Generating Privacy and Security Threat Summary for Internet of Things

Ray Yan
Dept. of Electrical and Computer Engineering
Purdue University
yan195@purdue.edu

Logan Lebanoff
Dept. of Computer Science
University of Central Florida
Logan.Lebanoff@knights.ucf.edu

Dr. Fei Liu
Dept. of Computer Science
University of Central Florida
feiliu@cs.ucf.edu

Abstract

- Website privacy policies are too long and vague for normal people to have the time to go through and fully understand. As a result, people usually skip over reading these documents.
- To help deal with this issue, we are doing research into developing a GAN (Generative Adversarial Network) capable of processing a document's natural language and give a vagueness rating similar to that of a human.
- The data set the GAN is trained on is a set of 5000 sentences taken from privacy policies with vagueness ratings done by humans hired via Amazon Mechanical Turk. For example, a sentence like "By agreeing to this policy you are willing to share your information with third-parties" may be rated with a vagueness score of 3 out of 5 by someone.

Data Analysis

- In order to be able to train a machine learning model, there must be a rich and robust set of data for it to learn from.
- To help verify data quality, I programmed some analytics work in Python on the annotated vagueness ratings.
- I worked on determining overall score distribution, words frequently cited as vague (as seen below), and most importantly, inter-rater reliability. Inter-rater reliability is the score of homogeneity between raters.
- There were some difficulties involved in determining inter-rater reliability due to issues finding a suitable model that fits with the relatively piecemeal way the data was gathered using Amazon Mechanical Turk.

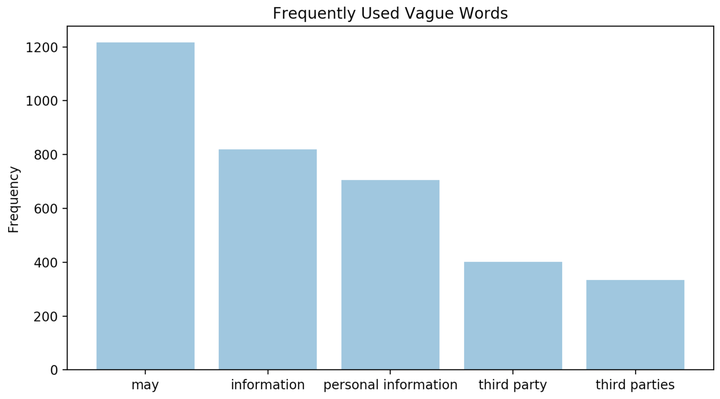


Figure 1: Python generated word frequency graph

How GAN Works

- A Generative Adversarial Network is a means of machine learning that uses two neural networks competing against each other.
- For our project, one generates sentences with a certain vagueness score based on a set of human ratings of privacy policy fragments, and tries to make it as real as possible.
- The other neural network attempts to discriminate between artificial and real sentences, while also assigning a vagueness rating as close as possible to the original.
- Ideally, once the process is finished, the generator and discriminative networks will have competed against each other and evolved to the point where the discriminator network can give human-quality vagueness ratings of privacy policy sentences.

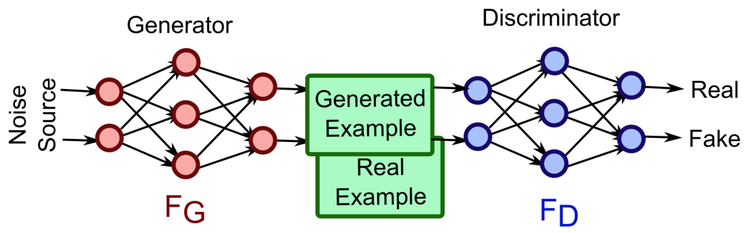


Figure 2: Model of how GAN neural networks function, [3]

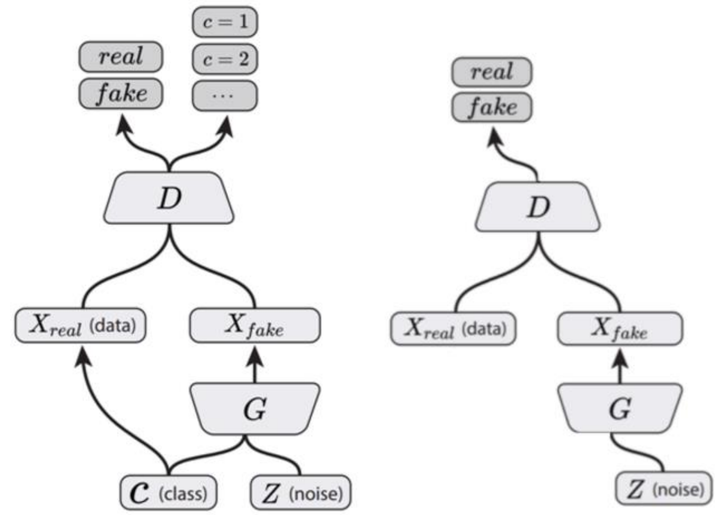
References

1. Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In Advances in neural information processing systems. pages 2672–2680.
2. Fei Liu, Nicole Lee Fella, and Kexin Liao. 2016. Modeling language vagueness in privacy policies using deep neural networks. In 2016 AAAI Fall Symposium Series.
3. [Guttenburg, Nicholas. "Stability of Generative Adversarial Networks." *Araya*. N.p., n.d. Web. <http://www.araya.org/archives/1183>.

Future Work

- We may try to improve capability of judging inter-rater reliability by imposing stricter rating standardization guidelines on human data gathered from Amazon Mechanical Turk, in order to better confirm reliability of data set for training.
- Continue working on the capabilities of the GAN, especially on refining the auxiliary classifier.
- In the future, a semi-supervised learning model may be implemented in order to generate more example sentences that can be used for training.

Figure 3: AC-GAN vs GAN comparison, [2]



Acknowledgments

The support for this work was provided by the National Science Foundation REU program under Award No. 1560302.

Any opinions, findings, and conclusions and recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Many thanks to Dr. Fei Liu and Logan Lebanoff for helping me learn a lot about machine learning and Python programming.