MEDS - projet méthodologie

Réalisé par:
Titouan Guerin
Yacine Chettab

Encadré par:
Olivier Schwander

03 Décembre 2025

Contents

1 Introduction

L'analyse et l'exploitation automatique de données tabulaires constituent un enjeu majeur en intelligence artificielle et en traitement du langage naturel. Contrairement aux textes libres, les tables combinent une structure explicite (lignes et colonnes) et un contenu sémantiquement riche, souvent hétérogène, ce qui rend leur compréhension algorithmique particulièrement complexe. Les questions posées sur des tables nécessitent un raisonnement multi-niveaux : identification des colonnes pertinentes, filtrage des lignes, application d'opérations arithmétiques ou logiques, et extraction ou synthèse de la réponse correcte.

Ce travail vise à évaluer la capacité de modèles pré-entraînés spécialisés pour les tables à généraliser sur des structures inédites. En effet, les modèles modernes de *Question Answering* (QA) tabulaire, tels que TAPAS Herzig et al., 2020 et TAPEX Liu et al., 2022 sont entraînés sur des ensembles de tables spécifiques et peuvent rencontrer des difficultés lorsqu'ils sont confrontés à de nouvelles structures ou à des entités jamais vues. L'objectif principal de cette étude est donc double : d'une part, quantifier la performance de ces modèles sur des tables et questions non rencontrées lors de l'apprentissage ; d'autre part, identifier leurs limites méthodologiques, en termes de généralisation sémantique, de sensibilité aux variations syntaxiques et de contraintes computationnelles.

Pour atteindre cet objectif, nous adoptons une approche expérimentale systématique: nous analysons les performances des modèles sur un corpus standardisé, *WikiTableQuestions* Pasupat and Liang, 2015. Cette étude permet de dresser un bilan critique des modèles actuels de QA tabulaire et de dégager des pistes d'amélioration pour le développement de systèmes capables de raisonner de manière flexible et efficace sur des données tabulaires.

2 Tâche

2.1 Question Asnwering en NLP

La tâche de *Question Answering* (QA) consiste à produire automatiquement une réponse à une question formulée en langage naturel. Chaque question est accompagnée par des données tabulaires, d'où cette tâche nécessite un raisonnement symbolique et structurel : le modèle doit interpréter la structure de la table, identifier les colonnes pertinentes, effectuer d'éventuelles opérations arithmétiques ou logiques, et enfin extraire ou déduire la réponse. Ce type de raisonnement dépasse la simple correspondance lexicale, et constitue un défi majeur pour les modèles NLP actuels.

Le dataset *WikiTableQuestions* (WTQ) constitue aujourd'hui une référence pour l'étude du QA sur tables semi-structurées. Il vise à tester la capacité des modèles à généraliser vers des structures tabulaires et des entités jamais vues lors de l'apprentissage en répondant à des questions parfois complexes.

2.2 Dataset : WikiTableQuestions

Le dataset WikiTableQuestions a été conçu pour l'apprentissage et l'évaluation de systèmes de QA sur des tables extraites automatiquement de Wikipédia. Il comporte environ 22 033 paires question–réponse réparties sur 2 108 tables distinctes. Chaque table provient d'un article Wikipédia contenant au moins huit lignes et cinq colonnes, afin de garantir une structure minimale d'analyse.

Chaque exemple du dataset contient : un identifiant unique, une question en langage naturel (*utterance*), une table HTML/CSV servant de contexte (*context*) et une ou plusieurs réponses exactes (*targetValue*).

Les données sont fournies au format `TSV`, accompagnées de métadonnées `HTML/CSV` et d'annotations linguistiques issues de *CoreNLP* (tokenisation, lemmatisation, reconnaissance d'entités nommées).

Le découpage officiel du dataset est le suivant :

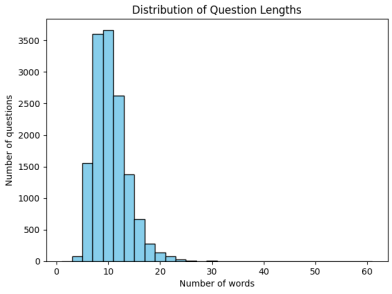
- L'ensemble d'apprentissage : 14 152 exemples;
- Pristine-unseen-tables : 4 344 exemples;
- Pristine-seen-tables : 3 537 exemples;
- 5 fichiers dev: 14114 exemples.

Les tables du jeu de test pristine-unseen n'apparaissent pas lors de l'apprentissage, ce qui permet d'évaluer la capacité de généralisation des modèles à des structures et entités inédites. Les questions couvrent une large diversité d'opérations sémantiques : Lookup (extraction directe d'une cellule), superlatif, comparatif, agrégation ou arithmétique.

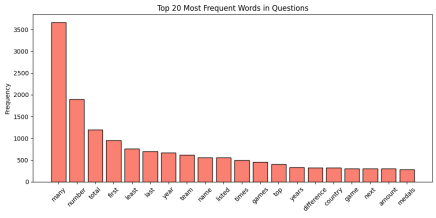
Table 1: Typologie des questions selon leur complexité logique

Type de question	**Fréquence**	**Exemple**
Simple extraction	13.5 %	*Who won the 2012 award?*
Superlatif/Comparatif	24.5 %	*Which country has the highest GDP?*
Agrégation	15.0 %	*What teams participated in 2010?*
Arithmétique	20.5 %	*What is the total duration in years?*
Autre	21.0 %	*What country has 4 consecutive drivers on the roster?*

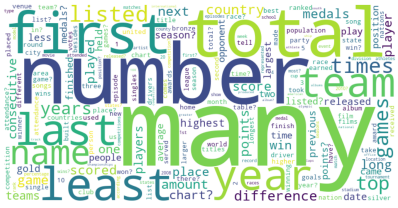
2.2.1 Analyse syntaxique du dataset



(a) Longueur des questions



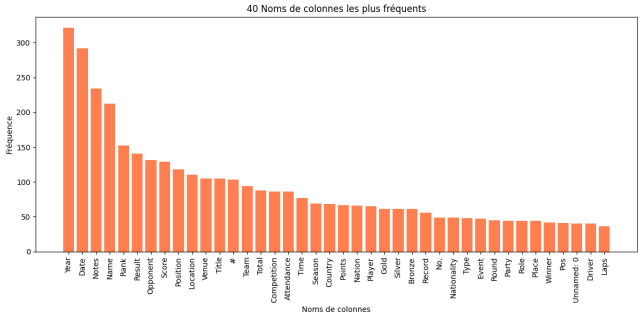
(b) Les 20 mots les plus présents dans les questions



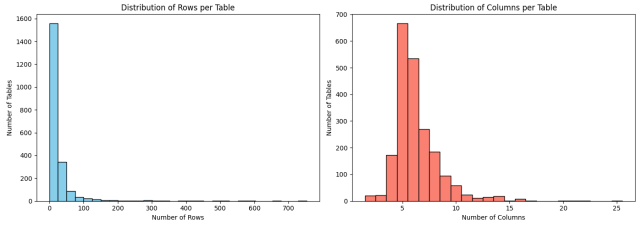
(c) Wordcloud des mots les plus présents dans les questions

Figure 1: Analyses sur les questions dans le dataset

Selon la distribution de la longueur des questions exprimée en nombre de mots, illustrée dans la figure 1a, nous observons que la majorité des questions comptent entre 5 et 15 mots. Ceci suggère que les annotateurs ont privilégié des formulations courtes et directes, souvent en langage journalistique ou encyclopédique. La présence minoritaire de questions très longues (30 mots ou plus) reflète néaumoins la diversité sémantique des questions. Les termes les plus fréquents dans les questions dans la figure 1b, nous révélent la prédominance des thèmes sportifs et historiques dans les tables. La décroissance de leur fréquences illutre un comportement conforme à la loi de Zipf , typique des coprus de language naturel.



(a) Les noms de colonnes les plus commun



(b) Tailles des colonnes

Figure 2: Analyses sur les colonnes dans le dataset

La fréquence des termes le plus courants dans les noms de colonnes des table diminue selon une loi de Zipf confirmant la présence d'un certain nombre de schémas tabulaires réccurents, comme l'indique la figure 2a. La majorité des tables comporte moins de 100 lignes comme l'indique la figure 2b voire souvent moins de 25 lignes, mais certaines peuvent même dépasser 700 lignes. En parallèle, le nombre de colonnes se situe le plus souvent entre 4 et 10 avec une forte concentration autour de 5 et 6 colonnes. Cette structure compacte permet des raisonnements locaux et explicites malgré la limitation de la richesse sémantique exprimée. Nous avons également chercher à savoir si la réponse à une question est une cellule de la table associée, soit via une extraction directe, une question d'un type superlatif ou comparatif. Le taux était à 57.55% ce qui est cohérent avec la table. 1.

2.2.2 Analyse sémantique du dataset

Dans cette section, nous avons entrepris une analyse thématique du dataset à partir de la similarité sémantique des questions. Nous ferons un clustering par question, par table, par la concatenation de la question avec sa table correspondante et enfin par type de question.

Le clustering est fait sur le dataset du train et les clusters obtenus seront imposés sur les autres parties du dataset. Pour le faire, nous avons utilisé le modèle `all-MiniLM-L6-v2` de la bibliothèque `SentenceTransformer`, lequel permet de

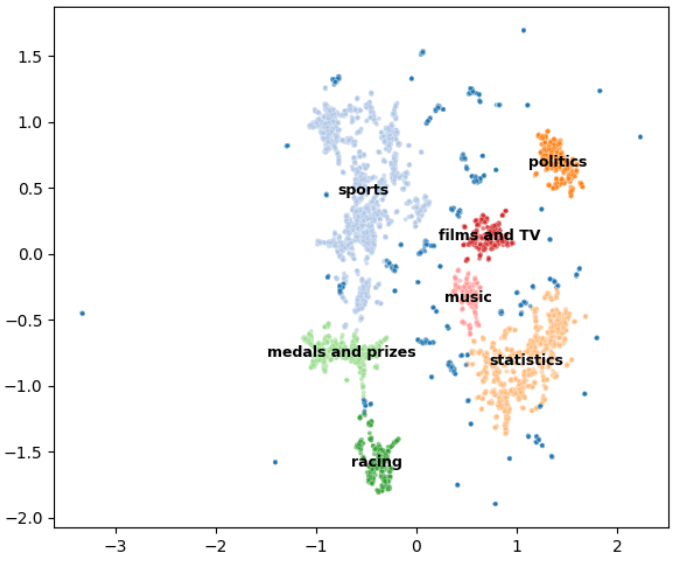


Figure 3: Clustering des questions contextualisées du dataset train

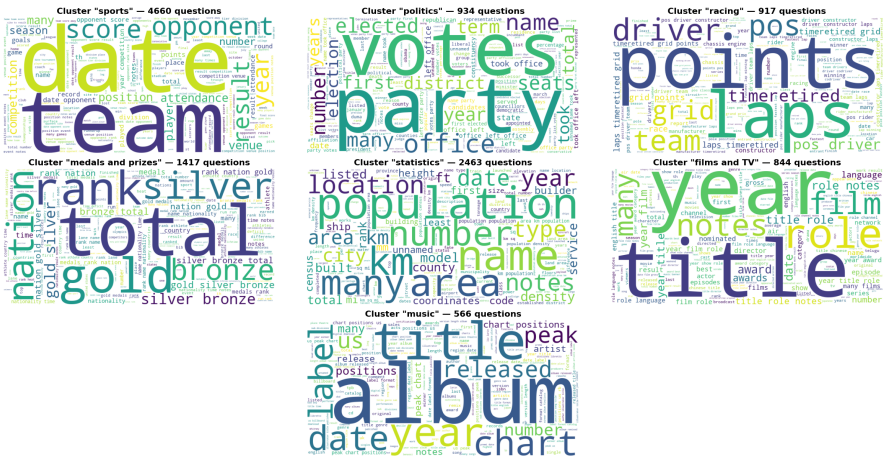


Figure 4: Wordclouds des clusters de train

représenter chaque texte sous la forme d'un vecteur d'embedding de dimension 384. Ces embeddings ont été générés à partir des questions issues du fichier `training.tsv`, concaténées avec les informations de la table correspondante, préalablement aplatie (flatten). Afin de visualiser la structure latente de ces embeddings, une réduction de dimension a été effectuée à l'aide de l'algorithme `UMAP` (60 voisins), ramenant ainsi les embeddings à 2 dimensions. Les données ont été normalisées avant d'être soumises à `DBSCAN`, avec un rayon de voisinage $\varepsilon = 0.158$ et `min_points` $= 200$ avec une distance euclidienne. Ces valeurs d'hyperparamètres ont été déterminées empiriquement après plusieurs itérations expérimentales.

Le clustering obtenu révèle la présence de 7 clusters distincts, auxquels s'ajoute un ensemble de points considérés comme du bruit. Afin d'interpréter le contenu sémantique de chaque groupe, nous avons généré des wordclouds par cluster (voir la figure 4), ce qui nous a permis de leur attribuer un titre. La distribution des questions montre une prédominance marquée du thème *sport*, comme l'illustre la figure 3.

2.2.3 Limites et difficultés du dataset

L'analyse du dataset *WikiTableQuestions* met en évidence un certain nombre de limites structurelles, linguistiques et méthodologiques susceptibles d'influer sur la performance et la généralisabilité des modèles de QA. Les caractéristiques du dataset en font un banc d'essai pertinent pour évaluer la robustesse des modèles de QA hybrides alliant raisonnement logique et représentation continue.

Biais de domaine et déséquilibres thématiques. Les tables du dataset proviennent exclusivement de Wikipédia, ce qui induit une forte concentration autour de thématiques encyclopédiques telles que le sport et la politique. Ce déséquilibre thématique se manifeste dans la distribution des questions, majoritairement centrées sur ces domaines comme l'a montré l'analyse sémantique par *clustering*. Une telle surreprésentation peut conduire les modèles à développer des biais de spécialisation et à moins bien généraliser sur des contextes moins fréquents ou absents du dataset.

Biais linguistique et diversité limitée des formulations. C'est un biais au niveau de la création du dataset. Ces questions ont été rédigées par des annotateurs anglophones via la plateforme *Amazon Mechanical Turk*, dans un style souvent contraint par des consignes d'annotation incitant à la complexité logique (la question doit nécessiter un calcul ou une comparaison). Cela conduit à un langage parfois artificiel, bien que syntaxiquement correct, qui ne reflète pas nécessairement la diversité des formulations naturelles. En outre, la présence marginale de textes multilingues (Afrikaans, Néerlandais, Allemand, etc.) introduit une hétérogénéité lexicale très marginale qui risque d'être considérée comme un bruit.

Biais de structure et de prétraitement. Les tables présentent des incohérences de formattage et de typage : unités mélangées (par exemple, des colonnes *duration* exprimées à la fois en *months* et en *years*), ponctuation irrégulière ou valeurs manquantes. De telles disparités compliquent les étapes de normalisation et de tokenisation, et peuvent altérer la qualité des embeddings textuels et tabulaires. Ces problèmes relèvent moins d'erreurs ponctuelles que d'une variabilité structurelle inhérente aux données réelles issues de Wikipédia.

Problème de généralisabilité. Le découpage officiel du dataset impose une séparation stricte entre les tables vues et non vues durant l'apprentissage. Les modèles sont ainsi confrontés à des structures et entités inédites au moment du test, ce qui constitue un défi majeur en matière de généralisation. Cette propriété rend le dataset scientifiquement pertinent, mais accroît la difficulté du raisonnement symbolique et de la composition sémantique.

Complexité combinatoire et ambiguïtés sémantiques. Certaines questions requièrent des opérations logiques complexes (agrégations imbriquées, filtrages multiples, comparaisons croisées), dont la recherche d'expressions formelles correctes s'apparente à un problème combinatoire à haute dimension. Par ailleurs, de nombreuses questions comportent des ambiguïtés lexicales ou pragmatiques. Ainsi, dans *"Which performer was last in the series?"* (question nu-7 de prestine-unseen-tables), le terme *last* peut renvoyer aussi bien à une position ordinale qu'à un ordre dans un sens temporel, rendant la désambiguïsation difficile même pour des modèles avancés.

Contraintes de représentation et d'intégration tabulaire. Le format `HTML/CSV` des tables ne se prête pas directement aux architectures neuronales, nécessitant une étape préalable de linéarisation ou de conversion en graphe. Or, ces transformations impliquent des compromis : la linéarisation tend à effacer les relations structurelles entre colonnes, tandis que la représentation en graphe accroît la complexité computationnelle et la difficulté d'apprentissage.

3 Approches expérimentales

3.1 Objectifs

Cette étude vise à évaluer de manière rigoureuse les performances de deux modèles spécialisés dans le *Question Answering* sur données tabulaires : **TAPEX** Liu et al., 2022 dans sa version `tapex-large-finetuned-wtq` et **TAPAS** Herzig et al., 2020 dans sa version `tapas-large-finetuned-wtq`. L'analyse porte à la fois sur le cas général, sur des regroupements thématiques (clusters) de paires question-table, ainsi que sur la cardinalité des réponses, qui peuvent être uniques ou multiples.

RQ1. Les modèles sont-ils capables de fournir les bonnes, toutes les bonnes et uniquement les bonnes réponses ?

RQ2. Les modèles présentent-ils un temps d'inférence compatible avec une utilisation pratique ?

Afin d'encadrer conceptuellement l'étude, nous formulons deux hypothèses principales :

(H1) Les modèles pré-entraînés sur des tables structurées surpassent les approches textuelles classiques. Les modèles spécifiquement conçus pour les données tabulaires capturent non seulement le contenu lexical, mais également la structure bidimensionnelle des tables. Les architectures telles que `TAPAS` et `TAPEX` reposent sur des objectifs adaptés (prédiction de cellules, alignement inter-colonnes, complétion tabulaire), leur conférant une meilleure capacité de raisonnement sur les relations internes aux tableaux.

(H2) Les architectures encoder–decoder sont plus robustes aux variations lexicales. Les modèles génératifs comme `TAPEX` disposent d'une flexibilité accrue pour produire des réponses, ce qui les rend potentiellement plus résistants aux reformulations, synonymies ou ambiguïtés. À l'inverse, les modèles encoder-only tels que `TAPAS` reposent sur des mécanismes de classification ou de sélection, limitant leur adaptabilité au langage naturel.

3.2 Expériences

Avant l'application des modèles, un ensemble de traitements préliminaires communs a été réalisé. Les questions et les tables ont été normalisées selon les mêmes règles : mise en minuscules, suppression de la ponctuation non pertinente, harmonisation des formats numériques et uniformisation des pourcentages. Les prédictions finales ont subi une procédure identique afin d'assurer une comparaison équitable. L'évaluation s'appuie principalement sur la métrique d'*Exact Match Accuracy* (EMA), qui mesure la proportion de correspondances exactes entre réponses prédites et réponses de référence pour les réponses singulières. Pour les questions à réponse multiples, nous considèrerons que les réponses partielles sont des réponses correctes.

Les deux modèles diffèrent toutefois par leur mécanisme interne de production de la réponse.

TAPAS encode conjointement table et question suivant une architecture bimodale. Le modèle ne génère pas de texte : il prédit des cellules ainsi qu'un opérateur d'agrégation explicite (None, Count, Sum, Average). La réponse finale doit donc être reconstruite à partir de ces éléments.

TAPEX, à l'inverse, adopte une approche entièrement générative de type encoder–decoder. Le décodeur produit directement une réponse textuelle conditionnée sur la table et la question. Cette formulation permet de traiter naturellement des requêtes impliquant comparaisons, superlatifs ou agrégations implicites, sans dépendre d'une prédiction explicite de cellules.

3.3 Critères d'évaluation

L'évaluation a été menée sur plusieurs milliers d'exemples des jeux d'entraînement et de test. Les deux indicateurs principaux sont :

- EMA pour répondre à **RQ1:** *Les modèles sont-ils capables de fournir les bonnes, toutes les bonnes et uniquement les bonnes réponses ?* ;
- le temps d'inférence moyen par échantillon pour répondre à **RQ2:** *Les modèles présentent-ils un temps d'inférence compatible avec une utilisation pratique ?*

Afin d'identifier les forces et faiblesses propres à chaque modèle selon la nature du raisonnement requis et d'étudier les comportements dans les cas les plus difficiles plutôt que de se limiter à des moyennes globales, plusieurs évaluations parallèles ont été réalisées :

- performances par **cluster sémantique**;
- performances selon la **cardinalité de la réponse** (une valeur unique ou une liste de valeurs) ;
- performances par **type d'opération tabulaire**, à partir d'une annotation manuelle d'un sous-échantillon : agrégation, superlatif, comparatif, lookup, extraction séquentielle (*next*), etc sur deux échantillons de taille 100 chacun, tirés du dataset de l'apprentissage et l'évaluation puis annotés manuellement.

4 Résultats

Le tableau 2 regroupe les résultats permettant de répondre à **RQ1**: *Les modèles sont-ils capables de fournir les bonnes, toutes les bonnes et uniquement les bonnes réponses ?* Les scores d'accuracy sont présentés par cluster, par cardinalité de la réponse et par type d'opération tabulaire.

		TAPEX		TAPAS	
		Train	Test	Train	Test
Par cluster	Sports	0.758	0.699	**0.416**	0.300
	Statistics	0.713	0.741	**0.428**	**0.292**
	Medals and prizes	0.789	**0.679**	0.468	0.315
	Politics	0.736	0.717	0.525	0.368
	Racing	0.777	**0.679**	0.454	0.330
	Films and TV	0.773	**0.567**	**0.424**	**0.254**
	Music	0.768	-	**0.401**	-
	noise	**0.708**	0.720	0.436	0.314
Par cardinalité de la réponse	1	0.77	0.719	0.452	0.329
	≥ 1	0	0	0	0
Par type d'opération tabulaire **(sur un échantillon de taille 100)**	Extraction	**0.66**	0.78	0.66	0.68
	Suivant/Précédent	1	0.75	0.83	0.75
	Comparative	0.73	0.78	0.53	0.42
	Superlative	0.77	0.78	0.66	0.78
	Agrégation	0.75	**0.55**	**0.16**	**0.02**
	Arithmétique	0.76	**0.50**	**0.11**	**0**
	autres	**0.50**	-	**0**	-
Toutes les données		0.746	0.696	0.437	0.318
Temps d'inférence moyen par question			0.128s		0.156s

Table 2: Le temps d'inférence et les scores d'accuracy par cluster, cardinalité de la réponse et type d'opération tabulaire. Les valeurs en **gras** indiquent les cas où le modèle obtient des résultats suffisamment inférieurs à l'accuracy baseline moyenne, tandis que le caractcaractère "-" indique l'absence d'échantillons correspondant à ce cluster dans le jeu en question.

Analyse par cluster Les performances varient significativement selon les clusters thématiques, et ces variations peuvent être interprétées à la lumière des caractéristiques propres à TAPEX et TAPAS. TAPEX obtient ses meilleurs scores à l'entraînement pour les thèmes *Medals and prizes*, *Racing* et *Films and TV* (0,789 ; 0,777 et 0,773). Cependant, ces mêmes thèmes présentent les performances les plus faibles lors de l'évaluation. Cette dégradation s'explique en partie par une forte variabilité lexicale, la présence de noms propres et des structures tabulaires hétérogènes, qui compliquent l'alignement texte-cellule et le raisonnement tabulaire. TAPAS, en revanche, rencontre des difficultés sur les clusters *Statistics* et *Films and TV*, tant à l'entraînement qu'au test. Pour *Statistics*, la forte densité de valeurs numériques et la nécessité de conversions d'unités ou de normalisations complexes limitent sa capacité à effectuer correctement les opérations mathématiques, car TAPAS est principalement conçu pour extraire des informations directement depuis les cellules et n'intègre pas de mécanismes sophistiqués de calcul ou de raisonnement arithmétique. Pour

Films and TV, la variabilité textuelle et la présence fréquente de noms propres compliquent la correspondance exacte entre la question et les cellules du tableau, ce qui illustre la limite de TAPAS dans le traitement de tableaux moins structurés ou fortement hétérogènes.

Analyse par cardinalité de la réponse Les modèles montrent de bonnes performances pour les réponses de cardinalité = 1, TAPEX atteignant 0,77 à l'entraînement et 0,719 au test. Les scores nuls pour la cardinalité ≥ 1 résultent d'une erreur dans notre fonction d'évaluation, qui normalisait incorrectement les réponses multiples, entraînant une comparaison érronée. Malgré cette erreur, la comparaison avec l'accuracy globale suggère que la performance réelle pour les réponses multiples devrait rester inférieure à la baseline moyenne. Cette observation met en évidence la difficulté intrinsèque à générer correctement plusieurs éléments de réponse, et indique que des stratégies dédiées (post-traitement, décodage séquentiel ou agrégation) sont nécessaires pour améliorer les performances sur ce type de requêtes.

Analyse par type d'opération tabulaire Pour cette analyse, nous avons sélectionné deux échantillons (train et test) de 100 données, annoté manuellement. Les différentes classes sont presque équidistribuées, à l'exception de la catégorie *autres*, qui est moins représentée. Les résultats montrent que TAPEX est généralement plus performant sur les opérations plus complexes, telles que l'agrégation et les calculs arithmétiques. Cette supériorité s'explique par l'architecture de TAPEX, conçue pour le raisonnement multi-étapes ce qui lui permet de combiner, croiser et transformer plusieurs cellules pour produire la réponse correcte. À l'inverse, TAPAS présente des performances très limitées pour ces mêmes opérations, avec une *accuracy* de seulement 2% pour les agrégations et 0% pour les calculs arithmétiques. Cela reflète la conception de TAPAS, principalement orientée vers l'extraction directe d'informations à partir des cellules, sans mécanismes sophistiqués de raisonnement ou de calcul sur plusieurs éléments du tableau ou un croissement de l'information.

Temps d'inférence Pour répondre à **RQ2:** *Les modèles présentent-ils un temps d'inférence compatible avec une utilisation pratique?*, nous avons mesuré les durées de traitement par échantillon sur nos ensembles complets. TAPEX présente un temps d'inférence moyen d'environ 0,13s par échantillon lorsqu'il est exécuté sur GPU, calculé à partir d'un temps total d'environ 30 min 9 secondes pour 14 111 échantillons. En revanche, TAPAS affiche des temps d'inférence un peu plus élevés, d'environ 0.156s par échantillon. Ceci montre qu'en moyenne, TAPEX est plus rapide que TAPAS.

5 Conclusion

Ce travail avait pour objectif d'évaluer TAPEX et TAPAS sur des tâches NLP et RI sur des données tabulaire en partie, tant du point de vue de la qualité des

réponses que des temps d'inférence. Nos résultats montrent que les performances varient selon les thématiques mais surtout selon le type des opérations tabulaires et la structure implicite des tables. TAPEX se distingue par une meilleure stabilité sur les opérations nécessitant du raisonnement computationnel, alors que TAPAS demeure plus sensible aux variations lexicales et à l'hétérogénéité des tableaux.

Ces observations ouvrent plusieurs perspectives. Une première consiste à introduire des perturbations lexicales et structurelles (ajout, suppression ou permutation de lignes/colonnes) afin d'évaluer systématiquement la robustesse des architectures. De telles manipulations permettraient de tester l'hypothèse selon laquelle les modèles encodeur–décodeur comme TAPEX sont plus résistants au bruit que les modèles purement encodeurs tels que TAPAS. Une seconde piste réside dans l'intégration de l'apprentissage par renforcement avec retour humain (RLHF), afin d'améliorer la cohérence, la complétude et la sensibilité tabulaire des réponses, notamment pour les questions à réponses multiples.

TAPEX et TAPAS apparaissent comme deux approches pertinentes mais encore limitées pour la modélisation des données tabulaires. L'étude future de leur robustesse face à des perturbations contrôlées, combinée à des méthodes d'ajustement plus fines comme le RLHF, constitue une voie prometteuse pour développer des modèles plus fiables, plus stables et mieux adaptés aux tables bruitées ou imparfaitement structurées.

References

Herzig, J., Nowak, P., Mueller, T., Piccinno, F., & Eisenschlos, J. M. (2020). Tapas: Weakly supervised table parsing via pre-training. *Proceedings of ACL*, 4320–4333.

Liu, Q., Chen, B., Lou, J.-G., Chen, Z., Fu, Y., & Chen, W. (2022). Tapex: Table pre-training via learning a neural sql executor. *Proceedings of ICLR*.

Pasupat, P., & Liang, P. (2015). Compositional semantic parsing on semi-structured tables. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, 1470–1480.