VADE - Video Anomaly Detection with Embeddings

By: Thomas Foltz

Problem Definition and Dataset Curation

Video Anomaly Detection (VAD) is a critical task in computer vision, aiming to automatically identify unusual occurrences in video data. Traditionally, VAD approaches have relied heavily on visual features and complex deep learning architectures, which can be computationally expensive and challenging to deploy in real-time scenarios. This project leverages the power of multi-modal foundational models to perform VAD using solely text-based classification, potentially enabling more efficient real-time processing.

This approach utilizes foundational models to generate textual descriptions of video frames, which are then used to detect anomalies. Specifically, we employ the Llama-3.2-Vision-Instruct 11B parameter model to generate frame descriptions for the UCSD Ped2 and CUHK Avenue datasets. These text descriptions serve as a compact and semantically rich representation of the visual content, allowing for efficient encoding and classification of anomalies. By shifting the computational burden to the preprocessing stage and leveraging the advanced natural language understanding capabilities of LLMs, we aim to develop a VAD system that can operate in near-real-time with reduced computational requirements compared to traditional visual feature-based methods.

Word Embeddings, Algorithm, and Training

In our approach to video anomaly detection using text-based classification, we explore various word embedding techniques to effectively represent the textual descriptions of video frames. These embeddings play a crucial role in transforming the natural language descriptions into dense vector representations that can be processed by our machine learning models. This work tests two pre-trained word embedding models, Word2Vec (Google News 300) and GloVe (Wikipedia 2014 + Gigaword 5).

Word2Vec, developed by Google, is a widely used word embedding technique that learns vector representations of words based on their context in large text corpora. We utilize the pre-trained Word2Vec model trained on Google News, which provides 300-dimensional vectors for a vocabulary of 3 million words and phrases.

Global Vectors for Word Representation (GloVe), developed by Stanford NLP, is another popular word embedding technique. We employ the pre-trained GloVe model based on Wikipedia 2014 and Gigaword 5 datasets, which also provide 300-dimensional word vectors.

Both Word2Vec and GloVe embeddings are used to represent the textual descriptions of video frames in our system. The process involves the following steps:

- Tokenization: NLTK's word_tokenize function is used to split the frame descriptions into individual words.
- Word vector lookup: For each token, its corresponding 300-dimensional vector is retrieved from the pre-trained embedding model.
- Averaging: The mean of all word vectors is computed in a description to create a single 300-dimensional vector representing the entire frame description.

This video anomaly detection system employs a CNN-based model (VADE_CNN) for classification. The training process involves a few key components. On each of the datasets, the data is split into training and testing sets using an 80-20 ratio. Then for faster training and testing times, the model input is preprocessed by generating frame descriptions using the Llama-3.2 Vision-Instruct 11B parameter foundational model. The VADE_CNN binary classification model is designed to process the 300-dimensional frame description embeddings. It includes three lightweight 1D-CNN layers along with corresponding ReLU activation functions. Binary Cross-Entropy with logit loss is used during training, which combines a sigmoid layer and binary cross-entropy loss. Class weights are calculated to handle potential class imbalance in the dataset to support the learning process. The AdamW optimizer is used to extend upon the Adam optimizer by decoupling weight decay from the gradient updates. The learning rate, weight decay, and epoch amount are set as hyperparameters specified in the config file for each dataset. To ensure robust model performance, k-fold cross-validation with 5 folds is used. The early stopping patience is set to 3 epochs by default.

Results

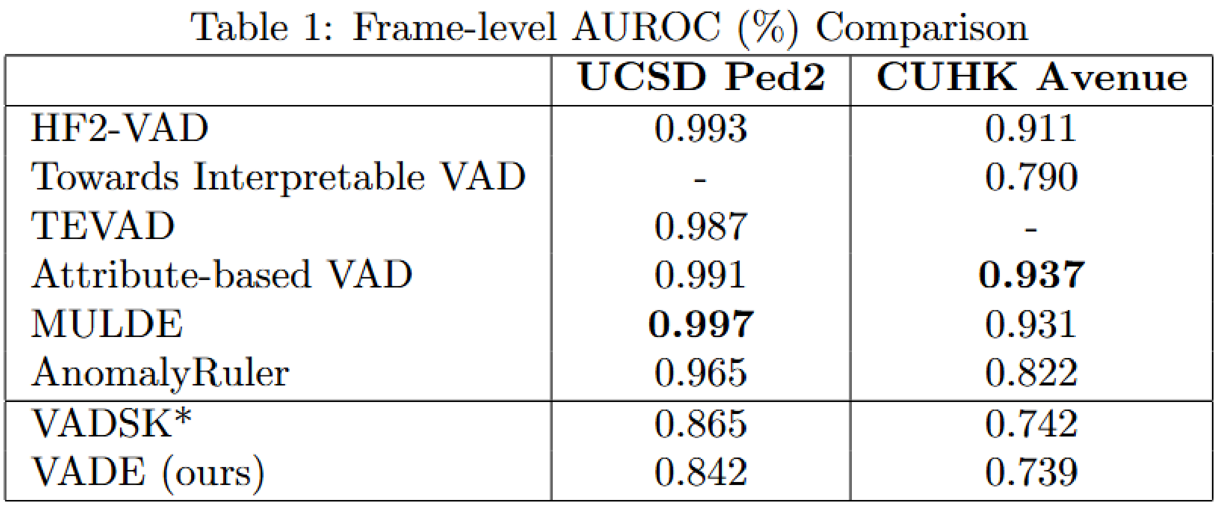


Table 1: Frame-level AUROC (%) Comparison

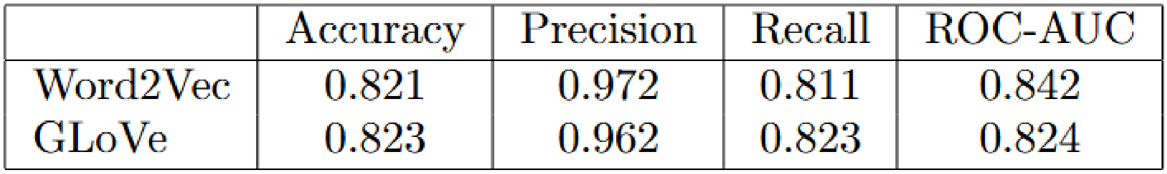
	UCSD Ped2	**CUHK Avenue**
HF2-VAD	0.993	0.911
Towards Interpretable VAD	-	0.790
TEVAD	0.987	-
Attribute-based VAD	0.991	**0.937**
MULDE	**0.997**	0.931
AnomalyRuler	0.965	0.822
VADSK*	0.865	0.742
VADE (ours)	0.842	0.739

The performance of VADE in Table 1 demonstrates promising results in the context of video anomaly detection, achieving ROC AUC scores of 0.842 and 0.739 on UCSD Ped2 and CUHK Avenue datasets, respectively. While these scores are lower than state-of-the-art methods like MULDE and Attribute-based VAD, VADE's performance is notable considering its significantly simplified pipeline that relies solely on foundational models and text embeddings for classification. The slight performance decrease compared to more complex approaches is a reasonable trade-off given VADE's ability to operate in near real-time with reduced computational requirements.

In-depth Analysis and Experiments

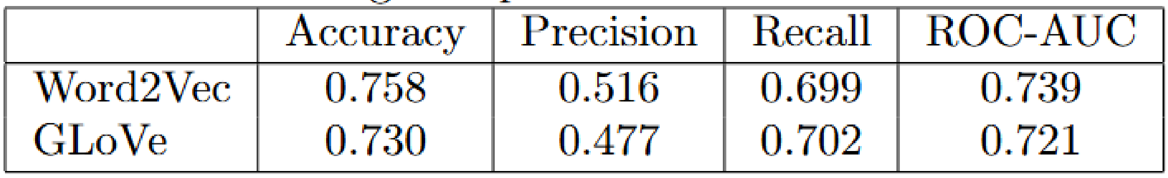
Interestingly, in Table 1, VADE's performance is comparable to its predecessor VADSK (0.865, 0.742), which also utilizes foundational models but employs a different architecture (MLP instead of CNN) and incorporates specific anomaly-indicative keywords with scoring mechanisms. Even though VADE has decreased interpretability to VADSK since we are encoding the text into embeddings, the marginal difference in performance between VADE and VADSK suggests that the CNN-based approach can effectively process text embeddings for anomaly detection, even without the explicit keyword extraction and anomaly scoring strategy from VADSK. This comparison highlights that while there may be a small sacrifice in accuracy compared to more complex visual feature-based methods, VADE maintains competitive performance while offering advantages in terms of computational efficiency and real-time processing capabilities.

Table 2: Embedding Comparison on UCSD Ped2 Dataset



	Accuracy	Precision	Recall	ROC-AUC
Word2Vec	0.821	0.972	0.811	0.842
GLoVe	0.823	0.962	0.823	0.824

Table 3: Embedding Comparison on CUHK Avenue Dataset



	Accuracy	Precision	Recall	ROC-AUC
Word2Vec	0.758	0.516	0.699	0.739
GLoVe	0.730	0.477	0.702	0.721

Tables 2 and 3 provide a detailed comparison between Word2Vec and GloVe embeddings across both datasets. On the UCSD Ped2 dataset, both embedding types showed strong performance with similar metrics. Word2Vec achieved slightly better results with an ROC-AUC of 0.842 compared to GloVe's 0.824. Word2Vec also demonstrated marginally higher precision, though GloVe showed better recall.

For the CUHK Avenue dataset, Word2Vec again performed slightly better with an ROC-AUC of 0.739 compared to GloVe's 0.721. The performance gap was more noticeable in precision metrics, where Word2Vec achieved 0.516 versus GloVe's 0.477. This consistent superior performance of Word2Vec might be attributed to its training on the Google News dataset, which likely contains more diverse and contextually rich vocabulary relevant to describing visual scenes and activities. The context-based learning approach of Word2Vec, which directly models word relationships based on their surrounding context, appears to be particularly effective for capturing the semantic nuances needed for anomaly detection in surveillance footage.

However, it's worth noting that the performance differences between the two embedding types are relatively small, suggesting that both pre-trained embedding models are capable of effectively capturing the semantic information needed for this task.

Lessons and Experience

Developing this video anomaly detection system yielded valuable insights into the application of text-based methods for classification tasks. Pre-trained word embeddings like GloVe and Word2Vec proved effective for representing frame descriptions, with the simple approach of averaging word vectors to create frame-level embeddings showing surprising power. While the choice between GloVe and Word2Vec had a modest impact on performance, both provided useful semantic representations.

Adapting text-based methods to video anomaly detection presented several challenges. Generating accurate and relevant frame descriptions using the Llama-3.2-Vision-Instruct model was crucial, requiring a delicate balance between description detail and computational efficiency. Additionally, handling class imbalance necessitated careful consideration of loss functions and class weights to ensure effective learning.

The project revealed interesting trade-offs between embedding complexity and model performance. The lightweight VADE_CNN model demonstrated that simple architectures can be effective for text-based video anomaly detection, while the use of 300-dimensional embeddings struck a good balance between feature richness and computational efficiency.
This approach showed both potential and limitations. It achieved comparable performance to visual feature-based methods on benchmark datasets, enabling real-time processing and improved interpretability. However, the lack of temporal context may limit the detection of complex anomalies spanning multiple frames.

Looking ahead, there are many ways to improve upon this work. These include incorporating temporal information to capture multi-frame anomalies, exploring fine-tuning language models specifically for VAD tasks, investigating more advanced text embedding techniques, and developing methods to further improve classification interpretability.