A Survey of Techniques for Verifying Deep Neural Networks

BEN TROVATO* and G.K.M. TOBIN*, Institute for Clarity in Documentation, USA
LARS THØRVÄLD, The Thørväld Group, Iceland
VALERIE BÉRANGER, Inria Paris-Rocquencourt, France
APARNA PATEL, Rajiv Gandhi University, India
HUIFEN CHAN, Tsinghua University, China
CHARLES PALMER, Palmer Research Laboratories, USA
JOHN SMITH, The Thørväld Group, Iceland
JULIUS P. KUMQUAT, The Kumquat Consortium, USA

A clear and well-documented LaTeX document is presented as an article formatted for publication by ACM in a conference proceedings or journal publication. Based on the "acmart" document class, this article presents and explains many of the common variations, as well as many of the formatting elements an author may use in the preparation of the documentation of their work.

CCS Concepts: • **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

Additional Key Words and Phrases: datasets, neural networks, gaze detection, text tagging

ACM Reference Format:
Ben Trovato, G.K.M. Tobin, Lars Thørväld, Valerie Béranger, Aparna Patel, Huifen Chan, Charles Palmer, John Smith, and Julius P. Kumquat. 2018. A Survey of Techniques for Verifying Deep Neural Networks. *J. ACM* 37, 4, Article 111 (August 2018), 3 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

ACM's consolidated article template, introduced in 2017, provides a consistent LaTeX style for use across ACM publications, and incorporates accessibility and metadata-extraction functionality necessary for future Digital Library endeavors. Numerous ACM and SIG-specific LaTeX templates have been examined, and their unique features incorporated into this single new template.

If you are new to publishing with ACM, this document is a valuable guide to the process of preparing your work for publication. If you have published with ACM before, this document provides insight and instruction into more recent changes to the article template.

*Both authors contributed equally to this research.

Authors' addresses: Ben Trovato, trovato@corporation.com; G.K.M. Tobin, webmaster@marysville-ohio.com, Institute for Clarity in Documentation, P.O. Box 1212, Dublin, Ohio, USA, 43017-6221; Lars Thørväld, The Thørväld Group, 1 Thørväld Circle, Hekla, Iceland, larst@affiliation.org; Valerie Béranger, Inria Paris-Rocquencourt, Rocquencourt, France; Aparna Patel, Rajiv Gandhi University, Rono-Hills, Doimukh, Arunachal Pradesh, India; Huifen Chan, Tsinghua University, 30 Shuangqing Rd, Haidian Qu, Beijing Shi, China; Charles Palmer, Palmer Research Laboratories, 8600 Datapoint Drive, San Antonio, Texas, USA, 78229, cpalmer@prl.com; John Smith, The Thørväld Group, 1 Thørväld Circle, Hekla, Iceland, jsmith@affiliation.org; Julius P. Kumquat, The Kumquat Consortium, New York, USA, jpkumquat@consortium.net.

0004-5411/2018/8-ART111 $15.00
https://doi.org/10.1145/1122445.1122456

The "acmart" document class can be used to prepare articles for any ACM publication — conference or journal, and for any stage of publication, from review to final "camera-ready" copy, to the author's own version, with *very* few changes to the source.

2 BACKGROUND

2.1 Deep Neural Networks

Activation Function.

2.2 An Example

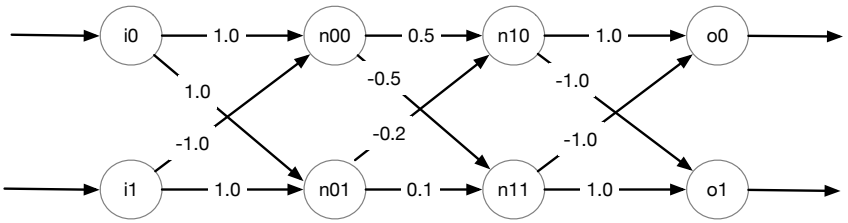

Fig. 1. A DNN example with 2 inputs, 2 hidden layers, and 2 outputs.

Fig. 1 shows a simple DNN that we will use to demonstrate various techniques throughout this paper. This DNN is composed of an input layer with two inputs i_0, i_1, an output layer with two outputs o_0, o_1, and two hidden layers with two nodes for each layer. The weights are shown for each edge and in this example we do not use bias (i.e., they are set to 0). This DNN model is fully connected (no weights having value 0) and uses the ReLU activation function.

2.3 Problem: Verifying DNN

2.3.1 DNN Verification Problem. Let N be a DNN with ReLU's and let α be a property on the inputs x's and β a property on the outputs y's of N. Our verification problem asks if $\alpha(x) \implies \beta(y)$ is a property of N. That is, every assignment σ for x that satisfies α, the result of propagating σ through N is an output that satisfies β. In other words, every input satisfying the precondition α produces in an input satisfying the postcondition β.

2.3.2 Complexity: NP-Complete. Show the reduction and use an example

2.4 Testing and Verifying DNNs

3 TESTING TECHINQUES

3.1 Symbolic Execution

4 VERIFICATION TECHNIQUES

4.1 Reluplex and Marabu

4.2 ReluVal and Neurify

4.3 Eran

5 PROPERTY INFERENCE TECHNIQUES

The title of your work should use capital letters appropriately - https://capitalizemytitle.com/ has useful rules for capitalization. Use the `title` command to define the title of your work. If your work has a subtitle, define it with the `subtitle` command. Do not insert line breaks in your title.

If your title is lengthy, you must define a short version to be used in the page headers, to prevent overlapping text. The `title` command has a “short title” parameter:

```
\title[short title]{full title}
```