[**GitHub URL Repository**](https://github.com/WaliSiddiqui1/BPM)

**Paper’s Github:** [**https://github.com/jackwu502/NSVA/tree/main**](https://github.com/jackwu502/NSVA/tree/main)

**BPM (Bron Performance Metrics)**

**Meeting w/ TA Johnny:**

For the meeting with Johnny, we have to have as thorough of an understanding as possible of the paper we’re replicating, be ready with ideas on what data we’ll need to use and how to access it, have a rough idea on what kind of architecture we plan on implementing, and have a proposal for our base, target, and stretch goals. All of the above should be adequately described in the following outline.

**The Team:**

The four group members are the following: Robayet Hossain (rmhossai), Angel Alvarado Reyes (agalvara), Wali Siddiqui (wasiddq), Alan Lucero (alucero). None of these individuals are capstoning this course.

**Introduction:**

**Prompt:** What problem are you trying to solve and why? If you are implementing an existing paper, describe the paper’s objectives and why you chose this paper. If you are doing something new, detail how you arrived at this topic and what motivated you. What kind of problem is this? Classification? Regression? Structured prediction? Reinforcement Learning? Unsupervised Learning? Etc.

**Response:** In the realm of sports video analysis, automating the generation of descriptive content has garnered significant attention. Traditional methods often rely on datasets from non-official sources and necessitate extensive annotations, such as pixel-level segmentation of players and the ball, which hampers their applicability in real-world scenarios. To address these challenges, Wu et al. introduced the NBA Sports Video Analysis (NSVA) dataset, a large-scale collection of official NBA game footage designed to facilitate tasks like video captioning, fine-grained action recognition, and salient player identification. Their approach minimizes manual labeling by leveraging a transformer-based architecture to process raw videos into meaningful features.

For our deep learning final project, we aim to implement and extend the methodologies presented in this paper. The primary objective is to develop a model capable of generating accurate and contextually rich captions for NBA game videos, thereby enhancing the accessibility and analysis of sports content. This task falls under the category of sequence-to-sequence learning, a subset of supervised learning, where the input is a sequence of video frames, and the output is a sequence of descriptive words forming a caption.

The motivation behind selecting this topic stems from the increasing demand for automated sports analysis tools that can provide insights without the need for labor-intensive annotations. By building upon the NSVA dataset and the SportsFormer model, this project seeks to contribute to the advancement of machine-generated sports commentary, offering potential applications in broadcasting, coaching, and fan engagement.

**Related Work:**

**Prompt:** Are you aware of any, or is there any prior work that you drew on to do your project? Please read and briefly summarize (no more than one paragraph) at least one paper/article/blog relevant to your topic beyond the paper you are re-implementing/novel idea you are researching. In this section, also include URLs to any public implementations you find of the paper you’re trying to implement. Please keep this as a “living list”–if you stumble across a new implementation later down the line, add it to this list.

**Response:** One related paper is found here: <https://arxiv.org/abs/2102.05095>. This paper implements the TimeSFormer that is used in the paper we’re implementing. The TimeSFormer is a convolution-free architecture approach to video classification. It uses “divided space-time attention”, which means it applies temporal and spatial attention separately.  
  
 Additionally, the code for the paper we’re implementing is open-source. The Github repository can be found at this link: [**https://github.com/jackwu502/NSVA/tree/main**](https://github.com/jackwu502/NSVA/tree/main)

**Data:**

**Prompt:** What data are you using (if any)? If you’re using a standard dataset (e.g. MNIST), you can just mention that briefly. Otherwise, say something more about where your data come from (especially if there’s anything interesting about how you will gather it). How big is it? Will you need to do significant preprocessing?

**Response:** Our data will be approximately 32,000 videos in the form of JPEG files. The paper “Design(s) a unified approach to process raw videos into a stack of meaningful features with minimum labelling efforts.” Because of this, extensive work will be done in order to ensure that the videos can be properly processed. This will involve converting the JPEG files into readable CSV data.

We will use data from the NSVA that includes basketball videos 32,019 with 44,649 sentences and 84.8 hours of video. The data is split into training, validation, and test sets based on 132 NBA games, with 32 games held out for validation and testing, ensuring diverse matchups in the training set. In addition to captioning, NSVA supports fine-grained action recognition with 172 action categories and player identification with 184 identities.

The dataset is processed using a pipeline that extracts fine-grained visual features from video clips, including object detection (for players, the ball, and the basket) and court-line segmentation. These features are then input into a vision transformer model for feature extraction. The global context is provided by TimeSformer, and a transformer decoder with task-specific heads is used to generate outputs for video captioning, action recognition, and player identification. The NSVA dataset is publicly accessible.

With the sheer amount of data and processing needed, we will need to process and train the data using Oscar through Brown. We will need to connect to Brown’s GPU network. The paper mentioned using 1 GPU and running the training in 6 hours, however, due to the quick turnaround time of the project and the fact that we will be using TensorFlow and not Jax, we will likely require 2 GPUs.

**Methodology:**

**Prompt:** What is the architecture of your model? How are you training the model? If you are implementing an existing paper, detail what you think will be the hardest part about implementing the model here. If you are doing something new, justify your design. Also note some backup ideas you may have to experiment with if you run into issues.

**Response:** We will train the model by first pre-training the model using an instructional video set so that we receive better results than initializing with nothing.

The goal is to predict an arbitrary-length sequence of word captions {y} for a given video clip X ∈ ℝ^H×W×3×N, using a multi-level feature extraction approach. The model is based on the UniVL framework, which incorporates four transformer backbones: coarse feature encoding, fine-grained feature encoding, cross-attention, and decoding. Initially, the model uses TimeSformer to extract spatiotemporal features by decomposing video frames into patches and processing them with a transformer. For fine-grained object modeling, an object detector (YOLOv5) identifies key objects like the ball, basket, and players, which are then processed by a vision transformer (ViT) to capture detailed regional features. The system also includes a position-aware module using court line segmentation to estimate distances between players and the basket.

To integrate both coarse and fine features, two transformers are used: a coarse encoder for video features and a finer encoder for object and position-aware features. These features are fused in a cross-encoder to create a joint representation. A transformer decoder then autoregressive generates captions based on this representation. The model is trained using the negative log-likelihood of the correct caption at each time step, and during inference, a beam search algorithm is used to generate the best possible caption. This approach combines global and local feature extraction techniques to generate accurate, detailed captions for sports video analysis.

**Metrics:**

**Prompt:** What constitutes “success?” What experiments do you plan to run? For most of our assignments, we have looked at the accuracy of the model. Does the notion of “accuracy” apply for your project, or is some other metric more appropriate? If you are implementing an existing project, detail what the authors of that paper were hoping to find and how they quantified the results of their model. If you are doing something new, explain how you will assess your model’s performance. What are your base, target, and stretch goals?

**Response:** Because this project focuses on video captioning rather than a traditional classification or regression task, accuracy alone isn’t a meaningful way to measure performance. Instead, we’ll evaluate the quality of the generated captions using several established Natural Language Generation (NLG) metrics that compare model outputs to human-written reference captions. Down below are the main metrics our team has looked into and is thinking about using to see how they apply to the SportsFormer model:

* BLEU (Bilingual Evaluation Understudy): measures how much overlap there is between the words (and short sequences of words, or n-grams) in the model-generated caption and the reference. It’s good for checking whether the model gets the “right words”, but it doesn’t account for semantics or sentence meaning very well.
* METEOR (Metric for Evaluation of Translation with Explicit ORdering): like BLEU, but it also considers synonyms and word order, which makes it a bit more flexible. It’s generally better aligned with human judgment, so we’ll use it to get a sense of how readable or natural the captions are.
* ROUGE (Recall-Oriented Understudy for Gisting Evaluation): primarily measures how much of the reference caption is “covered” by the model’s output. It focused more on recall than precision, and it’s useful for seeing if the model is leaving out important information.
* CIDEr (Consensus-based Image Description Evaluation): designed specifically for image and video captioning. It weighs common words less and rewards words that are important and unique to each scene. This will help us see whether the model is actually describing what matters in the clip.
* SPICE (Semantic Propositional Image Caption Evaluation): looks at the underlying meaning of a sentence by building scene graphs and comparing them. It’s the best option for checking whether the caption captures the relationships and actions in the video, which is key for game highlights like “Player X assists Player Y”.

After training the model, we’ll run it on the validation and test sets of the NSVA dataset. For each video segment, the model will generate a caption, and we’ll compare it to the ground-truth captions using the metrics above. This will provide us with some quantitative insights, like how well the model performs across different benchmarks, and some qualitative examples, like our team can choose a few good and bad examples to show how well the model does no real clips, especially for common NBA scenarios like dunks, steals, assists, or fouls.

To keep the project realistic and manageable, we’ve broken it into three tiers of goals based on time, scope, and compute resources, as well as the project check-in guidelines:

* Base Goal (achievable no matter what):
  + Successfully preprocess the NSVA data and run the SportsFormer codebase.
  + Train a base model and evaluate it using at least BLEU and METEOR.
  + Generate and analyze a few example captions on test videos.
* Target Goal (what we aim to achieve):
  + Reproduce the results from the original paper on all five metrics (BLEU, METEOR, ROUGE, CIDEr, SPICE).
  + Experiment with tuning hyperparameters and improving caption fluency.
  + Visualize metric trends over training epochs and include qualitative comparisons (e.g., side-by-side caption evaluations).
* Stretch Goal (if things go better than expected):
  + Modify the transformer architecture to include audio signals (e.g., crowd noise or commentary).
  + Implement a more complex attention mechanism that better captures player dynamics.
  + Explore captioning in another language or generating short highlight summaries instead of single sentences.

**Ethics:**

**Prompt:** Choose 2 of the following bullet points to discuss; not all questions will be relevant to all projects so try to pick questions where there’s interesting engagement with your project. (Remember that there’s not necessarily an ethical/unethical binary; rather, we want to encourage you to think critically about your problem setup.)

**Response:** The chosen questions and responses are down below:

**Why is Deep Learning a good approach to this problem?** Deep learning is a good approach to this problem because the model is working on learning from a defined environment in every clip. Since the NBA has a standardized viewing angle for the broadcasts of their games, the model can more effectively learn to detect and identify key aspects of the game. Similarly, the sequential nature of the input (a video) with slight changes in each frame would make for good learning in a deep learning model.

**What is your dataset? Are there any concerns about how it was collected, or labeled? Is it representative? What kind of underlying historical or societal biases might it contain?** Our dataset is non-entirely representative because it is game footage and therefore subject to the inherent biases associated with the footage. For instance, if a player receives more playing time they are likely to be better recognized by the model. Essentially, the data could be skewed towards high profile teams and players, such as LeBron and the Lakers. The model might also have an easier time identifying players who are minorities in the NBA, as they would have more observable distinguishable features as observed by the model.

**Division of Labor:**

**Prompt:** Briefly outline who will be responsible for which part(s) of the project.

**Response:** The proposed division of labor is the following: Robayet for dataset integration, Angel for feature extraction, Wali for transformer architecture, and Alan for evaluation. All in all, the entire project will be worked on equally by each team member as needed.

**Significant References:**

<https://github.com/jackwu502/NSVA/tree/main>

<https://arxiv.org/abs/2102.05095>

<https://arxiv.org/abs/2208.04897>

<https://medium.com/nlplanet/two-minutes-nlp-learn-the-bleu-metric-by-examples-df015ca73a86#:~:text=BLEU%2C%20or%20the%20Bilingual%20Evaluation,as%20paraphrasing%20and%20text%20summarization>.

<https://huggingface.co/spaces/evaluate-metric/meteor>

<https://huggingface.co/spaces/evaluate-metric/rouge>

<https://arxiv.org/abs/1411.5726>

<https://github.com/peteanderson80/SPICE>