Explainable Neural Binary Analysis
Michael Davinroy and Jane Adams
CS7295 Visualization for Machine Learning
Professor Enrico Bertini
Fall 2022

Motivation: What is the main goal of your project? What problem do you want to solve or what idea do you want to explore?

In the field of cybersecurity, analyzing raw binary code is an important step in both malware analysis and reverse engineering. Traditionally, this process has been done with both deterministic and heuristic methods in tools such as Ghidra [10] and IDA Pro [11]. Recently, however, researchers have started to use machine learning to solve some basic problems in this space, such as malware classification, i.e. mapping a binary to good or bad, and function boundary detections, i.e. mapping bytes in a binary to function starts.

However, because the models used in this space tend to be extremely complex and therefore black-box, researchers are currently asking, "are the models we are training learning actual semantics or just syntax?" We currently use a variety of metrics common in the field of machine learning to evaluate these models, such as precision, recall, and F1 scores, but we believe visualizing the influence of each feature (in this case bytes, instructions, operands, etc.) can help answer this question more thoroughly. In particular, we believe using SHAP values for visualizing the influence of features would work extremely well in this discrete, non-differentiable domain. Since neural binary analysis often uses ideas from NLP, we hope that we can take inspiration from some visualization ideas that have been used in that domain as well as inspire new methods that can apply to this space.

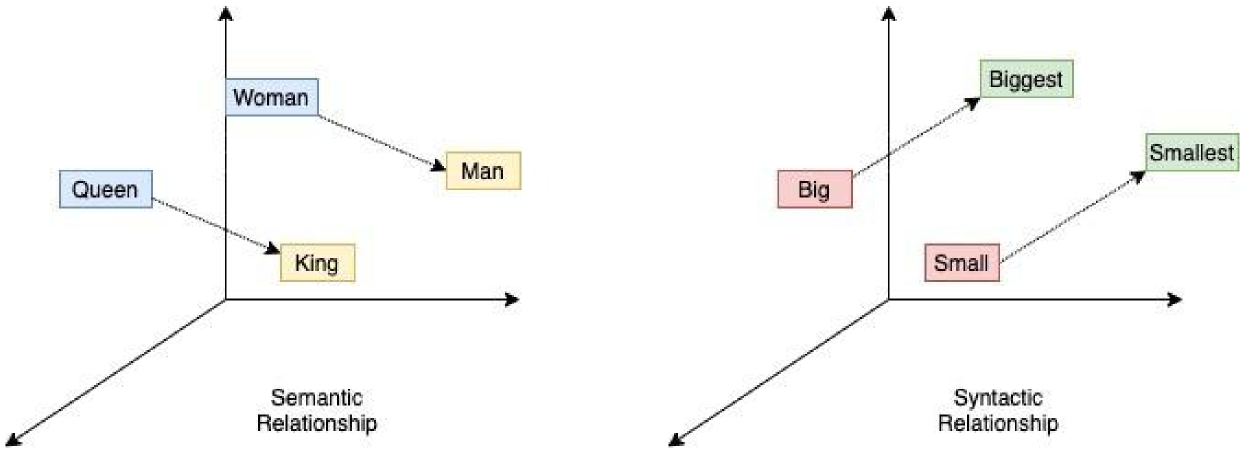
We propose to create an "interactive explainer" in the style of submissions to VISxAI, to be submitted to the 2023 HCxAI [13] workshop at ACM CHI (estimated deadline 2/22, based on last year's deadline), or, if rejected, submitted to the VISxAI [12] workshop (estimated deadline 6/22, based on last year's deadline).

Data+Model: What kind of data and model will you use? Are these data and models available?

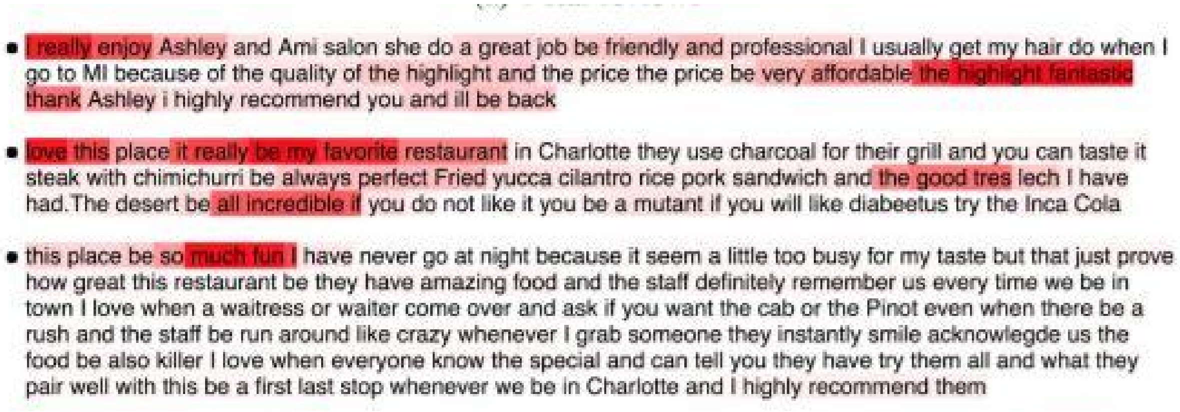
For data, we plan to use the BinKit dataset [1], as this is a large dataset representative of a large number of binaries compiled with different compilers, optimizations, and architectures. Michael is currently working on developing a new model to overcome the limitations found in current models (as outlined in his paper under review, "Attacking Neural Binary Function Detection" [2] trained with different versions of this dataset. Other models we can look at include the open source model for function boundary detection, XDA: Accurate, Robust Disassembly with Transfer Learning [3], and a closed source model for the same task, DeepDi [4]. Other models that compute varying functions also exist in this space, such as Malware Detection by Eating a Whole EXE [8]. However, we have used and are more experienced with working with the first two, so we will likely use them to keep the project pickup overhead low for the semester.

Related Work: What have others done in this space? Is there anything similar or related that others have done?

For visualization, there are a few existing encoding paradigms from the field of Natural Language Processing generatlly, which could potentially be applied in similar ways to Neural Binary Analysis. For example:

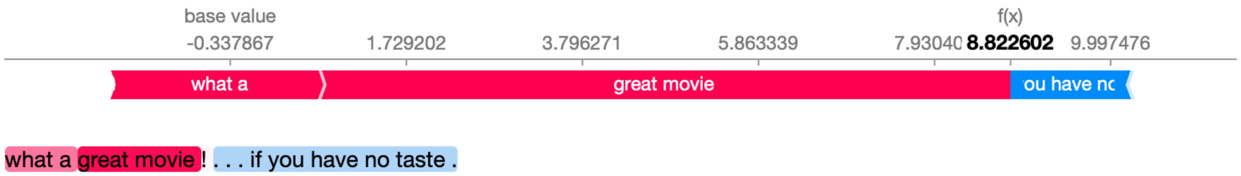


1. **Word2Vec** [7] and other embedding methods are commonly used to cluster tokens based on their co-occurrence or similar usages in linguistic contexts. By embedding tokens in low-dimensional space, semantic or syntactic relationships can be better explored.



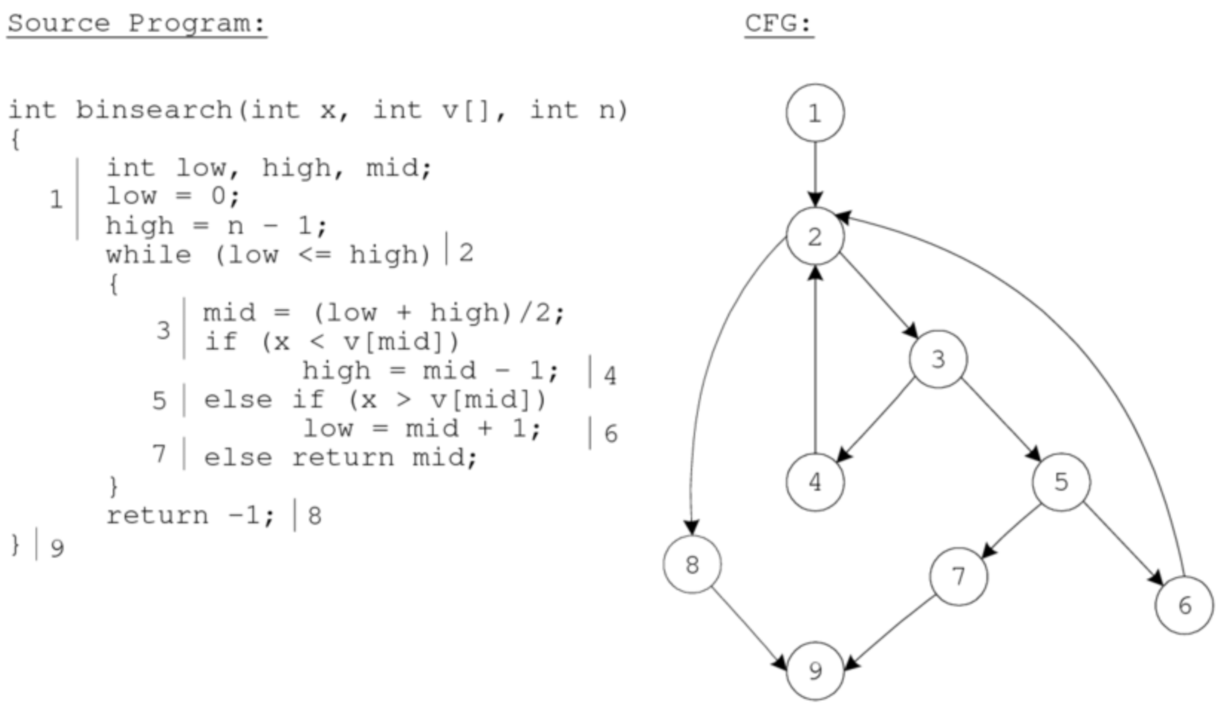
- I really enjoy Ashley and Ami salon she do a great job be friendly and professional I usually get my hair do when I go to MI because of the quality of the highlight and the price the price be very affordable the highlight fantastic thank Ashley i highly recommend you and ill be back
- love this place it really be my favorite restaurant in Charlotte they use charcoal for their grill and you can taste it steak with chimichurri be always perfect Fried yucca cilantro rice pork sandwich and the good tres lech I have had.The desert be all incredible if you do not like it you be a mutant if you will like diabeetus try the Inca Cola
- this place be so much fun I have never go at night because it seem a little too busy for my taste but that just prove how great this restaurant be they have amazing food and the staff definitely remember us every time we be in town I love when a waitress or waiter come over and ask if you want the cab or the Pinot even when there be a rush and the staff be run around like crazy whenever I grab someone they instantly smile acknowlegde us the food be also killer I love when everyone know the special and can tell you they have try them all and what they pair well with this be a first last stop whenever we be in Charlotte and I highly recommend them

2. **Attention maps** [6] for natural language processing allow users to explore the ‘reasons’ that a classification model or predictive text made its assertions, by highlighting the specific tokens in the corpus that had the greatest impact on the model output.



3. **SHAP** [5] **and other classification explainers** support visual representations of what best explains the tokens or groups of tokens that a model used to make a decision. For example, above, we might imagine instead that the highlighted “words” are portions of the code that are either causing a model to classify the binary as malicious or benign.

Additionally, some existing visual encoding paradigms from the field of binary analysis (neural or otherwise) may provide useful graphical metaphors for explanation. In particular, Control Flow Graphs (CFGs) [9] are a common means of conceptualizing how various “basic blocks” (each block is a set of instructions that execute sequentially, with no jumps in or out) interact with one another, as shown below:


Source Program:

```
int binsearch(int x, int v[], int n)
{
     | int low, high, mid;
   1 | low = 0;
     | high = n - 1;
       while (low <= high) | 2
       {
          3 | mid = (low + high)/2;
            | if (x < v[mid])
                     high = mid - 1;  | 4
          5 | else if (x > v[mid])
                     low = mid + 1;   | 6
          7 | else return mid;
       }
       return -1; | 8
} | 9
```

Evaluation: How will you evaluate your work? How do you know if you succeed?

This project is a joint effort between Dr. William K. Robertson’s Diverge Lab in the Cybersecurity and Privacy Institute, and the Data Visualization lab in the Human-Computer Interaction group, both at Khoury College, Northeastern University. The factual correctness of the explainers will be assessed and confirmed by members of the Diverge Lab, including senior faculty. The visualization context and efficacy will be assessed by members and faculty in the Data Visualization Lab.

Once we have drafted the explainer, we plan to invite other PhD students in computer science (outside of cybersecurity or visualization) to read and interact with the explainer. We will ask questions of them about the usefulness and clarity of the article. It is our objective to give readers with no familiarity with neural binary analysis (but some experience with computer science and machine learning topics generally) a fundamental understanding of the field of Neural Binary Analysis, so that they might engage with papers on the topic in the future. We understand that this kind of survey may require IRB approval and therefore the completion of Northeastern's Human Subject Protection Training; Jane already has her CITI training complete, and Michael will be completing by 18 November.

The final evaluation(s) will come in the form of our submission to the HCxAI [13] workshop at ACM CHI 2023 in February, and, if there are too many review comments for acceptance there, to re-assess and re-submit to VISxAI [12] at IEEE VIS 2023 in June.

Works Cited

[1] Kim, Dongkwan et al. 'Revisiting Binary Code Similarity Analysis using Interpretable Feature Engineering and Lessons Learned'. *IEEE Transactions on Software Engineering* (2022): 1–23. Web.

[2] Bundt, Joshua et al. 'Attacking Neural Binary Function Detection'. 2022. Web.

[3] Pei, Kexin et al. 'XDA: Accurate, Robust Disassembly with Transfer Learning'. 2020. Web.

[4] Yu, Sheng et al. 'DeepDi: Learning a Relational Graph Convolutional Network Model on Instructions for Fast and Accurate Disassembly'. *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, 2022. 2709–2725. Web.

[5] Lundberg, Scott M. and Lee, Su-In. 'A Unified Approach to Interpreting Model Predictions'. *Advances in Neural Information Processing Systems 30*. I. Guyon and U. V. Luxburg and S. Bengio and H. Wallach and R. Fergus and S. Vishwanathan and R. Garnett,, 2017. 4765–4774. Web.

[6] Galassi, Andrea, Marco Lippi, and Paolo Torroni. 'Attention in Natural Language Processing'. *IEEE Transactions on Neural Networks and Learning Systems* 32.10 (2021): 4291–4308. Web.

[7] Mikolov, Tomas et al. 'Efficient Estimation of Word Representations in Vector Space'. 2013. Web.

[8] Raff, Edward et al. 'Malware Detection by Eating a Whole EXE'. 2017. Web.

[9] Al-Ekram, R. and Kontogiannis, Kostas. 'Source code modularization using lattice of concept slices'. N.p., 04 2004. 195–203. Web.

[10] NSA. Ghidra. https://ghidra-sre.org/

[11] Hex Rays. IDA Pro. https://hex-rays.com/ida-pro/

[12] VISxAI Workshop at IEEE VIS. https://visxai.io/

[13] HCxAI Workshop at ACM CHI. https://hcxai.jimdosite.com/

[14] Human Subject Protection Training for Northeastern reseearchers, via CITI.