Technische Universiteit
Eindhoven
University of Technology

Department of Mathematics and Computer Science
Architecture of Information Systems Research Group

Adversarial Noise Benchmarking On Image Caption

Bachelor Thesis

H.J.M. van Genuchten

Supervisors:
C. de Campos
Z.M. van Cauter

Intermediate Draft

Eindhoven, April 2022

Abstract

TODO Abstract

1 Introduction

Introduction:

- Introduce Image Caption task (Metrics?)
- Introduce Adversarial attacks/samples
- Introduce attention
- Introduce Research topic (i.e. combination of these things)

2 Methodology

2.1 Notation

- X, clean image in the range [0,1] retrieved from the dataset.
- X^{adv}, adversarial image generated from X by applying some perpetration to it. Clipped to be in the range [0,1].
- $Clip_{X,\epsilon}(A)$ is the element-wise clipping of A such that $X-\epsilon \leq A \leq X+\epsilon$ holds element-wise.

2.2 Dataset

As clean dataset the well known MSCOCO (Lin et al., 2015) dataset will be used. It contains 35 thousand images, of which 30 thousand are part of the train set, and 5 thousand of the testing set. Due to the computational limitations, only the test set is used.

2.3 Model

The model used, as already introduced in section 1, will be Show Attend and Tell. It is an interesting model as it uses attention, which can be visualized, to focus on most important places of the image.

2.4 Generating Adversarial Samples

Generating adversarial input images can be done by using the Fast Method (EQ. 1) proposed by Goodfellow, Shlens and Szegedy.

$$X^{adv} = X + \epsilon * sign(\nabla_x J(X, y_{true})) \tag{1}$$

With X being the input image, ϵ a hyperparameter determining much the original image can be perpetrated and $J(X, y_{true})$ the loss function which to, in the adversarial case, maximize. Finally, the image is clipped ensuring the vector stays within the 0 to 1 input range of the model. As can be seen in Figure 1 (and bigger size in the appendix A), using this method images up to and including $\epsilon = 0.02$ are indistinguishable and up to and including $\epsilon = 0.16$ recognizable to humans. The sign in combination with the epsilon ensures $L_\infty(X - X^{adv}) \leq \epsilon$.

In practice applying this a single time is often not enough to successfully attack S.A.T. therefore the iterative method will be used as proposed by Kurakin, Goodfellow and Bengio. Which repeatedly applies the Fast Gradient Sign Method for N iterations

$$X_0^{adv}, X_{n+1}^{adv} = Clip_{X,\epsilon}(X_n^{adv} + \alpha * sign(\nabla_x J(X_n^{adv}, y_{true}))) \tag{2}$$

In which, α is a hyperparameter which naively can be set to ϵ/N. Images generated using this method are usually less visually disturb for the same epsilons. Here an epsilon of 0.040 is nearly indistinguishable, as can be seen in figure 2

Distracting Adversarial Sample

Distraction is a powerful technique often used by adversaries in the real world. As S.A.T. employs attention to generate sentences, it is possible to try and distract it by creating an adversarial sample that makes the model hyperfocused on only part of the image. During training S.A.T. learns to divide the attention roughly equally over the whole image during the generation of a single caption. It does this by including the loss shown in equation 3.

$$L_{attention} = \sum_{i}^{L}(1 - \sum_{t}^{C} \alpha_{ti}^2) \tag{3}$$

(a) ϵ=0.000 (b) ϵ=0.020

(c) ϵ=0.000 (d) ϵ=0.160

Figure 1: Clean (left) and Adversarial images (right) for varying epsilon values of 0.020 and 0.160. Generated using equation 1. More values of epsilon can be found in appendix A.

(a) ϵ=0.000 (b) ϵ=0.020

Figure 2: Clean (left) and Adversarial images (right) for varying epsilon values of 0.020 and 0.160. Generated using equation 2. More values of epsilon can be found in appendix A.

With C equal to the amount of words generated by S.A.T., L equal to the amount of latent[1] pixels, and α_{ti} the attention given to latent pixel i for generating word t.

Using categorical cross-entropy we can craft an adversarial example which focuses the attention of S.A.T. to a single latent pixel.

$$L_{distraction} = CrossEntropy(d, \alpha) \tag{4}$$

With d, $\alpha \in \mathbb{R}^{LxC}$ and d be constructed to focus attention on a specific latent pixel. Combining it with the Iterative Method 2, results in equation 5

$$X_0^{adv}, X_{n+1}^{adv} = Clip_{X,\epsilon}(X_n^{adv} + \alpha * sign(\nabla_x J(X_n^{adv}, \alpha))) \tag{5}$$

As can be seen in figure 3 the images are visually less perturbed even with a higher epsilon. With an image with a perturbation of $\epsilon = 0.160$ almost indistinguishable from the clean image. Although $\epsilon = 0.640$ is visually distorted it is still very recognizable and would still be described the same by a human.

2.5 Evaluation

To determine if the model is indeed susceptible to distraction the BLEU-4 score (Papineni, Roukos, Ward & Zhu, 2001) will be calculated, as it is a widely reported metric within the image captioning task. Because the BLEU score checks for direct word occurrences it does not give a complete view on the success of the adversarial attack, as the model can still give a correct description using synonyms. This would result in a low BLEU score, where in fact the model is still performing correctly. To combat this the cosine similarity of the original and adversarial output will be calculated using universal sentence embedding proposed by Cer et al.. It is a separately learned model that embeds an entire sentence. In the case of distraction, the average attention the model applies on the dataset is also analyzed.

[1]S.A.T. uses a CNN architecture as feature extractor before the attention layer. The output of the CNN is described by the authors as the latent pixel space.

(a) ϵ=0.000 (b) ϵ=0.160

(c) ϵ=0.000 (d) ϵ=0.640

Figure 3: Clean (left) and Adversarial images (right) for varying epsilon values of 0.160 and 0.640. Generated using equation 5. More values of epsilon can be found in appendix **??**.

3 Motivation

Motivation

- why?
  - Safety?
  - Useful for adversarial Training
  - Better understanding network
- Never done on attention layer as attack vector

Research Questions

- WHY?
  - Is it possible to attack attention?
- Research Questions

4 Related work

Related work

- FSGM and other methods
- Introduce Show-and-Fool

5 Results

Adversarial Samples

Adversarial samples are generated using the iterative version of Fast Gradient Sign method as shown in equation 2. With $N = 10$ satisfactory results can be achieved, however higher N results in even better results for the same ϵ. Higher epsilons did not have an effect on BLEU score beyond 0.08 as can be seen in figure 4. This is in contrast to the cosine similarity as it does decrease further for the higher ϵ.

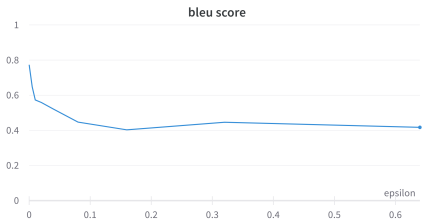


Figure 4: Average BLEU score

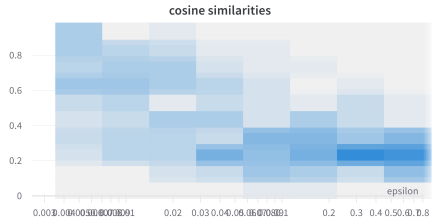


Figure 5: Cosine similarity vs epsilon (Axis is not correct yet)

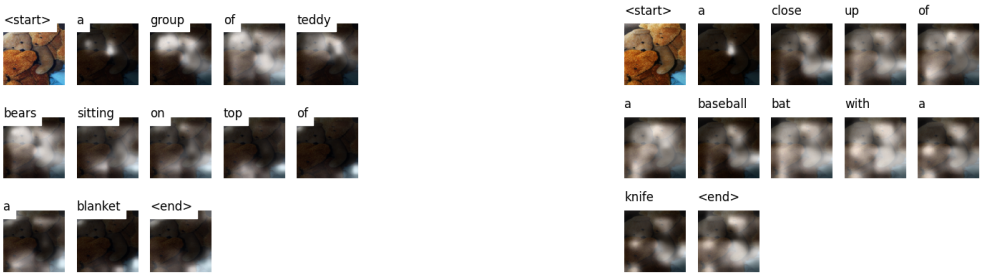


Figure 6: Clean Image (left), Adversarial Image $\epsilon = 0.02, N = 10$ (right)

As can be seen in figure 6 the attention of S.A.T, even though not explicitly attacked, is not as focused as on the clean image. This is especially visible in images that are successfully attacked. Images for which the model still is able to generate decent captions, still have a good focus on the main subjects in the image (7).

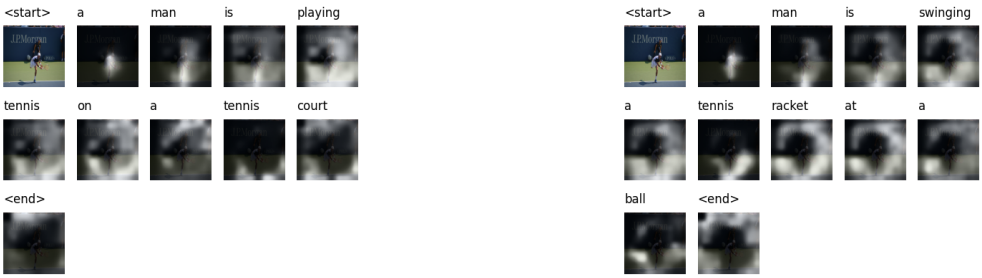


Figure 7: Clean Image (left), Adversarial Image $\epsilon = 0.02, N = 10$ (right)

Distracting Samples

To distract the model, adversarial samples are created using the iterative method (EQ. 2) and the distraction adversarial loss (EQ. 4). The amount of iterations was experimentally found to be good enough in most cases at 100, in which more would result in better distraction at the cost of longer running times. The top left pixel was chosen to focus the attention on as the model focus least on it (8) (albeit slightly) during the clean images. With an epsilon of 0.04 satisfactory results are achieved. The attention of the model clearly focused on the top left on average as can be seen in figure 9. With the perturbation at most 0.04 the image is visually almost identical to the human eye (figure 10).

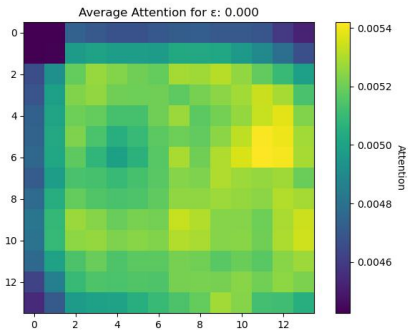


Figure 8: Average attention on clean images

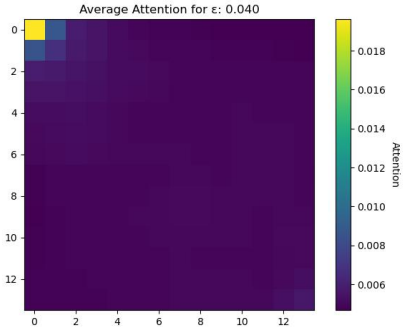


Figure 9: Average attention on adversarial images with ϵ=0.04 at 100 iterations

Figure 10: Clean Image (left), Adversarial Image $\epsilon = 0.04, N = 100$ (right)

The attention and sentence generation for figure 10 are visualized in figure 11. The model is not completely distracted and still attends to other parts of the image, however they are not clearly a single object relating to the word that is generated. During the generations of the last few words the attention is focused almost solely on the top left part.

Figure 11: Attention on Clean Image (left) and Adversarial Image $\epsilon = 0.04, N = 100$ (right)

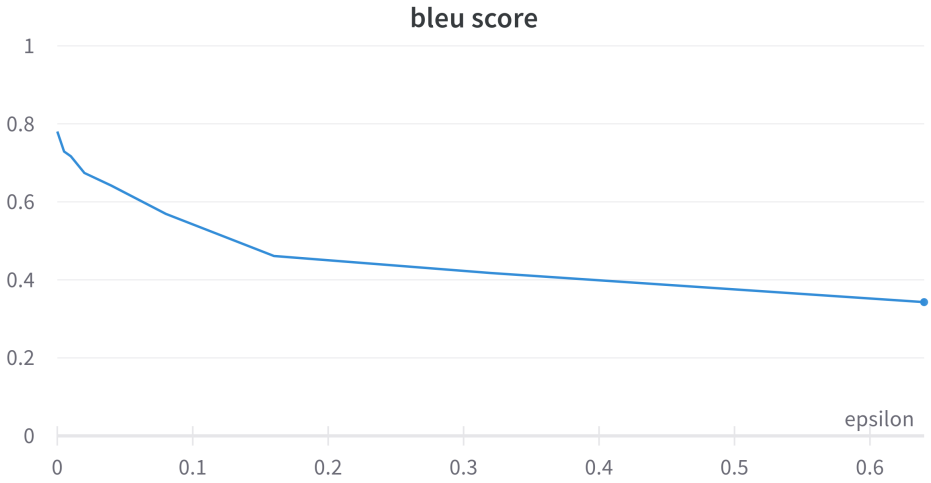


Figure 12: BLEU score during distraction over epsilon

6 Discussion

- Cosine similarity is based on a learned value, it is not a watertight value. (But neither is BLEU)
- How useful is the iterative method for training, as it takes a significant time to compute.
- Future work:

  If the adversarial samples generated are included during training is the model more robust.

  Are the adversarial samples transferable to other models, even ones not employing attention?

7 Conclusions

- S.A.T. is susceptible to adversarial samples, with the most important part in generating them is the amount of iterations. (This could use some more plots)
- Successful attacks are visible in the attention layer, even if not attack explicitly.
- The attention layer is susceptible to attacks.
- Attacking the attention is harder than attacking the sentence.
- Summarizing in attacking using the (iterative) fast gradient sign method, the most important part is iterations.

References

Cer, D., Yang, Y., Kong, S., Hua, N., Limtiaco, N., John, R. S., ... Kurzweil, R. (2018). Universal sentence encoder. *CoRR*, *abs/1803.11175*. Retrieved from http://arxiv.org/abs/1803.11175 4

Goodfellow, I. J., Shlens, J. & Szegedy, C. (2015). *Explaining and harnessing adversarial examples.* 2

Kurakin, A., Goodfellow, I. & Bengio, S. (2016). *Adversarial examples in the physical world.* arXiv. Retrieved from https://arxiv.org/abs/1607.02533 doi: 10.48550/ARXIV.1607.02533 2

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., ... Dollár, P. (2015). *Microsoft coco: Common objects in context.* 2

Papineni, K., Roukos, S., Ward, T. & Zhu, W.-J. (2001). Bleu. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02*. doi: 10.3115/1073083.1073135 4

A Bigger adversarial images

Clean image
Prediction by S.A.T.: A group of people standing around a tennis court.

Adversarial Image with $\epsilon = 0.005$
Prediction by S.A.T.: A group of people sitting in a room with a bunch of different colored vases.

Clean image
Prediction by S.A.T.: A group of people standing around a tennis court.

Adversarial Image with $\epsilon = 0.010$
Prediction by S.A.T.: A group of vases sitting on top of a table.

Clean image
Prediction by S.A.T.: A group of people standing around a tennis court.

Adversarial Image with $\epsilon = 0.020$
Prediction by S.A.T.: A group of vases sitting on top of a table.

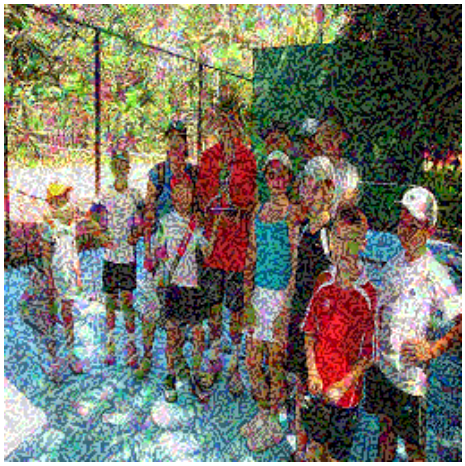
Clean image
Prediction by S.A.T.: A group of people standing around a tennis court.

Adversarial Image with $\epsilon = 0.040$
Prediction by S.A.T.: A large glass vase with a bunch of flowers on it.

Clean image
Prediction by S.A.T.: A group of people standing around a tennis court.

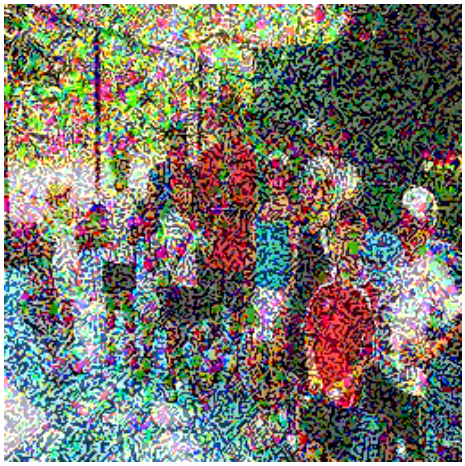
Adversarial Image with $\epsilon = 0.080$
Prediction by S.A.T.: A bathroom with a toilet and a sink.

Clean image
Prediction by S.A.T.: A group of people standing around a tennis court.



Adversarial Image with $\epsilon = 0.160$
Prediction by S.A.T.: A red wall with a red and white design.

Clean image
Prediction by S.A.T.: A group of people standing around a tennis court.



Adversarial Image with $\epsilon = 0.320$
Prediction by S.A.T.: A large red object with a red and white background.

B More Adversarial Samples

Figure 13: Prediction by Show Attend and Tell on a normal image

Figure 14: Prediction on an adversarial image with $\epsilon = 0.2$ (roughly 5% of original range)

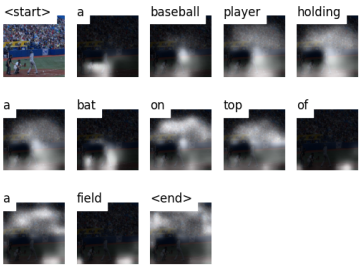


Figure 15: Prediction by Show Attend and Tell on a normal image

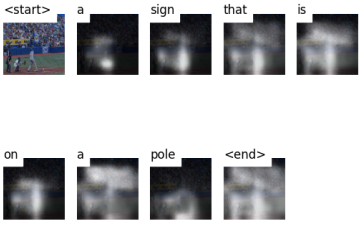


Figure 16: Prediction on an adversarial image with $\epsilon = 0.2$ (roughly 5% of original range)

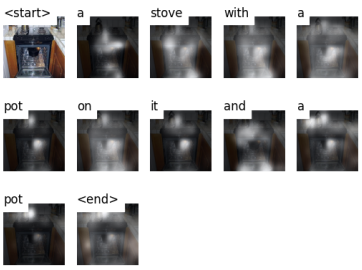


Figure 17: Prediction by Show Attend and Tell on a normal image

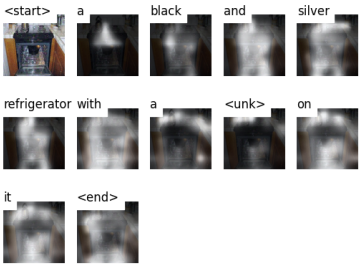


Figure 18: Prediction on an adversarial image with $\epsilon = 0.2$ (roughly 5% of original range)

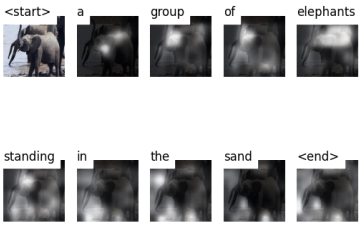


Figure 19: Prediction by Show Attend and Tell on a normal image

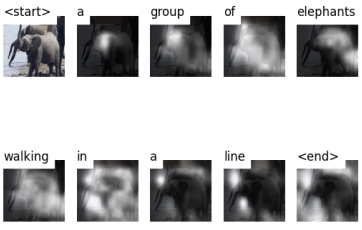


Figure 20: Prediction on an adversarial image with $\epsilon = 0.2$ (roughly 5% of original range)