

Détection de subjectivité dans les textes : Compte-Rendu de Projet

IAFA-tigable

Oleiwane Joe, Delmote Adrien, Meunier Mortimer, Bienassis Jules, Mouysset Martin

15/04/2024



M1 Informatique
S8
UE Management de projet

Table des matières

I	Présentation	3
I.1	Données	3
I.2	Modèle	3
II	Travail Réalisé	4
II.1	Première implementation sur OSIRIM	4
II.2	Premier ajustement fin	4
II.3	Second ajustement fin	7
III	Pistes d'Améliorations	9
III.1	Temps d'entraînement plus long	9
III.2	Augmentation des données d'entraînement	9
III.3	Amélioration du prompt	9
III.4	Amélioration de l'ajustement fin	9
IV	Bibliographie	10

I – Présentation

I.1 Données

Les données utilisées sont issues de la tâche 2 de CheckThat 2024 [1]. Nous n'avons utilisé que les données du jeu de données anglais, ce dernier est divisé en train, dev, test et test_gold (identique au fichier test et contenant les étiquettes "subj" pour subjectif et "obj" pour objectif), voici la répartition :

- train : 830 phrases. 532 objectives, 298 subjectives
- dev : 219 phrases. 106 objectives, 113 subjectives
- test : 243 phrases. 116 objectives, 127 subjectives

Nous avons utilisé le jeu de données "train" pour entraîner notre modèle, "dev" pour obtenir ses performances pendant la phase d'affinement du modèle ("Fine-tuning" en anglais). Enfin, "test" est utilisé pour en obtenir les performances finales.

I.2 Modèle

Nous avons opté pour le modèle Mistral7B Instruct [3], en raison de sa flexibilité en adaptation via l'apprentissage par instruction. Toutefois, l'utilisation de ce modèle requiert énormément de ressources pour un ajustement complet. C'est pourquoi nous avons procédé à sa quantification (quantization en anglais) avec LoRa [5] afin de minimiser ses besoins en ressources et d'accélérer son fonctionnement pour n'utiliser que 22 Go de VRAM sur la plateforme OSIRIM de l'IRIT (2x GTX1080TI, soit 2x 11 Go de VRAM). Cette technique de "quantization" [6] réduit le nombre d'opérations nécessaires à l'optimisation des paramètres.

II.1 Première implementation sur OSIRIM

Afin d’optimiser le temps à notre disposition, nous avons autant que possible essayé de paralléliser les processus de développement. C’est pourquoi nous avons commencé à implémenter Mistral dans notre environnement de travail OSIRIM, sans ajustement fin, afin de commencer à expérimenter et de découvrir les limites de ce modèle. Cette implémentation a effectivement donné des résultats médiocres : le modèle ne renvoyait pas que les labels requis (SUBJ et OBJ) mais aussi des phrases explicatives. Mais cela nous a permis de gagner du temps et d’expérimenter différents prompts pendant qu’une autre personne de l’équipe commençait la tâche de “Fine-tuning”.

II.2 Premier ajustement fin

Afin d’ajuster notre modèle nous avons utilisé SFTTrainer (SFT = supervised Fine-tuning) de la librairie python trl (transformer reinforcement learning) et nous nous sommes basés sur les hyperparamètres utilisés pour un ajustement fin de 100 pas (steps) avec le jeu de données "ultrachat" contenant des dialogues générés par chatGPT [5]. Cependant, même en combinant les expérimentations de prompt et le “Fine-tuning”, les sorties du modèle comportaient encore trop de phrases à la place des labels. Nous nous sommes donc concentrés sur l’amélioration des prompts fournis afin d’améliorer nos résultats.

```
global_step=100  
training_loss=2.7287393188476563  
metrics:  
  - train_runtime: 1292.2778  
  - train_samples_per_second: 1.857  
  - train_steps_per_second: 0.077  
total_flos: 6997820682797752320  
train_loss: 2.7287393188476563  
epoch: 2.86
```

FIGURE II.1 – Tracé général du premier ajustement fin

A l'origine, nous nous sommes inspirés des articles scientifiques qui utilisaient l'API de GPT [7] "Our task is to evaluate whether a text is subjective or objective. Subjectivity is a characteristic of language : by uttering a statement, the speaker simultaneously expresses his or her position, attitude and feelings towards the statement, thus leaving his or her own mark. If the text is not subjective, it is considered objective. There are a few points to clarify : 1. emotions, considered objective because there is no more objective way of describing them. 2. quotations, considered objective whatever their content. 3. Intensifiers, considered subjective 4. Speculation, considered subjective. Answer only OBJ for objective or SUBJ for subjective followed by a semicolon, then explain the reasoning and assumptions behind your answer. For example : "SUBJ ; the text is in the first person."

Nous l'avons écrit en anglais car le modèle Mistral a été pré-entraîné principalement en anglais. Nous avons récupéré la définition de [2] sur la subjectivité. L'utilisation de ce prompt n'a pas permis de résoudre le problème des sorties différent de SUBJ et OBJ. Nous avons donc émis l'hypothèse que notre prompt était trop grand, nous avons donc testé le modèle avec une multitude de prompt plus court et contenant moins d'information pour trouver en sélectionner un qui était efficace : "Your task is to evaluate whether a text is subjective or objective. Subjectivity is a characteristic of language : when uttering a statement, the speaker simultaneously expresses their position, attitude, and feelings towards the statement, thus leaving their own imprint. If the text is not subjective, it is considered objective. Your output is only OBJ for objective or SUBJ for subjective."

Grâce à ce nouveau prompt, nous avons vu la quantité de labels corrects dans les réponses du modèle passer à 100%. Les résultats restaient cependant "instables" : avec le même prompt et les mêmes données, le nombre de labels dans les sorties variait entre 95% et 100%. Ce résultat est donc suffisant pour commencer à analyser les scores de classification mais à revoir afin d'améliorer encore cette "stabilité". De plus, les résultats du modèle n'est pas satisfaisant, le modèle donne 0,27 de score macro F1 et 0,68 de score subj F1 sur les données qui ont servi à son entraînement. La matrice de confusion montre que le modèle avait tendance à sur-représenter la classe subjective. Résultat étonnant puisque nous nous attendions à un surapprentissage lié à la surreprésentation de la classe objective dans les données fournies.

Résultats sur les données utilisées pour l'entraînement

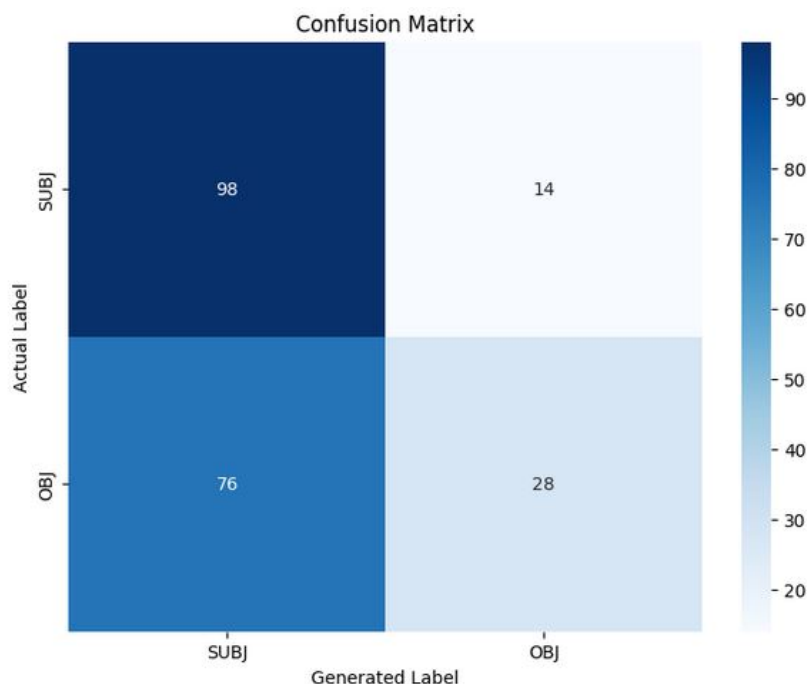


FIGURE II.2 – Matrice de confusion sans erreurs sur les données d'entraînement

Nous avons malgré tout souhaité tester le modèle sur les données de test.
Résultats sur les données de Test :

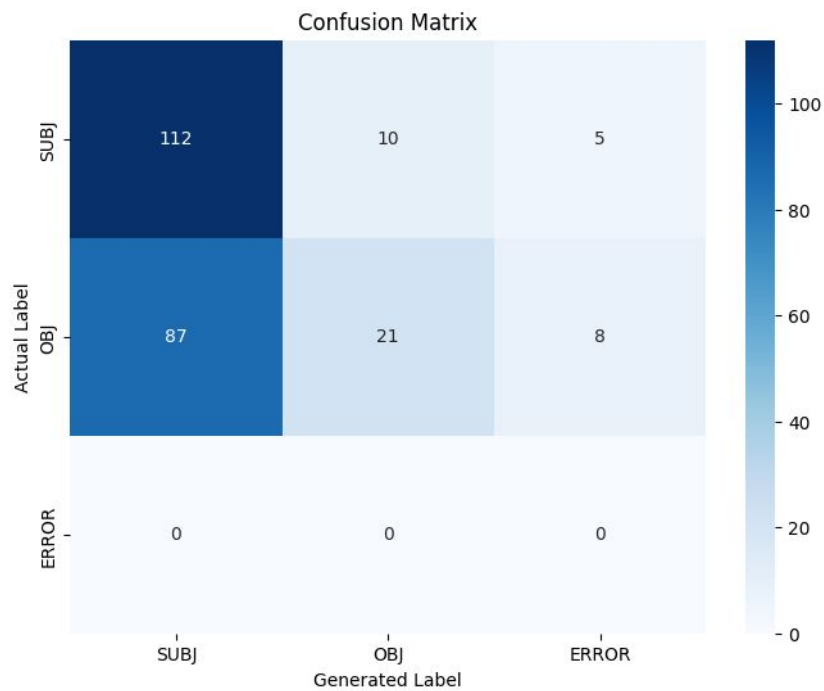



FIGURE II.3 – Matrice de confusion avec erreurs sur les données de test

	precision	recall	f1-score	support
ERROR	0.00	0.00	0.00	0
OBJ	0.68	0.18	0.29	116
SUBJ	0.56	0.88	0.69	127
accuracy			0.55	243
macro avg	0.41	0.35	0.32	243
weighted avg	0.62	0.55	0.50	243

FIGURE II.4 – Scores du modèle avec erreurs sur les données de test



Step	Training Loss	Validation Loss
20	3.029800	3.076159
40	2.737400	3.073150
60	2.361600	3.157660
80	1.634100	3.573520
100	1.618300	3.461876
120	0.878100	3.883179
140	0.942000	3.899183

FIGURE II.5 – Tracé de la loss sur le second ajustement fin

```

global_step=140
training_loss=2.0074230194091798
metrics:
  - train_runtime: 1882.9445
  - train_samples_per_second: 1.763
  - train_steps_per_second: 0.074
total_flos: 9746652063596544.0
train_loss: 2.0074230194091798
epoch: 4.0

```

FIGURE II.6 – Tracé général du second ajustement fin

II.3 Second ajustement fin

Ayant maintenant des résultats sous un format convenable, nous avons tenté d’améliorer les scores du modèle ; en commençant par effectuer de nouveaux ajustements fin en essayant de trouver de bons hyperparamètres dans le but de diminuer la perte. Pour ce faire, nous avons utilisé les mêmes données en remplaçant les “steps” par des “epochs” (4 epochs, plus précisément). Cela a eu pour effet de diminuer la perte sur les données d’entraînement mais de l’augmenter sur les données de validation.

Arrivant au terme du temps disponible, nous avons simplement pu tester le nouveau modèle avec les données d'entraînement afin de regarder la cohérence de ses résultats. Les scores sont similaires au modèle précédents et bien en dessous de ceux de la Baseline fournie par CheckThat :

Baseline (SBERT + LR) en anglais :

- Macro F1 : 0.74
- Macro P (précision moyenne) : 0.74
- Macro R (rappel moyen) : 0.74
- F1 pour la subjectivité (SUBJ F1) : 0.74
- Précision pour la subjectivité (SUBJ P) : 0.75
- Rappel pour la subjectivité (SUBJ R) : 0.73
- Précision globale (Accuracy) : 0.74

Notre modèle Mistral7B sur les données de test :

- $\text{Macro F1} = 0.29 + 0.69 / 2 = 0.635$
- F1 pour la subjectivité (SUBJ F1) : 0.69
- Précision pour la subjectivité (SUBJ P) : 0.56
- Rappel pour la subjectivité (SUBJ R) : 0.88
- Précision globale (Accuracy) : 0.55

Précision : La précision pour la subjectivité de notre modèle (0.56) est notablement inférieure à celle de la baseline (0.75). Cela suggère que la baseline est mieux à même de correctement identifier les vrais exemples positifs de subjectivité parmi toutes les instances classées comme subjectives.

F1-Score : Notre modèle a un F1-score de 0.69 pour la subjectivité, ce qui est légèrement inférieur à la baseline pour l'anglais (0.74). Le F1-score est une mesure harmonique de la précision et du rappel, donc même si notre rappel est supérieur, la précision plus basse de notre modèle tire cette métrique vers le bas.

III – Pistes d’Améliorations

III.1 Temps d’entraînement plus long

Augmenter le temps d’entraînement pourrait permettre à notre modèle d’atteindre de meilleurs résultats, cependant le risque de sur-apprentissage est élevé avec un jeu de données d’entraînement si restreint.

III.2 Augmentation des données d’entraînement

Notre jeu de données d’entraînement est déséquilibré ce qui introduit des biais de classification comme vu dans les travaux précédents sur cette tâche. De plus il reste aussi de taille très modeste (moins de 1000 phrases) par rapport aux jeux habituellement utilisés pour l’ajustement fin des modèles aussi large qui ont besoin d’une grande quantité de données. Réussir à augmenter grandement la taille de ce jeu pourrait être une piste très efficace d’amélioration du modèle. On pourrait utiliser la traduction d’autres langues vers l’anglais, voire de la traduction aller-retour si on trouve un modèle de traduction efficace, on peut aussi faire de la reformulation et de la génération ou alors utiliser des benchmarks sur la subjectivité.

III.3 Amélioration du prompt

Nous avons centré notre prompt sur sa capacité à nous renvoyer un output de la forme demandée, à savoir “OBJ” ou “SUBJ”. Cependant il serait certainement possible de l’améliorer de manière à ce qu’il soit plus efficace dans la tâche de classification elle-même.

III.4 Amélioration de l’ajustement fin

L’apprentissage basé sur les invites que l’on utilise uniquement lorsque l’on prompt le modèle en few-shot, l’apprentissage basé sur les patrons et l’apprentissage basé sur les tâches proxy (similaires) :

- Recent Advances in Natural Language Processing via Large Pre-Trained Language Models : A Survey <https://arxiv.org/abs/2111.01243>

peuvent être utilisées soit comme des stratégies lors de la phase d’ajustement fin d’un modèle de langage, soit comme des approches de génération lors de l’utilisation du modèle après son entraînement.

La méthode d’ajustement fin utilisée consiste à donner aux modèles les données et les labels correspondants au jeu d’entraînement. Cependant, le prompt n’est pas utilisé à cette étape. Nous pensons qu’il s’agit d’un frein important aux performances du modèle. Intégrer le prompt aux données d’entraînement pourrait être une solution envisageable.

Par exemple, un jeu de données pour l’ajustement fin pourrait inclure des exemples spécifiques où l’apprentissage est guidé par des instructions directes, des patrons, ou des tâches proxy. Le but ici est d’adapter le modèle à mieux comprendre et générer des réponses en fonction de ces méthodes spécifiques. Le jeu de données d’entraînement contiendrait donc des exemples directement reliés à ces méthodes.

IV – Bibliographie

- [1] Lien des données du laboratoire CheckThat : https://gitlab.com/checkthat_lab/clef2024-checkthat-lab/-/tree/main/task2/data
- [2] Ruggeri, F., Antici, F., Galassi, A., Korre, K., Mutti, A. et Barron-Cedeno, A. : On the Definition of Prescriptive Annotation Guidelines for Language-Agnostic Subjectivity Detection, 103–111 (2023)
- [3] Lien du modèle de mistral : <https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1>
- [4] Lien qui a permis de quantizer : <https://towardsdatascience.com/mistral-7b-recipes-for-fine-tuning-and-quantization-on-your-computer-631401583f77>
- [5] Quantization avec LoRa ou GPTQ : <https://towardsdatascience.com/mistral-7b-recipes-for-fine-tuning-and-quantization-on-your-computer-631401583f77>
- [6] Lien de la vidéo expliquant le principe de la quantization <https://www.youtube.com/watch?v=mii-xFaPCrA&t=544s&pp=ygUMenVhbnRpemF0aW9u>
- [7] Tornberg, P. : How to use Large Language Models for Text Analysis, 1–6 (2023)