Video Text Matching: A Deep Learning Model for Video to Descriptive Text Matching

ANTHONY ANAZO, University of Twente, The Netherlands

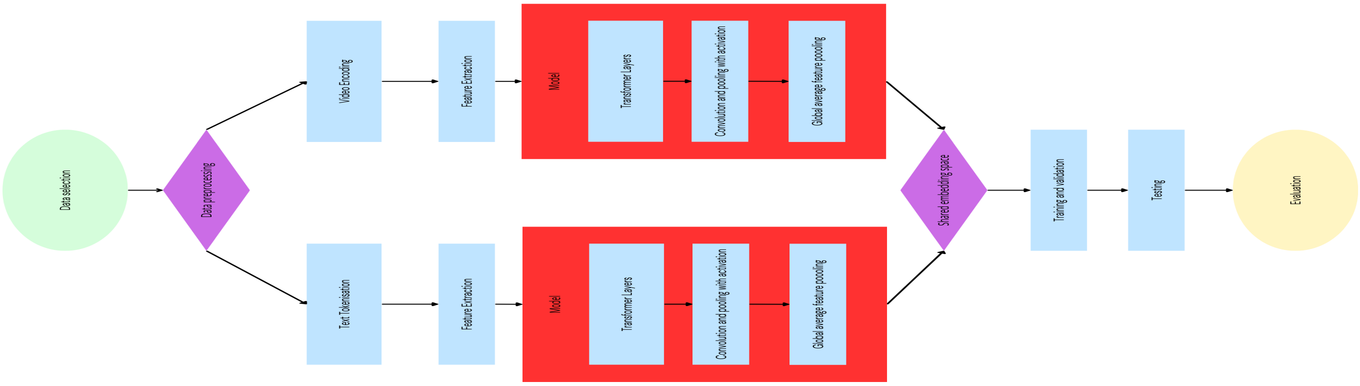


Fig. 1. Model pipeline. For processing, text is tokenised and videos are cropped to 224^2 without keeping the aspect ratio and encoded to normalised tensors. To extract features, text is put through a BERT model and video is put through an MViT model. These features are then put through the model giving an output of shape $B \times C$, where B is the batch size and C is the channel size which is 768. Since both modalities have the same size, there is a shared embedding space, allowing for the training and validation of the model.

The goal is to develop a functional multi-modal model that can retrieve short videos based on a text description provided by a user and also give a textual description based on a user-provided video. This will be done by processing textual descriptions and video clips and designing a feature space that will be shared for both text and video, thus enabling the matching of the two data types using contrastive learning. The model is trained and tested on the Microsoft Research Video Description Corpus (MSVD).

Additional Key Words and Phrases: Transformers, Contrastive Learning, Multi-modal model, Pytorch, MSVD, Video Text Matching.

1 INTRODUCTION

Attention [8] has revolutionised Natural Language Processing (NLP), Computer Vision (CV) and many more machine-learning fields. In CV, attention is performed on images. An example is the Vision Transformer (ViT) [4] and approaches have been further designed to address the spatio-temporal complexity of videos. This has resulted in a Video Vision Transformer (ViViT) that attends videos over both space and time jointly [1]. In addition, multiple variations of ViTs such as the Multiscale Vision Transformer (MViT) [5] have been made and contribute to improving the state of the art. Building upon these advancements, this research project aims to develop a multi-modal model capable of simultaneously processing videos and textual descriptions. The model will be trained using contrastive learning [9]. This training objective aims to make the model capable of distinguishing between positive and negative video text pairs, where positive pairs correspond to the same data source, while negative pairs do not. The development of this model aims to provide a functional approach, addressing video-to-text and text-to-video retrieval. With this context in mind, the main goal of this paper is to develop an efficient and functional multi-modal model for retrieving videos and relevant text. This will be achieved through pre-trained video and text encoders. The model will be trained and tested on the Microsoft Research Video Description (MSVD) Corpus[2].

2 RELATED WORK

2.1 Transformers

A transformer is a deep-learning architecture that encodes input to tokens, adds positional information to specify token positions, and uses attention patterns that highlight the relation between tokens [8]. Multi-head attention is running attention in parallel, where each attention head attends to a different context, capturing contextual relationships more effectively.

2.2 Vision Transformers

Vision transformers are an extension of transformers for images. Here images are split into multiple smaller non-overlapping patches and vectorised before being put into the transformer. Since these patches' position is unknown, positional encoding is added to the embedding to maintain spatial awareness [1].

2.3 Multi-Modal Contrastive Learning for Video-Text Understanding

Contrastive Language-Image Pre-Training (CLIP) is a joint image and text embedding model which uses the shared embedding space of both modalities to match images to text and vice versa. This is achieved through contrastive representation learning, which aims

TScIT 41, July 5, 2024, Enschede, The Netherlands

to learn an embedding space where positive pairs are closer to each other compared to negative pairs. Similar to CLIP, Multi-modal contrastive learning for video-text understanding is the training objective that uses the embedding space differences between positive and negative video-text pairs[9]. A pair is positive when the video and text correspond to the same data source, while a negative pair contains video and text that do not correspond to the same data source.

3 METHODOLOGY

To develop a functional multi-modal model, key challenges need to be addressed: processing text and video data effectively, utilising a contrastive loss, and ensuring efficient training and evaluation of the model. In sections 3.1 and 3.2 the methodologies used for text and video processing will be explained. Sections 3.3 and 3.4 describe the model architecture and contrastive learning objective, and sections 3.5 and 3.6 outline the training and evaluation process used to assess model performance.

3.1 Text Processing

The descriptive text $t = \{t^1, t^2, \ldots, t^i\}$ will be attended using a Bidirectional Encoder Representations from Transformers (BERT) [3]. The resulting output of this transformer will be $p_t = \{p_t^1, p_t^2, \ldots, p_t^i\}$, where each vector will contain contextualised information.

3.2 Video Processing

To process videos of size $T \times H \times W$, where T is the duration, H is the height, and W is the width of the video. Each video v will be split into non-overlapping patches $v = \{v^1, v^2, ..\}$. The output of this transformer will produce an output $p_v = \{p_v^1, p_v^2, \ldots, p_v^i\}$.

3.3 Model architecture

The Multi-Modal model M consists of two modules, one for video M_v and the other for text M_t. Each module contains 5 transformer layers, two convolution and pooling blocks with GeLU activation and a global average pooling block. The modules M_v and M_t have inputs p_v and p_t and give outputs z_v and z_t, which have dimensionality $B \times C$. In addition, z_v and z_t have the same shape, enabling the shared embedding space for both modalities.

3.4 Contrastive Learning

The training objective of the model is based on contrastive learning to differentiate positive and negative video text pairs. Using Noise-Contrastive Estimation (NCE) loss [6], Two losses $NCE(z_v, z_t)$ and $NCE(z_t, z_v)$, will be calculated for video-to-text similarity and text-to-video similarity of which the former is given by Equation 1.

$$\mathrm{NCE}(z_v, z_t) = \frac{\exp(z_v \cdot z_t^+/\tau)}{\sum_{z \in \{z_t^+, z_t^-\}} \exp(z_v \cdot z/\tau)} \tag{1}$$

Here, τ is a temperature hyperparameter. z_v^+ and z_t^+ are positive embedded video and text. z_t^- are negative embedded text that is z_t^+ but flipped. The contrastive loss of $NCE(z_t, z_v)$ is defined symmetrically. Combining both of these losses gives us our loss function which is given by Equation 2

$$\mathcal{L} = \mathrm{NCE}(z_v, z_t) + \mathrm{NCE}(z_t, z_v) \tag{2}$$

3.5 Training

Algorithm 1 Tensor Flipping Training

```
Require: $V$ (video-text set), $M$ (model), $num\_epochs$, $batch\_size$
for $epoch = 1$ to $num\_epochs$ do
    for each batch $B$ in $V$ do
        $V^+ \leftarrow$ Positive samples from batch $B$
        $O^+ \leftarrow M(V^+)$                              ▷ $O = (z_v^+, z_t^+)$
        $O^- \leftarrow$ Generate negatives by flipping $O^+$
        Calculate $\mathrm{NCE}(z_v, z_t)$
        Calculate $\mathrm{NCE}(z_t, z_v)$
        $\mathcal{L} = \mathrm{NCE}(z_v, z_t) + \mathrm{NCE}(z_t, z_v)$
        Update model $M$ parameters based on the loss $\mathcal{L}$
    end for
end for
```

As shown in Algorithm 1, a training epoch for the model involves flipping each video text set to create negatives used to calculate both loss functions. These loss functions are used to train the model as they align positive video-text pairs while distinguishing them from negative pairs.

3.6 Evaluation

The model will be evaluated using two metrics: the model's precision, and the Recall at K (R@K) for k=1 and k=5. These metrics are measured for each video text combination in the test split. The precision will show how capable the model is in finding the ground truth text representation for each video. If the positive text pair of the video is in the R@5, this would mean it was not far off; if it is not in the R@5, it would be safe to assume the model could not predict the correct positive pair. Besides these metrics, there will also be two heatmaps that visualise the L2 distance between each video text combination. The first indicates the matrix with a red diagonal indicating the positive pair, while the second in addition, highlights the R@5 scores per video.

4 EXPERIMENTS

4.1 Pre-Training

For pre-training, I stream the text through the $\mathrm{BERT}_{Base-uncased}$[3] model which is done through the use of the huggingface BertTokenizer and BertModel [1]. Subsequently for video pre-training, I put the videos through an MViT model [5]. The MVit model used is pytorch mvit_v1_b [2].

4.2 Implementation Details

4.2.1 Video Processing. From the videos, 16 frames are uniformly sampled over a fixed interval. Without keeping the aspect ratio, these frames are cropped to 224^2 and converted to a normalised

[1] Both the BertTokenizer and the BertModel can be found at https://huggingface.co/docs/transformers/model_doc/bert

[2] https://pytorch.org/vision/main/models/generated/torchvision.models.video.mvit_v1_b.html#torchvision.models.video.mvit_v1_b

tensor of size 16 × 3 × 224 × 224 and dimensions $T \times C \times H \times W$. Here T is the duration, C is the amount of colour channels, H is the height, and W is the width of the video.

4.2.2 Streamlining of Data. I Utilise a DataFrame[3], containing the paths to the pickle files, the textual descriptions and video title. This is used in combination with a PyTorch custom dataset and dataloader [4] to facilitate the streamlining of the data for feature extraction of both modalities as well as the training of the model.

4.2.3 Training details. The models are trained NVIDIA T4 GPUs (With 16GB RAM) provided by Google Colab [5]. The optimiser used during training is Adam [7], with a learning rate of $2e - 9$. And the temperature of the loss function is set to $\tau = 0.003$. The batch size for these models is 32 and the loss function $\mathcal{L}$ (Equation 2) of this model is calculated by calculating the NCE for each model with the detached output from the other model.

4.3 (Main) Results

The evaluation of the model is performed as discussed in section 3.5.

4.3.1 Precision. The precision of the model is quite low as it is approximately 0.0045. This means the model correctly predicted three video text pairs out of 670 video text pairs.

4.3.2 Heatmap. Figure 2 is a heatmap indicating the Euclidean distance of each video text combination in the test split after both have been put through the multi-modal model. The diagonal of the heatmap is the positive video text pair. Darker shades in the heatmap indicate smaller distances while lighter shades indicate bigger distances between video and text. The distances for all video-text pairs lie in a consistent range as indicated by the colour distribution across the heatmap. The distances are similar which can also be seen when looking at the mean and standard deviation which are approximately 6.03 and 0.078 respectively. The low deviation implies that the model thinks that most video text pairs are similar. Moreover, a pattern can be observed in figure 3 where the top 5 predictions of the model per video in the heatmap are black. These top 5 predictions create vertical lines in the heatmap, indicating that each video ranks the same top 5 texts.

Table 1. Recall Rates for MSVD

Dataset	R@1	R@5
MSVD	0.0045	0.119

4.3.3 Recall. In table 1 we see the performance of the model on MSVD [2]which indicates that the model rarely ranks the correct result at the very top. In addition, the correct result is seldom found in the top 5. The R@5 on a heatmap of all video text combinations can be found in figure 3.

4.3.4 True and False Positive Comparison. figure 4 displays four images, two correct and incorrect predictions of the model. The false predictions had the highest correspondence out of all predictions which signifies that the model thought these text and video combinations represented each other best out of the entire test split. The annotations of both negatives describe the same phenomenon, strengthening the assumption that there is convergence to a limited set of texts for each video. Despite the low precision, there were still enough true predictions to showcase how true and false predictions should be envisioned.

4.4 Ablations

Table 2. Mean of L2 distances between positive and negative pairs per sampling method

Ablation	Positive Value	Negative Value
Negative flipping sampling	0.0481	0.0490
Smallest distance sampling	0.0500	0.0499
Random top 4 sampling	0.5450	0.5450

In section 4.3.1 the precision is low, meaning the training pipeline is not implemented correctly. Table 2 summarises the results of two ablations. In these ablations, the sampling of negatives is modified. The results which are L2 distances between positive and negative pairs over the entire test split, show that with different methods of negative retrieval, the model can still not differentiate between positives and negatives. This indicates that the current training pipeline is not effective in differentiating between positive and negative pairs. In addition, the method of negative sampling does not affect the difference between positive and negative pairs, which means that the cause is likely another component of the training process.

Table 3. Text Predictions Counted from All Videos. Text 567 is predicted for most videos

Text ID	Count
Text_567	583
Text_204	54
Text_305	14
Text_618	3
Text_202	7
Text_122	9
Total	**670**

Table 3 contains the counted text predictions for all videos. This shows that the model predicts text 567 for most videos. Text 567 has a z-score of 1.64 when looking at the number of textual descriptions each video in the test split has, meaning the test split is not imbalanced.

[3] I use pandas: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html
[4] https://pytorch.org/tutorials/beginner/basics/data_tutorial.html
[5] https://colab.google/

5 DISCUSSION

The low precision and recall scores indicate that the model has a very low performance. The research goal was to create a functional multi-modal model. The results indicate that the task of retrieving relevant text pieces is difficult. The model might not be able to capture the complexity of the data resulting in poor performance. This performance indicates that parts of the training pipeline are not properly implemented. The part of the implementation that I suspect is incorrectly implemented is the loss function. Despite the loss decreasing and not over- or underfitting regarding the validation split during training, the distances between the positive and negative pairs are small or insignificant as seen in table 2.

6 CONCLUSION

I have presented an approach to developing a multi-modal contrastive learning model for video-text matching. This model uses a training objective that contrasts positives from negatives, allowing for video text matching. The creation of this model was done by first finding and processing the MSVD [2] corpus, encoding the videos, and storing the video-text tuples in a pandas Dataframe. After this initial step, I performed further preprocessing of the MSVD dataset, which included the grabbing of initial weights of videos as well as text through the implementation of a hook. This was followed by the creation of the model and the configuration of its optimization function. The results show that the model has poor performance. The model cannot generalise the features it was supposed to extract and is incapable of discriminating between video text pairs. In conclusion, the functional multi-modal model has poor performance and drastic improvements have to be made.

REFERENCES

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. 2021. ViViT: A Video Vision Transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, Montreal, QC, Canada, 6816–6826. https://doi.org/10.1109/ICCV48922.2021.00676

[2] David Chen and William Dolan. 2011. Collecting Highly Parallel Data for Paraphrase Evaluation. 190–200.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. http://arxiv.org/abs/1810.04805 arXiv:1810.04805 [cs].

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. https://doi.org/10.48550/arXiv.2010.11929 arXiv:2010.11929 [cs].

[5] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. 2021. Multiscale Vision Transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, Montreal, QC, Canada, 6804–6815. https://doi.org/10.1109/ICCV48922.2021.00675

[6] Michael U. Gutmann and Aapo Hyvärinen. 2012. Noise-Contrastive Estimation of Unnormalized Statistical Models, with Applications to Natural Image Statistics. *Journal of Machine Learning Research* 13, 11 (2012), 307–361. http://jmlr.org/papers/v13/gutmann12a.html

[7] Diederik P. Kingma and Jimmy Ba. 2017. Adam: A Method for Stochastic Optimization. https://doi.org/10.48550/arXiv.1412.6980 arXiv:1412.6980 [cs].

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. https://doi.org/10.48550/arXiv.1706.03762 arXiv:1706.03762 [cs].

[9] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. 2021. VideoCLIP: Contrastive Pre-training for Zero-shot Video-Text Understanding. http://arxiv.org/abs/2109.14084 arXiv:2109.14084 [cs].

A APPENDICES

During the preparation of this work, I used two AI tools, ChatGPT and Github Copilot.I used ChatGPT for generating LaTeXcode to achieve proper formatting. Subsequently, I used GitHub Copilot to accelerate the development of the model. The predictive capabilities of the tool offered me code suggestions which in turn I evaluated and modified to the point where they align with the requirements of this research project. After using both tools, I reviewed and edited the content as needed and take full responsibility for the content of the work.

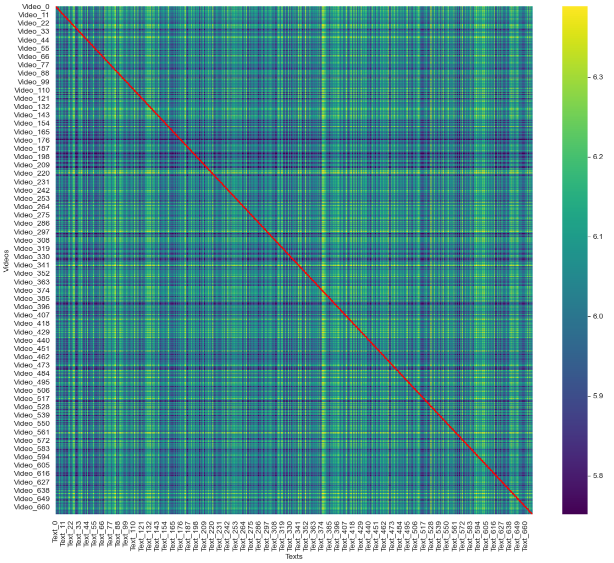


Fig. 2. Distance Matrix (Videos vs Text)

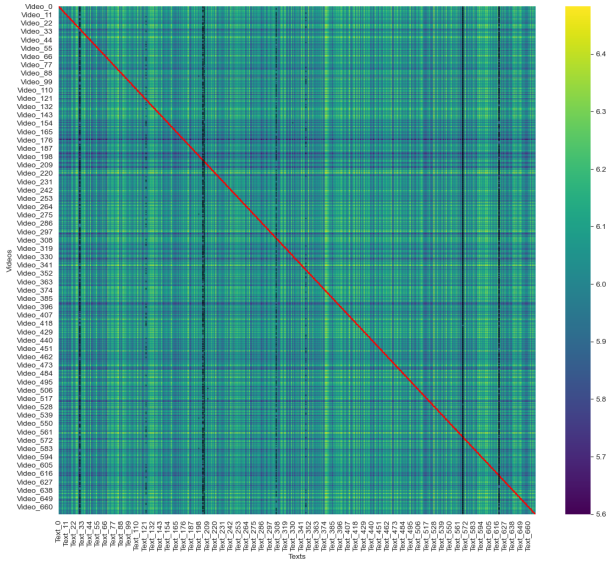
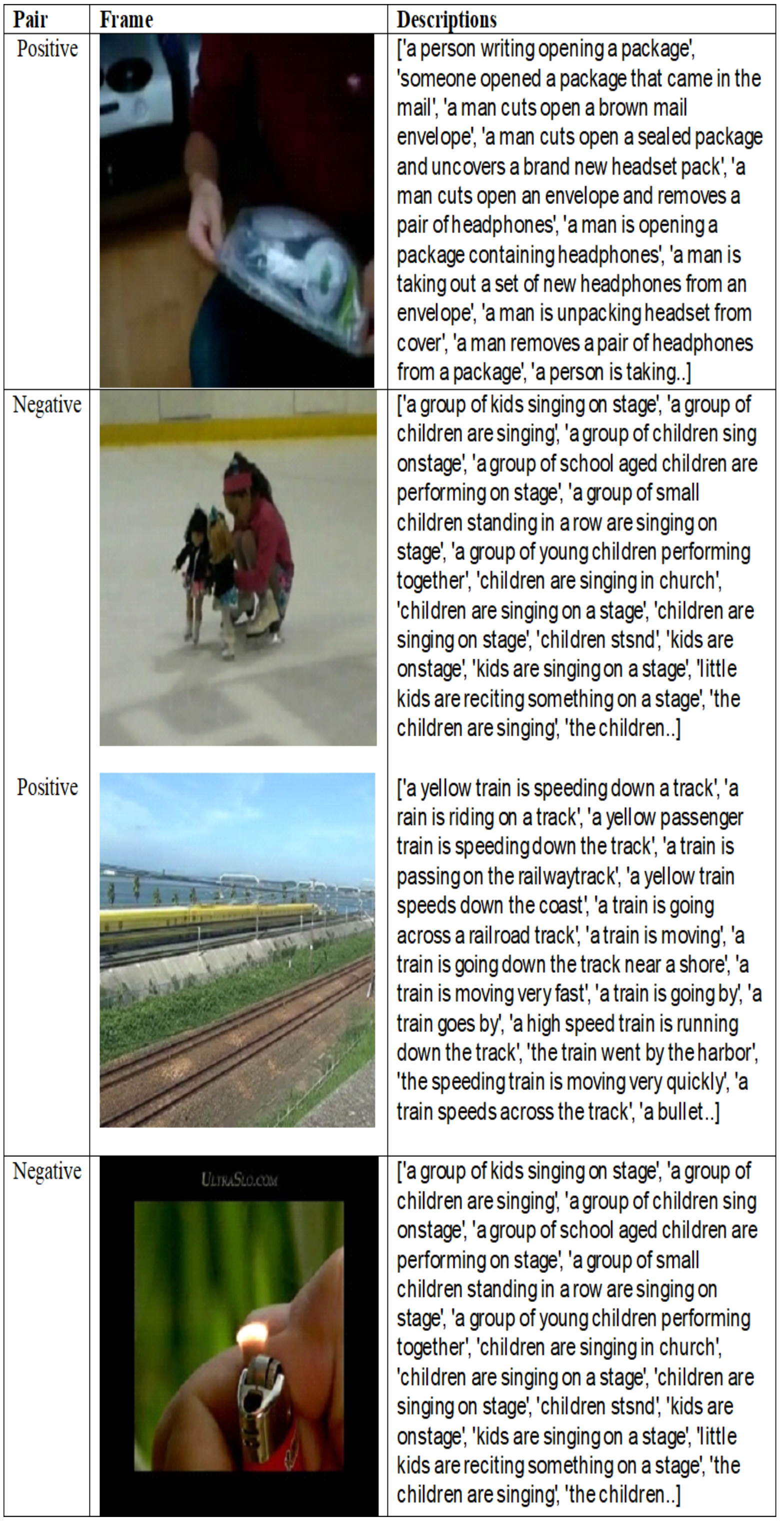


Fig. 3. Distance Matrix (Videos vs. Texts) with Top 5 Lowest Distances marked in black



Pair	Frame	Descriptions
Positive		['a person writing opening a package', 'someone opened a package that came in the mail', 'a man cuts open a brown mail envelope', 'a man cuts open a sealed package and uncovers a brand new headset pack', 'a man cuts open an envelope and removes a pair of headphones', 'a man is opening a package containing headphones', 'a man is taking out a set of new headphones from an envelope', 'a man is unpacking headset from cover', 'a man removes a pair of headphones from a package', 'a person is taking..]
Negative		['a group of kids singing on stage', 'a group of children are singing', 'a group of children sing onstage', 'a group of school aged children are performing on stage', 'a group of small children standing in a row are singing on stage', 'a group of young children performing together', 'children are singing in church', 'children are singing on a stage', 'children are singing on stage', 'children stsnd', 'kids are onstage', 'kids are singing on a stage', 'little kids are reciting something on a stage', 'the children are singing', 'the children..]
Positive		['a yellow train is speeding down a track', 'a rain is riding on a track', 'a yellow passenger train is speeding down the track', 'a train is passing on the railwaytrack', 'a yellow train speeds down the coast', 'a train is going across a railroad track', 'a train is moving', 'a train is going down the track near a shore', 'a train is moving very fast', 'a train is going by', 'a train goes by', 'a high speed train is running down the track', 'the train went by the harbor', 'the speeding train is moving very quickly', 'a train speeds across the track', 'a bullet..]
Negative		['a group of kids singing on stage', 'a group of children are singing', 'a group of children sing onstage', 'a group of school aged children are performing on stage', 'a group of small children standing in a row are singing on stage', 'a group of young children performing together', 'children are singing in church', 'children are singing on a stage', 'children are singing on stage', 'children stsnd', 'kids are onstage', 'kids are singing on a stage', 'little kids are reciting something on a stage', 'the children are singing', 'the children..]

Fig. 4. Four images with part of their predicted annotations: Two positive and two negative pairs.