Assignment 5

Now when you know what kind of data your study will collect, you can proceed to plan for how to analyze it so that the outcomes respond to your problem, aims, and questions. There are several things to think about. For example, if the aim was to *explore* then analysis will need to produce answers to questions like 'what' and 'how.' If the aim was to *evaluate* then analysis will need to produce answers to questions like 'how' and 'why'. Read Denscombe's book pp. 235-240 for a good introduction to the central questions of data analysis. Chapters 13 and 14 in Denscombe's book discuss quantitative and qualitative data analysis.

Do the following items:

1. Combine all your work from the previous weeks to this week's task.
2. Edit your previous texts according to the feedback you got from other course participants.
3. Continue your previous work with a description of your data analysis plan under five headings: "Data preparation plan", "Initial exploration plan", "Analysis plan", "Presentation plan", and "Tools." Include exact references (incl. page numbers) to literature in your text under each heading.
4. Choose a tool that allows you to take screen shots of the data analysis process and learn the basics of using the tool so that you'll be ready for it in the next weeks. (Examples of qualitative data analysis tools include Atlas.TI and Dedoose, and many others. Examples of quantitative data analysis tools include SPSS and R. Some web-based data collection instruments provide their own tools. If you work with computer-generated data, think of what tools you can use to analyze the data, draw the appropriate charts, do comparisons, and so forth.)

Assignment Information

Learning objectives: 1) To prepare for rigorous data analysis. 2) To understand the elements of analysis and how they are connected to the rest of the study. 3) To get acquainted with one data analysis software.

Estimated time to finish: 10 hours

========================= See Next Page =========================

Background and problem statement:

Cyber bullying and online harassment threaten the prosperity and existence of online communities [1]. According to a 2014 study conducted by Pew Research, 40% of internet users have faced harassment and 73% of users have seen others get harassed [2]. Human content moderation is neither scalable nor effective, which increases the need for automatic detection of abusive language. Deep Learning methods are reported to have the best performance on this task [5]. However, these methods can easily be fooled by systematically modified syntax, known as adversarial examples. A model that assigns a toxicity score of 90% to the sentence 'you're an idiot' may give a score as low as 20% to its adversarial counterpart 'you're an i d i o t'. More generally, adversarial sentences are highly toxic sentences receiving a low score because of a non-semantic modification of their text [3].

This project aims to explore methods to increase text-classifiers' resistance to such examples. Previous studies attempted to identify the types of text transformations most likely to affect toxicity scores ([3]). Other studies outlined general solutions to overcome this issue ([7]), some involving human intervention ([6]). This study will test the effect of one solution, adversarial training, on improving the robustness of deep learning models. If time permits, the study can also extend to test the effect of character-based embedding on the same task. The base model for this study would be a Recurrent Neural Network with attention mechanisms, as it is reported to have a state-of-the-art performance on non-adversarial datasets [4]. The dataset used in the study is a set of comments from Wikipedia's edit discussions, published by Conversation AI through a Kaggle competition in 2018. The project would report on any achieved results and point to areas for improvement to guide future research on this topic.

Project aim:

Improve the resistance of automatic toxicity detection models to adversarial attacks.

Objectives:

- Use adversarial training to increase the robustness of an attention RNN model for toxicity detection.
- Make the model and training procedure available as an open-source project for future research.
- Report on the approach(es) explored for adversarial training and the achieved performance (measured in accuracy of prediction).

Main Question:
Which text-classification methods can best predict toxic online comments?
Sub-questions:
Which text-classification techniques are relevant to the identification of abusive language in online texts?
Does the performance of our model meet state-of-the-art results?
How do the investigated methods compare to one another in terms of performance?
What guidelines does this project provide for future research on this topic?

Limitations:

Limitations of this study include access to a diversified dataset for comment classification. The only one we found so far with enough use is Conversation AI's. The applicability of our study may also be limited to English speaking environments, since we do not intend to achieve similar results in other languages for now. We also face time limitations due to the relatively short span of this course/project, which is why we tried to narrow down our research to a single model and a single counter-adversarial method. Finally, the lack of experience of our team in this topic may also be a hindrance to the timely development of the project.

Types of information needed:

1) A formal definition and/or taxonomy of toxic language, cyberbullying and online harassment.
2) An overview of text-classification models and a literature-supported justification of the chosen model (Recurrent Neural Network with Attention Mechanisms).
3) Dataset of online comments for training and testing the model.

Data collection methods:

Information needed in 1) and 2) can be collected through literature review in the field of NLP, particularly in tasks relevant to the detection of toxic language. This type of data would be qualitative, and its use would depend on the understanding and interpretation of the researchers. Since the data is available through published reports, our data collection method in this case is *document studies.*

For the information mentioned in 3), we can use *observation studies* or *document studies* [1.b]. Observation studies would allow us to collect data about the online behavior of a group of participants in a non-invasive study environment. Document studies would be existing online comments collected from the public profiles of consenting participants. Since we are solely interested in natural language texts generated online, and not in the behavior of participants in such situations, we believe document studies to be more suitable for our research. It is also notably easier to organize this data collection scheme given our constraints in time and resources.

One variant of document studies we can use is collecting data (online comments) directly from a group of participants, then having said data labeled by another (larger) group of participants based on the level of toxicity of the text. This process is time consuming and requires resources we may not have access to (especially for the labeling step). We therefore opted for a ready-made publicly available dataset, collected by Conversation AI, and published by Kaggle through a data science competition in 2018. This dataset has been used extensively in research like ours (see [4], [6] and [8] from the reference list).

We chose a quantitative data analysis method, since our research uses statistical methods to assign numerical scores to online texts based on their level of toxicity. [2.b]

Pros:

- Effortless and organizable considering time and resources.
- Based on genuine and intact comments of users.
- Low cost, well detailed and variety data.

- Unbiased collection process can be verified and cross-checked for errors.

Cons:

- Limitations with documentation itself.
- Size of data consume time and resources.

References:

[1] O'Brien, D., & Kayyali, D. (2015, January 22). Facing the Challenge of Online Harassment. Retrieved January 23, 2020, from https://www.eff.org/deeplinks/2015/01/facing-challenge-online-harassment

[2] Bottino, S., Bottino, C., Regina, C., Correia, A., & Ribeiro, W. (2015). Cyberbullying and adolescent mental health: systematic review. Cadernos De Saúde Pública. doi: 10.1590/0102-311X0003611

[3] Hosseini, H., Kannan, S., Zhang, B., & Poovendran, R. (2017, February 27). Deceiving Google's Perspective API Built for Detecting Toxic Comments. arXiv:1702.08138

[4] Kohli, M., Kuehler, E., & Palowitch, J. (2018). Paying attention to toxic comments online.

[5] Georgakopoulos, S. V., Vrahatis, A. G., Tasoulis, S. K., & Plagianakos, V. P. (2018). Convolutional Neural Networks for Toxic Comment Classification. In SETN '18: Proceedings of the 10th Hellenic Conference on Artificial Intelligence. New York, NY: Association for Computing Machinery.

[6] Dinan, E., Humeau, S., Zhang, B., Chintagunta, B., & Weston, J. (2019, August 17). Build it Break it Fix it for Dialogue Safety: Robustness from Adversarial Human Attack. arXiv:1908.06083

[7] Mehdad, Y., Tetreault, J. (2016, September). Do Characters Abuse More Than Words? Conference: Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue. doi: 10.18653/v1/W16-3638

Other potential references:

[8] Chakrabarty, N. (2012). A Machine Learning Approach to Comment Toxicity Classification

[9] Mehdad, Y., Tetreault, J. (2017, September). Deeper Attention to Abusive User Content Moderation. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. doi: 10.18653/v1/D17-1117

References for research methods:

[1.b] Johannesson, P., & Perjons, E., (2012, p.30). A design Science Primer. CreateSpace Independent Publishing Platform.

[2.b] Johannesson, P., & Perjons, E., (2012, p.31). A design Science Primer. CreateSpace Independent Publishing Platform.

Data Collection Strategy:

The first thing that is important to get clear from the beginning, is that our samples will not be retrieved by us personally, instead we will use samples that have been already collected. Since we are trying to improve the capability of neural networks for the detection of adversarial toxic comments, collecting a huge amount of conversations would be impossible for us, and to expect that in those conversations there are good examples of toxic comments would be even more optimistic. We plan to use a dataset of comments collected from Wikipedia's edit discussions. The dataset was prepared by Google's Conversation AI and was made available to the public through a Kaggle competition in 2018. This notebook provides a statistical overview of the dataset. We consider our sampling frame to be all edit comments available on Wikipedia.

Our sample will be representative, since we expect that with enough amount of conversations we will cover, of course not every situation of any context, but at least a great quantity of them, making the NN capable of understanding toxic comments in any different possibility and so representing as much as possible any conversation. Due to this our sample frame must match only two specific pre-requisites, apart of course from being English speakers, since our research is mainly done for English NLP, it should contain a decent amount of toxic conversations.

As I said the samples have already been collected (due to this , the time for data collection at least for us is minimum), but indeed inside this sample we expect to find some level of toxicity, therefore our data strategy for selecting the samples will be purposive sampling [2.c, p 1] since we are not going to discriminate inside the data with any specific filter, and any member of the population will have the same chances of being selected for the research, but this dataset has been selected by us due to its specific situation since it's known that it contains toxic comments that have been pre-rated by humans.

Between probability sampling and non-probability sampling following [1.c, p 15], our choice is non-probability sampling. The main reason extracted from this source is "it is not feasible to include a sufficiently large number of examples in the study", the cost would be too high and we wouldn't have any confidence on our dataset that it contains enough toxic comments. And for the last, as we said on "data collection methods" we have chosen our informants/dataset since it has been specifically developed for tasks of this topic (detection of toxic comments).

Finally, we identify topics of discussion as a potential bias of our dataset. Since our data is collected from a single source, we expect toxic sentences used to be relevant to particular topics (e.g., political differences). This may affect the generalization of our model especially in production environments. However, since this dataset has been used extensively in research ([4], [5], and [6] are only few examples), we believe this bias does not pose a serious threat to generalization and is thus of minimal importance.

References for data collection strategy:

[1.c] Denscombe, M. (2014). The good research guide: for small-scale social research projects. McGraw-Hill Education (UK).

[2.c] Kandace Landreneau. Sampling strategies:
http://www.natco1.org/research/files/SamplingStrategies.pdf

[3.c] Marshall, M. N. (1996). Sampling for qualitative research. Family practice, 13(6), 522-526.

Data collection protocol:

- **Model choice and justification**

Our model will be a Bidirectional Long Short-Term Memory Recurrent Neural Network (BiLSTM RNN for short). RNNs, particularly BiLSTMs, are reported to perform best in text-classification according to [1]. Furthermore, in a Kaggle competition launched by Conversation AI, RNN models reached an accuracy score of 98%. Since our focus is on increasing classifiers' robustness against adversarial examples, we chose as a baseline model a classifier that is performing as well as possible on standard datasets. Another motivation for using this model is the availability of open-source examples on tasks like ours (e.g., [2]), and the abundance of libraries facilitating the implementation of such models (Keras, Tensorflow, PyTorch to only name a few). This is criteria is particularly relevant given our restrained time constraints.

- **Adversarial training**

In order to make the model resistant to adversarial attacks, we plan to explore the following methods:

- **Algorithms generating adversarial examples:** the aim of this type of algorithms is to introduce systematic modifications to text data to make it undetectable by computer models but still identifiable by human classifiers. We chose to focus on 'black-box' algorithms, which operate without knowledge of the architecture (and therefore strengths and weaknesses) of the model they are attacking. DeepWordBug is one such algorithm [3]. We plan to look up more algorithms if time permits and/or we face issues using this one.
- **Introducing a limited number of syntactic changes:** in this scenario, we will generate the adversarial examples ourselves by applying 4 syntactic modifications to toxic words. These modifications are adding space between letters, adding random punctuation in the middle of a word, repeating and swapping letters. These methods were found to fool Google's Perspective AI, a state-of-the-art model in toxicity detection with current industrial applications, which shows the effectiveness of these attacks [4]. [4] also proposes augmenting the training set with such examples as a possible fix to the classifier(s)' weakness, despite the computational cost of this approach.
- **Using human attackers:** [5] proposes a strategy called 'build it, break it, fix it' inspired by security testing of software products. Following this method, an initial classifier M1 is built, achieving as high a performance as possible on a regular dataset. Human attackers are then asked to try to break the model (by giving it a toxic sentence it would not detect). The examples generated by the attackers are added to the training dataset, resulting in a new model M2. In the next round, participants should try to fool both M1 and M2. This strategy goes on for a predefined number of rounds and is reported to improve the classifier's ability to detect linguistically advanced toxic expressions (using metaphors or requiring cultural knowledge for instance). We would use the team members as attackers if we were to implement this strategy. However, since our focus is on syntactically modified examples only, and due to time limitations, this method would be our last resort.

The methods above are in preferential order. If time permits, we might try multiple methods and compare their results. We will report on all our findings either way.

- **Training plan**

As explained in our data collection strategy, our input would be Conversation AI's publicly available dataset made of edit discussion comments from Wikipedia. We plan to use 80% of said data for training, 10% for validation, and 10% for testing. This training strategy is conforming with the guidelines given in the Kaggle competition where the dataset was presented. At an initial stage, we plan to train the model on a regular dataset to match its reported accuracy. We will then assess how the model, without additional training, performs on an adversarial dataset, obtained through one of the methods mentioned in the previous section. This initial accuracy would be used as a baseline to measure any subsequent improvement in the model's resistance to adversarial attacks. We would then retrain our model using a portion of the adversarial dataset and reassess its accuracy. This process will be repeated a predefined number of rounds and all results/observations will be reported adequately.

- **Evaluation**

In all runs of the program, we will use accuracy as a measure of performance. More specifically, we plan to use ROC-AUC measure as it provides a more robust estimation of accuracy. This measure was also proposed as an evaluation metric in Kaggle's competition. Overall, we hope to present an adversarial training method that increases the model's ability to detect adversarial toxic language and mention any observations on the training strategy and/or model we make along the way.

Resources

[1] Kohli, M., Kuehler, E., & Palowitch, J. (2018). Paying attention to toxic comments online.

[2] Pukar Acharya, Toxic Comments Classsification, (2018), GitHub repository, https://github.com/iampukar/toxic-comments-classification/

[3] Gao, J., Lanchantin, J., Soffa & Qi, Y. (2018). Black-box Generation of Adversarial Text Sequences to Evade Deep Learning Classifiers

[4] Hosseini, H., Kannan, S., Zhang, B., & Poovendran, R. (2017, February 27). Deceiving Google's Perspective API Built for Detecting Toxic Comments. arXiv:1702.08138

[5] Dinan, E., Humeau, S., Zhang, B., Chintagunta, B., & Weston, J. (2019, August 17). Build it Break it Fix it for Dialogue Safety: Robustness from Adversarial Human Attack. arXiv:1908.06083

Data Analysis Plan and Tools

Data Preparation Plan

The data being used is already in the format required for analysis. Hence, there is no need to do the preparation as such. Since the data is produced by a third party, we need to familiarize ourselves with the format of the data, i.e. the range of values, know whether data is in tabular form, know if it can be easily processed by a data processing program etc. After tackling these issues, the data must be transformed into numerical format so that it can be processed further. To this aim, word embeddings will be used [1].

Initial Exploration Plan

Initially, the data needs to be explored in order to get an idea about internal format of the data. This can be done using Exploratory Data Analysis (EDA). Since this is a classification problem, information such as the number of classes, number of examples for each class will be useful. This will also help us in the identification of imbalances in number of examples belonging to a category ("class imbalance problem") that can affect the performance of the methods used [2]. At this stage, several visualizations of data on different parameters can be produced to better ascertain the methods which are more suited for the task. This source provides basic summary statistics of our dataset. We plan to generate a similar summary including any specific statistics we deem necessary for our study.

Analysis plan

For analysis, first the baseline model i.e. BiLSTM RNN will be trained and tested using the data [3]. About 80% of data will be kept for training and the rest will be for testing. Since, the aim of this study is to increase the robustness of models. The data will be systematically modified using the various approaches mentioned in the "Data Collection Protocol" section. Then we will train (using transfer learning) and test the model on this modified data and note the ROC-AUC (Receiver Operating Characteristics - Area Under the Curve) score. The ROC-AUC measures the classification performance of classifier. This will be repeated with different approaches of modifying data in adversarial fashion.

Presentation plan

The data itself is stored in *csv* format and can be read by a spreadsheet program. A text explaining the findings of the study will be provided. In addition, graphs and figures for relevant measures such as accuracy, F1-scores, ROC-AUC and the networks will be produced.

Tools

For the analysis, the tool of choice will be the *Python* programming language. It comes with many libraries that are well suited for the study. Specifically, libraries like *tensorflow*, *keras*, and *pytorch* make it easy to prototype models quickly [4][5]. This will be helpful for us because of the time constraints. Other libraries which will be used are matplotlib for presenting graphical information and Pandas for data management and handling. We plan to use a Jupyter notebook for collaborative programming which allows taking screenshots and generating data summaries as needed for our report.

References for this section:

[1] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111-3119).

[2] Japkowicz, N., & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent data analysis*, *6*(5), 429-449.

[3] Chen, T., Xu, R., He, Y., & Wang, X. (2017). Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Systems with Applications*, *72*, 221-230.

[4] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Kudlur, M. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)* (pp. 265-283).

[5] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., ... & Lerer, A. (2017). Automatic differentiation in pytorch.