Groupe 5 : SemEval-2025 Tâche 9
Extraction Expliquable de Risques Sanitaires
Rapport Final

Becard (becr7846)[1], **Skrzypczak (skra8835)**[1], **Maurel (maus5144)**[1], and **Jemmali (jemk6351)**[1]

[1]Université de Sherbrooke, Canada

Résumé

La contamination alimentaire représente une menace majeure causant environ 420 000 décès annuels selon Qian et al. (2023). Dans le cadre de la tâche 9 de SemEval-2025 (Randl et al., 2025b), nous proposons une architecture multi-tâches (MTL) basée sur RoBERTa-large pour l'extraction expliquable de risques sanitaires. Notre approche combine la classification sémantique avec une extraction de segments justificatifs au format BIO, optimisée par une pondération adaptative des pertes via l'incertitude homoscédastique. Nos expérimentations démontrent un score F1 global de 0,7937 pour la sous-tâche 1 (ST1) et un score combiné (ST1+BIO) de 0,7606, surpassant significativement les baselines tout en offrant une justification extractive robuste. L'analyse de l'explicabilité révèle une suffisance de 66,90 % et une précision fidèle de 67,96 %, montrant une relation entre la prédiction et le raisonnement interne du modèle.

1 Introduction

Selon Qian et al. (2023), la sécurité alimentaire est un enjeu de santé publique critique nécessitant une surveillance constante. Si l'automatisation de la détection des risques permet une réaction rapide des agences, l'explicabilité demeure vitale pour permettre aux experts de valider les alertes instantanément (Randl et al., 2025b). La tâche 9 de SemEval-2025 propose de classifier les dangers (*Hazards*) et les produits (*Products*) à partir de rapports d'incidents brefs. Notre objectif est de lier ces prédictions à des segments textuels précis afin de réduire le "fossé de fidélité" entre le raisonnement interne des modèles et leurs sorties d'après Randl et al. (2025a).

2 Ressources et Données

2.1 Description du Jeu de données

Le projet s'appuie sur le dataset officiel de SemEval-2025 Task 9, constitué de 6 644 rapports de rappels de produits annotés manuellement en anglais (Randl et al., 2025c). Ces rapports incluent des titres (moyenne de 88 caractères) et des descriptions textuelles (moyenne de 2329 caractères).

La distribution en longue traîne (Figure 2) complexifie la ST2, qui compte plus de 1000 produits majoritairement sous-représentés.

Comme illustré en Figure 3, les étiquettes se divisent en deux niveaux :

- **Sous-tâche 1 (ST1)** : 10 catégories de risques et 22 catégories de produits.
- **Sous-tâche 2 (ST2)** : Classification fine incluant 128 risques et 1142 produits (Randl et al., 2025b).

3 Travaux connexes

Randl et al. (2025b) révèlent que la détection des risques alimentaires lors du challenge **SemEval-2025** s'est structurée autour d'approches de classification et de stratégies robustes contre le déséquilibre extrême des données. Le et al. (2025) indiquent que leur équipe (**Anastasia**), classée première en ST1, a démontré l'efficacité d'un ensemble de modèles DeBERTa-v3 et RoBERTa-large optimisés par une *Focal Loss* pour traiter le déséquilibre « longue traîne ».

Parallèlement, l'approche de **BitsAndBites** a mis en lumière l'intérêt d'une architecture multi-tâches (MTL) séquentielle pour capturer les dépendances hiérarchiques entre les catégories générales et les détails fins des produits et dangers (Gensale et al., 2025). La stabilisation des performances sur les classes minoritaires a également nécessité des techniques d'augmentation variées, allant de la génération synthétique par LLM à l'insertion contextuelle via BERT comme proposée par l'équipe **BrightCookies** de Papadopoulou et al. (2025).

Cependant, malgré ces avancées en précision, Randl et al. (2025a) soulignent l'existence d'un « fossé de fidélité » : les auto-explications générées par les modèles s'avèrent souvent plausibles pour l'humain sans refléter fidèlement leur raisonnement interne réel.

4 Méthodologie

C'est dans ce contexte que notre approche se distingue en intégrant l'étiquetage **BIO** établi par Ramshaw and Marcus (1995) directement dans une boucle MTL afin de fournir une justification extractive ancrée directement dans le texte.

4.1 Architecture Multi-Tâches

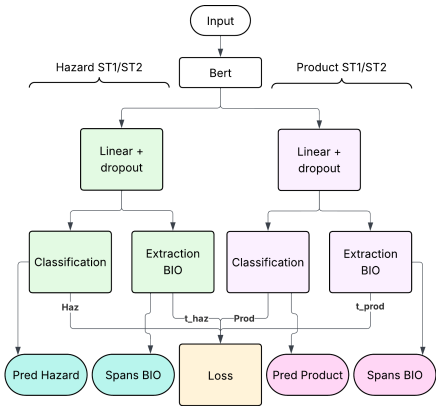

Figure 1: Schéma simplifié de l'architecture du modèle

S'inspirant des travaux de l'équipe BitsAndBites de Gensale et al. (2025), nous utilisons un encodeur BERT partagé alimentant deux branches distinctes pour les risques et les produits. Chaque branche dispose d'une tête de classification multiclasse (ST1) et d'une tête de classification de jetons (BIO tagging) pour l'extraction de segments.

4.2 Optimisation et Fonctions de Perte

Pour assurer la convergence du modèle dans un contexte de déséquilibre de classes et de multi-tâches, nous avons implémenté un système de perte hybride.

4.2.1 Classification via Focal Loss

Afin de pallier le déséquilibre extrême des données entre les catégories de risques et de produits, nous appliquons une Focal Loss comme Le et al. (2025). Contrairement à une entropie croisée standard, cette fonction permet de réduire l'importance des exemples "faciles" pour concentrer l'apprentissage sur les cas ambigus :

$$L_{focal} = \alpha \cdot (1 - p_t)^{\gamma} \cdot CE \tag{1}$$

Ici, p_t représente la probabilité prédite pour la classe correcte. Le facteur de pondération α équilibre l'importance globale des classes, tandis que l'exposant de focalisation γ ajuste la vitesse à laquelle les exemples bien classés sont écartés de la fonction de perte.

4.2.2 Extraction via Soft Dice Loss

Pour l'extraction des segments justificatifs (BIO tagging), nous utilisons la Soft Dice Loss, inspirée de Li et al. (2020). Cette fonction est particulièrement adaptée à la segmentation car elle maximise l'intersection entre les segments prédits et réels, tout en ignorant le tag majoritaire « O » (Outside) qui représente la majeure partie du texte :

$$\text{SoftDiceLoss} = 1 - \frac{2\sum_{i=1}^{N} p_i t_i + \epsilon}{\sum_{i=1}^{N} p_i + \sum_{i=1}^{N} t_i + \epsilon} \tag{2}$$

Dans cette équation, p_i est la probabilité prédite de la classe positive pour le token i et t_i est la valeur cible. Le terme ϵ assure la stabilité numérique en évitant toute division par zéro.

4.2.3 Pondération par Incertitude Homoscédastique

La combinaison de tâches de nature différente (classification de texte et étiquetage de jetons) rend l'équilibrage manuel des poids de perte (hyperparamètres) complexe et sous-optimal. Nous adoptons donc une approche d'apprentissage de l'incertitude homoscédastique proposée par Kendall et al. (2018):

$$L_{total} = \sum_{i \in haz, prod, t_{haz}, t_{prod}} \frac{1}{2} \exp(-s_i) L_i + \frac{1}{2} s_i \tag{3}$$

Le modèle apprend, pour chaque tâche i, un paramètre s_i représentant son incertitude intrinsèque. Le terme $\exp(-s_i)$ agit comme un poids adaptatif : plus une tâche est incertaine ou "bruyante", plus son poids dans la perte globale est réduit, tandis que le terme de régularisation $\frac{1}{2}s_i$ empêche le modèle d'annuler purement et simplement une tâche.

4.3 Métriques d'Évaluation

L'évaluation de la performance de notre modèle repose sur des indicateurs capables de mesurer à la fois la précision de la classification hiérarchique et la fidélité de l'extraction de segments.

4.3.1 Classification (ST1 et ST2)

Pour les sous-tâches de classification, nous utilisons le score F1 macro pour les catégories de dangers et de produits. Le F1 macro, calculé comme la moyenne simple des scores F1 de chaque classe, est privilégié car il permet de détecter les échecs du modèle sur les classes rares, un aspect critique compte tenu du déséquilibre du jeu de données. La performance globale sur ces tâches est mesurée par un score synthétique pondérant la réussite de la classification des dangers et celle des produits, sous condition de justesse du danger:

$$\begin{aligned} ST1_{score} = \frac{1}{2}\big[&F1_{macro}(\text{hazard}) + \\ &F1_{macro}(\text{product}|\text{hazard correct})\big] \end{aligned} \tag{4}$$

4.3.2 Extraction BIO et Performance Combinée

L'efficacité de l'extraction de segments justificatifs est évaluée via un score spécifique (Bio_{score}), fondé sur la moyenne des scores F1 macro (Dice) pour les étiquettes BIO des dangers (Bio_H) et des produits (Bio_P):

$$Bio_{score} = \frac{1}{4} \times F1_{macro}(Bio_H) + \frac{1}{4} \times F1_{macro}(Bio_P) \tag{5}$$

Enfin, pour obtenir une vue d'ensemble de la capacité du modèle à classifier correctement tout en fournissant une

explication textuelle valide, nous définissons une métrique combinée pour la ST1 et l'extraction BIO:

$$ST1\_BIO\_combined = \frac{1}{2} \times ST1_{score} + Bio_{score} \quad (6)$$

Cette métrique finale assure que l'architecture ne sacrifie pas la précision sémantique au profit de l'extraction, ou inversement.

4.3.3 Fidélité de l'Explication : Suffisance et Complétude

Afin de quantifier la fiabilité des justifications extraites et de répondre directement au problème du « fossé de fidélité » soulevé dans la littérature, nous introduisons une évaluation lors de l'inférence. Cette méthode teste la robustesse du raisonnement du modèle en mesurant l'impact des segments extraits sur la prédiction finale à travers deux critères, s'inspirant de DeYoung et al. (2020) :

- **Suffisance (*Sufficiency*)** : Évalue si l'extrait textuel (le segment étiqueté BIO) contient à lui seul l'information nécessaire pour justifier la prédiction. Elle est validée si le modèle, recevant uniquement les mots extraits (le reste du texte étant remplacé par des masques), parvient à formuler des prédictions de danger et de produit strictement identiques à celles obtenues sur le texte complet :

$$S = (\hat{y}_{suff}^{(haz)} = \hat{y}_{orig}^{(haz)}) \wedge (\hat{y}_{suff}^{(prod)} = \hat{y}_{orig}^{(prod)}) \quad (7)$$

- **Complétude (*Comprehensiveness*)** : Mesure si l'extrait est indispensable à la décision. Elle est vérifiée si la suppression de cet extrait du texte introductif altère la confiance du modèle et modifie sa prédiction (soit pour la catégorie du risque, soit pour celle du produit) :

$$C = (\hat{y}_{comp}^{(haz)} \neq \hat{y}_{orig}^{(haz)}) \vee (\hat{y}_{comp}^{(prod)} \neq \hat{y}_{orig}^{(prod)}) \quad (8)$$

Un échantillon est alors considéré comme présentant **une explication fidèle** s'il valide de concert les propriétés de suffisance et de complétude. La fidélité globale du système correspond au pourcentage d'échantillons fidèles évalués sur le jeu de test.

4.4 Annotations des données

Nous introduisons une nouvelle sous-tâche : l'extraction exacte (classification BIO) des segments justifiant les prédictions ST1 (catégorie de produit et catégorie de danger). Les 6 644 rapports de rappels de produits n'étant pas annotés pour cette sous-tâche, une annotation manuelle a été écartée en raison du volume de données.

Nous avons opté pour une annotation par LLM : chaque modèle reçoit le texte brut ainsi que les prédictions ST1, et retourne les extraits correspondants, convertis ensuite au format BIO. La nature subjective de cette tâche nous a conduits à adopter une annotation douce multi-modèles : plutôt qu'une classe unique par token, on associe à chaque token une distribution de probabilité sur les classes B, I et O, construite à partir des prédictions agrégées de plusieurs modèles.

Une première expérimentation avec des modèles "instruct" quantifiés (Qwen2.5-7B, Mistral-7B-v0.3, LLaMA-3-8B, tous en AWQ) s'est révélée infructueuse : les modèles tendaient soit à corriger les erreurs du texte source, soit à reproduire directement les labels ST1, produisant des segments invalides.

L'annotation finale a été réalisée via OpenRouter avec quatre modèles : Gemini 2.0 Flash, Qwen3 80B A3B Instruct, DeepSeek V3.2 et Mistral Small 3.1 24B.

4.5 Augmentation de données

Afin de compenser le déséquilibre des classes, en s'inspirant de Le et al. (2025) (Anastasia) et Papadopoulou et al. (2025) (BrightCookies), nous avons décidé d'augmenter les données sur les catégories minoritaires en combinant deux méthodes : le paraphrasage à l'aide du modèle Flan-T5, et le bruitage contextuel à l'aide de BERT (insertion contextuelle de nouveaux mots + remplacement de termes existants). Les premiers résultats montrent une diminution des scores ST1 et une augmentation des scores ST2 (cf. Table 1).

Une analyse manuelle des exemples générés a montré que certaines variantes perdaient des informations critiques, modifiaient légèrement le sens initial ou introduisaient du bruit lexical. Nous avons donc conçu une seconde version moins agressive privilégiant la qualité grâce à des mécanismes de validation : conservation des mots clés liés aux dangers et aux produits, seuil minimal de similarité avec le texte source, contrôle de longueur, suppression des doublons et limitation du nombre de variantes par exemple. Grâce à cela, nous avons amélioré la baseline ST1 de 0,7670 à 0,7877.

5 Expérimentations et Résultats

5.1 Expérimentations pour ST1

Pour mieux comprendre les données du challenge, à l'aide d'un notebook fourni dans le répertoire GitHub des organisateurs, nous avons mené des analyses statistiques.

Ensuite, en s'inspirant des notebooks du dépôt des organisateurs, nous avons mis en place une structure de code modulaire réglable par des configurations YAML, facilitant les expérimentations et le prototypage de modèles. Grâce à cela, nous avons pu mener nos premières expérimentations sur ST1 et ST2 et en sauvegarder les résultats (cf. Table 2 et Table 3). Bien que notre projet ne se concentre pas sur ST2, nous avons tout de même testé notre modèle sur cette sous-tâche pour voir ce que nous pouvions obtenir.

1. **Baseline ST1 & ST2** : Conformément aux résultats de la tâche, nous avons testé et validé les baselines (TF-IDF + LR et BERT) de Randl et al. (2025b). Après plusieurs expérimentations, nous avons observé que BERT surpassait systématiquement le modèle TF-IDF + LR. Nous ferons donc de BERT notre baseline. C'est également la baseline du challenge.

2. **Expérimentation sur BERT** : Nous avons entraîné le modèle BERT sur les deux sous-tâches (ST1 et ST2) avec différents paramètres et configurations. Nous sommes parvenus à augmenter le score F1 macro de notre baseline (0,71 $\rightarrow$ 0,78 pour ST1, et 0,26 $\rightarrow$ 0,28 pour ST2) sur le jeu de validation en concaténant le titre avec le texte - en supprimant les balises HTML, les caractères spéciaux, et les espaces superflus - en utilisant une focal loss - et en utilisant deux têtes de sortie sur le même modèle (*Hazard* et *Product*).

5.2 Expérimentations et résultats pour l'extraction BIO

Pour l'extraction BIO, un premier entraînement sur les données fraîchement annotées a permis d'obtenir des performances encourageantes avec un score F1 macro (Dice) de 0,678 sur hazard et 0,727 sur product pour notre jeu de validation, sans observer de perte pour notre score ST1 (0,7853) (Table 4), permettant de fixer une baseline.

Nous remarquons que nous avons un sur-apprentissage de notre modèle au vu de résultats (Figure 4), c'est pourquoi la suite des expérimentations s'est concentrée sur l'optimisation des hyper-paramètres. Nous avons notamment réglé : le taux de régularisation L2, le taux d'apprentissage du modèle ainsi que celui de la pondération des pertes, les probabilités de dropout à chaque branche et à chaque tête, la taille du batch, le nombre de tokens d'entrée. Pour finalement obtenir la configuration visible en Table 5.

À cela, s'ajoute le passage du modèle BERT-base à RoBERTa-large qui a permis de passer, après correction du sur-apprentissage, d'un score de 0,697 à 0,760 pour ST1 + BIO combiné, sur notre jeu de validation.

Notre optimisation de paramètres nous a permis d'obtenir des résultats plus robustes (Table 4). Notre entraînement final obtient de meilleurs résultats que notre baseline ST1 tout en corrigeant le sur-apprentissage. Notre entraînement final stabilise RoBERTa-large à -0,072, tandis que notre baseline BIO stabilisait bert-base à -0,138.

Comme illustré par la Figure 5, le modèle final assure une décroissance monotone de la perte. L'équilibrage dynamique des tâches converge vers 0,25, confirmant une progression conjointe des branches sans prédominance de l'une sur l'autre.

Notre modèle final surpasse les résultats que nous avions pour ST1 seul (F1 0,7838 < 0,7937) sur notre jeu de validation. Et notre approche permet d'obtenir des scores F1 macro (Dice) très prometteurs : 0,7577 pour product et 0,7606 pour hazard.

5.3 Analyse de l'Explicabilité

Les résultats les plus significatifs pour valider notre approche sont les scores de suffisance et de complétude obtenus sur le jeu de validation (562 échantillons). Notre prédiction est suffisante 66,90 % du temps et elle est complète à 57,12 % du temps (cf. Table 6). Sur les échantillons satisfaisant ces deux critères, en moyenne la classification BIO extrait 17,89 % du texte, confirmant l'isolation du signal utile. Seulement 36,65 % de nos échantillons sont fidèles, et parmi eux, seulement 67,96 % ont produit une prédiction ST1 parfaitement exacte.

6 Analyse Qualitative et Limites

Bien que le système cible efficacement l'information pertinente avec une excellente concision (environ 18% de mots extraits pour $\sim 67\%$ de suffisance), l'évaluation révèle une limite au niveau de la complétude ($\sim 57\%$). En effet, même lorsque les termes clés sont masqués, le modèle parvient souvent à maintenir sa prédiction initiale dans $\sim 43\%$ des cas. Cela démontre que l'architecture exploite la forte redondance contextuelle des documents (corrélations statistiques, vocabulaire périphérique) pour deviner la classe, plutôt que de s'appuyer exclusivement sur son extraction. Cela met en évidence notre principal problème, malgré un bon score d'extraction BIO, notre analyse a montré que nous avions encore un fossé de fidélité important entre nos étiquettes et le raisonnement interne du modèle. Enfin, le bruit et le manque de validation des annotations BIO limitent l'alignement parfait entre les segments extraits fournis et le véritable raisonnement interne du modèle.

7 Conclusion

Ce projet valide l'efficacité d'une architecture multi-tâches RoBERTa-large, stabilisée par l'incertitude homoscédastique. Nos résultats montrent une progression nette de notre score F1 de 0,7937 en ST1, sur notre jeu de validation, malgré une surcharge de l'architecture. L'explicabilité que nous proposons avec notre tâche d'extraction BIO permet d'obtenir des résultats encourageants avec une forte suffisance (66,90 %), malgré une forte complétude (57,12 %). Notre approche constitue une avancée dans l'objectif de réduire le fossé de fidélité entre la prédiction et le raisonnement interne du modèle, et d'améliorer l'explicabilité du modèle, nécessaire dans le domaine alimentaire. L'intégration de l'augmentation des données, et une annotation manuelle des données sont des pistes d'amélioration possibles.

References

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A Benchmark to Evaluate Rationalized NLP Models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online. Association for Computational Linguistics.

Aurora Gensale, Irene Benedetto, Luca Gioacchini, Luca Cagliero, and Alessio Bosca. 2025. BitsAndBites at SemEval-2025 task 9: Improving food hazard detection with sequential multitask learning and large language models. pages 718–725.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. *Preprint*, arxiv:1705.07115 [cs].

Tung Thanh Le, Tri Minh Ngo, and Trung Hieu Dang. 2025. Anastasia at SemEval-2025 Task 9: Subtask 1, Ensemble Learning with Data Augmentation and Focal Loss for Food Risk Classification. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 141–147, Vienna, Austria. Association for Computational Linguistics.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. 2020. Dice loss for data-imbalanced NLP tasks. *Preprint*, arxiv:1911.02855 [cs].

Foteini Papadopoulou, Osman Mutlu, Neris Özen, Bas Van Der Velden, Iris Hendrickx, and Ali Hurriyetoglu. 2025. BrightCookies at SemEval-2025 Task 9: Exploring Data Augmentation for Food Hazard Classification. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 914–930, Vienna, Austria. Association for Computational Linguistics.

C. Qian, S.I. Murphy, R.H. Orsi, and M. Wiedmann. 2023. How can AI help improve food safety? 14(1):517–538.

Lance Ramshaw and Mitch Marcus. 1995. Text Chunking using Transformation-Based Learning. In *Third Workshop on Very Large Corpora*.

Korbinian Randl, John Pavlopoulos, Aron Henriksson, and Tony Lindgren. 2025a. Mind the gap: from plausible to valid self-explanations in large language models. *Machine Learning*, 114(10):220.

Korbinian Randl, John Pavlopoulos, Aron Henriksson, Tony Lindgren, and Juli Bakagianni. 2025b. SemEval-2025 Task 9: The Food Hazard Detection Challenge. *arXiv preprint*. ArXiv:2503.19800 [cs].

Korbinian Randl, John Pavlopoulos, Tony Lindgren, Aron Henriksson, and Giannis Stoitsis. 2025c. Food Recall Incidents.

A Figures

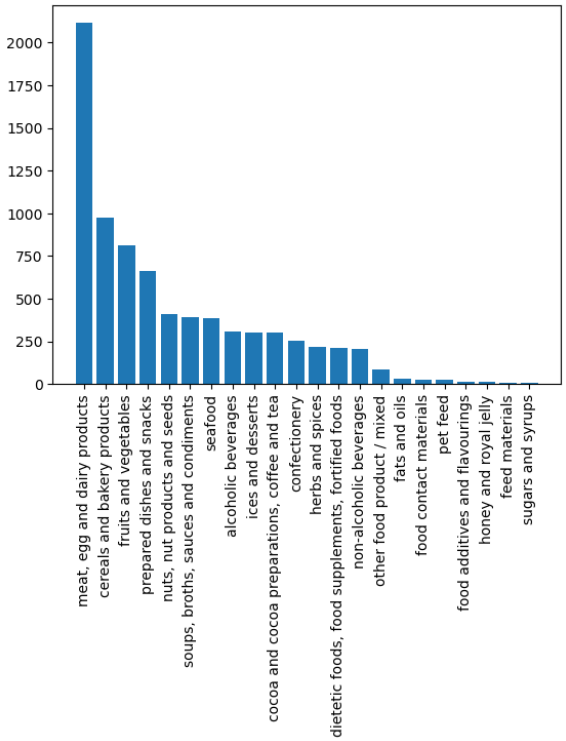

Figure 2: Distribution des fréquences des 22 catégories de produits (ST1), illustrant le déséquilibre du jeu de données.

"Randsland brand Super Salad Kit recalled due to Listeria monocytogenes"	
hazard:	listeria monocytogenes
hazard-category:	biological
product:	salads
product-category:	fruits and vegetables

Figure 3: Exemple de rapport d'incident illustrant les types de données traitées (Titre, Texte et étiquettes ST1/ST2).

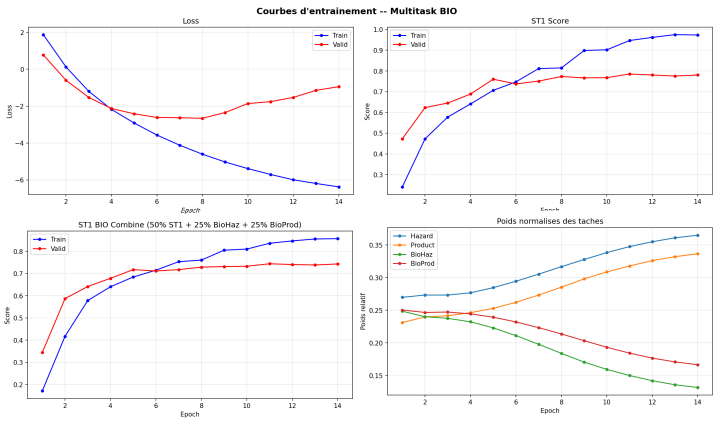

Figure 4: Courbe d'entraînement de la première expérimentation avec l'extraction BIO (baseline BIO).

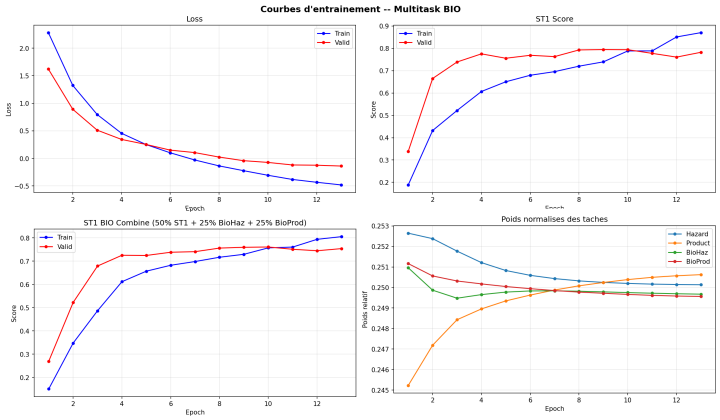

Figure 5: Courbe d'entraînement de la dernière expérimentation avec l'extraction BIO (entraînement final).

B Tableaux pour les expérimentations ST1 ST2

Aug.	Configuration	Haz. M.	Prod. M.	Global
Résultats pour ST1				
Sans	BERT base	0,7087	0,6526	0,7052
	MTL* BERT	0,8153	0,7078	0,7838
Avec	BERT base	0,7042	0,6317	0,6874
	MTL* BERT	0,7928	0,7267	0,7499
Résultats pour ST2				
Sans	BERT base	0,2098	0,0763	0,1567
	MTL* BERT	0,4766	0,0875	0,2819
Avec	BERT base	0,2571	0,0711	0,1817
	MTL* BERT	0,5241	0,0980	0,3198

*MTL : BERT Multi-tâches.

Table 1: Scores F1 Macro pour ST1 et ST2 avec augmentation de données (première approche), sur le jeu de validation.

Configuration	Haz. M.	Prod. M.	Global
TF-IDF + LR			
Title	0,5153	0,4650	0,4977
Text	0,5446	0,4107	0,4884
Title + Text	0,5472	0,4424	0,5022
BERT Base Uncased			
Title	0,7087	0,6526	0,7052
Text	0,7870	0,6195	0,7289
Title + Text	0,8353	0,6564	0,7670
+ Nettoyage	0,8156	0,6588	0,7589
+ Perte Focale	0,8061	0,6705	0,7621
+ Nett. + Per. Foca.	0,8321	0,6561	0,7659
+ N + Fl + Multi-tâches	0,8153	0,7078	0,7838

Table 2: Scores F1 Macro (Hazard/Product) et Score Global pour ST1, sur le jeu de validation

Configuration	Haz. M.	Prod. M.	Global
TF-IDF + LR			
Title	0,1997	0,1219	0,1832
Text	0,1944	MI*	-
Title + Text	0,2068	MI*	-
BERT Base			
Title	0,2098	0,0763	0,1567
Text	0,4307	0,0784	0,2631
Title + Text	0,4242	0,1036	0,2641
+ Nettoyage	0,4624	0,0921	0,2810
+ Perte Focale	0,4468	0,1007	0,2764
+ Nett. + Per. Foca.	0,4565	0,0795	0,2713
+ N + Fl + Multi-tâches	0,4766	0,0875	0,1900

*MI: mémoire insuffisante.

Table 3: Scores F1 Macro (Hazard/Product) et Score Global pour ST2, sur le jeu de validation.

C Tableaux pour expérimentation et résultats ST1 + BIO

Métriques (Validation)	Baseline	Modèle Final
Perte Finale	-1,7544	**-0,0724**
F1 Score ST1	0,7853	**0,7937**
Bio Hazard Dice	0,6781	**0,6972**
Bio Product Dice	0,7271	**0,7577**
ST1 + BIO	0,7440	**0,7606**

Table 4: Synthèse comparative entre la baseline et l'architecture finale pour ST1 + BIO, sur le jeu de validation.

Configuration	Ancienne	Nouvelle
`weight_decay`	0,05	0,10
`uncertainty_lr`	1×10^{-3}	1×10^{-4}
`hidden_dropout`	0,15	0,2
`attention_dropout`	0,1	0,15
`dropout_branch`	0,15	0,25
`dropout_head_st1`	0,3	0,4
`dropout_head_bio`	0,2	0,2
`learning_rate`	2×10^{-5}	1×10^{-5}
`pretrained_model`	*bert-base-uncased*	*RoBERTa-large*

Table 5: Résumé des configurations intermédiaires pour l'optimisation des hyperparamètres.

Indicateur de Fidélité	Résultat (Validation)
Nombre d'échantillons évalués	562
Proportion moyenne extraite	17,89 %
Suffisance globale (S)	66,90 %
Complétude globale (C)	57,12 %
Échantillons Fidèles ($S \wedge C$)	**36,65 %**
ST1	0,7546
Précision sur échantillons fidèles	67,96 %

Table 6: Synthèse de l'évaluation de l'explicabilité (Suffisance et Complétude), sur le jeu de validation, montrant la cohérence entre performance et fidélité.

D Environnement Technique

Afin de garantir la reproductibilité et l'efficacité de nos expérimentations, notre pipeline d'apprentissage a été construit autour de diverses technologies :

- **Frameworks de Deep Learning :** Les modèles ont été implémentés et entraînés en utilisant **PyTorch** pour le calcul tensoriel et l'autograd. La gestion des architectures de Transformers (BERT, RoBERTa) a été assurée par la bibliothèque **transformers** (Hugging Face).
- **Gestion des Données et Métriques :** La manipulation des jeux de données et le calcul des scores (F1 scores) ont été réalisés à l'aide de **pandas**, **NumPy**.
- **Gestion des Configurations :** Pour permettre une exécution modulaire et le suivi des hyperparamètres, nous avons utilisé **Hydra** avec des fichiers YAML.
- **Annotation NLP :** Les appels aux modèles LLM externes pour la génération de la sous-tâche BIO ont été orchestrés via l'API **OpenRouter**. Bien qu'un essai d'inférence directe des modèles sur **Google Colab** ait été testé avec **vLLM**.
- **Matériel et Infrastructure :** La majorité de nos entraînements et expérimentations ont été exécutés sur des environnements Cloud via la plateforme **Google Colab** (utilisant des GPUs NVIDIA T4 et L4). L'hébergement et le versionnage du code source ont été hébergés sur **GitHub**.

E Contributions

- **A. Skrzypczak** : - Revue de littérature focalisée sur les résultats de la compétition semeval 2025 task 9 - Mise en place de la structure modulaire du projet pilotée par les fichiers de configuration YAML (DataLoader, évaluation, pipeline entraînement, système de logging) - Exploration et statistiques des données pour comprendre les difficultés de la tâche - Entraînement et expérimentation TF-IDF + LR, BERT, comparaison des approches - Approfondissement documentaire sur l'augmentation des données - Affinage de l'architecture en consultant de nouveaux papiers - Essai et réflexion sur l'évaluation et l'annotation de l'extraction de segments BIO - Rédaction d'une partie des rapports de projet - Configurer et lancer de nouveaux entraînements en ajustant les hyperparamètres du modèle. Notamment le nombre d'époques, la taille de lot et le taux d'apprentissage. - Adapter l'infrastructure et la configuration aux besoins. - Gérer la stratégie d'annotation des données manquantes pour l'extraction de segments via un ensemble de LLM (vllm, openrouter). Et exporter les annotations au format BIO (soft labeling). - Extension du MultiTaskTrainer pour les données BIO (alignement des tokens, soft dice loss, checkpointing, logging) - Expérimentation du modèle ST1 + BIO, avec un notebook d'expérimentation depuis Colab. - Interprétation des résultats pour apporter les conclusions du projet aux rapports, réalisation du readme. -
- **R. Becard** : - Conception de l'architecture multi-tâches (*MultiTaskModel*) basée sur les Transformers DeBERTa et RoBERTa-large - Intégration de trois têtes de prédiction : classification des dangers (*Hazard*), classification des produits (*Product*) et étiquetage de jetons (*BIO tagging*) qui ont été changés par la suite. - Développement de la *MultiTaskLoss* combinant une *Focal Loss* pour le déséquilibre des classes et une *Dice Loss* pour l'extraction de segments. - Mise en place de la pondération adaptative des pertes via l'apprentissage de l'incertitude homoscédastique. - Implémentation du *MultiTaskTrainer* incluant la gestion dynamique des étiquettes et des masques de padding. - Création et implémentation de la métrique d'évaluation de la fidélité des explications basée sur les critères de Suffisance, de Complétude et la plausibilité (ratio). - Revue de littérature approfondie sur les écarts de plausibilité, le *text chunking* et les standards CoNLL-2003. - Rédaction intégrale des trois livrables (rapport de proposition, d'avancement et rapport final) selon le format *ACL*. - Configuration de l'environnement d'entraînement Cloud (Google Colab, Kaggle) et optimisation de l'utilisation mémoire GPU.
- **S. Maurel** : Implémentation d'un module de métriques BIO non utilisé - Expérimentations et comparaisons BERT/DeBERTa (configuration, entraînement, analyse des résultats) - Aide à la rédaction des rapports.
- **K. Jemmali** : Conception et implémentation d'une stratégie d'augmentation de données hybride (Flan-T5/BERT) - Développement des filtres de validation - Rééquilibrage des classes minoritaires - Expérimentations et analyse de l'impact sur les performances ST1.