Détection d'anomalies textuelles à base de l'ingénierie d'invite

Yizhou Xu, Kata Gábor, Leila Khouas, Frédérique Segond

Actes de la 29e Conférence sur le Traitement Automatique des Langues Naturelles
Avignon, France, 27 juin au 1er juillet 2022

Détection d'anomalies textuelles à base de l'ingénierie d'invite

Yizhou Xu[1,2] Kata Gábor[1] Leila Khouas[2] Frédérique Segond[1,3]
(1) ERTIM, INaLCO, 2 Rue de Lille, 75007 Paris, France
(2) Flandrin IT, 4 rue du Port Aux Vins, 92150 Suresnes, France
(3) INRIA, 860 Rue Saint Priest, 34095 Montpellier, France
yxu@chapsvision.com, kata.gabor@inalco.fr, lcollenot@chapsvision.com,
frederique.segond@inria.fr

RÉSUMÉ

La détection d'anomalies textuelles est une tâche importante de la fouille de textes. Plusieurs approches générales, visant l'identification de points de données aberrants, ont été appliqués dans ce domaine. Néanmoins, ces approches exploitent peu les nouvelles avancées du traitement automatique des langues naturelles (TALN). L'avènement des modèles de langage pré-entraînés comme BERT et GPT-2 a donné naissance à un nouveau paradigme de l'apprentissage automatique appelé ingénierie d'invite (*prompt engineering*) qui a montré de bonnes performances sur plusieurs tâches du TALN. Cet article présente un travail exploratoire visant à examiner la possibilité de détecter des anomalies textuelles à l'aide de l'ingénierie d'invite. Dans nos expérimentations, nous avons examiné la performance de différents modèles d'invite. Les résultats ont montré que l'ingénierie d'invite est une méthode prometteuse pour la détection d'anomalies textuelles.

ABSTRACT

Prompt Engineering-Based Text Anomaly Detection

Text anomaly detection is an important text mining task. Many outlier identification methods have been applied in this field. However, these approaches have hardly benefited from recent Natural Language Processing (NLP) advances. The advent of pre-trained language models like BERT and GPT-2 has given rise to a new machine learning paradigm called prompt engineering, which has shown good performance on many NLP tasks. This article presents an exploratory work aimed at examining the possibility of detecting text anomaly using prompt engineering. In our experiments, we have examined the performance of different prompt templates. The results showed that prompt engineering is a promising method for text anomaly detection.

MOTS-CLÉS : Détection d'anomalies textuelles, Ingénierie d'invite, Modèle de langage pré-entraîné, GPT-2.

KEYWORDS: Text Anomaly Detection, Prompt Engineering, Pre-trained Language Model, GPT-2.

1 Introduction

Les anomalies sont des points de données qui ne se conforment pas au comportement attendu ou qui s'écartent tellement des éléments connus qu'ils sont suspectés d'être générés par un mécanisme différent (Chandola *et al.*, 2009; Hawkins, 1980). Inconnus et inattendus, ces points de données anomaux sont également appelés nouveautés, valeurs aberrantes, observations discordantes ou surprises dans la

littérature. La détection d'anomalies est généralement considérée comme une tâche de classement unitaire (Pimentel *et al.*, 2014). Différente du classement traditionnel (binaire ou multi-classe), la détection d'anomalies concerne essentiellement les jeux de données extrêmement déséquilibrés qui contiennent une classe dominante (normale ou positive). Dans ce cas, les échantillons négatifs sont rares et difficiles à obtenir. Par conséquent, les techniques utilisées sont souvent non supervisées ou semi-supervisées. Dans notre travail, nous faisons une distinction entre deux types d'anomalies, à savoir les nouveautés et les valeurs aberrantes, du point de vue de l'apprentissage automatique :

— La détection de nouveautés consiste à identifier des données qui ne sont pas vues dans la phase d'entraînement, autrement dit il s'agit de déterminer si une observation est nouvelle. En général, les modèles sont entraînés uniquement sur les données normales. Ainsi, la détection de nouveautés est également appelée détection d'anomalies semi-supervisée.

— La détection de valeurs aberrantes est également connue sous le nom de détection d'anomalies non supervisée. Elle consiste à identifier des points de données qui se trouvent loin des autres. Dans ce cas, les données d'entraînement non étiquetées sont contaminées par des anomalies.

L'importance de la détection d'anomalies réside dans le fait que les anomalies servent souvent de signal d'un événement significatif ou d'une situation critique qui nécessite des actions correspondantes appropriées en fonction des scénarios d'applications telles que la détection de fraude (Aleskerov *et al.*, 1997), la détection d'intrusion et la cybersécurité (Labib & Vemuri, 2002; Jyothsna *et al.*, 2011), le diagnostic médical (Quinn & Williams, 2007; Clifton *et al.*, 2011) ou la vidéosurveillance (Diehl & Hampshire, 2002).

En ce qui concerne les données textuelles, la détection d'anomalies est une tâche importante de la fouille de textes et largement appliquée aux domaines comme la veille stratégique et l'intelligence économique (Wang & Chen, 2019; Kim *et al.*, 2019; Barrett *et al.*, 2019), la sécurité et la défense (Amato *et al.*, 2016; de la Torre-Abaitua *et al.*, 2021), etc. Néanmoins, la détection d'anomalies dans les textes présente des difficultés particulières liées aux spécificités de la donnée textuelle, notamment :

— l'espace de caractéristiques des données textuelles est généralement de grande dimension et très clairsemé ;

— les anomalies peuvent survenir à différents niveaux linguistiques (lexical, syntaxique, sémantique ou pragmatique), ce qui augmente la difficulté de distinguer les anomalies d'intérêt du bruit ;

— pour certaines applications comme la détection de nouvelles thématiques dans la presse, la normalité (par exemple, la thématique « normale » ou « dominante ») change constamment, et la « norme » sur la base de laquelle les anomalies sont déterminées est donc difficile à définir.

Plusieurs approches générales, visant l'identification de points de données aberrants, ont été appliqués aux données textuelles, certaines ayant été développées spécifiquement pour le texte. Cependant, elles exploitent peu les nouvelles avancées du traitement automatique des langues naturelles (TALN).

En effet, le domaine du TALN a connu ces dernières années une importante progression des modèles de langage pré-entraînés (MLPs) comme BERT (Devlin *et al.*, 2019), GPT-2 (Radford *et al.*, 2019), GPT-3 (Brown *et al.*, 2020) ou encore T5 (Raffel *et al.*, 2020). L'émergence de ces modèles a donné naissance à un nouveau paradigme de l'apprentissage automatique appelé « *prompt engineering* » (Liu *et al.*, 2021) que nous proposons de traduire par « ingénierie d'invite ». Ce nouveau paradigme permet d'utiliser des modèles génératifs comme *few-shot learners* et a montré de bonnes performances sur plusieurs tâches du TALN (Brown *et al.*, 2020).

Cet article présente un travail exploratoire visant à examiner la possibilité de détecter des anomalies

textuelles à l'aide de l'ingénierie d'invite.

2 Travaux Connexes

La détection d'anomalies a été largement étudiée par différentes communautés de recherche et selon différentes approches théoriques (Chandola *et al.*, 2009; Pimentel *et al.*, 2014). Cependant, la problématique spécifique de détection d'anomalies textuelles n'a bénéficié que d'un nombre limité de travaux. Les travaux existants se concentrent essentiellement sur deux axes, à savoir les différentes techniques permettant de distinguer les anomalies des données normales ainsi que les différentes représentations des données textuelles. Typiquement, le système de détection d'anomalies est un système d'apprentissage automatique (supervisé, semi-supervisé ou non supervisé) qui prend comme entrée un texte et produit, en sortie, soit une étiquette, soit un score d'anomalie.

Approches à base de classement unitaire Une des approches courantes pour la détection d'anomalies est le classement unitaire où l'objectif est d'entraîner un modèle qui décrive la « normalité » à partir des données d'entraînement n'ayant que l'étiquette « normale ». Dans ce cadre, plusieurs algorithmes sont proposés, notamment ceux basés sur la frontière de décision, tels que des variants de *One-Class Support Vector Machines* (OCSVM) (Schölkopf *et al.*, 2001) et de *Support Vector Data Description* (SVDD) (Tax & Duin, 2004). Manevitz & Yousef (2001) ont effectué un classement unitaire sur le jeu de données Reuters [1] (Lewis, 1997) en utilisant plusieurs versions de SVM avec différents noyaux et différentes représentations de données comme *One Hot*, TF-IDF, etc. Yu *et al.* (2002, 2004) ont proposé une technique basée sur SVM appelée PEBL (*Positive Example Based Learning*) pour classer les pages Web. Ruff *et al.* (2019) ont proposé la technique dite *Context Vector Data Description* (CVDD), qui est une extension des classifieurs unitaires utilisant des modèles de langage (plongements de mots) pré-entraînés, tels que word2vec (Mikolov *et al.*, 2017), GloVe (Pennington *et al.*, 2014) et FastText (Joulin *et al.*, 2016). Avec l'essor de l'apprentissage profond (*Deep Learning*) au cours de ces dernières années, plusieurs variants du modèle SVDD intégrant ces techniques sont proposés, tels que DSVDD (*Deep* SVDD) (Ruff *et al.*, 2018), mSVDD (*multi-modal deep* SVDD) (Hu *et al.*, 2021), etc.

Approches à base de proximité Les méthodes à base de proximité, qui ont déjà fait leurs preuves dans plusieurs domaines (Munz *et al.*, 2007; Papadimitriou *et al.*, 2003; Tian *et al.*, 2014), sont également utilisées pour détecter des anomalies dans le texte. Ces approches considèrent un point de données comme anomal lorsque son voisinage est peu peuplé (Aggarwal, 2017). Selon les différentes définitions de la proximité, nous pouvons regrouper ces techniques en trois sous-classes :

— Les méthodes à base de *cluster*. Srivastava & Zane-Ulman (2005); Srivastava *et al.* (2006) ont utilisé diverses techniques de clustering comme *k-means clustering* (Macqueen, 1967) et *spectral clustering* pour détecter des anomalies dans des rapports de maintenance de systèmes complexes.

— Les méthodes à base de distance, telles que les k plus proches voisins (k-NN) (Cover & Hart, 1967). Allan *et al.* (2000b) ont utilisé les k-NN pour détecter des nouvelles thématiques dans le flux d'actualités.

— Les méthodes à base de densité, telles que *Local Outlier Factor* (LOF) (Breunig *et al.*, 2000) et

1. https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester *et al.*, 1996). Song & Suh (2019) ont utilisé LOF pour détecter des conditions anomales dans des rapports d'accidents.

Approches à base de reconstruction Les méthodes à base de reconstruction tentent de trouver un sous-espace de dimension inférieure des données originales où les anomalies et les données normales sont censées être séparées les unes des autres. Cette représentation de données compressée est utilisée ensuite pour la reconstruction des données originales. La différence entre les données reproduites et les données originales, autrement dit l'erreur de reconstruction, est utilisée comme score d'anomalie. Les points de données mal reconstruits sont considérés comme des anomalies. Plusieurs techniques s'appuyant sur ce type d'approche ont été proposées pour la détection d'anomalies, parmi lesquelles *Principal Component Analysis* (PCA), *Replicator Neural Networks* (RNNs) (Hecht-Nielsen, 1995), *Autoencoders* (AEs) et *Generative Adversarial Networks* (GANs) (Goodfellow *et al.*, 2014). Par exemple, Yap (2020) a proposé une méthode appelée ARAE-AnoGAN qui combine l'AE et le GAN pour détecter des anomalies textuelles.

3 Détection d'anomalies à base de l'ingénierie d'invite

3.1 Ingénierie d'invite

Au cours des deux dernières décennies, le TALN basé sur l'apprentissage automatique a connu une évolution constante de paradigme, de l'ingénierie des caractéristiques au *pre-train* et *fine-tune* en passant par l'ingénierie de l'architecture (de réseaux de neurones). Le paradigme de *pre-train* et *fine-tune* est devenu la norme *de facto* pour le TALN et a permis d'obtenir d'excellents résultats dans de multiples tâches. Dans le paradigme de *pre-train* et *fine-tune*, un modèle de langage est pré-entraîné en appliquant un apprentissage non supervisé ou autosupervisé sur des corpus non étiquetés à grande échelle, ce qui lui permet de capturer des connaissances riches (lexicales, syntaxiques, sémantiques et factuelles) et des caractéristiques d'usage général de la langue. Les modèles de langage sont entraînés avec une tâche de pré-entraînement dont l'objectif principal est crucial pour apprendre la représentation du langage. Parmi ces tâches, on retrouve *Standard Language Modeling* et *Masked Language Modeling* (Qiu *et al.*, 2020). Le modèle de langage pré-entraîné (MLP) sera ensuite adapté à différentes tâches en aval en introduisant des paramètres supplémentaires et en les ajustant (*fine-tuning*) à ces tâches à l'aide de fonctions d'objectif spécifiques.

Comme l'évolution technique se poursuit, un nouveau paradigme appelé « ingénierie d'invite (*prompt engineering*) » a récemment été proposé (Liu *et al.*, 2021). Contrairement au paradigme de *pre-train* et *fine-tune*, qui vise à adapter le MLP à la nouvelle tâche (par exemple, via l'ingénierie de fonction objectif), le principe de base de l'ingénierie d'invite est de reformuler la nouvelle tâche pour la faire ressembler à celle résolue lors de l'entraînement initial du modèle de langage à l'aide d'une invite textuelle (Liu *et al.*, 2021). Prenons, par exemple, la tâche de classement des thématiques : étant donné un texte « *The 2017 French presidential election was held on 23 April and 7 May 2017.* », nous pouvons poursuivre le texte en posant une question comme « *What is this text about ?* », exprimée sous la forme de « *The text is about* ___. » et laisser le MLP remplir le trou avec un mot portant la thématique. Le modèle a plus de chances de produire le mot « *politics* » que d'autres mots comme « *sports* ». De cette façon, à l'aide d'une amorce textuelle bien choisie, c'est-à-dire d'une invite

(*prompt*), le MLP peut prédire le résultat que l'on attend, car nous avons transformé la tâche de classement de thématique en une tâche de génération de texte pour laquelle le MLP est conçu.

3.2 Détection d'anomalies à base d'invite

La détection d'anomalies est traditionnellement considérée comme une tâche de détection ou de classement unitaire où les techniques du TALN, notamment les MLPs, jouent un rôle secondaire. Notre travail consiste à transformer la détection d'anomalie en problème de génération de texte à l'aide de l'ingénierie d'invite, ce qui nous permet de profiter des avantages des MLPs. La conception de notre méthode suit la méthodologie proposée par (Liu *et al.*, 2021). La méthode est constituée de trois composants : la sélection de MLP, l'ingénierie d'invite et l'ingénierie de réponse.

Modèle pré-entraîné Comme dans le paradigme de *pre-train* et *fine-tune*, le MLP joue un rôle crucial dans celui de l'ingénierie d'invite. Plus précisément, la tâche de pré-entraînement d'un MLP est déterminant pour son applicabilité à une tâche cible donnée. Dans le cadre de ce travail, nous nous intéressons en particulier à la génération conditionnelle de texte. Ainsi choisissons-nous le modèle GPT-2 qui est pré-entraîné pour un objectif simple : prédire le mot suivant (Radford *et al.*, 2019).

Ingénierie d'invite L'ingénierie d'invite consiste à définir une fonction d'invite $f_{prompt}(x)$ qui transforme l'entrée en une forme spécifique. Pour créer une invite textuelle, la fonction insère l'entrée x dans un modèle (*template*) (par exemple, « $[x]$ The text is about $[z]$ ») et ajoute un trou $[z]$ dans lequel la réponse z est censée être remplie par le MLP. Étant donné que l'invite spécifie la tâche, le choix d'une invite appropriée a un effet important sur les résultats. Il existe deux types d'invite :

— l'invite comme un trou (*cloze prompt*), qui invite le MLP à remplir le trou dans un texte (par exemple, « $[x]$ The text is about $[z]$ ») ;
— l'invite comme un préfixe (*prefix prompt*), qui invite le MLP à continuer un préfixe (par exemple, « TEXT : $[x]$ TOPIC : $[z]$ »).

Ingénierie de réponse L'ingénierie de réponse consiste à définir l'espace de réponse (à l'invite) ainsi qu'un mappage de l'espace de réponse vers l'espace d'étiquette. La réponse peut prendre plusieurs formes plus ou moins complexes comme un mot ({"*normal*", "*anomalous*"}) ou un document (par exemple, un résumé). L'espace de réponse peut être limité ({"*yes*", "*no*"}) ou illimité (par exemple, les thématiques des textes).

Généralement, les réponses ne coïncident pas avec les sorties finales (les étiquettes attendues) du système. Nous devons également créer une fonction de mappage pour projeter les réponses dans l'espace de sortie (par exemple, f_{map} : {"*politics*", "*sports*", "*business*", $\cdots$}$\rightarrow$ {"*True*", "*False*"}).

Configuration Dans cette étude, nous avons examiné deux configurations (Voir Table 1). Les invites sont manuellement conçues, et nous nous focalisons essentiellement sur leur forme. Les réponses générées par le MLP sont présentées sous forme d'adjectifs décrivant la normalité du texte ({"*normal*", "*anomalous*"}). Ces réponses sont converties en étiquette booléenne à la sortie.

Dans nos premières expérimentations, nous avons aussi envisagé d'utiliser deux autres configurations avec un espace de réponses illimitées, à savoir les thématiques de texte (une séquence de mots comme « *Asian Economic Crisis* » et « *Monica Lewinsky Case* »), mais nous n'avons pas pu aller jusqu'au

bout, car le comportement imprévu du MLP ne nous permet pas de transformer correctement les résultats en sortie binaire. Ces résultats seront expliqués davantage dans la section 4.3.

ID	Type d'invite	Modèle	Mappage (Réponse [z] ->Sortie)
Config. 1	trou	$[x]$ *This is* $[z]$ *text.*	*(an) anomalous* $\rightarrow$ *True* *(a) normal* $\rightarrow$ *False*
Config. 2	préfixe	*TEXT :* $[x]$ *NORMALITY :* $[z]$	

TABLE 1 – Ingénierie d'invite et ingénierie de réponse pour la détection d'anomalies

4 Expérimentations

4.1 Données et modèle du langage pré-entraîné

Nous avons mené des expérimentations sur le corpus TDT-2 Multilanguage V4.0 [2] (Cieri *et al.*, 1999). Il s'agit d'une collection de textes journalistiques provenant de multiples sources. Les textes sont annotés selon leur thématique et chaque texte a une thématique unique. Le corpus contient environ 10 000 documents avec plus de 200 thématiques annotées. TDT-2 est largement utilisé dans des tâches de détection de nouveautés dans les textes comme *First Story Detection* (Allan *et al.*, 2000a). Dans nos expérimentations, nous ne prenons en compte que les documents en anglais contenant moins de 300 mots et les 40 premières thématiques (T1-T40) [3] ayant plus de 10 documents. Les documents de T1 (« *Asian Economic Crisis* ») sont utilisés comme les textes normaux et tous les autres textes (T2-T40 : « *Monica Lewinsky Case* », « *1998 Winter Olympics* », etc.) sont considérés comme anomalies. Nous distinguons deux types de jeu de données :

— Le jeu de données d'entraînement : 800 documents de T1 sont utilisés comme données d'entraînement pour les méthodes semi-supervisées ; 200 documents complémentaires de T2-T20 sont ajoutés pour les méthodes non supervisées et les méthodes à base d'invite.
— Le jeu de données de test : les données de test comportent 200 documents de T1, 400 documents de T2-T20 (valeurs aberrantes) et 400 documents de T21-T40 (nouveautés).

Tous les documents sont choisis avec un échantillonnage aléatoire stratifié. Nous avons utilisé le modèle « gpt-2 » fourni par HuggingFace [4]. Les paramètres pour la génération de texte sont comme suit :

— temperature : 0
— top_k : 50
— top_p : 0.9

4.2 Bases de référence et métriques d'évaluation

Afin de mesurer la performance de notre méthode, nous la comparons avec les méthodes existantes. Nous avons choisi les méthodes OCSVM, LOF et *K-means Clustering*, car elles sont largement

2. https://catalog.ldc.upenn.edu/LDC2001T57
3. https://catalog.ldc.upenn.edu/docs/LDC2001T57/tdt2topics.html
4. https://huggingface.co/byeongal/gpt2

utilisées comme bases de référence dans la tâche de détection d'anomalies textuelles (Barrett *et al.*, 2019; Ruff *et al.*, 2019). Contrairement à la méthode à base d'invite qui est supervisée, ces méthodes sont généralement utilisées dans les scénarios non supervisés ou semi-supervisés. Les textes sont représentés par un modèle vectoriel de TF-IDF dont l'IDF est calculé sur les textes normaux.

La détection d'anomalies peut être évaluée sous deux perspectives différentes : soit comme une tâche de détection, soit comme une tâche de classement (unitaire ou binaire). Dans notre étude, nous la considérons comme une tâche de classement binaire et utilisons les métriques les plus courantes, à savoir la précision, le rappel, le F-score et l'*Area Under the Curve* (AUC). Par ailleurs, étant donné que les données sont déséquilibrées (les anomalies sont rares) dans les scénarios réels, nous calculons aussi la mesure *Matthews Correlation Coefficient* (MCC) (Matthews, 1975) qui est connue pour sa capacité de traiter les cas de données déséquilibrées.

4.3 Résultats et discussion

La Table 2 montre les résultats obtenus dans nos expérimentations. En ce qui concerne les bases de référence, la méthode LOF surpasse les deux autres méthodes sur les cinq métriques en obtenant un F score de $0{,}83$ et un MCC score de $0{,}48$.

Pour les méthodes à base de l'ingénierie d'invite, dans nos premières expérimentations, en plus des configurations 1 & 2 dont l'espace de réponse est limité, nous avons aussi exploré des configurations dont l'espace de réponse est illimité. Les résultats de ces dernières n'ont pas été retenus, car le MLP manifeste un comportement imprévu. En effet, les expérimentations réalisées ont montré que l'utilisation de ce type de configuration impliquant une réponse de type contenu illimité générait souvent des séquences inexploitables, constituées par exemple de beaucoup de mots vides, de longues répétitions d'un même mot. De plus, les modèles ainsi construits présentaient une importante instabilité et sensibilité aux différents paramétrages (notamment longueur maximale de réponse). Ces comportements imprévus ont empêché l'évaluation des résultats.

Pour les configurations 1 et 2, nous constatons que les modèles sont en fait capables de prédire parfaitement l'anormalité du texte. Ils peuvent compléter le texte avec une réponse (qui est convertie en une valeur booléenne) puis arrêter l'exécution par un token de fin de phrase. La configuration 2 a obtenu un F score de $0{,}98$, ce qui est sensiblement supérieur à celui obtenu par la méthode LOF ($0{,}79$). Il convient également de noter que la configuration 2 a obtenu un score MCC de $0{,}92$, ce qui signifie que le modèle a des bons résultats pour les quatre catégories de la matrice de confusion (vrais positifs, faux négatifs, vrais négatifs et faux positifs) même si les données sont déséquilibrées. De plus, la Table 3 montre que les résultats pour les thématiques non vues (nouveautés) en apprentissage sont aussi bons que ceux des thématiques connues, ce qui signifie que l'invite d'ingénierie est une nouvelle étape vers le *Zero-shot Learning* (Xian *et al.*, 2017).

5 Conclusion et perspectives

La détection d'anomalies utilise traditionnellement des techniques semi-supervisés ou non supervisés à cause du manque d'échantillons négatifs étiquetés. Dans cette étude, nous avons présenté une méthode supervisée de détection d'anomalies textuelles à l'aide de l'ingénierie d'invite. Nous avons examiné deux configurations conçues de manière manuelle. Les résultats obtenus ont largement

Méthode	P	R	F	AUC	MCC
OCSVM + TFIDF	0.92	0.49	0.64	0.65	0.25
K-means + TFIDF	0.88	0.63	0.73	0.65	0.24
LOF + TFIDF	0.95	0.74	0.83	0.79	0.48
Config. 1 + GPT-2	**1.00**	**0.92**	**0.96**	**0.96**	0.82
Config. 2 + GPT-2	**1.00**	**0.97**	**0.98**	**0.98**	**0.93**

TABLE 2 – Résultats par méthode, avec P (Précision), Rappel (R), F (score), AUC et MCC, pour la détection d'anomalies dans le corpus de presse (TDT-2).

Thématique	Config. 1			Config. 2		
	P	R	F	P	R	F
T2-T20	1.00	0.92	0.96	1.00	0.98	0.99
T21-T40	1.00	0.92	0.96	1.00	0.97	0.98

TABLE 3 – Résultats sous les configurations 1 et 2 : comparaison des résultats entre les thématiques anomales observées en apprentissage (T2-T20) et les anomalies non vues (T21-T40).

surpassé les bases de référence, ce qui montre que le MLP est un outil prometteur pour la détection d'anomalies textuelles grâce au paradigme de l'ingénierie d'invite. Vu que l'anomalie peut être définie différemment selon le scénario d'application, nous envisageons, dans nos travaux futurs, de tester la méthode sur d'autres corpus comme Reuters-21578 (Lewis, 1997), *20 NewsGroups* [5] et *Amazon Reviews* (Keung *et al.*, 2020) pour valider notre hypothèse. Comme perspectives, nous envisageons également d'explorer les différents aspects de l'ingénierie d'invite, tels que la conception automatique d'invite et de réponse et l'utilisation de multi-invites. Par ailleurs, nous chercherons à appliquer d'autres techniques récentes du TAL comme SBERT (Reimers & Gurevych, 2019) à la détection d'anomalies.

Remerciements

Ce travail a été réalisé dans le cadre d'une convention CIFRE, financée par l'Association nationale de la recherche et de la technologie (ANRT), et établie entre l'Equipe de Recherche Textes, Informatique, Multilinguisme (ERTIM) de l'INaLCO et la société Flandrin IT.

Références

AGGARWAL C. C. (2017). *Outlier Analysis*. Cham : Springer International Publishing. DOI : 10.1007/978-3-319-47578-3.

ALESKEROV E., FREISLEBEN B. & RAO B. (1997). CARDWATCH : A neural network based database mining system for credit card fraud detection. In *Proceedings of the IEEE/IAFE 1997 Computational Intelligence for Financial Engineering (CIFEr)*, p. 220–226, New York City, NY, USA : IEEE. DOI : 10.1109/CIFER.1997.618940.

5. http://qwone.com/~jason/20Newsgroups/

ALLAN J., LAVRENKO V. & JIN H. (2000a). First story detection in TDT is hard. In *Proceedings of the Ninth International Conference on Information and Knowledge Management - CIKM '00*, p. 374–381, McLean, Virginia, United States : ACM Press. DOI : 10.1145/354756.354843.

ALLAN J., LAVRENKO V., MALIN D. & SWAN R. (2000b). Detections, bounds, and timelines : Umass and tdt-3. In *Proceedings of Topic Detection and Tracking Workshop*, p. 167–174 : Citeseer.

AMATO F., COZZOLINO G., MAZZEO A. & ROMANO S. (2016). Detecting anomalies in Twitter stream for public security issues. In *2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a Better Tomorrow (RTSI)*, p. 1–4, Bologna, Italy : IEEE. DOI : 10.1109/RTSI.2016.7740574.

BARRETT L., FLETCHER S. & KINGAN R. (2019). Textual Outlier Detection and Anomalies in Financial Reporting. In *2nd KDD Workshop on Anomaly Detection in Finance*, p.6̃.

BREUNIG M. M., KRIEGEL H.-P., NG R. T. & SANDER J. (2000). LOF : Identifying density-based local outliers. *ACM SIGMOD Record*, **29**(2), 93–104. DOI : 10.1145/335191.335388.

BROWN T. B., MANN B., RYDER N., SUBBIAH M., KAPLAN J., DHARIWAL P., NEELAKANTAN A., SHYAM P., SASTRY G., ASKELL A., AGARWAL S., HERBERT-VOSS A., KRUEGER G., HENIGHAN T., CHILD R., RAMESH A., ZIEGLER D. M., WU J., WINTER C., HESSE C., CHEN M., SIGLER E., LITWIN M., GRAY S., CHESS B., CLARK J., BERNER C., MCCANDLISH S., RADFORD A., SUTSKEVER I. & AMODEI D. (2020). Language Models are Few-Shot Learners. *Advances in neural information processing systems*, **33**, 1877–1901.

CHANDOLA V., BANERJEE A. & KUMAR V. (2009). Anomaly detection : A survey. *ACM Computing Surveys*, **41**(3), 1–58. DOI : 10.1145/1541880.1541882.

CIERI C., GRAFF D., LIBERMAN M., MARTEY N., STRASSEL S. *et al.* (1999). The TDT-2 text and speech corpus. In *Proceedings of the DARPA Broadcast News Workshop*, p. 57–60.

CLIFTON L., CLIFTON D. A., WATKINSON P. J. & TARASSENKO L. (2011). Identification of Patient Deterioration in Vital-Sign Data using One-Class Support Vector Machines. In *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)*, p. 125–131 : IEEE.

COVER T. & HART P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, **13**(1), 21–27. DOI : 10.1109/TIT.1967.1053964.

DE LA TORRE-ABAITUA G., LAGO-FERNÁNDEZ L. F. & ARROYO D. (2021). A Compression-Based Method for Detecting Anomalies in Textual Data. *Entropy*, **23**(5), 618. DOI : 10.3390/e23050618.

DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2019). BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv :1810.04805 [cs]*.

DIEHL C. & HAMPSHIRE J. (2002). Real-time object classification and novelty detection for collaborative video surveillance. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)*, p. 2620–2625, Honolulu, HI, USA : IEEE. DOI : 10.1109/IJCNN.2002.1007557.

ESTER M., KRIEGEL H.-P. & XU X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *KDD'96 : Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, p. 226–231 : AAAI Press.

GOODFELLOW I. J., POUGET-ABADIE J., MIRZA M., XU B., WARDE-FARLEY D., OZAIR S., COURVILLE A. & BENGIO Y. (2014). Generative Adversarial Networks. *arXiv :1406.2661 [cs, stat]*.

HAWKINS D. M. (1980). *Identification of Outliers*. Dordrecht : Springer Netherlands. DOI : 10.1007/978-94-015-3994-4.

HECHT-NIELSEN R. (1995). Replicator Neural Networks for Universal Optimal Source Coding. *Science*, **269**(5232), 1860–1863. DOI : 10.1126/science.269.5232.1860.

HU C., FENG Y., KAMIGAITO H., TAKAMURA H. & OKUMURA M. (2021). One-class Text Classification with Multi-modal Deep Support Vector Data Description. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, volume Main Volume, p. 3378–3390 : Association for Computational Linguistics. DOI : 10.18653/v1/2021.eacl-main.296.

JOULIN A., GRAVE E., BOJANOWSKI P., DOUZE M., JÉGOU H. & MIKOLOV T. (2016). FastText.zip : Compressing text classification models. *arXiv :1612.03651 [cs]*.

JYOTHSNA V., V. RAMA PRASAD V. & MUNIVARA PRASAD K. (2011). A Review of Anomaly based Intrusion Detection Systems. *International Journal of Computer Applications*, **28**(7), 26–35. DOI : 10.5120/3399-4730.

KEUNG P., LU Y., SZARVAS G. & SMITH N. A. (2020). The Multilingual Amazon Reviews Corpus. *arXiv :2010.02573 [cs]*.

KIM K. H., HAN Y. J., LEE S., CHO S. W. & LEE C. (2019). Text Mining for Patent Analysis to Forecast Emerging Technologies in Wireless Power Transfer. *Sustainability*, **11**(22), 6240. DOI : 10.3390/su11226240.

LABIB K. & VEMURI R. (2002). NSOM : A Real-Time Network-Based Intrusion Detection System Using Self-Organizing Maps. *Networks and Security*, **21**, 6.

LEWIS D. D. (1997). Reuters-21578 Text Categorization Collection Data Set. https ://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection.

LIU P., YUAN W., FU J., JIANG Z., HAYASHI H. & NEUBIG G. (2021). Pre-train, Prompt, and Predict : A Systematic Survey of Prompting Methods in Natural Language Processing. *arXiv :2107.13586 [cs]*.

MACQUEEN J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, p. 281–297, Oakland, CA, USA.

MANEVITZ L. M. & YOUSEF M. (2001). One-Class SVMs for Document Classification. p.1̃6.

MATTHEWS B. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, **405**(2), 442–451. DOI : 10.1016/0005-2795(75)90109-9.

MIKOLOV T., GRAVE E., BOJANOWSKI P., PUHRSCH C. & JOULIN A. (2017). Advances in Pre-Training Distributed Word Representations. *arXiv :1712.09405 [cs]*.

MUNZ G., LI S. & CARLE G. (2007). Traffic Anomaly Detection Using K-Means Clustering. In *GI/ITG Workshop MMBnet*, p.8̃.

PAPADIMITRIOU S., KITAGAWA H., GIBBONS P. & FALOUTSOS C. (2003). LOCI : Fast outlier detection using the local correlation integral. In *Proceedings 19th International Conference on Data Engineering (Cat. No.03CH37405)*, p. 315–326, Bangalore, India : IEEE. DOI : 10.1109/ICDE.2003.1260802.

PENNINGTON J., SOCHER R. & MANNING C. (2014). Glove : Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language*

Processing (EMNLP), p. 1532–1543, Doha, Qatar : Association for Computational Linguistics. DOI : 10.3115/v1/D14-1162.

PIMENTEL M. A., CLIFTON D. A., CLIFTON L. & TARASSENKO L. (2014). A review of novelty detection. *Signal Processing*, **99**, 215–249. DOI : 10.1016/j.sigpro.2013.12.026.

QIU X., SUN T., XU Y., SHAO Y., DAI N. & HUANG X. (2020). Pre-trained Models for Natural Language Processing : A Survey. *Science China Technological Sciences*, **63**(10), 1872–1897. DOI : 10.1007/s11431-020-1647-3.

QUINN J. A. & WILLIAMS C. K. I. (2007). Known Unknowns : Novelty Detection in Condition Monitoring. In J. MARTÍ, J. M. BENEDÍ, A. M. MENDONÇA & J. SERRAT, Éds., *Pattern Recognition and Image Analysis*, volume 4477, p. 1–6. Berlin, Heidelberg : Springer Berlin Heidelberg. DOI : 10.1007/978-3-540-72847-4_1.

RADFORD A., WU J., CHILD R., LUAN D., AMODEI D. & SUTSKEVER I. (2019). Language Models are Unsupervised Multitask Learners. *OpenAI blog*, **1**, 24.

RAFFEL C., SHAZEER N., ROBERTS A., LEE K., NARANG S., MATENA M., ZHOU Y., LI W. & LIU P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *arXiv :1910.10683 [cs, stat]*.

REIMERS N. & GUREVYCH I. (2019). Sentence-BERT : Sentence Embeddings using Siamese BERT-Networks. *arXiv :1908.10084 [cs]*.

RUFF L., VANDERMEULEN R. A., GÖRNITZ N., DEECKE L., SIDDIQUI S. A., BINDER A., MÜLLER E. & KLOFT M. (2018). Deep One-Class Classification. In *Proceedings of Machine Learning Research*, p. 4393–4402 : PMLR.

RUFF L., ZEMLYANSKIY Y., VANDERMEULEN R., SCHNAKE T. & KLOFT M. (2019). Self-Attentive, Multi-Context One-Class Classification for Unsupervised Anomaly Detection on Text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, p. 4061–4071, Florence, Italy : Association for Computational Linguistics. DOI : 10.18653/v1/P19-1398.

SCHÖLKOPF B., PLATT J. C., SHAWE-TAYLOR J., SMOLA A. J. & WILLIAMSON R. C. (2001). Estimating the Support of a High-Dimensional Distribution. *Neural Computation*, **13**(7), 1443–1471. DOI : 10.1162/089976601750264965.

SONG B. & SUH Y. (2019). Narrative texts-based anomaly detection using accident report documents : The case of chemical process safety. *Journal of Loss Prevention in the Process Industries*, **57**, 47–54. DOI : 10.1016/j.jlp.2018.08.010.

SRIVASTAVA A., AKELLA R., DIEV V., KUMARESAN S., MCINTOSH D., PONTIKAKIS E., ZUOBING XU & YI ZHANG (2006). Enabling the Discovery of Recurring Anomalies in Aerospace Problem Reports using High-Dimensional Clustering Techniques. In *2006 IEEE Aerospace Conference*, p. 1–17, Big Sky, MT, USA : IEEE. DOI : 10.1109/AERO.2006.1656136.

SRIVASTAVA A. & ZANE-ULMAN B. (2005). Discovering recurring anomalies in text reports regarding complex space systems. In *2005 IEEE Aerospace Conference*, p. 3853–3862, Big Sky, MT, USA : IEEE. DOI : 10.1109/AERO.2005.1559692.

TAX D. M. & DUIN R. P. (2004). Support Vector Data Description. *Machine Learning*, **54**(1), 45–66. DOI : 10.1023/B :MACH.0000008084.60811.49.

TIAN J., AZARIAN M. H. & PECHT M. (2014). Anomaly Detection Using Self-Organizing Maps-Based K-Nearest Neighbor Algorithm. In *PHM Society European Conference*, p.9̃.

WANG J. & CHEN Y.-J. (2019). A novelty detection patent mining approach for analyzing technological opportunities. *Advanced Engineering Informatics*, **42**, 100941. DOI : 10.1016/j.aei.2019.100941.

XIAN Y., SCHIELE B. & AKATA Z. (2017). Zero-Shot Learning — The Good, the Bad and the Ugly. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 3077–3086, Honolulu, HI : IEEE. DOI : 10.1109/CVPR.2017.328.

YAP T. Y. (2020). Text Anomaly Detection with ARAE-AnoGAN. *Honors Projects*, **22**.

YU H., HAN J. & CHANG K. C.-C. (2002). PEBL : Positive example based learning for Web page classification using SVM. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '02*, p. 239, Edmonton, Alberta, Canada : ACM Press. DOI : 10.1145/775047.775083.

YU H., HAN J. & CHANG K. C.-C. (2004). Pebl :web page classification without negative examples. *IEEE Transactions on Knowledge and Data Engineering*, **16**(1), 70–81. DOI : 10.1109/TKDE.2004.1264823.