic parrots : Can language models be too big ? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, p. 610–623, New York, NY, USA : Association for Computing Machinery.

BOCKLISCH T., FAULKER J., PAWLOWSKI N. & NICHOL A. (2017). Rasa : Open source language understanding and dialogue management. In *NIPS 2017 Conversational AI workshop*, p. 1–9.

BOULANGER H. (2020). Évaluation systématique d'une méthode commune de génération. In *6e conférence conjointe Journées d'Études sur la Parole (JEP, 33e édition), Traitement Automatique des Langues Naturelles (TALN, 27e édition), Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL, 22e édition). Volume 3 : Rencontre des Étudiants Chercheurs en Informatique pour le TAL*, p. 43–56, Nancy, France : ATALA.

BOULANGER H., LAVERGNE T. & ROSSET S. (2022). Generating unlabelled data for a tri-training approach in a low resourced NER task. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, p. 30–37, Hybrid : Association for Computational Linguistics.

BUNK T., VARSHNEYA D., VLASOV V. & NICHOL A. (2020). Diet : Lightweight language understanding for dialogue systems. *arXiv : 2004.09936 [cs]*.

CASTELLUCCI G., BELLOMARIA V., FAVALLI A. & ROMAGNOLI R. (2019). Multi-lingual intent detection and slot filling in a joint bert-based model. *arXiv : 1907.02884 [cs]*.

CATTAN O., GHANNAY S., SERVAN C. & ROSSET S. (2022). Benchmarking Transformers-based models on French Spoken Language Understanding tasks. In *Proc. Interspeech 2022*, p. 1238–1242.

CATTAN O., SERVAN C. & ROSSET S. (2021). On the usability of transformers-based models for a French question-answering task. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, p. 244–255, Held Online : INCOMA Ltd.

CHEN Q., ZHUO Z. & WANG W. (2019). Bert for joint intent classification and slot filling. *arXiv : 1902.10909 [cs]*.

CHEN Y.-N., HAKANNI-TÜR D., TUR G., CELIKYILMAZ A., GUO J. & DENG L. (2016). Syntax or semantics ? knowledge-guided joint semantic frame parsing. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, p. 348–355.

COUCKE A., SAADE A., BALL A., BLUCHE T., CAULIER A., LEROY D., DOUMOURO C., GISSELBRECHT T., CALTAGIRONE F., LAVRIL T., PRIMET M. & DUREAU J. (2018). Snips voice platform : an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv : 1805.10190 [cs]*.

DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2019). BERT : Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, p. 4171–4186, Minneapolis, Minnesota : Association for Computational Linguistics.

GOO C.-W., GAO G., HSU Y.-K., HUO C.-L., CHEN T.-C., HSU K.-W. & CHEN Y.-N. (2018). Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 2 (Short Papers)*, p. 753–757, New Orleans, Louisiana : Association for Computational Linguistics.

GUO D., TUR G., YIH W.-T. & ZWEIG G. (2014). Joint semantic utterance classification and slot filling with recursive neural networks. In *2014 IEEE Spoken Language Technology Workshop (SLT)*, p. 554–559.

HAKKANI-TÜR D., TUR G., CELIKYILMAZ A., CHEN Y.-N., GAO J., DENG L. & WANG Y.-Y. (2016). Multi-Domain Joint Semantic Frame Parsing Using Bi-Directional RNN-LSTM. In *Proc. Interspeech 2016*, p. 715–719.

HEMPHILL C. T., GODFREY J. J. & DODDINGTON G. R. (1990). The ATIS spoken language systems pilot corpus. In *Speech and Natural Language : Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*.

HENDERSON M., VULIĆ I., GERZ D., CASANUEVA I., BUDZIANOWSKI P., COOPE S., SPITHOURAKIS G., WEN T.-H., MRKŠIĆ N. & SU P.-H. (2019). Training neural response selection for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, p. 5392–5404, Florence, Italy : Association for Computational Linguistics.

HUI Y., WANG J., CHENG N., YU F., WU T. & XIAO J. (2021). Joint intent detection and slot filling based on continual learning model. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, p. 7643–7647.

JADERBERG M., DALIBARD V., OSINDERO S., CZARNECKI W. M., DONAHUE J., RAZAVI A., VINYALS O., GREEN T., DUNNING I., SIMONYAN K., FERNANDO C. & KAVUKCUOGLU K. (2017). Population based training of neural networks. *arXiv : 1711.09846 [cs]*.

JAPKOWICZ N. & STEPHEN S. (2002). The class imbalance problem : A systematic study. *Intell. Data Anal.*, **6**(5), 429–449.

JEONG M. & LEE G. G. (2008). Triangular-chain conditional random fields. *IEEE Transactions on Audio, Speech, and Language Processing*, **16**(7), 1287–1302.

KOHAVI R. (1995). A study of cross-validation and Bootstrap for accuracy estimation and model selection. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, p. 1137–1143 : Morgan Kaufmann Publishers Inc.

LAFFERTY J. D., MCCALLUM A. & PEREIRA F. C. N. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, p. 282–289, San Francisco, CA, USA : Morgan Kaufmann Publishers Inc.

LAMPLE G., BALLESTEROS M., SUBRAMANIAN S., KAWAKAMI K. & DYER C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 260–270, San Diego, California : Association for Computational Linguistics.

LAN Z., CHEN M., GOODMAN S., GIMPEL K., SHARMA P. & SORICUT R. (2020). Albert : A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

LIU B. & LANE I. (2016). Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling. In *Proc. Interspeech 2016*, p. 685–689.

LIU Y., OTT M., GOYAL N., DU J., JOSHI M., CHEN D., LEVY O., LEWIS M., ZETTLEMOYER L. & STOYANOV V. (2020). Roberta : A robustly optimized bert pretraining approach.

LOSHCHILOV I. & HUTTER F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

MARTIN L., MULLER B., ORTIZ SUÁREZ P. J., DUPONT Y., ROMARY L., DE LA CLERGERIE É., SEDDAH D. & SAGOT B. (2020). CamemBERT : a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 7203–7219, Online : Association for Computational Linguistics.

MOOSAVI N. S., FAN A., SHWARTZ V., GLAVAŠ G., JOTY S., WANG A. & WOLF T., Éds. (2020). *Proceedings of SustaiNLP : Workshop on Simple and Efficient Natural Language Processing*, Online. Association for Computational Linguistics.

ORTIZ SUÁREZ P. J., ROMARY L. & SAGOT B. (2020). A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 1703–1714, Online : Association for Computational Linguistics.

RAMSHAW L. & MARCUS M. (1995). Text chunking using transformation-based learning. In *Third Workshop on Very Large Corpora*.

SHAW P., USZKOREIT J. & VASWANI A. (2018). Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 2 (Short Papers)*, p. 464–468, New Orleans, Louisiana : Association for Computational Linguistics.

SMITH S. L., KINDERMANS P.-J. & LE Q. V. (2018). Don't decay the learning rate, increase the batch size. In *International Conference on Learning Representations*.

STRUBELL E., GANESH A. & MCCALLUM A. (2019). Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, p. 3645–3650, Florence, Italy : Association for Computational Linguistics.

VALSOV V., MOSIG J. E. M. & NICHOL A. (2019). Dialogue transformers. *arXiv : 1910.00486 [cs]*.

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.

WELD H., HUANG X., LONG S., POON J. & HAN S. C. (2022). A survey of joint intent detection and slot filling models in natural language understanding. *ACM Comput. Surv.*, **55**(8).

WU L., FISCH A., CHOPRA S., ADAMS K., BORDES A. & WESTON J. (2018). Starspace : Embed all the things ! *Proceedings of the AAAI Conference on Artificial Intelligence*, **32**(1).

WU Y., SCHUSTER M., CHEN Z., LE Q. V., NOROUZI M., MACHEREY W., KRIKUN M., CAO Y., GAO Q., MACHEREY K., KLINGNER J., SHAH A., JOHNSON M., LIU X., KAISER L., GOUWS S., KATO Y., KUDO T., KAZAWA H., STEVENS K., KURIAN G., PATIL N., WANG W., YOUNG C., SMITH J., RIESA J., RUDNICK A., VINYALS O., CORRADO G., HUGHES M. & DEAN J. (2016). Google's neural machine translation system : Bridging the gap between human and machine translation. *arXiv : 1609.08144 [cs]*.

XU P. & SARIKAYA R. (2013). Convolutional neural network based triangular crf for joint intent detection and slot filling. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, p. 78–83.

XU W., HAIDER B. & MANSOUR S. (2020). End-to-end slot alignment and recognition for cross-lingual NLU. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 5052–5063, Online : Association for Computational Linguistics.