Triple-VAE: A Triple Variational Autoencoder to Represent Events in One-Class Event Detection

Marcos P. S. Gôlo[1], **Rafael G. Rossi**[2], **Ricardo M. Marcacini**[1]

[1]Institute of Mathematics and Computer Sciences - University of São Paulo (USP),
PO Box 668 – 13.560-970 – São Carlos – São Paulo – Brazil

[2]Federal University of Mato Grosso do Sul (UFMS),
PO Box 210 – 79620-080 – Três Lagoas – Mato Grosso do Sul – Brazil

marcosgolo@usp.br, rafael.g.rossi@ufms.br

ricardo.marcacini@icmc.usp.br

Abstract. *Events are phenomena that occur at a specific time and place. Its detection can bring benefits to society since it is possible to extract knowledge from these events. Event detection is a multimodal task since these events have textual, geographical, and temporal components. Most multimodal research in the literature uses the concatenation of the components to represent the events. These approaches use multi-class or binary learning to detect events of interest which intensifies the user's labeling effort, in which the user should label event classes even if there is no interest in detecting them. In this paper, we present the Triple-VAE approach that learns a unified representation from textual, spatial, and density modalities through a variational autoencoder, one of the state-of-the-art in representation learning. Our proposed Triple-VAE obtains suitable event representations for one-class classification, where users provide labels only for events of interest, thereby reducing the labeling effort. We carried out an experimental evaluation with ten real-world event datasets, four multimodal representation methods, and five evaluation metrics. Triple-VAE outperforms and presents a statistically significant difference considering the other three representation methods in all datasets. Therefore, Triple-VAE proved to be promising to represent the events in the one-class event detection scenario.*

1. Introduction

Nowadays, social networks and news portals share and publish on different events affecting our daily lives [Chen and Li 2020]. Social protests, pandemic effects, natural disasters, political and economic actions are examples of events that occur in a specific time and place [Deng et al. 2020]. Event analysis is the field that investigates how to organize and extract knowledge from large event databases [Radinsky and Horvitz 2013, Zhao 2021]. Such knowledge is useful for exploratory analysis tasks, building decision-making indicators, and improving machine learning models by providing new (extra) features on the world's external factors. A crucial step in event analysis is filtering which events are interesting for a given application, as thousands of events are published daily. Event classification methods usually carried out this step considering the textual information of an event, as well as its geographic information and other metadata [Setty and Hose 2018, Zhao 2021].

Recent event classification methods have some limitations [Zhou et al. 2020, Chen and Li 2020, Zhao 2021]. The first limitation is to propose event classification considering a multi-class scenario, a decision that makes the practical use of the model unfeasible. In this case, the event dataset's volume, diversity, and frequent update rate surpass the human capacity to label and maintain a training set. Some studies model the event classification as a binary problem [Zhou et al. 2020], in which the positive class identifies events of interest and the negative class defines non-relevant events. However, both classes require significant labeling of training examples. In this sense, the one-class learning paradigm is a promising alternative as it requires labeling only of events of interest [Alam et al. 2020].

A second limitation is the event representation model [Zhou et al. 2020]. Events are composed of textual information, geographic location, names of people, organizations, and other metadata. Traditional methods usually concatenate these different features into a single representation that is used to train a model. More recent methods explore representation learning, such as deep autoencoders, to extract a new latent space (embeddings) from the concatenated representation of features [Blandfort et al. 2019]. Although both strategies obtain competitive results, few studies evaluate the performance of these representation strategies for event analysis. We argue that different information from events represents different data modalities misused through concatenation strategies. Thus, our focus is to explore event representation as a multimodal representation learning task.

This paper presents an approach to learning multimodal representation for one-class classification of events. Our approach is called Triple-VAE and explores three main event modalities: textual information, geographic location, and topic metadata. First, we propose a multimodal variational autoencoder capable of learning a single representation from triple modalities. Unlike concatenation-based methods, our approach merges modalities more naturally, automatically learning the importance of modalities in the final representation. Second, we also argue that our approach is more appropriate for one-class classification since it learns a representation space that approximates events of interest in high-density regions — which significantly improves the event classification step. In short, our proposed approach has the following contributions:

- We naturally incorporate latitude and longitude data into the embedding space, along with textual and topic information. Previous works use geographic location only as extra features in the concatenated representation.
- We leverage pre-trained neural language models to represent events. In particular, we use the DistilBERT Multilingual model, which is trained in a large textual corpus and has some general-purpose knowledge. In practice, this is a strategy to carry out transfer learning from the pre-trained model for our multimodal representation learning.
- We explore event topic information as a visual modality, in which each topic represents high-density information in a given dimensional space. In fact, density information facilitates visual exploration of the spatial distribution of events, thus providing useful information about related events.

We carried out an experimental evaluation involving ten real-world event datasets. We compared our proposed approach with the other three multimodal strategies, from simple feature concatenation to representation learning strategies via autoencoders and variational autoencoders. Our proposal outperforms the other three strategies considering the precision, F_1-Score, accuracy, and area under the receiver operating characteristic curve metrics in all datasets. Furthermore, a statistical analysis of the results indicated that our approach is statistically different from the other three strategies in one-class event classification tasks.

2. Related Works

The proposal presented in [Zeppelzauer and Schopfhauser 2016] uses texts and images as modalities to perform event detection. Both modalities are unstructured and therefore need preprocessing. The authors preprocessed the text using the Bag-of-Words (BoW) and the dimensionality reduction technique Latent Dirichlet Allocation. The work uses the bag-of-visual-words for representation in the image modality. After representing the text and image, the authors explored both modalities through early and late fusions. Early fusion is made through the concatenation of the representations, while late fusion is made through additive and hierarchical late fusion. The approach uses a binary classifier to ignore non-relevant events, and then multi-class learning is applied to built classification models. The experimental evaluation shows that using two modalities improves the event classification task. It is worth point out that early fusion outperforms late fusion.

In [Kang and Kang 2017], the authors use multi-class learning to predict crime events defined by visual (neighborhood appearance), spatial and temporal information. First, the work uses a Convolutional Neural Network (CNN) to represent the images, and the spatial and temporal data are already structured. Then, a Deep Neural Network (DNN) is used to predict if the event is a crime. In the DNN, the authors use the early fusion with the concatenation operator and a softmax activation function in the output layer. Results show that the DNN outperforms the Kernel Density Estimation and Support Vector Machines (SVM) algorithms with a simple concatenation of modalities.

[Blandfort et al. 2019] explores the detection of gang violence events through Twitter. The authors use text and image modalities to represent the events. The work uses Linguistic features, word embeddings, and a CNN to learn text representations. Furthermore, the authors use the Faster R-CNN and global image features generated by the deep convolutional model Inception-v3 to represent the images. The work uses the early fusion considering the concatenation operator, and the late fusion is performed considering an ensemble of algorithms trained on each modality. The authors use the multi-class SVM learning algorithm. Results show that multimodal representations outperform unimodal ones. Early fusion presents high results than late fusion.

[Zhou et al. 2020] is a survey of multimodal event detection. The authors compare works that make event detection through multi-class or binary learning, clustering, and other tasks considering unimodal and multimodal representations. The work shows that multimodality can be promising with representation learning through artificial neural networks. Furthermore, the work concludes that concatenate the modalities can result in a representation with a high dimension which can negatively affect event detection. The survey also points that future work should involve information enrichment, i.e., news modalities to enrich the representation learned and the use of different state-of-the-art neural network architectures in representation learning.

We observed that existing multimodal studies for event detection (i) use multi-class learning, which generates more user's effort on labeling, and if a new class arises, the classifier will make wrong predictions since it was not trained on that event class; (ii) use binary learning, which generates less user's effort on labeling, and the chance of the user not labeling events of one of the classes is smaller in comparison with the multi-class learning. However, labeling uninteresting events requires knowing a wide range of classes so that the user cannot label enough examples; (iii) do not explore other early fusion operators such as addition, subtraction, multiplication, and average; (iv) use the event image as a modality and, consequently, its models only work on events with images (note that many events do not have associated images); and (v) shows that early fusion outperforms late fusion.

Given all these facts and gaps mentioned in this section and the future works presented in the survey [Zhou et al. 2020], we propose an event detection approach considering one-class learning (OCL) over a multimodal representation. OCL avoids multi-class, and binary learning limitations since the user labels only one class and classifies a new example as belonging to the class of interest or not. Also, we propose the generative model variational autoencoder (VAE) to learn a representation from a set of three modalities: the event text, geolocation, and density information. We considered a VAE because it is a powerful method to learn representations since it is one of the state-of-the-art in representation learning. Furthermore, we propose to use early fusion with different fusion operators. We present the details of the proposed approach in the next section.

3. Proposal: One-Class Multimodal Event Detection

According to [Zhou et al. 2020], an event is: *"A story related to some news topic comprising of patterns that occurred at some specific time and space"*. Based on this definition, we define an event ($\boldsymbol{e}$) as its text representation ($\boldsymbol{\gamma}$), its density information ($\boldsymbol{\lambda}$), its geolocation ($\boldsymbol{\iota}$), and its date (τ). Therefore, we formally define an event $\boldsymbol{e}_i$ by the quadruple:

$$\boldsymbol{e_i} = \{\boldsymbol{\gamma}_i, \boldsymbol{\lambda}_i, \boldsymbol{\iota}_i, \tau_i\} \tag{1}$$

In this multimodal scenario, we propose a pipeline to detect events through one-class learning via multimodal representation (Figure 1). The pipeline has six steps. The first step consists of collecting events with description, geolocation, and date. In the second step, we represent the event's text through a neural language model. In the third step, we use a modality based on density information generated from the text's representation. In the fourth step, a variational autoencoder learns a multimodal representation from the three modalities ($\boldsymbol{\gamma}, \boldsymbol{\lambda}$ and $\boldsymbol{\iota}$), considering events that occurred prior to date τ. We use events that occur after date τ to evaluate the event classification model. The fifth step consists in using one-class learning to learn a decision function. Finally, in the sixth step, we make the detection of the events of interest.

3.1. Event Geolocation

We obtain the event geolocation through the Latitude and Longitude coordinates. Latitude and Longitude refer to the position or geographic coordinates of a place on Earth. Latitude ranges from -90 to 90, in which -90 represents the south pole, 90 represents the north pole, and 0 represents the Earth's equator. Longitude ranges from -180 to 180, in which values $\in [-180, 0)$ represent places in the west, values $\in (0, 180]$ represent places in the

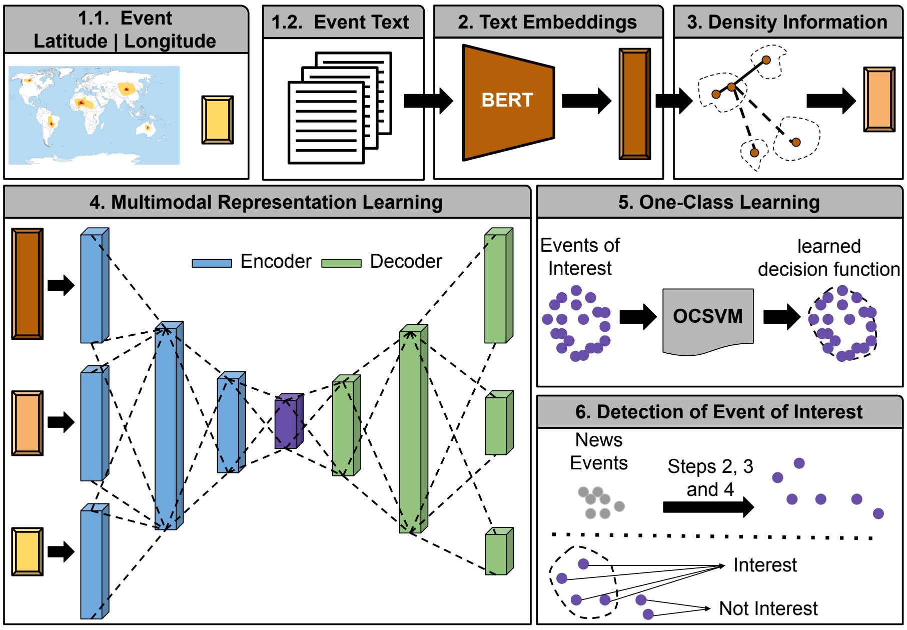

Figure 1. The Pipeline of multimodal representation Learning to detect events of interest through one-class learning.

east, and 0 represents the Greenwich meridian. Therefore, modality ι is a vector with two dimensions in which the first is the Latitude and the second is the Longitude.

3.2. Text Embeddings

One of the states-of-the-art to represent text is the context-dependent neural language model Bidirectional Encoder from Transformers (BERT) [Devlin et al. 2019]. It is noteworthy that this model obtains better results in natural language processing tasks than other models, such as based on word embeddings models or traditional models (e.g., BoW) [Otter et al. 2020]. Therefore, we use the Distilled version of BERT in its multilingual version (DBERTML) [Reimers and Gurevych 2020] to represent the event's text. First, we use the sentence-transformers library[1] to use the model DBERTML. Then, we make the preprocessing, providing the text to the pre-trained DBERTML model, and it returns an embedding γ_i with 512 real values. Details of the model DBERTML and its training parameters to obtain the embeddings are available in [Reimers and Gurevych 2020].

3.3. Density Information

We explore a modality based on density information. This modality is based on a one-class learning assumption, which assumes that high-density regions contain examples of the interest class [Krawczyk et al. 2014, Sharma et al. 2018]. To obtain the events density information, we apply a clustering algorithm and use a statistical technique that calculates the consistency within data clusters.

We use the silhouette coefficient [Rousseeuw 1987] to generate the density information ($\boldsymbol{\lambda}$). In this modality, the density representation $\boldsymbol{\lambda}_i$ of an event $\boldsymbol{e}_i$ is given by the concatenation of silhouette values computed considering a different number of clusters.

[1] https://www.sbert.net/

Thus, given u different clustering settings, $\boldsymbol{\lambda}_i = \{s_{i,1}, s_{i,2}, \dots, s_{i,u}\}$, in which $s_{i,j}$ is the silhouette of $\boldsymbol{\gamma}_i$ in j-th setting, and $s(\boldsymbol{\gamma}_i, k)$ is given by:

$$s(\boldsymbol{\gamma}_i, k) = \frac{\beta(\boldsymbol{\gamma}_i) - \alpha(\boldsymbol{\gamma}_i)}{\max(\alpha(\boldsymbol{\gamma}_i), \beta(\boldsymbol{\gamma}_i))} \tag{2}$$

in which k is the number of clusters, $2 \leq k < m$ and m is the number of events, $\alpha(\boldsymbol{\gamma}_i)$ is the average distance of $\boldsymbol{\gamma}_i$ to the centroid of its cluster, and $\beta(\boldsymbol{\gamma}_i)$ defines the average distance of $\boldsymbol{\gamma}_i$ to all $\boldsymbol{\gamma}$ of the closest cluster.

The event scenario has high-density regions representing well-defined topics of the events [Bide and Dhage 2021]. Furthermore, density information can be explored as a visual modality to analyze the spatial distribution of events. However, we highlight that the use of density information as a modality for events is still unexplored in literature.

3.4. Multimodal Representation Learning

After we have $\boldsymbol{\iota}$, $\boldsymbol{\gamma}$ and $\boldsymbol{\lambda}$ represented, we use a variational autoencoder with multimodal representation learning to learn a joint representation. An autoencoder (AE) is a neural network that learns data representations using two steps: encoding and decoding [Aggarwal 2018]. First, the encoder ($f()$) compresses $\boldsymbol{e}_i$ to a latent space $\boldsymbol{z}_{e_i}$. Then, the decoder ($g()$) reconstructs $\boldsymbol{e}_i$ from $\boldsymbol{z}_{e_i}$ in the output $\boldsymbol{r}_{e_i}$. Thus, the training of AE consists of making $\boldsymbol{r}_{e_i} \approx \boldsymbol{e}_i$. The encoder and decoder, and the AE optimization function are respectively given by:

$$autoencoder = \begin{cases} \mathbf{z}_{e_i} = f(\Phi; \mathbf{e}_i) \\ \mathbf{r}_{e_i} = g(\Theta; \mathbf{z}_{e_i}) \end{cases} \tag{3}$$

$$J(\Phi; \Theta) = \frac{1}{m} \sum_{e_i} \|\mathbf{e}_i - \mathbf{r}_{e_i}\|^2 \tag{4}$$

in which Φ is the weights and biases of neurons in the encoder, Θ is the weights and biases of neurons in the decoder. Thus, the AE is adequate in scenarios with examples belonging to one class because it trains in an unsupervised way.

There are variations of the AE that impose constraints on the hidden units [Aggarwal 2018]. For instance, the Variational Autoencoder (VAE) constraint that the activation in the hidden units should be drawn from the standard Gaussian with zero mean and unit variance [Xu and Durrett 2018]. This constraint also allows generating samples of the training data just feeding the decoder with samples generated from a normal distribution. Formally, the VAE assumes that a variable $\mathbf{z}_{e_i}$ generates the data $\mathbf{e}_i$ (Equation 5).

$$p(\mathbf{z}_{e_i}|\mathbf{e}_i) = \frac{p(\mathbf{e}_i|\mathbf{z}_{e_i})p(\mathbf{z}_{e_i})}{p(\mathbf{e}_i)} \tag{5}$$

$$p(\mathbf{e}_i) = \int p(\mathbf{e}_i|\mathbf{z}_{e_i})p(\mathbf{z}_{e_i})d\mathbf{z}_{e_i} \tag{6}$$

VAE approximates $p(\mathbf{z}_{e_i}|\mathbf{e}_i)$ to another treatable distribution $q(\mathbf{z}_{e_i}|\mathbf{e}_i)$ using the Kullback-Leibler (KL) divergence, which is responsible for measuring the divergence between two distributions. To optimize the marginal likelihood ($p(\mathbf{e}_i)$), you can use the log of the marginal likelihood [Xu and Durrett 2018]:

$$\log p_\Theta(\mathbf{e}_i) = KL(q_\Phi(\mathbf{z}_{e_i}|\mathbf{e}_i)||p_\Theta(\mathbf{z}_{e_i}|\mathbf{e}_i)) + \mathcal{L}(\Theta, \Phi; \mathbf{e}_i) \tag{7}$$

in which

$$\mathcal{L}(\Theta, \Phi, \mathbf{e}_i) = \mathbb{E}_{q_\Phi(\mathbf{z}_{e_i}|\mathbf{e}_i)} \log p_\Theta(\mathbf{e}_i|\mathbf{z}_{e_i}) - KL(q_\Phi(\mathbf{z}_{e_i}|\mathbf{e}_i)||p_\Theta(\mathbf{z}_{e_i})) \tag{8}$$

We implement a VAE using a neural network. Thus, it learns the encoder's $\varPhi$ parameters and the decoder's $\varTheta$ parameters through the weights of the neurons of the neural network layers. The first term of Equation 8 is related to the neural network reconstruction error. In the second term, we want to minimize the difference between the learned distribution $q_{\varPhi}(\mathbf{z}_{e_i}|\mathrm{e}_i)$ and $p_{\varTheta}(\mathbf{z}_{e_i})$ (prior knowledge). It is worth mentioning that literature studies replace the term $p_{\varTheta}(\mathbf{z}_{e_i})$ by a multivariate Gaussian distribution $\mathcal{N}(\mathbf{z}_{e_i}; 0, 1)$.

In this paper, we propose a Triple-VAE: a VAE that learns from three modalities. Therefore, the Triple-VAE has three inputs and three outputs. To learn a representation from three modalities, Triple-VAE combines them through early fusion. We opt to use the early fusion because of the advantage of using only one representation for the events in the classification step. Furthermore, to deal with the challenge of combine modalities with different dimensions, we use three dense layers with the same number of neurons that receive the inputs of Triple-VAE (Figure 1). The proposed architecture allows us to combine the modalities with different literature fusion operators.

Our proposed Triple-VAE aims to maximize Equation 9. Given an event $\boldsymbol{e}_i$, the first term calculates the reconstruction errors of $\boldsymbol{\gamma}_{e_i}, \boldsymbol{\lambda}_{e_i}, \boldsymbol{\iota}_{e_i}$. The second term wants to approximate the learned distribution $q_{\varPhi}(\boldsymbol{z}_{e_i}|\boldsymbol{\gamma}_{e_i}, \boldsymbol{\lambda}_{e_i}, \boldsymbol{\iota}_{e_i})$ from $p_{\varTheta}(\boldsymbol{z}_{e_i})$.

$$\begin{aligned}\mathcal{L}(\varTheta, \varPhi, \boldsymbol{\gamma}_{e_i}, \boldsymbol{\lambda}_{e_i}, \boldsymbol{\iota}_{e_i}) = \mathbb{E}_{q_{\varPhi}(\boldsymbol{z}_{e_i}||\boldsymbol{\gamma}_{e_i}, \boldsymbol{\lambda}_{e_i}, \boldsymbol{\iota}_{e_i})} \log p_{\varTheta}(|\boldsymbol{\gamma}_{e_i}, \boldsymbol{\lambda}_{e_i}, \boldsymbol{\iota}_{e_i}|\boldsymbol{z}_{e_i}) \\ -KL(q_{\varPhi}(\boldsymbol{z}_{e_i}|\boldsymbol{\gamma}_{e_i}, \boldsymbol{\lambda}_{e_i}, \boldsymbol{\iota}_{e_i})||p_{\varTheta}(\boldsymbol{z}_{e_i}))\end{aligned} \tag{9}$$

3.5. One-Class Learning to Detect Events of Interest

After we have a multimodal representation for events, we are able to classify them. We use one-class learning (OCL) [Tax 2001] to detect events of interest. In the OCL, the algorithms train using only examples of the class of interest. Thus, we do not suffer from new event categories or not knowing a non-relevant event category. Moreover, even if the user is interested in a single event category, the OCL is most appropriate since OCL does not label examples of classes that are not the class of interest [Alam et al. 2020]. Another advantage of OCL over multi-class or binary learning is (i) the user has less effort in data labeling; and (ii) it is more appropriate in unbalanced scenarios [Fernández et al. 2018].

Let the domain of events be $\mathcal{E}$, and the domain of labels be $\mathcal{Y}$, in which $y_i = \{+1, -1\}$ for $y_i \in \mathcal{Y}$ and +1 represents the label of the interest class, while -1 represents the label of the non-interest class. Then, given a set of m training events $\{(\mathrm{e}_j, y_j)\}_{j=1}^{m}$, in which $y_j = +1$, the goal of OCL is to learn a function $f : \mathcal{E} \rightarrow \mathcal{Y}$ given only labeled events from the interest class. After learning the function f, the classifier is able to predict y_i for a new event e_i comparing $f(\mathrm{e}_i)$ with a threshold as presented in Equation 10.

$$y_i = \begin{cases} +1 \textit{ (Interest)} & \text{if } f(\mathrm{e}_i) \leq threshold \\ -1 \textit{ (Non Interest)} & \text{otherwise} \end{cases} \tag{10}$$

Among the OCL algorithms [Gôlo et al. 2019], we chose the One-Class Support Vector Machine (OCSVM) [Tax and Duin 2004] since it is considered state-of-the-art in OCL [Alam et al. 2020]. The training of OCSVM consists of finding and hypersphere of minimum volume that involves the training events. The center of the hypersphere is defined in Equation 11 [Tax and Duin 2004]:

$$\mu_{(c)} = \underset{\mu \in U}{\arg\min} \max_{1 \leq i \leq m} ||\varphi(\boldsymbol{e}_i) - \mu||^2 \tag{11}$$

in which m is the number of events of interest, U is the feature space associated with the function kernel φ, $\mu_{(c)}$ is the center of the hypersphere. Since the goal is to obtain the hypersphere with minimum volume, we minimize the radius (r), i.e., r^2. Slack variables $(\varepsilon_i \geq 0)$ can also allow a trade-off between hypersphere volume and coverage of the training events. Then, the constraint that almost all training events are within the sphere is given by Equation 12. OCSVM aims to minimize Equation 13 subject to Equation 12.

$$\begin{gathered} \|\varphi(\boldsymbol{e}_i) - \mu_{(c)}\|^2 \leq r^2 + \varepsilon_{\boldsymbol{e}_i}, \\ \forall i = 1, ..., m, \ \varepsilon_{\boldsymbol{e}_i} \geq 0 \end{gathered} \tag{12}$$

$$\min_{\mu, \varphi, r} \quad r^2 + \frac{1}{m} \sum_{i=1}^{m} \frac{\varepsilon_{\boldsymbol{e}_i}}{\nu} \tag{13}$$

In which $\nu \in (0, 1]$ is a parameter to control the trade-off between the radius and the errors so that the hypersphere is not too large and the false positive rate increases. We will consider an event as belonging to the class of interest if its distance from the center is less than the radius r of the hypersphere, i.e., $f(\boldsymbol{e}_i) = dist(\varphi(\boldsymbol{e}_i), \mu_{(c)})$ and $threshold = r$.

4. Experimental Evaluation

We compared our proposed Triple-VAE with the other three literature multimodal representation methods in the experimental evaluation. We want to demonstrate that the representations generated by Triple-VAE outperform others usually explored in the literature for event detection. We use the OCSVM algorithm to compare all the events representation methods. The next subsections present the event collections, experimental settings, results, and discussion. All source codes that we use in the experimental evaluation are available online[2].

4.1. Event Collections and Experimental Settings

We obtain the event collections from the GDELT project, which monitors real-time events worldwide. Each dataset represents a theme and contains 6000 events. We populate the datasets by using the google cloud big query.

We compare our Triple-VAE with three multimodal strategies. The first consists of the concatenation of the modalities ($\boldsymbol{\gamma}$, $\boldsymbol{\lambda}$, and $\boldsymbol{\iota}$). The other two learn a representation using an AE and a VAE from the concatenated representation. The parameters from the three multimodal strategies, our proposal (Triple-VAE), and the OCSVM algorithm are:

- **Triple-VAE, VAE, AE and Concatenate**: we used the k-Means with the sets of k = $\{\{3, 6, 9, 12\}, \{2, 4, 6, 8, 10\}, \{3, 5, 7, 9, 11\}, \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}\}$;
- **Triple-VAE, VAE and AE**: learning rate = 0.001, optimization algorithm = $\{$Adam$\}$, linear activation function, dimensions of the dense layers = $\{(512, 384, 128, 384, 512), (512, 256, 64, 356, 512), (512, 64, 2, 64, 512)$ and $(512, 128, 512)\}$, tensorflow seed = 1, maximum number of epochs = $\{5, 10, 50\}$ and batch_size = 32;
- **Triple-VAE**: fusion operators = $\{$addition (add), subtraction (subtract), concatenation (concat), average and multiplication (multiply)$\}$;
- **OCSVM**: $kernels = \{rbf, linear, sigmoid, polynomial\}$, the kernel coefficients $degree$ =$\{2, 3, 4\}$ and $gamma = \{1/(na), 1/n\}$, in which n is the dimension of the input data and a is the variance of the representations, and $\nu = \{0.001, 0.01, 0.05 * h, h \in [1..18]\}$.

[2] https://github.com/GoloMarcos/TripleVAE-OCL-ENIAC2021.git

We used 2000 events with the oldest dates for the training set and the other 4000 for the test set for each event dataset. Also, we randomly selected 4000 events from different event datasets and added them to the test set in order to have counter-examples of the interest class during the evaluation process.

The classification performances were analyzed using the precision (Equation 15), recall (Equation 16), F_1-Score (Equation 14), accuracy (Equation 19), and Area Under Curve Receiver Operating Characteristic (AUC-ROC). F_1-Score is a harmonic average between precision and recall. AUC-ROC (Equation 18) computes the area under curve ROC. A ROC curve presents the relation between the tpr (equivalent to recall) and false positive rate (fpr) (Equation 17) at different threshold settings.

$$f1 = \frac{2 \cdot p \cdot r}{p + r} \quad (14) \qquad p = \frac{tp}{tp + fp} \quad (15) \qquad r = \frac{tp}{tp + fn} \quad (16) \qquad fpr = \frac{fp}{fp + tn} \quad (17)$$

$$auc\text{-}roc = \int_{\infty}^{-\infty} tpr(t) fpr'(t) \,\mathrm{d}t \quad (18) \qquad acc = \frac{tp + tn}{tp + tn + fp + fn} \quad (19)$$

In the equations presented above, tp (true positives) is the number of events of interest that the algorithm has correctly classified; tn (true negatives) is the number of non-interest events that the algorithm has correctly classified; fp (false positives) is the number of non-interest events that have been classified as interest; fn (false negatives) is the number of events of interest classified as non-interest; and t is a classification threshold.

4.2. Results and Discussion

Table 1 presents the best results in the ten event datasets and four event representation methods. The results consist in the highest F_1-Score (f1), accuracy (acc), and AUC-ROC (auc), among all representations method parameters and OCSVM parameters. The precision (p) and recall (r) values are the ones that generated the highest F_1-Score. Bold values indicate that the method obtained the highest value considering each metric.

Table 1. Results in ten event datasets considering the OCSVM algorithm and the metrics precision, recall, accuracy, auc-roc and F_1-Score.

	Concatenate					**Concatenate-Autoencoder**					**Concatenate-VAE**					**Triple-VAE**				
	p	**r**	**f1**	**auc**	**acc**	**p**	**r**	**f1**	**auc**	**acc**	**p**	**r**	**f1**	**auc**	**acc**	**p**	**r**	**f1**	**auc**	**acc**
Earthquake	0.53	**0.92**	0.67	0.63	0.64	0.55	0.85	0.67	0.67	0.65	0.56	0.86	0.68	0.69	0.66	**0.81**	0.84	**0.82**	**0.83**	**0.82**
Agriculture	0.52	**0.93**	0.67	0.59	0.57	0.62	0.88	0.73	0.75	0.68	0.63	0.87	0.73	0.74	0.68	**0.82**	0.90	**0.86**	**0.90**	**0.85**
Terrorism	0.54	**0.92**	0.68	0.65	0.63	0.54	0.91	0.68	0.68	0.64	0.54	0.91	0.68	0.68	0.64	**0.89**	0.89	**0.89**	**0.94**	**0.89**
Immigration	0.57	0.89	0.69	0.69	0.65	0.68	**0.93**	0.78	0.79	0.74	0.74	0.91	0.81	0.81	0.79	**0.83**	0.89	**0.86**	**0.92**	**0.86**
Racism	0.59	0.85	0.69	0.71	0.65	0.63	**0.94**	0.75	0.70	0.70	0.68	**0.94**	0.79	0.74	0.75	**0.91**	0.92	**0.91**	**0.96**	**0.91**
Inflation	0.55	0.89	0.68	0.62	0.61	0.52	0.90	0.66	0.61	0.58	0.51	**0.93**	0.66	0.60	0.58	**0.85**	0.88	**0.86**	**0.87**	**0.86**
Corruption	0.58	0.86	0.69	0.67	0.64	0.55	0.93	0.69	0.70	0.63	0.54	**0.95**	0.69	0.69	0.63	**0.86**	0.88	**0.87**	**0.89**	**0.87**
Covid	0.66	0.84	0.74	0.81	0.76	0.69	0.87	0.77	0.85	0.76	0.68	0.89	0.77	0.85	0.78	**0.95**	**0.94**	**0.94**	**0.98**	**0.94**
War	0.53	**0.95**	0.68	0.63	0.61	0.57	0.88	0.69	0.69	0.63	0.55	0.92	0.69	0.68	0.63	**0.67**	0.89	**0.77**	**0.76**	**0.74**
Tsunami	0.58	0.79	0.67	0.64	0.67	0.62	0.77	0.68	0.67	0.65	0.55	**0.89**	0.68	0.69	0.62	**0.76**	0.88	**0.81**	**0.84**	**0.81**

In general, considering the recall metric, the Concatenate and VAE methods get the highest values in four datasets, outperforming AE and Triple-VAE that get the highest values in two datasets and one dataset, respectively. However, considering precision, F_1-Score, accuracy, and ROC-AUC, Triple-VAE outperforms all other methods. Therefore, Triple-VAE achieves a better balance among false positives and false negatives, given by F_1-Score and ROC-AUC, and a better classification considering both interest and non-interest class, given by the accuracy.

We performed Friedman's statistical test with Nemenyi's post-test to compare the approaches considering all metric scenarios and datasets [Trawinski et al. 2012]. Figure 2 presents a critical difference diagram[3] generated through the results of the Friedman test with Nemenyi's post-test.

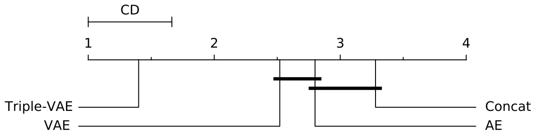

Figure 2. Critical difference diagram with the average rankings of the Friedman test with Nemenyi's post-test considering all metrics and datasets.

In addition to obtaining the highest results, Triple-VAE presented a statistically significant difference in relation to other methods. Furthermore, VAE also has a statistically significant difference considering the concatenation method. These results show that Triple-VAE was better than the other methods in learning highly non-linear relationships, redundancies, and dependencies between modalities. Thus, our proposal structures events with more representativeness of their modalities in relation to the other three methods.

Analyzing the representation parameters that provided the best results for each dataset, we highlight: (i) 5 and 10 epochs generate the higher results - 50 % of the datasets each one, indicating that Triple-VAE behaves better when it uses the lowest number of epochs, i.e., our model may be suffering overfitting when it uses a high number of epochs; (ii) each clustering set provides the highest results in at least two datasets, i.e., any one of the clustering sets can be an adequate choice; (iii) the concatenation operator in the neural network provided the best results for most of the cases. We highlight that the operator most used in the literature, i.e., concatenation, generates higher results in 50 % of the datasets. On the other hand, other operators, which are scarce in the multimodal learning literature for events, generate higher results in the other 50 %. Thus, the use of different operators can improve multimodal representation learning. Another interesting point is that the concatenation operator always gets higher results together with the architecture with fewer layers ($\{512, 128, 512\}$), while the other operators always get higher results together with the architectures with more layers.

5. Conclusions and Future Works

Event detection can be used to sense, analyze and comprehend important events that happen in our society. These social events have textual, geographical, and temporal components. Thus, multimodal representations have been investigated to represent the events since these components directly influence the detection of events.

[3] The diagram presents the methods' average rankings, and the methods connected by a line do not present statistically significant differences between them.

This paper proposes a multimodal method (Triple-VAE) to learning a representation from three modalities (text, density, and geolocation), with different early fusion operators (concatenate, add, subtract, and multiply), and using a variational autoencoder to learn a unified representation from those modalities. We also applied the OCSVM in order to perform OCL in the generated representations. The results obtained in our experimental evaluation show that Triple-VAE outperforms literature methods to represent events on the OCL scenario considering precision, F_1-Score, accuracy, and AUC-ROC. Our proposal also presented better results with statistically significant differences concerning all other techniques. It is noteworthy that the models built through OCL and considering the representations generated by Triple-VAE were able to differentiate events of interest and non-interest satisfactorily.

In future works, we intend to extend our Triple-VAE to handle incomplete modalities. We note that some events may have incomplete or inaccurate information regarding geographic information and other metadata. Thus, a multimodal representation learning method must be robust to these scenarios. We also intend to use semi-supervised OCL algorithms (Positive and Unlabeled Learning [Bekker and Davis 2020]) with the representations obtained by Triple-VAE.

References

Aggarwal, C. (2018). *Neural Networks and Deep Learning: A Textbook*. Springer.

Alam, S., Sonbhadra, S. K., Agarwal, S., and Nagabhushan, P. (2020). One-class support vector classifiers: A survey. *Knowledge-Based Systems*, 196:1–19.

Bekker, J. and Davis, J. (2020). Learning from positive and unlabeled data: a survey. *Machine Learning*, 1(Apr):1–45.

Bide, P. and Dhage, S. (2021). Similar event detection and event topic mining in social network platform. In *6th Int. Conf. for Convergence in Technology*, pages 1–11. IEEE.

Blandfort, P., Patton, D. U., Frey, W. R., Karaman, S., Bhargava, S., Lee, F.-T., Varia, S., Kedzie, C., Gaskell, M. B., Schifanella, R., et al. (2019). Multimodal social media analysis for gang violence prevention. In *Proc. of the Int. AAAI Conf. on web and social media*, volume 13, pages 114–124.

Chen, X. and Li, Q. (2020). Event modeling and mining: a long journey toward explainable events. *The VLDB Journal*, 29(1):459–482.

Deng, S., Rangwala, H., and Ning, Y. (2020). Dynamic knowledge graph based multi-event forecasting. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1585–1595.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL 2019: North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minnesota. ACL.

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., and Herrera, F. (2018). *Learning from imbalanced data sets*, volume 11. Springer.

Gôlo, M., Marcacini, R., and Rossi, R. (2019). An extensive empirical evaluation of preprocessing techniques and supervised one class learning algorithms for text classification. In *ENIAC 2019: Proc. of the XVI Encontro Nacional de Inteligência Artificial e Computacional.*, pages 262–273, Brazil. SBC.

Kang, H.-W. and Kang, H.-B. (2017). Prediction of crime occurrence from multi-modal data using deep learning. *PloS one*, 12(4):e0176244.

Krawczyk, B., Woźniak, M., and Cyganek, B. (2014). Clustering-based ensembles for one-class classification. *Information sciences*, 264:182–195.

Otter, D. W., Medina, J. R., and Kalita, J. K. (2020). A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):604–624.

Radinsky, K. and Horvitz, E. (2013). Mining the web to predict future events. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 255–264.

Reimers, N. and Gurevych, I. (2020). Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*.

Setty, V. and Hose, K. (2018). Event2vec: Neural embeddings for news events. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1013–1016.

Sharma, S., Somayaji, A., and Japkowicz, N. (2018). Learning over subconcepts: Strategies for 1-class classification. *Computational Intelligence*, 34(2):440–467.

Tax, D. M. and Duin, R. P. (2004). Support vector data description. *Machine Learning*.

Tax, D. M. J. (2001). *One-class classification: Concept learning in the absence of counter-examples*. PhD thesis, Technische Universiteit Delft.

Trawinski, B., Smetek, M., Telec, Z., and Lasota, T. (2012). Nonparametric statistical analysis for multiple comparison of machine learning regression algorithms. *Applied Mathematics and Computer Science*, 22(4):867–881.

Xu, J. and Durrett, G. (2018). Spherical latent spaces for stable variational autoencoders. In *EMNLP 2018: Proc. Conf. on Empirical Methods in Natural Language Processing*, pages 4503–4513, Belgium. Association for Computational Linguistics.

Zeppelzauer, M. and Schopfhauser, D. (2016). Multimodal classification of events in social media. *Image and Vision Computing*, 53:45–56.

Zhao, L. (2021). Event prediction in the big data era: A systematic survey. *ACM Computing Surveys (CSUR)*, 54(5):1–37.

Zhou, H., Yin, H., Zheng, H., and Li, Y. (2020). A survey on multi-modal social event detection. *Knowledge-Based Systems*, 195:105695.