Étude de l'influence des représentations textuelles sur la détection d'évènements non supervisée dans des flux de données

Elliot Maître, Zakaria Chemli, Max Chevalier, Bernard Dousset, Jean-Philippe Gitto, Olivier Teste

Étude de l'influence des représentations textuelles sur la détection d'évènements dans des flux de données

Elliot Maître [1, 2], **Zakaria Chemli** [2], **Max Chevalier** [1], **Bernard Douset** [1], **Jean-Philippe Gitto** [2], **Olivier Teste** [1]

1. IRIT
118, Route de Narbonne, 31062 Toulouse cedex 04, France
prenom.nom@irit.fr

2. Scalian,
Bâtiment Oméga, 22, bd Déodat de Séverac, 31770 Colomiers, France
prenom.nom@scalian.com

RÉSUMÉ. *La détection d'évènements à partir des données postées sur internet est un sujet important de la recherche d'information. Les sources de données potentiellement intéressantes sont multiples et peuvent prendre la forme de flux de données textuelles plus ou moins structurées. Nous étudions dans cet article la détection d'évènements dans les flux de données textuelles et plus particulièrement l'impact de la représentation du texte sur la qualité des évènements détectés. Nous comparons différentes approches de traitement du langage dans deux contextes : supervisé et non supervisé. Nous étudions la question de l'efficacité des modèles basés sur les architectures Transformer pour la détection d'évènements dans les documents courts. Cette étude nous permet de conclure que, contrairement à ce qui avait pu être précédemment montré, les architectures Transformer peuvent être compétitives par rapport aux méthodes classiques.*

ABSTRACT. *Detection of real-world events using online data sources is a trending topic in the information retrieval domain. Multiple data sources are potentially of interest and some of them are data streams. There are multiple data sources that are potentially interesting, and some of them are textual data streams, structured or unstructured. We propose to analyse the problem of event detection from text data stream and to focus particularly on the importance of the representation of the textual data. To do so, we compare multiple approaches in different context: supervised and unsupervised. We focus on the performances of Transformer-based architectures for event detection on short text documents, and we conclude that, contrary to previous studies, these architectures can be competitive compared to classical methods.*

MOTS-CLÉS : *Fouille de texte, Recherche d'Information, Détection d'évènements, Traitement automatique du langage naturel, Partitionnement de données*

KEYWORDS: *Text mining, Information Retrieval, Event Detection, Natural Language Processing, Clustering*

DOI:10.3166/INFORSID...1-**??**

1. Introduction

De nombreux évènements se produisent constamment et sont à l'origine de perturbations potentiellement importantes dans de nombreux domaines. Si l'exemple de la pandémie liée au virus Sars-cov2 est évidemment l'évènement venant le plus naturellement à l'esprit cette année, d'autres types d'évènements peuvent aussi avoir des impacts importants, comme les évènements politiques (élections présidentielles aux Etats-Unis), les évènements météorologiques ou encore les catastrophes naturelles. Malgré la facilité actuelle d'accès aux masses d'information, il est difficile d'avoir une vue exhaustive de l'ensemble des évènements se déroulant sur la planète, notamment du fait de la profusion d'informations. De manière à faciliter cette tâche de veille, des systèmes automatisés ont vu le jour afin de détecter les contenus importants. Une des manières d'aborder cette tâche est la détection d'évènements dans les données textuelles (Atefeh, Khreich, 2015), (Hasan *et al.*, 2018). En effet, un des principaux vecteurs de communication de la presse et sur Internet de manière générale sont des données constituées majoritairement de texte. Il est possible d'accéder à ces documents textuels via des flux, qu'ils soient issus de réseaux sociaux ou de journaux. La difficulté qui se présente est alors de réussir à trouver les sources intéressantes mais aussi d'être capable d'extraire l'information pertinente de ces flux.

Nous proposons au cours de cette étude une méthode de détection d'évènements dans les flux de données textuelles. Ce problème est très étudié dans la littérature (Sakaki *et al.*, 2010), (Weng, Lee, 2011), (Hasan *et al.*, 2019) et constitue un problème important de la fouille de données textuelles. Cette tâche peut se décomposer en différentes parties, notamment la détection, le suivi et l'extraction des évènements (Allan, 2012). Différentes approches sont possibles pour chacune de ces tâches. Nous nous focalisons ici sur la tâche de détection d'évènements. Cette tâche est souvent abordée comme un problème de partitionnement, dynamique ou non, où chacun des partitionnement correspond à un évènement ou à une sous-partie d'un évènement (Allan, 2012).

Nous souhaitons évaluer la pertinence de l'utilisation des modèles de langage basés sur des architectures Transformer, qui ont prouvé leur efficacité dans de nombreux domaines du TALN et qui tendent à remplacer les architectures basées sur les réseaux de neurones récurrents dans ces domaines, (Reimers, Gurevych, 2019), (Cer *et al.*, 2018) pour la détection d'évènements dans des textes courts. En effet, les performances de ces modèdles n'ont pas été évaluées dans un cadre de partitionnement classique et ont même été évaluées comme moins performantes que TF-IDF dans le cadre d'un partitionnement dynamique (Mazoyer *et al.*, 2020). Nous cherchons à montrer l'intérêt du partitionnement classique par rapport au partitionnement dynamique dans ce contexte. Afin de répondre à cette problématique, nous proposons une méthode de détection d'évènements dans des flux de données textuelles basée sur le partitionnement de données où le flux de données est découpé en fenêtres contenant un nombre fixe de tweets, de manière à pouvoir appliquer des algorithmes de partitionnement classiques, et ainsi s'extraire des contraintes imposées par le partitionnement dynamique. Cela permet de considérer l'ensemble des documents publiés au moment du partitionne-

ment, et non de devoir travailler avec des informations fragmentaires au fil de l'eau. Nous comparons notre approche à des approches de partitionnement dynamiques reconnues afin d'en valider la pertinence. Enfin, nous comparons différentes méthodes de représentation des données textuelles. Plus particulièrement, nous nous intéressons aux approches basées sur les architectures Transformers qui sont actuellement reconnues comme disruptives dans le domaine du TALN (Traitement Automatisé du Langage Naturel) mais n'ont pas encore prouvé leur efficacité pour des documents courts et peu structurés comme ceux issus des réseaux sociaux. Notre approche montre que, contrairement aux études précédemment menées (Mazoyer *et al.*, 2020), les approches basées sur des architectures Transformers peuvent avoir des performances similaires aux approches classiques dans ce contexte.

Le reste de cet article s'organise de la manière suivante. La section 2 présente les travaux de la littérature. Ensuite, la section 3 détaille notre approche. Enfin, nous présentons et discutons nos résultats dans la section 4.

2. Etat de l'art

Dans un premier temps, nous présentons les différentes manières de représenter le contenu textuel, notamment les approches vectorielles. Dans un second temps, nous étudions différentes approches existantes pour la détection d'évènements à partir de textes, avec une attention particulière sur les documents issus des réseaux sociaux.

2.1. Représentation du contenu textuel

Les méthodes de représentation du contenu textuel constituent un des enjeux majeurs des travaux relatifs à la recherche d'informations (Baeza-Yates, Ribeiro-Neto, 1999). La méthode constituant actuellement la référence est TF-IDF (Jones, 1972) qui permet de prendre en compte l'importance des mots dans la représentation du document en pondérant chaque mot de manière inversement proportionnelle au nombre de documents dans lesquels il apparaît. Ainsi, un mot apparaissant dans un document alors qu'il n'apparaît que peu dans le corpus est considéré comme porteur de beaucoup d'informations. Sa pondération dans le cadre de TF-IDF est donc forte. Cette représentation est très utilisée, même de nos jours, dans la recherche d'information et obtient de très bonnes performances, même sur les textes courts du type réseaux sociaux.

Ces représentations statistiques sont actuellement complétées par des représentations vectorielles, appelées plongement de mots, basées sur des approches d'apprentissage profond. Les auteurs de (Mikolov *et al.*, 2013) introduisent le modèle Word2vec qui correspond à une approche neuronale permettant d'associer à un mot un vecteur, qui est calculé grâce au contexte dans lequel le mot apparaît dans le jeu d'entraînement. Ainsi, le vecteur représentant un mot contient de l'information à propos de celui-ci. L'hypothèse faite pour la constitution de ces vecteurs est que des mots dont l'utilisation contextuelle est proche, seront porteur d'un sens similaire et donc seront

représentés par un vecteur proche. Des variations existent, comme le modèle FastText (Bojanowski *et al*., 2016) qui découpe les mots en sous-mots, permettant de prendre en compte la construction des mots, notamment les suffixes et les préfixes. Les modèles les plus récents sont basés sur des architectures Transformers (Vaswani *et al*., 2017). Le plus notable d'entre eux est BERT (Devlin *et al*., 2018). L'architecture de BERT peut s'appliquer à toutes les tâches grâce à une approche d'apprentissage par transfert (transfer learning) (Pan, Yang, 2010). En effet, le modèle est d'abord pré-entraîné sur deux types de tâches, prédire les mots masqués dans une phrase et prédire la phrase suivante. Un affinage (fine-tuning) est ensuite possible sur la tâche spécifique pour laquelle le modèle doit être utilisé.

Tous ces modèles permettent de représenter des mots mais ne permettent pas nécessairement de représenter des phrases. Une des premières approche est Skip-Thought, proposée par (Kiros *et al*., 2015). C'est une architecture encodeur-décodeur, entraînée de manière non supervisée à prédire les phrases voisines d'une phrase donnée dans un texte. Une autre approche classique est l'utilisation de réseaux siamois, c'est-à-dire deux réseaux de neurones en parallèle, possédant la même architecture et les mêmes poids, mais qui ne prendront pas la même entrée (Bromley *et al*., 1994). C'est notamment ce qui a été proposé par (Conneau *et al*., 2017) avec leur modèle InferSent. C'est un réseau LSTM bi-directionnel siamois entraîné de manière supervisée sur le jeu de données SNLI (Bowman *et al*., 2015). Ce jeu de données contient 570 000 paires de phrases annotées selon trois catégories : implication entre la première et la deuxième phrase, contradiction de la première avec la deuxième phrase, les phrases sont neutres entre elles. Un autre moyen de représenter les phrases est d'utiliser une architecture basée sur les Transformers (Cer *et al*., 2018). Universal Sentence Encoder (USE) est entraîné sur deux types de tâches, une supervisée, basée sur le jeu de données SNLI de la même manière que Infersent, et sur des tâches non supervisées, comme Skip-Thought. Les architectures Transformers peuvent aussi être utilisées sous forme de réseaux siamois. C'est notamment l'approche suivie dans Sentence BERT (S-BERT) présentée par (Reimers, Gurevych, 2019). Cette approche consiste à créer un réseau siamois de deux modèles BERT qui seront entrainés avec l'objectif de produire des vecteurs similaires pour des phrases dont le sens est proche et des vecteurs dissimilaires pour des phrases dont le sens est éloigné. Ensuite, une dernière couche de neurones est rajoutée, de manière à pouvoir être affinée sur des tâches spécifiques.

Dans la suite de ce papier, nous menons une étude comparative des modèles basés sur TF-IDF et ceux basés sur des architectures Transformers, en particulier S-BERT et USE.

2.2. La détection d'évènements

La détection d'évènements sur les réseaux sociaux est une tâche de fouille de texte classique (Allahyari *et al*., 2017). Les réseaux sociaux sont particulièrement étudiés pour la détection d'évènements car ils sont très réactifs et des informations traitant

du court terme ou du long terme y sont discutées (Zubiaga *et al*., 2018). Le réseau le plus classiquement étudié est Twitter, car il est le plus performant pour la détection d'évènements (Hasan *et al*., 2018).

La détection d'évènements est un dérivé de la détection et du suivi de sujet (TDT : Topic Detection and Tracking), et peut être divisée en différentes sous tâches selon (Allan, 2012) : la segmentation de sujets, la détection de nouveaux sujets (FSD : First Story Setection), le partitionnement (Cluster Detection), le suivi et la détection de liens. Nous nous intéresserons plus particulièrement aux tâches de détection de nouveaux sujets et au partitionnement. Ces sous tâches peuvent être abordées de différentes manières, se divisant en deux grandes catégories : document-pivot et feature-pivot. La première consiste à travailler à l'échelle du document tandis que la seconde travaille à l'échelle du mot ou de groupement de mots. Nous choisissons de nous focaliser sur les approches document-pivot. En effet, ces approches permettent de considérer l'ensemble du contenu textuel du document et d'exploiter un maximum de sens.

L'algorithme de FSD a d'abord été introduit par (Allan *et al*., 2000) dans le système Umass puis a été amélioré par (Petrović *et al*., 2010) introduisant l'algorithme de FSD avec LSH (Locality Sensitive Hashing), permettant d'accélerer la recherche de plus proches voisins. L'objectif de cette méthode est de détecter le premier document faisant référence à un évènement. Le problème est ici abordé comme un problème de clustering dynamique des nouveaux documents. (Hasan *et al*., 2019) proposent d'utiliser l'algorithme de FSD pour évaluer la nouveauté d'un tweet et assigne ensuite le tweet à un cluster à l'aide de la différence entre ce tweet et la moyenne de représentation des clusters. Les représentations des tweets sont calculées à l'aide de TF-IDF. (Mazoyer *et al*., 2020) proposent de comparer les performances de Word2vec, TF-IDF, BERT et USE pour la détection de nouveaux sujets. Les auteurs de (Becker *et al*., 2011) proposent de grouper les tweets dans des clusters de messages similaires afin de déterminer quels messages parlent d'évènements ou non. Ils utilisent TF-IDF pour représenter les tweets puis calculent une similarité pour créer des clusters et les classer à l'aide d'un classifieur SVM (Machine à Vecteurs de Support). Dans (Boom *et al*., 2016), les auteurs prolongent les travaux de Becker et al. en utilisant un algorithme de clustering incrémental et en exploitant la sémantique des hashtags pour améliorer le clustering. Ils filtrent ensuite les évènements triviaux. (McMinn, Jose, 2015) utilisent aussi TF-IDF pour représenter les tweets et appliquent ensuite un algorithme de clustering incrémental se basant sur des critères de similarités et de taille des tweets pour les regrouper. Ils couplent cela avec des méthodes de filtrage pour permettre le passage à l'échelle de l'algorithme.

Dans la suite de cet article, nous proposons de comparer les méthodes basées sur les architectures Transformers à la méthode TF-IDF très majoritairement utilisée dans la littérature de manière à en évaluer les performances dans un contexte de partionnement classique. Nous comparons aussi les performances entre les contextes de partitionnement dynamique et de partitionnement classique. Pour cela, nous menons un comparatif similaire à celui proposé par (Mazoyer *et al*., 2020) avec notre méthode et nous comparons les résultats qu'ils obtiennent à ceux obtenus avec notre méthode.

3. Le moteur de détection d'évènements : EDF

3.1. Description de l'approche

Nous proposons d'aborder le problème de la détection d'évènements dans les flux de données textuelles comme une tâche de partitionnement de données (Allan, 2012). Cela permet de s'extraire des contraintes imposées par le partitionnement dynamique ; i.e., nous pouvons ainsi considérer l'ensemble des documents publiés au moment du partitionnement, et non de devoir travailler avec des informations fragmentaires au fil de l'eau. La méthode présentée est flexible, permettant d'utiliser tout type de représentation vectorielle du texte ainsi que tout algorithme classique de partitionnement de données. Cette flexibilité est particulièrement intéressante car il est important de pouvoir adapter la paire algorithme de représentation - algorithme de clustering. Afin de se ramener à un contexte de partitionnement classique, nous proposons de découper le flux entrant en fenêtre de densités de documents. Ainsi, il n'est pas nécessaire d'utiliser des algorithmes de partitionnement dynamiques. De plus, cette approche par fenêtre nous permet de nous assurer que les documents regroupés sont bien des documents dont la date de publication est proche. Cet élément est crucial afin que des évènements similaires mais totalement déconnectés dans le temps ne soient pas associés entre eux. Nous choisissons pour cette étude de découper en fenêtres de **w = 2000 tweets**. Ce choix nous assure d'avoir une fenêtre représentative en termes de nombres d'évènements présents dans la fenêtre, tout en conservant des intervalles de temps courts, afin de se rapprocher au plus juste du contexte de traitement d'un flux de données issu de Twitter. L'approche proposée de détection d'évènement proposée, appelée EDF (Event Detetection Framework) est constituée de trois étapes majeures, qui seront effectuées sur chaque fenêtre temporelle : (1) Représentation de chaque document sous forme vectorielle, (2) Calcul de la matrice de similarité entre tous les documents, (3) Application de l'algorithme de partitionnement. Ainsi, différentes combinaisons peuvent être envisagées pour chacune des trois étapes.

3.2. Algorithmes mis en oeuvre

Nous proposons de comparer différentes techniques de représentation des données textuelles dans deux contextes différents : le premier est le FSD, pour lequel nous nous basons sur les expérimentations réalisées dans (Mazoyer *et al.*, 2020). Le second est un contexte de partitionnement de données, pour lequel nous utiliserons l'approche EDF. Ainsi, pour chacun des deux contextes, nous effectuons les trois étapes présentées précédemment; i.e. la représentation des données textuelles, le calcul des similarités entre les représentations et le partitionnement des données.

L'algorithme de FSD consiste à assigner dynamiquement chaque nouveau document à un regroupement de données à l'aide d'un algorithme du plus proche voisin. Pour chaque nouveau document, la similarité avec le plus proche voisin est calculée. Si cette similarité est supérieure à un seuil défini, alors le document est assigné

au regroupement du plus proche voisin. Si non, un nouveau regroupement est créé, initialement composé uniquement du nouveau document.

Concernant EDF, nous proposons de travailler à l'échelle des documents. Cela permet notamment de considérer l'ensemble de l'information textuelle contenue dans le tweet et non uniquement des mots clefs. Notre approche consiste à calculer une représentation pour chaque document d'une fenêtre. La matrice de similarité entre documents est ensuite calculée. Cette matrice est utilisée pour constituer un graphe, dont les nœuds sont les documents et les arêtes entre chaque paire de document sont pondérées par la similarité entre ces documents. Les arêtes pour lesquelles les valeurs de similarités sont en dessous d'un seuil fixé sont supprimées. Une fois le graphe ainsi obtenu, un algorithme de partitionnement des données est utilisé, afin de calculer les regroupements. L'algorithme de clustering utilisé est l'algorithme de Louvain (Blondel *et al.*, 2008), utilisé notamment dans (Fedoryszak *et al.*, 2019). Cet algorithme a l'avantage de déterminer le nombre optimal de cluster sans qu'il soit spécifié au préalable. Nous utilisons comme mesure de similarité la Cosine Similarity, mesure la plus classique dans le clustering de texte (Aggarwal, Zhai, 2012). La Figure 1 illustre l'approche décrite.

Concernant les algorithmes de représentations des données, nous mettons en concurrence deux types de techniques : des approches statistiques, basées sur TF-IDF et des approches basées sur des architectures Transformers, USE et S-BERT. Nous avons choisi ces techniques car TF-IDF est un standard qui a prouvé plusieurs fois ses performances (Mazoyer *et al.*, 2020) tandis que S-BERT et USE sont des références pour la représentation de phrases basées sur les architectures Transformers (Reimers, Gurevych, 2019), (Cer *et al.*, 2018).

3.3. Présentation du jeu de données

Les auteurs de (McMinn *et al.*, 2013) proposent un jeu de données, nommé Event2012, constitué de 150 000 tweets annotés, au sein d'un corpus de plus de 120 millions de tweets issus du flux de données de Twitter, collectés entre Octobre et Novembre 2012. Ce dataset est particulièrement adapté à des approches document-pivot, puisque les labels portent sur les tweets complets (Fedoryszak *et al.*, 2019). Chaque tweet annoté est associé à un évènement et une catégorie d'évènements. Au total, 506 évènements ont été annotés et sont répartis dans 8 grandes catégories d'évènements. Afin de respecter les conditions d'utilisations de Twitter, seuls les identifiants des tweets sont partagés, permettant par la suite la récupération du contenu. Du fait de la suppression des tweets au cours du temps, nous avons pu récupérer 69875 tweets annotés, répartis dans 504 évènements.

De manière à simuler un contexte le plus proche possible d'un flux de données, nous avons choisi d'organiser le jeu de données par ordre de publication des tweets et de le découper en fenêtre de 2000 tweets. Ainsi, dans toutes les expériences présentées ci-après, les tweets sont présentés aux modèles dans leur ordre de publication initial. Ce paramètre est particulièrement important pour l'entraînement du modèle S-BERT

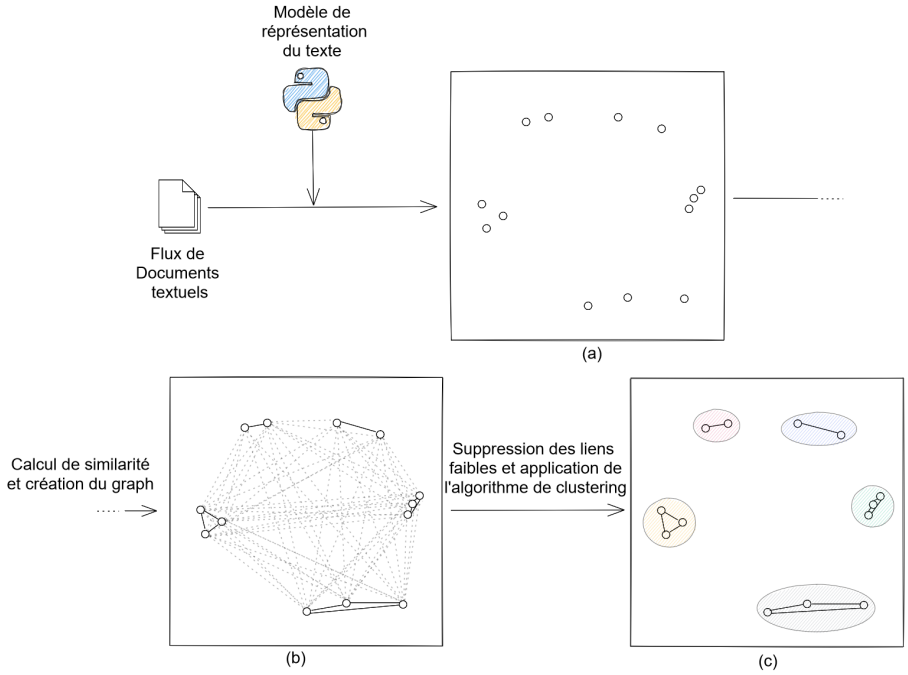

FIGURE *1. Processus de traitement des données réalisé par EDF à l'échelle d'une fenêtre. (a) Représentation des documents dans l'espace. Chacun point est un document. (b) Création du graphe à partir de la matrice de similarité. Chaque document est un sommet, chaque arête est pondérée par la similarité entre documents. (c) Création des partitionnements, en supprimant les arêtes dont le poids est trop faible.*

qui sera détaillé par la suite car la majorité des labels d'évènements présents dans le jeu d'entraînement ne sont pas présents dans le jeu de test. En effet, le jeu d'entraînement comporte 225 évènements, tandis que le jeu de test contient 303 évènements. Sur ces évènements, 24 sont en communs entre les deux collections. La répartition des données dans le dataset que nous avons pu récupérer est présentée dans la Figure 2. Nous pouvons donc constater que les fenêtres ne sont pas équivalentes entre elles, que ce soit en termes de nombres d'évènements étant discutés dans la fenêtre, mais aussi concernant la couverture temporelle de chaque fenêtre.

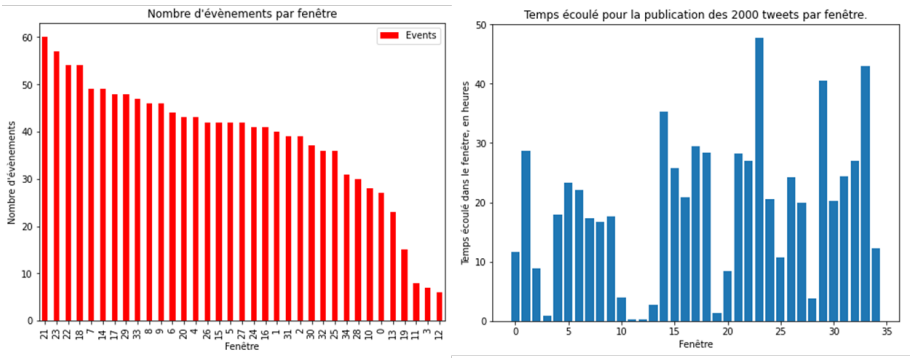

FIGURE 2. *Caractéristiques du dataset. La partie gauche illustre la disparité du nombre d'évènements par fenêtre de tweets. La partie droite de la figure illustre le temps en heures qu'il a fallu pour que les 2000 tweets de la fenêtre soient publiés.*

3.4. Modèles de représentation

Nous proposons deux variations de TF-IDF et de S-BERT, tandis que nous utilisons le modèle USE-large [1], que nous appellerons **USE**. Concernant TF-IDF, nous utilisons les implémentations proposées par (Mazoyer *et al.*, 2020). La première, que nous appellerons dans le reste de ce papier **TF-IDF dataset**, propose un IDF calculé sur les tweets labélisés du jeu de données. La seconde, **TF-IDF all tweets**, a un IDF calculé sur l'ensemble du jeu de données. Concernant S-BERT, la première version, nommée **S-BERT nli** est la version pré-entrainée sur le dataset NLI disponible dans les implémentations proposées par les auteurs du papier (Reimers, Gurevych, 2019) [2]. Ainsi, ce modèle est un réseau siamois, composé de deux modèles BERT égaux. Ces modèles ont été affinés sur le dataset SNLI discuté dans l'état de l'art. Nous avons choisi ce modèle de BERT notamment car le dataset NLI est connu pour améliorer les performances sur les tâches de clustering (Bowman *et al.*, 2015). Pour la seconde version, **S-BERT fine-tuned**, nous avons réalisé un affinage du modèle S-BERT sur le jeu d'entraînement, qui constitue la première moitié du jeu de données. Les évènements ont été utilisés comme labels d'entraînement. La particularité de cet entraînement vient de l'organisation temporelle du jeu de données : la majeure partie des évènements présents dans la collection de test ne le sont pas dans la collection d'entraînement, comme expliqué en partie 3.3. L'affinage a donc été réalisé sur 36 000 tweets, de manière à ce que la découpe jeu d'entraînement/jeu de test corresponde aux fenêtres de tweets que nous avons établies. Nous avons assigné à chaque tweet une paire de tweets : un tweet issu du même label et un tweet issu d'un label différent,

1. https://tfhub.dev/google/universal-sentence-encoder-large/5
2. https://github.com/UKPLab/sentence-transformers

conformément aux méthodes d'entraînement classique des réseaux sociaux siamois. Chacun de ces deux tweets est choisi de manière aléatoire dans le jeu d'entraînement, selon les règles concernant les labels que nous venons d'énoncer.

4. Expérimentations et Résultats

Trois expériences ont été menées afin d'évaluer différents aspects. La première expérience nous permettra de valider quelle est la meilleure approche de regroupement des données entre FSD et EDF tandis que les expériences 2 et 3 nous permettront de déterminer quelles sont les meilleurs techniques de représentation du texte, selon les contextes définis.

Pour chacune des trois expériences, nous présentons d'abord le protocole expérimental qui a été mis en oeuvre puis nous présentons les résultats. Nous étudions les tests de significativité avec une valeur α = 0,05. Nous procédons à ces évaluations à l'aide du "Wilcoxon signed-rank test", méthode de test de significativité statistique la plus adaptée à notre contexte (Yeh, 2000). En effet, nous utilisons des méthodes de test non paramétriques du fait des caractérstiques de nos données.

4.1. Première expérience

4.1.1. Protocole expérimental

La première expérience est la comparaison des 4 modèles de représentation de texte, **TF-IDF dataset**, **TF-IDF all tweets**, **S-BERT nli** et **USE** dans deux contextes différents : avec application de l'algorithme FSD ou avec application de notre solution EDF. Nous utilisons l'implémentation de FSD proposée par (Mazoyer *et al.*, 2020) [3], en adaptant cette solution. En effet, contrairement à l'implémentation proposée, nous appliquons l'algorithme à des fenêtres successives de 2000 tweets et utilisons comme mesure de performance la mesure B-Cubed, une adaptation de la précision, du rappel et de la F-mesure pour le partitionnement de données, reconnue comme étant la méthode d'évaluation des performances de partitionnement la plus complète (Amigó *et al.*, 2009). Ainsi, nous formulons l'hypotèse H0 suivante : "Il n'y a pas de différence statistiquement significative entre les performances des algorithmes dans le cadre du FSD et de EDF". Pour la valider, nous utilisons le test "Wilcoxon signed-rank test". Concernant les valeurs seuils de l'algorithme FSD proposé, nous avons utilisé les valeurs présentées par (Mazoyer *et al.*, 2020), c'est à dire t=0.65 pour TF-IDF dataset, t=0.75 pour TD-IDF all tweets et t=0,39 pour S-BERT et t=0,22 pour USE. Les valeurs de seuils utilisées pour la suppression des arêtes du graphe dans l'approche EDF sont les suivantes : t=0,39 pour les modèles basés sur TF-IDF, t=0,79 pour le S-BERT, t=0,59 pour USE. Pour rappel, les valeurs correspondent à des similarités calculées via Cosine Similarity. Ces valeurs seuils ont été déterminées empiriquement.

3. https://github.com/ina-foss/twembeddings

4.1.2. Résultats

Le *Tableau 1* résume les résultats. Les chiffres présentés sont les moyennes sur l'ensemble des fenêtres de chaque métrique ainsi que l'écart type. Dans la majorité des cas, l'approche EDF est plus performante que l'approche FSD. Les résultats des tests de significativité sont présentés dans le *Tableau 2*. Le test est réalisé entre les valeurs de chaque métrique, pour chaque méthode, pour toutes les fenêtres de tweets. Dans chacun des cas, on constate que la p-value est toujours inférieure α.

Tableau 1. Qualité des clusters créés selon la métrique B-cubed, pour chacune des représentations textuelles, en fonction de l'algorithme de partitionnement. L'approche EDF a quasi-systématiquement les meilleurs résultats.

Modèle	Approche	Précision	Rappel	F1 Score
TF-IDF dataset	FSD	0.727 $\pm$0.128	0.523 $\pm$0.184	0.573 $\pm$0.150
	EDF	**0.930** $\pm$0.048	**0.702** $\pm$0.276	**0.756** $\pm$0.240
TF-IDF all tweets	FSD	0.781 $\pm$0.107	0.552 $\pm$0.199	0.613 $\pm$0.161
	EDF	**0.929** $\pm$0.039	**0.751** $\pm$0.272	**0.805** $\pm$0.245
USE	FSD	**0.919** $\pm$0.001	0.379 $\pm$0.01	0.500 $\pm$0.01
	EDF	0.918 $\pm$0.01	**0.664** $\pm$0.01	**0.729** $\pm$0.01
S-BERT-nli	FSD	**0.968** $\pm$0.023	0.323 $\pm$0.159	0.460 $\pm$0.195
	EDF	0.880 $\pm$0.075	**0.611** $\pm$0.244	**0.680** $\pm$0.207

Tableau 2. P-value pour le test Wilcoxon signed-rank "FSD vs EDF". Dans chacun des cas, P-value<α.

	Précision	Rappel	F1 Score
TF-IDF dataset	2.47 e-07	1.14 e-06	8.21e-05
TF-IDF all tweets	2.47e-07	1.31e-07	2.21e-05
S-BERT nli	3.65e-07	2.47e-07	2.47e-07

Ainsi, dans la majeure partie des cas, les performances sont plus élevées pour l'algorithme EDF que FSD, et les tests de significativité indiquent une pvalue < α. En rejetant donc l'hypothèse H0, ceci nous permet de conclure que l'approche EDF a de meilleures performances dans notre contexte que l'algorithme FSD.

4.2. Seconde expérience

4.2.1. Protocole Expérimental

La seconde expérience consiste à comparer les performances de **TF-IDF dataset**, **TF-IDF all tweets**, **S-BERT nli** et **USE** dans le contexte EDF. Cette expérience nous sert à comparer les approches de représentation du texte entre elles, de manière à déterminer quelle solution de représentation des données textuelles est la plus efficace, en particulier, nous souhaitons examiner les performances des approches basées sur les Transformers et les comparer aux performances des modèles basés sur TF-IDF,

reconnus comme étant plus performants, notamment dans (Mazoyer *et al*., 2020). Les performances sont évaluées à l'aide de la métrique B-cubed. Nous formulons l'hypothèse H0 suivante : "Aucune des approches n'est significativement meilleure que les autres". Les valeurs de seuils utilisées pour cette expérience sont les mêmes que précédemment, c'est-à-dire t=0,39 pour les modèles basés sur TF-IDF, t=0,79 pour S-BERT et t=0,59 pour USE.

4.2.2. Résultats

Nous comparons chacune des méthodes en les appliquant à chacune des fenêtres définies précédemment, dans un contexte non-supervisé puisqu'aucun des modèles n'a nécessité de label pour un entraînement. Les résultats obtenus sont ceux présentés dans *le Tableau 1*, sur les lignes correspondant à l'approche EDF. Les résultats des tests de significativité sont présentés dans le *Tableau 3*.

Tableau 3. P-value pour le test Wilcoxon signed-rank. Dans chacun des cas, P-value<α.

	Précision	Rappel	F1 Score
S-BERT nli / TF-IDF dataset	7.19e-06	1.13 e-05	6.16e-06
S-BERT nli / TF-IDF all tweets	3.99e-07	2.43e-04	2.05e-05
USE / TF-IDF dataset	1.75e-02	1.77e-05	8.84e-04
USE / TF-IDF all tweets	6.51e-05	8.21e-07	1.32e-03

Ainsi, les performances sont en moyenne meilleures pour les approches basées sur TF-IDF comparées à S-BERT, mais sont proches avec USE. Les tests de significativité ont une valeur p-value < 0,05. Nous pouvons donc rejeter l'hypothèse H0 qui a été formulée, et conclure que les approches TF-IDF sont effectivement plus performantes que l'approche S-BERT, dans un cas non supervisé comme présenté ici. Cependant, les résultats sont proches entre les approches TF-IDF et USE.

4.3. Troisième expérience

4.3.1. Protocole expérimental

La troisième expérience compare **TF-IDF dataset**, **TF-IDF all tweets**, **S-BERT fine-tuned** et **USE** dans le contexte EDF, sur le jeu de test. Les performances sont évaluées à l'aide de la métrique B-cubed. Nous formulons l'hypothèse H0 suivante : "Aucune des approches n'est significativement meilleure que les autres". Les valeurs de seuils utilisées pour cette expérience sont les mêmes que précédemment, c'est-à-dire t=0,39 pour les modèles basés sur TF-IDF, t=0,79 pour S-BERT, t=0,59 pour USE. Cette expérience est similaire à la précédente mais nous voulons cette fois-ci valider l'intérêt de l'affinage dans un contexte d'application à un flux de données. Afin d'être cohérent avec ce contexte, nous avons entraîné sur le modèle S-BERT sur le jeu d'entraînement, dont les détails sont présentés en partie 3.3. Les détails de la phase d'entraînement sont quant à eux présentés dans la partie 3.4. Nous n'avons pas

affiné le modèle USE, nous projetons de le faire dans des travaux futurs. Nous avons choisi de nous focaliser sur un dérivé de BERT, qui est le modèle le plus standard actuellement. Nous appliquons néanmoins le modèle USE non affiné au jeu de test à des fins de comparaison.

4.3.2. Résultats

Les résultats sont présentés dans le *Tableau 4*. Les résultats des tests de significativité sont présentés dans le *Tableau 5*.

Tableau 4. Qualité des clusters créés selon la métrique B-cubed, pour chacune des représentations textuelles, dans un contexte supervisé, sur le jeu de test.

	Précision	Rappel	F1 Score
TF-IDF dataset	0.904 $\pm$0.044	0.769 $\pm$0.216	0.805 $\pm$0.170
TF-IDF all tweets	**0.929** $\pm$0.035	0.750 $\pm$0.215	0.805 $\pm$0.184
S-BERT fine tuned	0.851 $\pm$0.067	0.837 $\pm$0.170	0.828 $\pm$0.106
USE	0.875 $\pm$0.061	**0.855** $\pm$0.211	**0.839** $\pm$0.158

Tableau 5. P-value pour le test Wilcoxon signed-rank. Tous les résultats ne sont pas significatifs, notamment les F1 Score pour S-BERT et TF-IDF.

	Précision	Rappel	F1 Score
S-BERT nli fine-tuned / TF-IDF dataset	8.39e-04	6.65e-03	*0.963*
S-BERT nli fine-tuned / TF-IDF all tweets	7.62e-05	7.62e-05	*0.889*
USE / TF-IDF dataset	1.49e-02	1.34e-02	6.38e-02
USE / TF-IDF all tweets	3.81e-04	4.57e-05	2.32e-02

Nous pouvons constater que les résultats ne sont pas significatifs concernant les F1 score de S-BERT par rapport aux approches TF-IDF. En revanche, les performances de USE sont significativement meilleures que les autres, notamment concernant le F1-score et le rappel. S-BERT est plus performant que les approches TF-IDF en termes de rappel mais pas en termes de précision.

4.4. Discussion générale des résultats obtenus

L'expérience 1 nous a permis de montrer que notre approche EDF est supérieure à l'approche FSD dans la majorité des cas présentés. Ce constat est particulièrement vrai pour la mesure du rappel. Concernant la précision, notamment pour les architectures Transformers, les valeurs entre FSD et EDF sont proches. Nous pensons que l'algorithme FSD permet, dans ces cas là, d'obtenir des partitionnements cohérents (forte précision). Dans un même temps, FSD a tendance à segmenter les documents d'un même label dans plusieurs partitionnements entraînant une chute importante du rappel. Cela est probablement dû au fait que les partitionnements avec FSD peuvent être créés à l'arrivée d'un nouveau document sans tenir compte de la totalité des documents de la fenêtre. Cette segmentation est moins présente dans la méthode EDF conduisant à une meilleure valeur de rappel.

Nous avons également montré que les approches basées sur des architectures Transformers, particulièrement USE et S-BERT affiné, sont compétitives par rapport aux approches classiques (TF-IDF). Il est quand même notable que, dans un contexte non supervisé, S-BERT a des performances moindres que USE. Nous pensons que cela peut être expliqué par les données utilisées pour pré-entraîner les différents modèles Transformers. En effet, le modèle S-BERT que nous avons utilisé est basé sur BERT NLI, qui est entraîné sur le corpus Wikipedia anglais, BookCorpus et affiné sur SNLI. USE quant à lui est entraîné sur un panel de données plus diversifié, incluant des données de forums de discussion ou de sites de questions-réponses, plus proche dans leur formulation (moins formel) des données de Twitter que ne l'est le jeu d'entraînement de S-BERT. De ce fait, les données issues des réseaux sociaux, dont la syntaxe est très particulière notamment du fait de la déstructuration de la langue utilisée (français, anglais...), posent des problèmes à S-BERT non affiné car entraîné sur des données écrites dans un anglais "plus conventionnel". Une fois le modèle S-BERT affiné sur des données issues de réseaux sociaux, les performances de S-BERT augmentent et deviennent comparables aux autres modèles. Ainsi, nous pouvons souligner l'importance de la phase d'affinage du modèle et l'intérêt que pourrait représenter un pré-entraînement de S-BERT directement sur des données issues des réseaux sociaux pour obtenir de meilleurs résultats dans notre contexte.

5. Conclusion

Nous avons étudié le problème de la détection d'évènements dans les flux de données textuelles sous la forme d'une tâche de partitionnement. Dans un premier temps, nous avons montré la supériorité de notre approche EDF basée sur du partitionnement par rapport à l'approche FSD dans le cadre de fenêtres de densité de tweets. Ensuite, nous avons montré que dans un contexte non-supervisé, S-BERT est en deça des autres modèles, mais que USE reste compétitif. Enfin, nous avons montré que dans un contexte supervisé, les approches TF-IDF ne peuvent pas être déclarées comme supérieures aux approches Transformers, ouvrant des perspectives intéressantes, comme des approches d'apprentissage actif ou incrémental, concernant de potentiels développement de ces approches dans le contexte des flux de données issus des réseaux sociaux. En effet, cela permettrait d'adapter les Transformers, potentiellement performants lorsque affinés, à des contextes où il est difficile d'avoir accès à des données labélisées. Afin de compléter le travail présenté dans ce papier, une comparaison de différentes approches de partitionnement peut être menée, à la manière de ce qui a été fait ici pour les représentations textuelles. Enfin, ce papier s'inscrit dans un projet de recherche mené en collaboration avec l'entreprise Scalian. L'objectif est de travailler sur une chaîne plus complète (Maître *et al*., 2020) de détection, de suivi et de caractérisation des évènements, à la manière de la méthode présentées dans (Fedoryszak *et al*., 2019), notamment pour faire un suivi inter-fenêtre des évènements et déterminer les principaux composants de ces évènements.

Bibliographie

Aggarwal C. C., Zhai C. (2012). A survey of text clustering algorithms. In *Mining text data*, p. 77–128. Springer.

Allahyari M., Pouriyeh S., Assefi M., Safaei S., Trippe E., Gutierrez J. *et al*. (2017, 07). A brief survey of text mining: Classification, clustering and extraction techniques.

Allan J. (2012). *Topic detection and tracking: event-based information organization* (vol. 12). Springer Science & Business Media.

Allan J., Lavrenko V., Malin D., Swan R. (2000, 11). Detections, bounds, and timelines: Umass and tdt-3. *Proceedings of Topic Detection and Tracking Workshop*.

Amigó E., Gonzalo J., Artiles J., Verdejo F. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, vol. 12, n° 4, p. 461–486.

Atefeh F., Khreich W. (2015, février). A survey of techniques for event detection in twitter. *Comput. Intell.*, vol. 31, n° 1, p. 132–164. Consulté sur https://doi.org/10.1111/coin.12017

Baeza-Yates R. A., Ribeiro-Neto B. A. (1999). *Modern information retrieval*. ACM Press / Addison-Wesley.

Becker H., Naaman M., Gravano L. (2011, 01). Beyond trending topics: Real-world event identification on twitter. In, vol. 11.

Blondel V. D., Guillaume J.-L., Lambiotte R., Lefebvre E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, vol. 2008, n° 10, p. P10008.

Bojanowski P., Grave E., Joulin A., Mikolov T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Boom C. D., Canneyt S. V., Demeester T., Dhoedt B. (2016). Representation learning for very short texts using weighted word embedding aggregation. *CoRR*, vol. abs/1607.00570.

Bowman S. R., Angeli G., Potts C., Manning C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 conference on empirical methods in natural language processing (emnlp)*. Association for Computational Linguistics.

Bromley J., Guyon I., LeCun Y., Säckinger E., Shah R. (1994). Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, p. 737–737.

Cer D., Yang Y., Kong S.-y., Hua N., Limtiaco N., John R. S. *et al*. (2018). Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Conneau A., Kiela D., Schwenk H., Barrault L., Bordes A. (2017, septembre). Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, p. 670–680. Copenhagen, Denmark, Association for Computational Linguistics.

Devlin J., Chang M.-W., Lee K., Toutanova K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Fedoryszak M., Frederick B., Rajaram V., Zhong C. (2019). Real-time event detection on social data streams. *CoRR*, vol. abs/1907.11229. Consulté sur http://arxiv.org/abs/1907.11229

Hasan M., Orgun M., Schwitter R. (2019, 5). Real-time event detection from the twitter data stream using the twitternews+ framework. *Information Processing and Management*, vol. 56, n° 3, p. 1146–1165.

Hasan M., Orgun M. A., Schwitter R. (2018, août). A survey on real-time event detection from the twitter data stream. *J. Inf. Sci.*, vol. 44, n° 4, p. 443–463.

Jones K. S. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*.

Kiros R., Zhu Y., Salakhutdinov R., Zemel R. S., Torralba A., Urtasun R. *et al*. (2015). Skip-thought vectors. *arXiv preprint arXiv:1506.06726*.

Maître E., Chemli Z., Chevalier M., Dousset B., Gitto J.-P., Teste O. (2020). Event detection and time series alignment to improve stock market forecasting. In *Joint conference of the information retrieval communities in europe (circle 2020)*, vol. 2621, p. 1–5.

Mazoyer B., Hervé N., Hudelot C., Cage J. (2020, janvier). Représentations lexicales pour la détection non supervisée d'événements dans un flux de tweets : étude sur des corpus français et anglais. In *Extraction et Gestion des connaissances, EGC 2020.*

McMinn A. J., Jose J. M. (2015). Real-time entity-based event detection for twitter. In J. Mothe *et al*. (Eds.), *Experimental ir meets multilinguality, multimodality, and interaction*, p. 65–77. Cham, Springer International Publishing.

McMinn A. J., Moshfeghi Y., Jose J. M. (2013). Building a large-scale corpus for evaluating event detection on twitter. In *Proceedings of the 22nd acm international conference on information & knowledge management*, p. 409–418.

Mikolov T., Chen K., Corrado G., Dean J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Pan S. J., Yang Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, n° 10, p. 1345-1359.

Petrović S., Osborne M., Lavrenko V. (2010). Streaming first story detection with application to twitter. In *Human language technologies: The 2010 annual conference of the north american chapter of the association for computational linguistics*, p. 181–189. USA, Association for Computational Linguistics.

Reimers N., Gurevych I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *CoRR*, vol. abs/1908.10084. Consulté sur http://arxiv.org/abs/1908.10084

Sakaki T., Okazaki M., Matsuo Y. (2010, 01). Earthquake shakes twitter users: Real-time event detection by social sensors. In, p. 851-860.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N. *et al*. (2017). Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Weng J., Lee B.-S. (2011). Event detection in twitter. In *Proceedings of the international aaai conference on web and social media*, vol. 5.

Yeh A. (2000). More accurate tests for the statistical significance of result differences. *arXiv preprint cs/0008005*.

Zubiaga A., Aker A., Bontcheva K., Liakata M., Procter R. (2018, février). Detection and resolution of rumours in social media: A survey. , vol. 51, n° 2. Consulté sur https://doi.org/10.1145/3161603