MTEB-FR: une expérience à large échelle pour l'apprentissage de représentation en français

Wissam Siblini*, Mathieu Ciancone**
Imene Kerboua***, Marion Schaeffer**

*Sans affiliation
wissam.siblini92@gmail.com,
**Wikit, France
{mathieu, marion}@wikit.ai
***INSA Lyon, LIRIS - Esker, France
imene.kerboua@liris.cnrs.fr

Résumé. De nombreux modèles de représentation textuelle (*embedding*) sont aujourd'hui disponibles et utilisés pour diverses tâches de traitement du langage naturel. Le projet MTEB (Massive Textual Embedding Benchmark) a fortement simplifié le choix d'un modèle efficace pour l'anglais. Nous proposons de l'élargir en introduisant la première expérience à large échelle pour le français. Nous introduisons 3 nouveaux ensembles de données, et en rassemblons des existants pour constituer une évaluation globale sur 27 jeux associés à 8 tâches (e.g. classification, recherche d'information). Nous comparons 51 modèles soigneusement sélectionnés, selon diverses métriques et statistiques, afin d'identifier les plus performants et d'analyser la corrélation entre performance et caractéristiques. Bien qu'aucune méthode ne domine sur toutes les tâches, les modèles multilingues avec un grand nombre de paramètres, et spécialisés pour la tâche de similarité entre phrases, sont particulièrement performants. D'autres modèles beaucoup plus économes sont également très compétitifs. Notre travail est accompagné d'une librairie facilement utilisable, ouverte au public (open source), et d'un classement public évolutif [1] permettant des contributions externes.

1 Introduction

Le terme anglais *embedding* désigne des représentations vectorielles denses de textes dont l'objectif est de capturer leur sémantique. Word2Vec, introduit par Mikolov et al. (2013), est un exemple emblématique. Il s'agit d'un réseau de neurones entraîné à apprendre des représentations de mots à partir de leurs relations contextuelles dans une grande quantité de textes. Depuis son introduction, de nombreux modèles ont été proposés, certains exploitant l'architecture du Transformer (Vaswani et al., 2017) pour produire des représentations génériques, ou contextualisés à l'aide du mécanisme d'auto-attention. L'effort communautaire est tel qu'au-

1. Onglet "French" sur le lien suivant : `https://huggingface.co/spaces/mteb/leaderboard`

jourd'hui, on peut trouver des centaines de milliers de modèles avec diverses architectures, mono ou multi-lingues, pré-entraînés ou spécialisés (Naseem et al., 2021; Ding et al., 2023).

Dans ce travail, l'objectif principal est d'aider la communauté francophone à sélectionner les meilleurs modèles pour le français avec les résultats d'une expérience à large échelle. Nous souhaitons une comparaison globale, incluant les modèles de l'état de l'art, et ciblée, en couvrant des besoins spécifiques, e.g. pour une tâche particulière, ou ciblant un modèle open source, léger, etc. Pour atteindre cet objectif, nos efforts se portent d'abord sur une vaste collecte de données, suivi d'une sélection de modèles variés, puis sur une évaluation complète. Les jeux de données choisis couvrent plusieurs tâches. Nous en créons trois nouveaux pour compléter la collection. Les méthodes d'*embeddings* retenues sont diverses, comprenant les modèles français et multilingues les plus performants. Par l'analyse des résultats, nous confirmerons certaines conclusions connues telles que la corrélation entre les performances et les dimensions des modèles et nous découvrirons également des nuances intéressantes mettant en valeur des modèles plus légers, ouverts et économes en calculs. Notre implémentation est open source et est associée à un classement public, qui permettront aux résultats d'évoluer avec l'arrivée de nouveaux modèles ou l'ajout de nouveaux jeux d'évaluation à l'avenir.

2 Travaux connexes

Méthodes de représentation de phrases Les représentations de phrases sont nécessaires pour de nombreuses tâches de traitement automatique du langage (TALN), telles que la similarité textuelle sémantique (STS) et la recherche d'information. De nombreux modèles ont été proposés dans la littérature, exploitant des stratégies d'agrégation d'*embeddings* de mots (Devlin et al., 2019; Muennighoff, 2022) ou d'apprentissage orienté sur la similarité (Reimers et Gurevych, 2019) avec un objectif contrastif : augmenter le score de similarité entre paires de phrases sémantiquement proches et la réduire entre d'autres (Gao et al., 2021; Wang et al., 2022; Zhang et al., 2023). D'autres méthodes utilisent un apprentissage à deux étapes impliquant de la distillation de connaissances (Chen et al., 2024; Lee et al., 2024) ou encore du "*prompt engineering*" (Wang et al., 2023). Peu de modèles francophones ont été proposés dans la littérature (Martin et al., 2019; Le et al., 2020). La majorité des plus performants pour la représentation de phrases ont été développés par la communauté open source [2], en spécialisant des modèles comme *CamemBERT* (Martin et al., 2019) ou *CroissantLLM* (Faysse et al., 2024).

Benchmarks Les modèles d'*embeddings* sont souvent comparés sur des tâches spécifiques, telles que la recherche d'information ou le reclassement (reranking) (Thakur et al., 2021; Wang et al., 2021). Certains travaux évaluent ces modèles sur plusieurs tâches (Wang et al., 2018; et al., 2022; Conneau et Kiela, 2018) ou comparent des combinaisons d'*embeddings* (García-Ferrero et al., 2021). Le benchmark le plus complet à ce jour est MTEB (Muennighoff et al., 2022). Ce "Massive Text Embedding Benchmark" regroupe une cinquantaine de jeux d'évaluation ciblés sur 8 tâches de TALN, mais il présente une limite majeure : il est principalement concentré sur l'anglais. Certaines initiatives ont déjà étendu le benchmark à d'autres langues, comme le chinois (Xiao et al., 2024) et l'allemand (Wehrli et al., 2024). Nos travaux s'inscrivent dans la même direction avec pour cible le français.

2. C.f. hub Huggingface : *sentence-camenbert*, *sentence_croissant_alpha_v0.3*, *Solon-embeddings-large-0.1*, etc.

3 MTEB-FR

Cette section présente les jeux de données et modèles évalués dans notre benchmark. Nous listons également les questions de recherche à aborder dans la discussion. Note : le papier fait référence à de nombreuses annexes qui sont disponibles au lien `https://github.com/Lyon-NLP/mtebscripts/tree/main/paper_EGC_mteb_french`.

3.1 Les données du benchmark

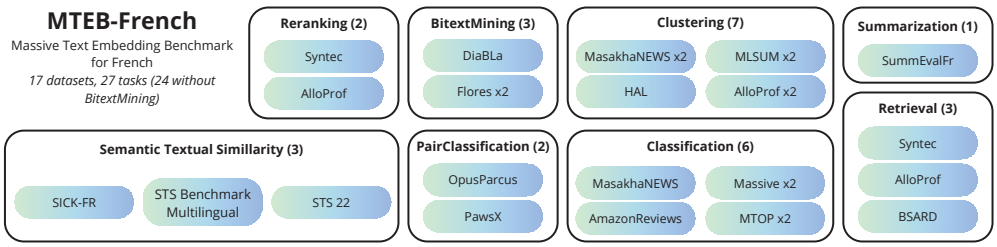

FIG. 1 – *Données du benchmark MTEB-FR. Certains jeux de données ont plusieurs évaluations possibles (e.g. Massive Intent et Massive Scenario).*

De la même manière que MTEB (Muennighoff et al., 2022), nous évaluons la capacité des modèles à produire des représentations pertinentes pour 8 tâches distinctes. Ces tâches sont évaluées à l'aide de 17 jeux de données sélectionnés ou créés dans ce travail.

3.1.1 Huit tâches de traitement automatique du langage

La tâche de **classification** évalue qu'un modèle produise des vecteurs qui aident les classifieurs à associer les textes à des classes pertinentes. Le **clustering** garantit que des clusters de phrases sémantiquement proches puissent être construits avec les représentations. La **classification de paires (pair classification)** évalue si un modèle génère des représentations vectorielles proches pour les textes qui contiennent les mêmes informations, et des représentations éloignées sinon. La tâche de **recherche d'information (retrieval)** encourage les modèles à créer des représentations de documents et de requêtes telles que la similarité requête-document soit corrélée avec la pertinence. Le **reranking** est très proche de la recherche d'information mais se focalise sur un sous-ensemble de documents déjà partiellement pertinents. La **similarité textuelle sémantique (STS)** évalue la capacité à générer des représentations pour lesquelles la distance détermine l'intensité de la relation. La tache de **résumé (summarization)** évalue la capacité d'un modèle à produire des vecteurs de représentation pour un texte et son résumé qui sont proches si le résumé est pertinent. Enfin, le **bitext mining** évalue la capacité d'un modèle à produire des représentations vectorielles avec une similarité élevée pour une paire de phrases équivalentes dans des langues différentes. Des informations plus détaillées sur les tâches sont disponibles dans l'article original de Muennighoff et al. (2022).

Jeu de données	Syntec	HAL	SummEvalFr
Taille	100 requêtes, 90 documents	26233 échantillons, 10 classes	100 textes, 1100 résumés humains, 1600 résumés automatiques
Processus de création	Scraping convention collective Syntec. Article utilisés comme documents. Requêtes créées.	Scraping articles HAL avec *id*, *titre* et *domaine*. Déduplication, filtrage linguistique et sous-échantillonnage.	Traduction de l'anglais vers le français avec Deepl du jeu SummEval original.
Processus d'annotation	4 annotateurs sélectionnent des articles et rédigent des questions en lien.	Annotations déjà fournies par les auteurs sur HAL.	Processus d'annotation détaillé par Fabbri et al. (2021).
Contrôles qualité	Vérification humaine des annotations.	Test de classification et de topic modeling avec des modèles de base.	Corrélation scores BLEU et ROUGE. Évaluation de la traduction par LLM-as-a-judge et vérification humaine.

TAB. 1 – *Détails sur les trois nouveaux jeux de données créés dans nos travaux.*

3.1.2 Évaluations et jeux de données

Nous avons identifié 7 jeux de données pertinents pour le français dans le banc d'évaluation MTEB original, et les avons complétés avec 7 jeux de données externes proposés dans la littérature, tels que BSARD (Louis et Spanakis, 2022) et Alloprof (Lefebvre-Brossard et al., 2023). Enfin, nous en avons créé 3 nouveaux : Syntec, HAL, et SummEvalFr[3]. L'ensemble des jeux d'évaluation de MTEB-FR est présenté sur la Figure 1. Le Tableau E en annexe donne des détails statistiques de chacun d'entre eux, ainsi que les références. Au total, les 17 jeux de données sont déclinés sur 27 évaluations car certains sont utilisés pour plusieurs tâches. Par exemple, MasakhaNEWS est utilisé à la fois pour la classification (*MasakhaNEWSClassification*) et le clustering (*MasakhaNEWSClusteringS2S* et *MasakhaNEWSClusteringP2P*).

Nous espérons que les nouveaux jeux de données introduits puissent être utilisés au delà du benchmark. Nous indiquons le processus de collecte, d'annotation et les nombreuses analyses de qualité réalisées dans l'Annexe 1, et les résumons dans le Tableau 1.

3.1.3 Note à propos de la tâche de reranking

Telle qu'évaluée dans MTEB, cette tâche nécessite un jeu de données composé d'un ensemble de requêtes, d'un ensemble de documents "liés" mais dont seule une partie est finalement pertinente. Malgré nos efforts, nous n'avons trouvé aucun jeu de données français qui présente nativement une telle structure. Ainsi, pour évaluer cette tâche, nous avons construit des données en se basant sur *Syntec* et *Alloprof*. Ces jeux comportent déjà des requêtes et des documents pertinents étiquetés. Des documents "liés" non pertinents sont ajoutés en utilisant le processus suivant : nous appliquons la méthode de recherche d'information BM25 (Robertson et Jones, 1976) basée sur les mots (pour éviter tout biais avec les modèles d'*embeddings*) afin de sélectionner les 10 documents avec le plus haut score et étiquetés comme non-pertinents pour constituer les échantillons négatifs. Nous rendons également les jeux de données obtenus *SyntecReranking* et *AlloprofReranking* disponibles sur le hub HuggingFace[3].

3. Les jeux de données sont disponibles à l'adresse suivante : `https://huggingface.co/lyon-nlp`

3.2 Modèles

Pour la comparaison sur MTEB-FR, nous avons sélectionné plusieurs modèles afin de remplir trois objectifs : **(i) Quantité** - l'idée est de comparer un nombre important de modèles (51 au total) pour fournir des résultats complets, facilitant la sélection pour la communauté francophone. **(ii) Pertinence** - Il est impératif d'inclure les meilleurs modèles. Nous avons choisi le haut du classement MTEB (Muennighoff et al., 2022), principalement parmi les modèles multilingues et quelques modèles anglais pour évaluer les capacités de transfert de langue. De plus, nous avons inclus des modèles emblématiques francophones basés sur le Transformer tels que *CamemBERT* (Martin et al., 2019), *FlauBERT* (Le et al., 2020) et même le récent *CroissantLLM* (Faysse et al., 2024). **(iii) Variété** - Divers types de modèles ont été inclus pour offrir une analyse approfondie de l'impact des caractéristiques sur la performance.

Pour ce troisième objectif, nous listons les caractéristiques étudiées et que nous discuterons avec les résultats. ***Dimension d'embeddings :*** Cet élément critique impacte l'expressivité de la représentation et, dans les applications pratiques, les coûts de stockage et de calculs. Nous avons sélectionné des modèles avec des dimensions allant de 384 à 4096. ***Longueur de séquence :*** C'est le nombre de 'tokens' qu'un modèle peut traiter en entrée. Il impacte l'unité qui peut être encodée (phrase, paragraphe, document). Parmi les méthodes sélectionnées, ce critère varie de 128 tokens à 32768. ***Paramètres du modèle :*** Souvent corrélé aux caractéristiques ci-dessus, le nombre de paramètres affecte la facilité d'utilisation sur des machines économes en ressources. Les modèles sélectionnés ont de 20 millions ($\sim$100 Mo en float32) à 7 milliards ($\sim$28 Go) de paramètres. ***Langue :*** Certains modèles sont monolingues, d'autres multilingues. La langue est généralement acquise lors du pré-entraînement, mais les modèles se familiarisent avec de nouvelles langues lors de la spécialisation (fine-tuning). Nous avons sélectionné des modèles français, ainsi que des modèles bilingues ou multilingues. Nous avons également inclus quelques modèles indiqués comme étant monolingue anglais (par exemple *all-MiniLM-L12-v2*[4]). ***Types de modèles :*** Il existe plusieurs stratégies pour générer des *embeddings* de texte, comme l'agrégation (par exemple la moyenne d'*embeddings* de mots), ou par apprentissage supplémentaire sur une tâche de similarité de phrases. Nous avons inclus des modèles de tous types, en résumant cette information selon deux critères : (i) spécialisé (fine-tuning) vs pré-entraîné, et (ii) entraîné pour la similarité de phrases.

Les modèles sélectionnés sont visibles sur la Figure 2, et toutes leurs caractéristiques sont résumées dans le Tableau F en annexe. En résumé, la sélection inclut les meilleurs modèles du framework SentenceTransformers (Reimers et Gurevych, 2019), les modèles de TALN français les plus populaires (Le et al., 2020; Martin et al., 2019), leurs variantes optimisées pour la similarité sémantique (Reimers et Gurevych, 2019), de nombreux modèles multilingues performants (e.g *e5* et *t5*), des variantes de *Bloom* (Zhang et al., 2023), des modèles récents basés sur des Large Language Models (LLMs) (Wang et al., 2023; Faysse et al., 2024) et enfin les modèles propriétaires d'OpenAI, Cohere et Voyage. Certains sont sélectionnés dans plusieurs tailles pour isoler efficacement l'effet de la dimension. Nous fournissons des informations sur les licences des modèles telles que rapportées dans le hub HuggingFace. Cependant, nous encourageons les lecteurs à effectuer des recherches plus approfondies avant d'utiliser un modèle.

4. https://huggingface.co/sentence-transformers/all-MiniLM-L12-v2

3.3 Évaluation et questions de recherche

Par souci d'homogénéité, les modèles sont évalués avec les métriques originales de MTEB : l'Accuracy pour la classification, le score F1 pour le bitext mining, la précision moyenne (AP) pour la classification de paires, la mesure V pour le clustering, la Mean Average Precision (MAP) pour le reranking, le NDCG@10 pour la recherche d'information, et la corrélation de Spearman entre score de vérité terrain et similarité du cosinus des embeddings pour les tâches de résumé et STS. La tâche bitext mining est exclue des scores de performance moyens [5], car elle évalue l'aspect cross-lingue, et nous souhaitons conclure sur la performance francophone.

En utilisant les résultats finaux, notre objectif sera de répondre aux questions de recherche suivantes. **Q1 :** Un modèle domine-t-il sur toutes les tâches ? Nous essayerons de savoir si un modèle est statistiquement meilleur que les autres pour le français, l'objectif sera également d'analyser les performances des modèles par tâche pour faciliter le choix dans des applications spécifiques. **Q2 :** Existe-t-il des liens entre les caractéristiques du modèle et ses performances ? Dans la section précédente, nous avons indiqué le travail important effectué pour rassembler les caractéristiques de tous les modèles évalués. L'objectif ici sera d'analyser leur impact sur les performances. **Q3 :** Les modèles monolingues ont-ils des capacités multilingues ? Nous interrogeons la capacité d'un modèle entraîné exclusivement dans une langue à généraliser dans une autre langue. **Q4 :** Existe-t-il des corrélations entre les jeux de données vis-à-vis du classement des modèles ? Nous regarderons les similitudes dans la façon qu'ont les jeux de données de classer les modèles.

4 Résultats et discussion

Les performances des modèles sur les différentes tâches sont présentées dans les Tableaux H, I, J, K et L en annexe. La Figure 2, elle, résume ici la comparaison globale à l'aide d'un diagramme de différence critique des rangs moyens des modèles. Dans cette section, nous présentons les résultats à travers le prisme de nos questions de recherche.

Q1 : Y a-t-il un modèle qui se démarque sur toutes les tâches ?

Aucun modèle ne domine sur toutes les tâches, même si le modèle *text-embedding-3-large* occupe aisément la première place en moyenne (voir Tableau H en annexe et Figure 2). Il est premier pour les tâches de classification et de reranking. Sur le clustering, c'est *text-embedding-ada-002* qui est le meilleur. Les modèles *voyage-code-2*, *text-embedding-3-small* et *mistral-embed* se partagent la première position pour la recherche d'information. Pour la classification de paires, c'est *laser2* qui domine. Enfin, *sentence-camembert-large* est en tête sur la tâche STS et *multilingual-e5-small* est le meilleur pour le résumé automatique.

Globalement, les modèles les plus performants sont ceux d'OpenAI *text-embedding-3-large*, *text-embedding-3-small* et *text-embedding-ada-002*. Il est intéressant de noter que de nombreux modèles ne présentent pas d'écarts de performance significatifs entre leurs versions de base et large. Certains modèles français open-source se démarquent, comme *Solon-embeddings-large-0.1*, *sentence_croissant_alpha_v0.3* et *sentence-camembert-large*.

5. Nous présentons quand même les résultats dans le tableau K en annexe pour information.

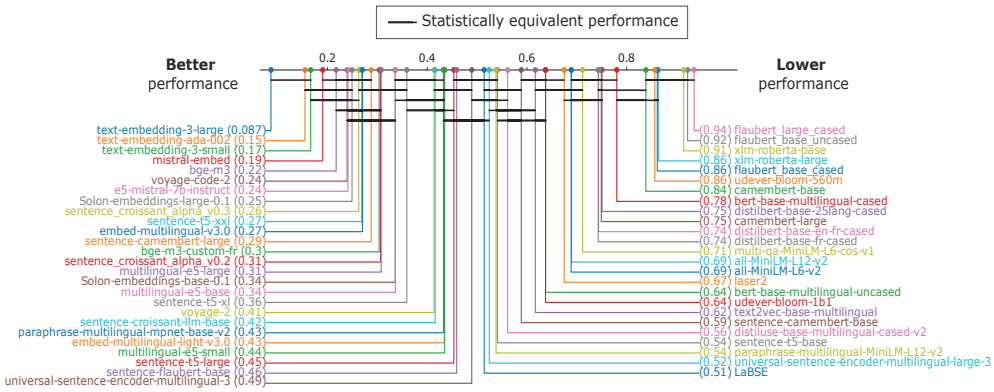

FIG. 2 – *Diagramme de différence critique des rangs des modèles. L'axe représente le rang moyen normalisé des modèles (les meilleurs modèles sont disposés à gauche). Les barres noires indiquent que la différence de rang des modèles n'est pas statistiquement significative, c'est-à-dire inférieure à la différence critique pour une p-value de 0.05.*

Q2 : Y a-t-il des liens entre caractéristiques des modèles et performance ?

Les corrélations de Spearman entre le rang moyen des modèles et leurs caractéristiques sont les suivantes : *entraîné pour la similarité de phrases* (0,727), *spécialisé vs pre-entraîné* (0,544), *nombre de paramètres* (0,49), *dimension des embeddings* (0,452), *modèle propriétaire* (0,449), *longueur de séquence* (0,336), *modèle multilingue* (0,103), *modèle anglais* (0,025), *modèle français* (-0,134), *modèle bilingue* (-0,135). Pour compléter ces chiffres, toutes les corrélations croisées entre les caractéristiques sont reportées dans la Figure H en annexe.

Comme attendu, le score est le plus positivement corrélé au fait que les modèles aient été entraînés pour la similarité de phrases, et au critère un peu plus général *spécialisé vs pré-entraîné*. Les seuls modèles très performants uniquement pré-entraînés sont de la famille *e5*, où le pré-entraînement est en fait optimisé pour la similarité. À l'inverse, les modèles pré-entraînés sur des tâches comme le "Masked Language Model" (Devlin et al., 2019) semblent moins bon. Nous observons aussi des corrélations entre performance et dimension, notamment la taille des *embeddings* et le *nombre de paramètres*. Cela apparaît clairement sur le classement relatif des modèles *e5* et *t5*. Certains petits modèles se comportent cependant très bien sur le benchmark, comme *Multilingual Universal Sentence Encoder* ou *Solon-embeddings-base-1.0*. La *longueur de séquence* est moins corrélée à la performance, ce qui s'explique en partie par le fait que de nombreux jeux contiennent des textes relativement petits (voir

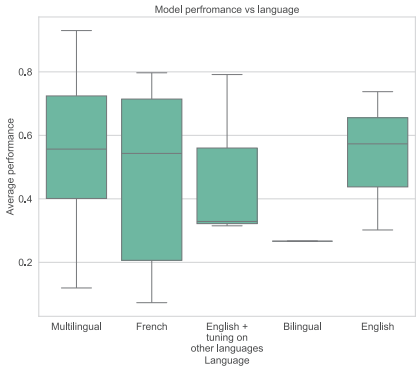

FIG. 3 – *Performances des modèles en fonction de la langue des données d'entraînement.*

Tableau E en annexe montrant que 15 jeux de données ont moins de 50 tokens par échantillon en moyenne). Les résultats relatifs à la langue sont surprenant mais, en réalité, dominés par l'effet des autres caractéristiques. Enfin, nous soulignons que les modèles propriétaires fonctionnent bien sur le benchmark (*text-embeddings*, *mistral-embed* et *voyage*) mais nous manquons d'information sur leurs caractéristiques et sur leur entraînement. À mesure que de nouveaux modèles open source performants seront ajoutés, nous pouvons nous attendre à ce que cette corrélation diminue comme c'est le cas dans le domaine des Large Language Models.

Q3 : Les modèles monolingues ont-ils des capacités multilingues ?

On constate l'absence de corrélation claire entre la langue d'entraînement et les performances sur le français, comme le montre l'écart interquartile important dans la Figure 3. En outre, certains modèles monolingues anglais tels que *voyage-code-2* montrent de très bons résultats sur des jeux de données français, à l'inverse de certains modèles français tels que *flaubert* et *distilbert-base-fr-cased*. Cela s'explique par le fait qu'une grande partie des modèles français sélectionnés sont uniquement pré-entraînés et génèrent des *embeddings* de phrase par agrégation. Seuls quelques modèles avec un entraînement spécifique à la modélisation de phrases conduisent à des plongements de bonne qualité (Gao et al., 2021). Ceci est confirmé par les excellents résultats de *Solon-embeddings-large-0.1*, *sentence_croissant_alpha_v0.3* et *sentence-camembert-large*. Enfin, il convient de noter qu'une partie importante des données utilisées pour spécialiser les modèles français provient en fait de données traduites automatiquement (May, 2021). Malgré les progrès considérables de la traduction automatique, il est souvent reconnu que ce processus entraîne une réduction des performances finales (Barbosa et al., 2021).

Pour aller plus loin sur l'analyse linguistique, nous avons regardé les autres benchmarks MTEB (Polonais/PL, Anglais/EN et Chinois/ZH). Certains modèles sont évalués sur plusieurs d'entre eux. Nous avons pu noter que les rangs normalisés des modèles varient beaucoup d'une langue à l'autre, et il y a notamment moins de corrélation entre les classements MTEB-ZH et MTEB-FR qu'entre MTEB-EN et MTEB-FR. Par exemple *text-embedding-3-large* performe très bien sur le français et l'anglais mais est dominé sur le chinois. D'autre modèles comme *multilingual-e5-large* se classent bien et de façon similaire sur tous les benchmarks.

Q4 : Certains jeux de données sont-ils redondants en terme de classement des modèles ?

La corrélation des jeux de données par rapport au classement des modèles est présentée sur la Figure 4. À l'exception de deux jeux *MasakhaNEWSClusteringP2P* et *SummEvalFr*, les corrélations sont en moyenne élevées car les bons modèles le sont de façon consistante sur le benchmark. Des petits groupes (e.g. *MassiveScenarioClassification* / *MTOPDomainClassification* / *MTOPIntentClassification*) présentent des corrélations particulièrement élevées ($\sim$0,95). A l'avenir, si le benchmark est complété par d'autres jeux de données, la suppression de certains d'entre eux pourra être envisagée. D'autre part, on peut noter des corrélations en blocs. Les données étant organisées par tâche sur la Figure, cela montre que les modèles se comportent de manière plus similaire au sein d'une même tâche (c.f. triangle de corrélations retrieval ou classification) qu'entre deux différentes. On souligne alors l'importance de cette variété dans le benchmark pour aider à sélectionner des modèles à usage général ou bien ciblés.

Nous réalisons aussi l'analyse complémentaire, i.e. des corrélations des modèles par rapport aux performances sur les différents jeux de données (Figure I en annexe). De fortes similitudes émergent entre les variantes de mêmes modèles (e.g. *DistilBERT*, *sentence-croissant*, *sentence-t5*, *e5*, etc.). Des corrélations sont aussi généralement observées parmi les modèles entraînés via la librairie SentenceTransformers (Reimers et Gurevych, 2019), ainsi que des modèles propriétaires, par exemple de Cohere et OpenAI. Les modèles spécialisés sur la similarité des phrases montrent une corrélation faible avec les modèles pré-entraînés.

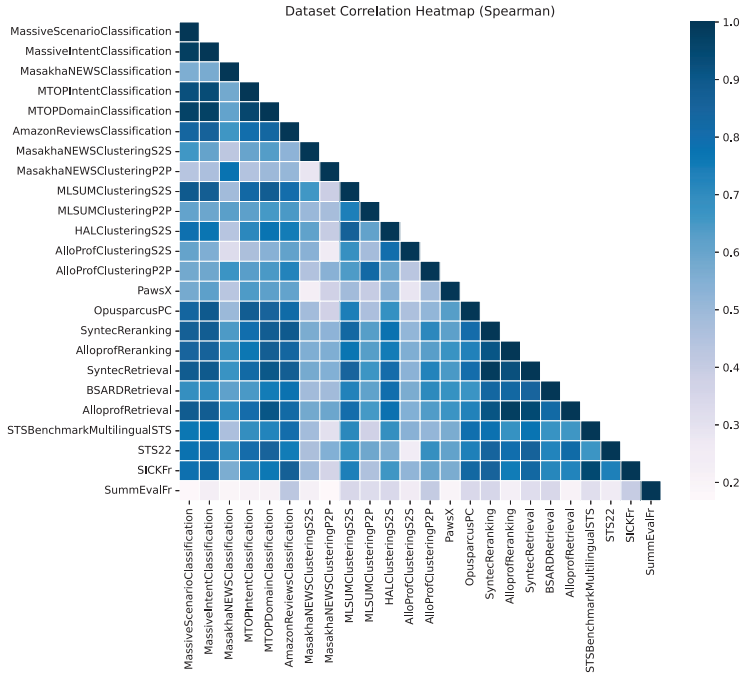

FIG. 4 – *Heatmap représentant les corrélations de Spearman entre jeux de données par rapport à l'ordre dans lequel ils classent les différents modèles.*

5 Conclusion et perspectives

Dans ce papier, nous introduisons une large comparaison de méthodes de représentation de texte en français afin de permettre à la communauté scientifique et à l'industrie de sélectionner les plus pertinentes en fonction de leurs besoins spécifiques. De nombreuses données (dont la qualité est contrôlée) sont collectées ou créés, pour constituer une évaluation globale sur 27 jeux de données. Celle-ci est réalisée sur une sélection de 51 modèles, comprenant l'état de l'art français et multilingues. Après une analyse approfondie des résultats, quelques modèles propriétaires comme ceux d'OpenAI se dégagent. Cependant, de très bon compétiteurs open source et produisant des vecteurs de petite dimension sont également mis en lumière. Notre travail ouvre plusieurs possibilités pour des améliorations futures.

D'abord, nous notons la disponibilité limitée de ressources nativement en français. Il y a évidemment beaucoup moins de modèles pour le français que pour l'anglais. La plupart des modèles français identifiés ont été entraînés avec des architectures ou des méthodes plus "anciennes", contrairement aux modèles multilingues récents (e.g. *e5-mistral-7b-instruct* par Wang et al. (2023)). Nous voyons cependant de nouvelles solutions émerger (Faysse et al., 2024) qui viennent concurrencer fortement les modèles propriétaires. Les limitations de ressources concernent également les jeux de données. Pour obtenir une diversité de tâches, nous avons par exemple construit des données pour le reranking en ré-utilisant des données de recherche d'information. Ce processus entraîne un biais avec une performance corrélée entre les deux tâches. Nous espérons que des ressources supplémentaires seront développées par la communauté francophone pour enrichir la comparaison, en ajoutant de la diversité.

La deuxième limite est le maintien de la validité des résultats dans le temps car le domaine évolue rapidement. Notre code, qui étend la librairie MTEB, ainsi que le classement des modèles sont ouvert à la communauté. Cet effort facilitera l'évaluation de nouvelles ressources (modèles et données) pour maintenir le travail à jour.

Troisièmement, le benchmark est exposé à la possibilité de "contamination". Les modèles qui utilisent les jeux d'entraînement des données du benchmark peuvent voir leurs performances s'améliorer considérablement. Ainsi, la performance des méthodes ne communiquant pas sur leur entraînement (e.g. modèles propriétaires) est difficile à interpréter. Il serait intéressant à l'avenir de proposer un outil calculant la similarité entre les données d'entraînement d'un modèle et les données de test du benchmark pour vérifier la capacité de généralisation.

Ensuite, comme dans la version originale de la MTEB, notre comparaison se concentre principalement sur les *embeddings* de phrases. De nouvelles tâches pourraient être ajoutées pour couvrir les *embeddings* de mots et, par conséquent, d'autres tâches de NLP. Pour terminer, nous espérons voir l'émergence d'autres variantes de MTEB dans de nouvelles langues pour effectuer des évaluations encore plus complètes des modèles multilingues.

Références

Barbosa, A., M. Ferreira, R. Ferreira Mello, R. Dueire Lins, et D. Gasevic (2021). The impact of automatic text translation on classification of online discussions for social and cognitive presences. In *LAK21 : 11th International Learning Analytics and Knowledge Conference*, LAK21, New York, NY, USA, pp. 77–87. Association for Computing Machinery, doi: 10.1145/3448139.3448147.

Chen, J., S. Xiao, P. Zhang, K. Luo, D. Lian, et Z. Liu (2024). Bge m3-embedding : Multilingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation.

Conneau, A. et D. Kiela (2018). Senteval : An evaluation toolkit for universal sentence representations. *ArXiv abs/1803.05449*.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2019). Bert : Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*.

Ding, N., Y. Lin, Z. Liu, et M. Sun (2023). Sentence and document representation learning. In *Representation Learning for Natural Language Processing*, pp. 81–125. Springer Nature Singapore Singapore.

et al., A. S. (2022). Beyond the imitation game : Quantifying and extrapolating the capabilities of language models. *ArXiv abs/2206.04615*.

Fabbri, A. R., W. Kryściński, B. McCann, C. Xiong, R. Socher, et D. Radev (2021). Summeval : Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics 9*, 391–409.

Faysse, M., P. Fernandes, N. M. Guerreiro, A. Loison, D. M. Alves, C. Corro, N. Boizard, J. Alves, R. Rei, P. H. Martins, A. B. Casademunt, F. Yvon, A. F. T. Martins, G. Viaud, C. Hudelot, et P. Colombo (2024). Croissantllm : A truly bilingual french-english language model.

Gao, T., X. Yao, et D. Chen (2021). Simcse : Simple contrastive learning of sentence embeddings. In *Conference on Empirical Methods in Natural Language Processing*.

García-Ferrero, I., R. Agerri, et G. Rigau (2021). Benchmarking meta-embeddings : What works and what does not. In M.-F. Moens, X. Huang, L. Specia, et S. W.-t. Yih (Eds.), *Findings of the Association for Computational Linguistics : EMNLP 2021*, Punta Cana, Dominican Republic, pp. 3957–3972. Association for Computational Linguistics, doi: 10.18653/v1/2021.findings-emnlp.333.

Le, H., L. Vial, J. Frej, V. Segonne, M. Coavoux, B. Lecouteux, A. Allauzen, B. Crabbé, L. Besacier, et D. Schwab (2020). Flaubert : Unsupervised language model pre-training for french.

Lee, C., R. Roy, M. Xu, J. Raiman, M. Shoeybi, B. Catanzaro, et W. Ping (2024). Nv-embed : Improved techniques for training llms as generalist embedding models.

Lefebvre-Brossard, A., S. Gazaille, et M. C. Desmarais (2023). Alloprof : a new french question-answer education dataset and its use in an information retrieval case study.

Louis, A. et G. Spanakis (2022). A statutory article retrieval dataset in French. In S. Muresan, P. Nakov, et A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, Dublin, Ireland, pp. 6789–6803. Association for Computational Linguistics, doi: 10.18653/v1/2022.acl-long.468.

Martin, L., B. Muller, P. O. Suarez, Y. Dupont, L. Romary, E. V. de la Clergerie, D. Seddah, et B. Sagot (2019). Camembert : a tasty french language model. In *Annual Meeting of the Association for Computational Linguistics*.

May, P. (2021). Machine translated multilingual sts benchmark dataset.

Mikolov, T., K. Chen, G. S. Corrado, et J. Dean (2013). Efficient estimation of word representations in vector space. In *International Conference on Learning Representations*.

Muennighoff, N. (2022). Sgpt : Gpt sentence embeddings for semantic search. *arXiv preprint arXiv :2202.08904*.

Muennighoff, N., N. Tazi, L. Magne, et N. Reimers (2022). Mteb : Massive text embedding benchmark. In *Conference of the European Chapter of the Association for Computational Linguistics*.

Naseem, U., I. Razzak, S. K. Khan, et M. Prasad (2021). A comprehensive survey on word

representation models : From classical to state-of-the-art word representation language models. *Transactions on Asian and Low-Resource Language Information Processing 20*(5), 1–35.

Reimers, N. et I. Gurevych (2019). Sentence-bert : Sentence embeddings using siamese bert-networks. In *Conference on Empirical Methods in Natural Language Processing*.

Robertson, S. E. et K. S. Jones (1976). Relevance weighting of search terms. *J. Am. Soc. Inf. Sci. 27*, 129–146.

Thakur, N., N. Reimers, A. Rücklé, A. Srivastava, et I. Gurevych (2021). BEIR : A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Vaswani, A., N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, et I. Polosukhin (2017). Attention is all you need. In *Neural Information Processing Systems*.

Wang, A., A. Singh, J. Michael, F. Hill, O. Levy, et S. R. Bowman (2018). Glue : A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP@EMNLP*.

Wang, K., N. Reimers, et I. Gurevych (2021). TSDAE : Using transformer-based sequential denoising auto-encoderfor unsupervised sentence embedding learning. In M.-F. Moens, X. Huang, L. Specia, et S. W.-t. Yih (Eds.), *Findings of the Association for Computational Linguistics : EMNLP 2021*, Punta Cana, Dominican Republic, pp. 671–688. Association for Computational Linguistics, doi: 10.18653/v1/2021.findings-emnlp.59.

Wang, L., N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, et F. Wei (2022). Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv :2212.03533*.

Wang, L., N. Yang, X. Huang, L. Yang, R. Majumder, et F. Wei (2023). Improving text embeddings with large language models. *arXiv preprint arXiv :2401.00368*.

Wehrli, S., B. Arnrich, et C. Irrgang (2024). German text embedding clustering benchmark.

Xiao, S., Z. Liu, P. Zhang, N. Muennighoff, D. Lian, et J.-Y. Nie (2024). C-pack : Packaged resources to advance general chinese embedding.

Zhang, X., Z. Li, Y. Zhang, D. Long, P. Xie, M. Zhang, et M. Zhang (2023). Language models are universal embedders. *ArXiv abs/2310.08232*.

Summary

Thousands of textual embedding models are available today. We introduce the first Massive Textual Embedding Benchmark for French so that models can be compared easily. We gather data and create new ones for a global evaluation on 27 datasets associated with 8 NLP tasks. We compare 51 carefully selected models to find the ones that dominate and conclude on the characteristics that make them competitive. Our work comes with an easily usable open source library, and a public leaderboard allowing external contributions.