"What Does My Classifier Learn?" A Visual Approach to Understanding Natural Language Text Classifiers

Jonas Paul Winkler[1], Andreas Vogelsang[2]

Abstract: Neural Networks have been utilized to solve various tasks such as image recognition, text classification, and machine translation and have achieved exceptional results in many of these tasks. However, understanding the inner workings of neural networks and explaining why a certain output is produced are no trivial tasks. Especially when dealing with text classification problems, an approach to explain network decisions may greatly increase the acceptance of neural network supported tools. We have developed an approach to visualize reasons why a classification outcome is produced by convolutional neural networks by tracing back decisions made by the network. The approach is applied to various text classification problems, including our own requirements engineering related classification problem. We argue that by providing these explanations in neural network supported tools, users will use such tools with more confidence and also may allow the tool to do certain tasks automatically.

Keywords: visual feedback, neural networks, artificial intelligence, machine learning, natural language processing, explanations, requirements engineering

1 Introduction

Artificial Neural Networks have become powerful tools for performing a wide variety of tasks such as image classification, text classification, and speech recognition. Recently, convolutional neural networks that were almost exclusively used for image processing tasks were also adapted to solve natural language classification tasks [Ki14]. However, neural networks usually do not explain why certain decisions are made. Especially when integrating neural networks into tools, users may not understand certain network decisions and consequently do not use the tool.

In [WV17], we proposed a technique to trace back decisions made by convolutional neural networks for text classification. We use this technique to create visual explanations by highlighting most important words in an input sentence. In this paper, we present a brief overview of the technique and its applications.

[1] Technische Universität Berlin, jonas.winkler@tu-berlin.de
[2] Technische Universität Berlin, andreas.vogelsang@tu-berlin.de

2 Tracing Back Network Decisions

Convolutional Neural Networks for text classification as described in [Ki14] classify examples by applying a set of filters to a sentence matrix (i.e., a sentence transformed into a numerical representation using word embeddings) and associating individual filters with output classes using fully connected layers. At each step, itermediate values are calculated, representing whether learned features have been detected or not. High intermediate values indicate that a feature was detected, and thus imply that a certain input part was important for deciding towards or against classifying an input example as a certain output class. We utilize these phenomena to calculate *Document Influence Matrices*. These matrices indicate which input elements were important for deciding the class of an input example.

Tab. 1: Examples

positive	both a successful adaptation and an enjoyable film in its own right .
negative	just a bunch of good actors flailing around in a caper that's neither original nor terribly funny .

Visual representations may be created based on these matrices. An example is provided in Tab. 1. A convolutional neural network has been trained on a dataset containing positive and negative movie reviews. Our tracing technique shows why these examples have been classified as positive and negative. Example 1 contains word groups („successful“, „an enjoyable film“, „its own“) usually accociated with positive sentiment, whereas example 2 contains word sequences with negative sentiment („neither ... nor terribly funny“).

3 Applications & Conclusions

At our industry partner, requirements documents are created to document the expected behavior of automotive systems. These documents contain both requirements and non-requirements (i.e., explanations, examples, references). A strict separation between these two classes is required. Requirements engineers usually classify the contents of a document manually. We have built a tool, incorporating the approach presented above to assist RE experts in this task.

We assume that by providing explanations, users may better understand the decisions made by the tool and consequently perform the given task more effectively and efficiently.

References

[Ki14] Kim, Yoon: Convolutional Neural Networks for Sentence Classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1746–1751, 2014.

[WV17] Winkler, Jonas P.; Vogelsang, Andreas: "What Does My Classifier Learn?" A Visual Approach to Understanding Natural Language Text Classifiers. In: Proceedings of the 22nd International Conference on Natural Language & Information Systems. NLDB, 2017.