"black and white dog jumps over bar."

IMAGE CAPTIONING PROJECT

Team: Furious Four

"man in black shirt is playing guitar."

"girl in pink dress is jumping in air."

"construction worker in orange safety vest is working on road."

Image(s) Credits: https://daniel.lasiman.com/post/image-captioning/

ABOUT THE PROJECT

Image Credits: http://www.haveyougotthatright.com/about-the-project#project

ABOUT

- ★ Automatic generation of image captions is a task close to the heart of scene understanding - one of the primary goals of computer vision
- ★ Caption generation involves challenges from both computer vision (determining the objects present in the scene) and NLP (expressing the information in a natural language) and hence has been viewed as a difficult problem for long
- ★ Recently - surge in interest in solving this problem
- ★ We refer to: Show, Attend and Tell

Image Credits: https://www.inovex.de/blog/end-to-end-image-captioning/

OBJECTIVES OF OUR PROJECT

Image Credits: https://businessfirstfamily.com/write-business-objectives-results/

OBJECTIVES

- ★ Develop an image captioning system - Given an input image, our system should be able to provide a logical caption to it.
- ★ If possible, add some novelty to the given method and improve upon it further
- ★ Make the implementation publically available for use by research community (after the course is over)
- ★ Increased understanding of CV, NLP and Attention-based ML models for all the team members

METHOD OVERVIEW

Image Credits:
https://www.businessanalystlearnings.com/blog/2015/5/9/4-process-improvement-methods-that-work-when-to-apply-them

METHOD: GENERAL OVERVIEW

- ★ The given paper uses a **attention model** (sequence to sequence model). It describes an analogy between it's encoder-decoder to machine translation systems, as image captioning can be considered an image to language translation problem
- ★ The paper implements **two** attention based **models**
- ★ One - A ***deterministic* soft** attention model which inputs a revised latent space vector representation of the image into the decoder.
- ★ The other - A **stochastic hard** attention model which inputs the latent space vector representation of a singular location in the image into the the decoder.

METHOD-1: Soft Attention

- ★ This attention mechanism takes the latent space vector representations for various locations all over the image and combines them based on the parameters α_i
- ★ The parameters are computed for each run of the LSTM, as an LSTM produces words one by one. These parameters are then fed to the attention mechanism which produces the input vector for the next LSTM step.
- ★ This method is **end to end differentiable**, hence can be trained as it is in the model.

METHOD-2: Hard Attention

- This attention mechanism takes the latent space vector representations of image locations and outputs a singular vector for a particular location into the decoder. $\Phi(\{\alpha_i\}, \{\alpha_i\})$ is a function that returns a sampled α_i at every point in time based upon a multinoulli distribution parameterized by α

- The parameters are computed for each LSTM cycle and the location in the image is changed accordingly. The method being stochastic in nature, it **can't be differentiated**. Hence a *multinoulli approximation* is made which makes this method differentiable

EXPERIMENTS

Image Credits:
https://www.monsterinsights.com/lesser-known-google-experiments-for-growing-your-business/

EXPERIMENTS

- **Different models for output prediction**: A couple of models were tried
  - **Initially**: CNN + LSTM: CNN outputs feature vectors, which were then fed into the LSTM
  - **Finally**: CNN output is sent to LSTM **each time along with the partial caption generated**

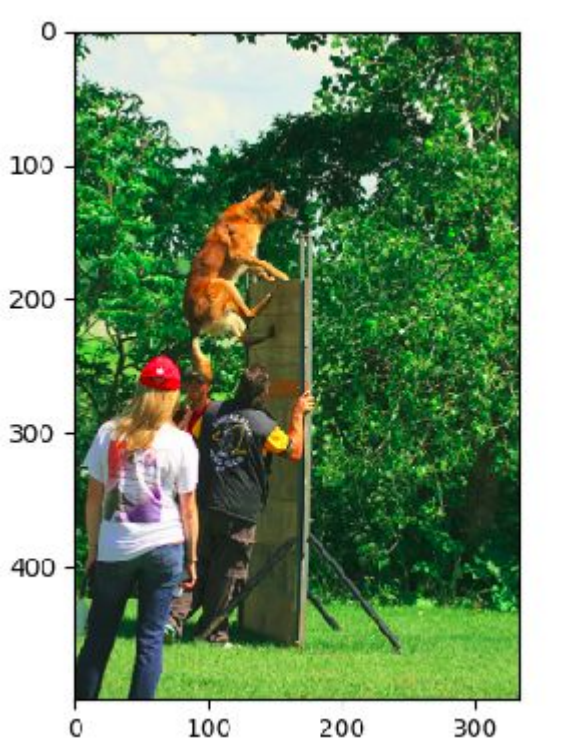
- **CNN vs Inception_V3:** We tried to build and train the whole network from scratch but the training was taking too long. So we used the pre-trained Inception_V3 model and used transfer learning for the CNN part

- **Model Parameters:** Had to play around with the parameters. Finally chose batch size as 3 and number of epochs as 10.

RESULTS

Image Credits: https://www.yourtrainingedge.com/powerful-written-goals-in-7-easy-steps/

RESULTS

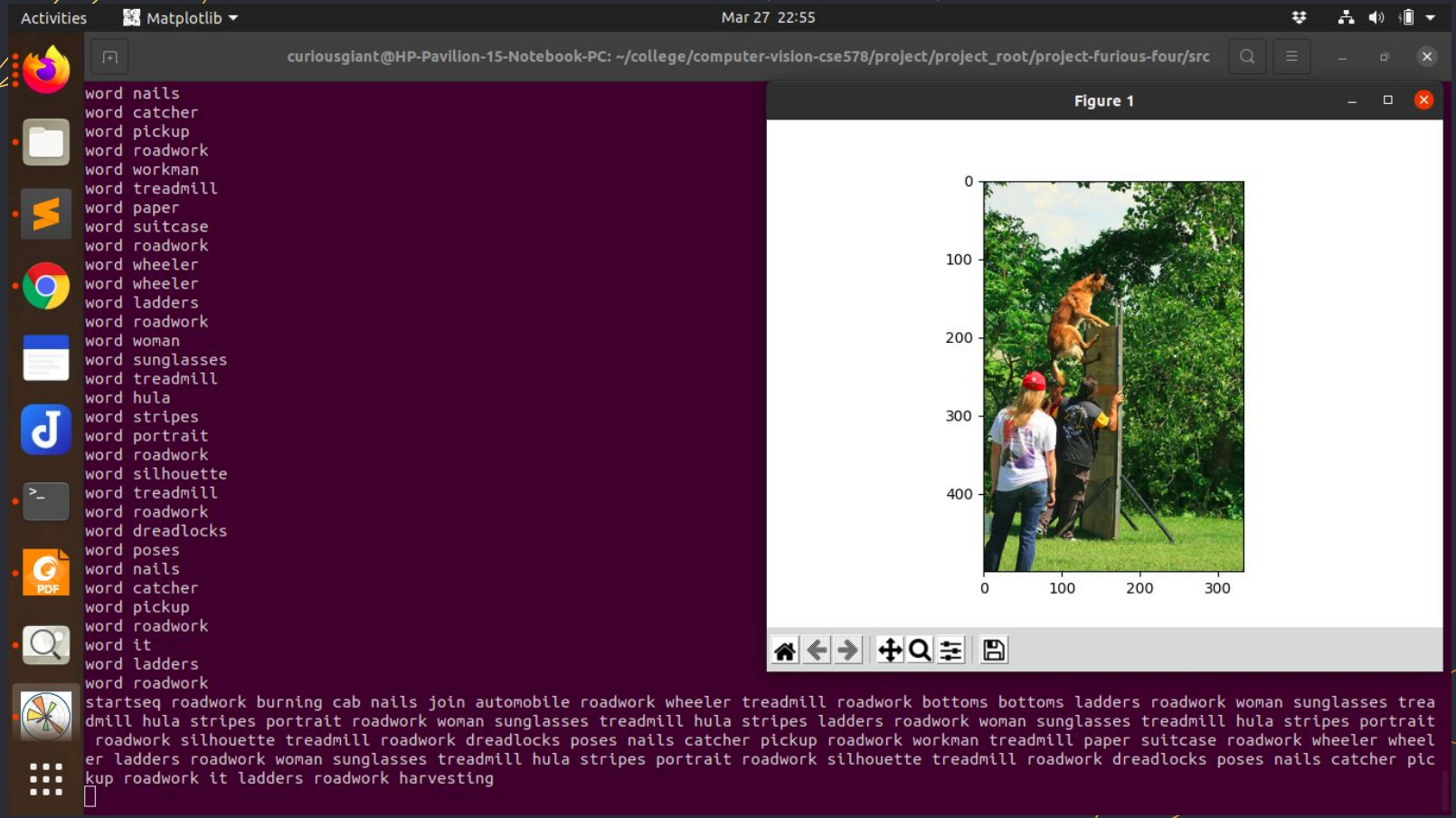


startseq roadwork burning cab nails join automobile roadwork wheeler treadmill roadwork bottoms bottoms ladders roadwork woman sunglasses treadmill hula stripes portrait roadwork woman sunglasses treadmill hula stripes ladders roadwork woman sunglasses treadmill hula stripes portrait roadwork silhouette treadmill roadwork dreadlocks poses nails catcher pickup roadwork workman treadmill paper suitcase roadwork wheeler wheeler ladders roadwork woman sunglasses treadmill hula stripes portrait roadwork silhouette treadmill roadwork dreadlocks poses nails catcher pickup roadwork it ladders roadwork harvesting

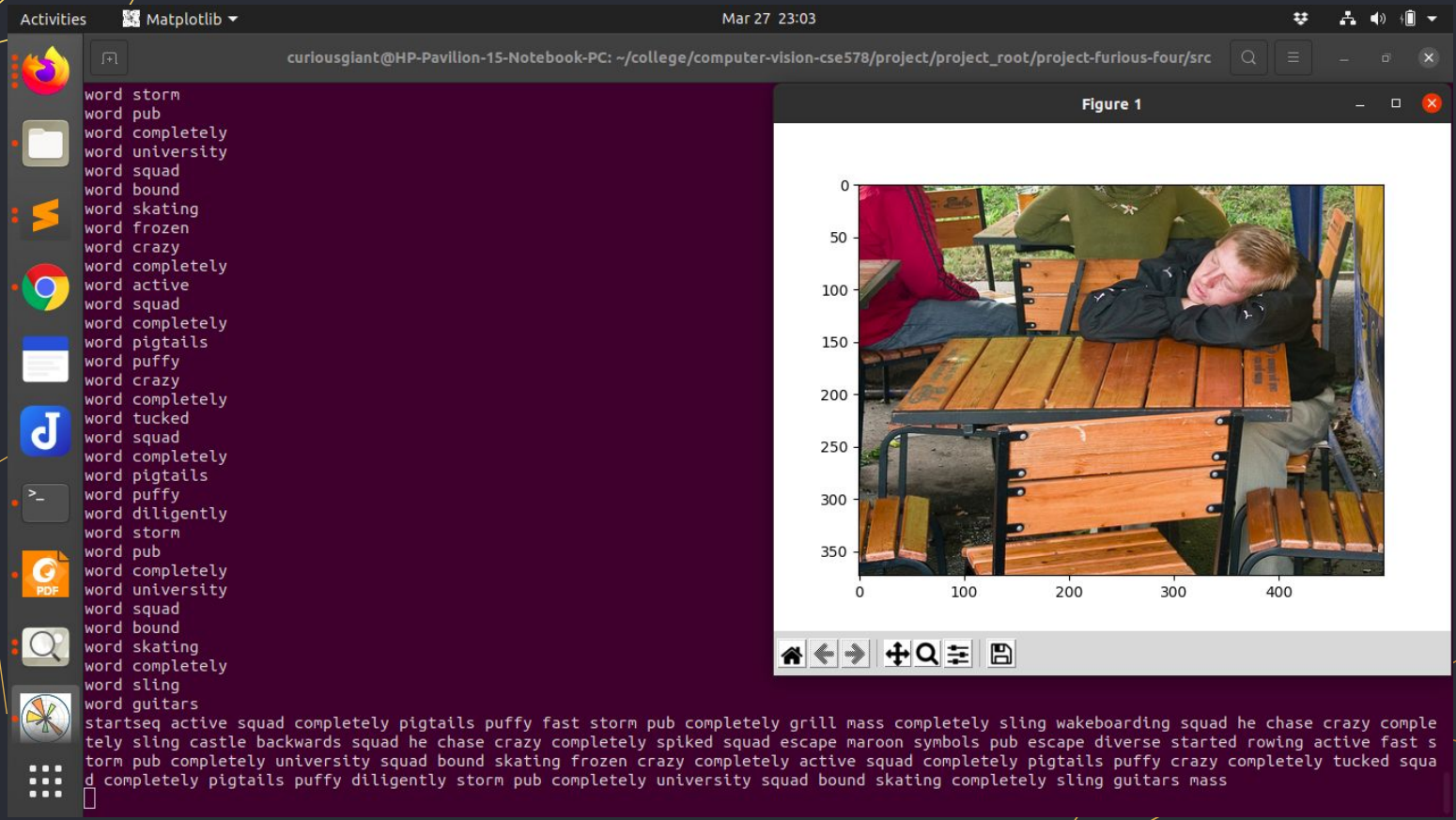
RESULTS (contd.)

startseq active squad completely pigtails puffy fast storm pub completely grill mass completely sling wakeboarding squad he chase crazy completely sling castle backwards squad he chase crazy completely spiked squad escape maroon symbols pub escape diverse started rowing active fast storm pub completely university squad bound skating frozen crazy completely active squad completely pigtails puffy crazy completely tucked squad completely pigtails puffy diligently storm pub completely university squad bound skating completely sling guitars mass

RESULTS (contd.)



RESULTS (contd.)



DIFFICULTIES FACED

Image Credits:
https://www.monsterinsights.com/lesser-known-google-experiments-for-growing-your-business/

DIFFICULTIES WE FACED AND HOW WE OVERCAME THEM

- **Training Platform:**
  - Options considered - Google Colab, Ada, Kaggle
  - Google Colab Issue - 12 Hrs reset; and Laptop hanging
  - Ada - Port forwarding issue
  - Large datasets
  - **Solution:** Eventually fixed ada
- **Feature Encoding Dictionary:**
  - Some images didn't get saved properly after creating the feature encoding dictionary and led to an error when we later loaded the encodings file
  - **Solution:** Encoding at runtime
- **Training Time:**
  - Huge training times due to redundant calculation of feature vectors.
  - **Solution:** Compute feature vectors just once and store them. Led to 25% reduction in time

TASKS COMPLETED

Image Credits:
https://medium.com/live-your-life-on-purpose/do-our-achievements-mean-anything-59d06fc9e74e

WHAT WE HAVE ACHIEVED

- ★ Dataset **cleaning and pre-processing**
- ★ **Train-test split**
- ★ **Dictionary of most frequent words**
- ★ Built a **basic model** comprising of **CNN** and **RNN**.
- ★ **Trained the model** and **tested** it with some **sample images**.

TASKS REMAINING

Image Credits: https://medium.com/simplemente/theres-a-long-road-ahead-23cbeb8b338

WHAT REMAINS

- ★ **Fine-Tune the present model to improve it further**
- ★ **Implement attention nodel**
- ★ **Perform training and testing for the whole dataset**
- ★ **Further fine-tune the models if needed**
- ★ **(BONUS)** Add some novelty if possible

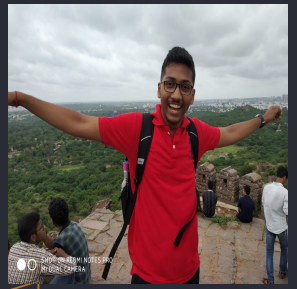
THE TEAM

Image Credits: https://iacsp.org/teamwork-within-the-surveillance-room/

TEAM

ABHISHEK SHAH
2018101052

TIRTH UPADHYAYA
2018101069



SAI TANMAY REDDY CHAKKERA
2018101054

AMOGH TIWARI
2018111003

#Team-FuriousFour

THANK YOU!

Image Credits: https://www.123rf.com/photo_28453044_stock-vector-thank-you-note-with-smiley.html

Presentation Credits: This presentation template was borrowed from SlidesGo, including icons by Flaticon, images and infographics by Freepik