Data Science Meet: Kolkata

Topic: Kaggle Toxic Comment Identification Challenge

Discussion led by: Rajneesh Tiwari

Kaggle Toxic Comment Identification Challenge

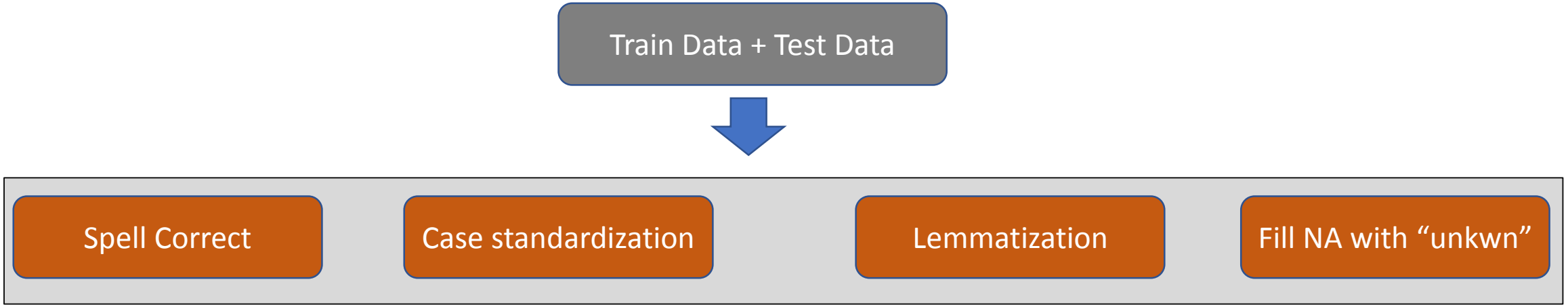
- Discussing things you care about can be difficult. The threat of abuse and harassment online means that many people stop expressing themselves and give up on seeking different opinions. Platforms struggle to effectively facilitate conversations, leading many communities to limit or completely shut down user comments

- The Conversation AI team, a research initiative founded by Jigsaw and Google (both a part of Alphabet) are working on tools to help improve online conversation. One area of focus is the study of negative online behaviors, like toxic comments (i.e. comments that are rude, disrespectful or otherwise likely to make someone leave a discussion). So far they've built a range of publicly available models served through the Perspective API, including toxicity. But the current models still make errors, and they don't allow users to select which types of toxicity they're interested in finding (e.g. some platforms may be fine with profanity, but not with other types of toxic content)

- In this competition, you're challenged to build a multi-headed model that's capable of detecting different types of of toxicity like threats, obscenity, insults, and identity-based hate better than Perspective's current models. You'll be using a dataset of comments from Wikipedia's talk page edits. Improvements to the current model will hopefully help online discussion become more productive and respectful

Source: https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge

Kaggle Toxic Comment Identification Challenge



Steps:

1. Combine test and train data into 1 single data frame
2. Spell correction
3. Convert to lowercase
4. Lemmatization
5. Fill empty/NA with ‘unknwn’ token

Source: Kaggle.com

Kaggle Toxic Comment Identification Challenge

Len of comments	Number of Capitals	Proportion of capitals	Number of exclamation marks	Number of question marks
Number of punctuation symbols	Number of symbols	Number of words	Number of unique words	Number of (happy) smilies
Number of Numeric	Number of alphanumeric	Word Vec (TF-IDF – N Grams)	Char Vec (TF-IDF – N Grams)	

Steps:

1. Create the above features for each comment in both test and train set
2. Features used vary from model to model

Source: Kaggle.com

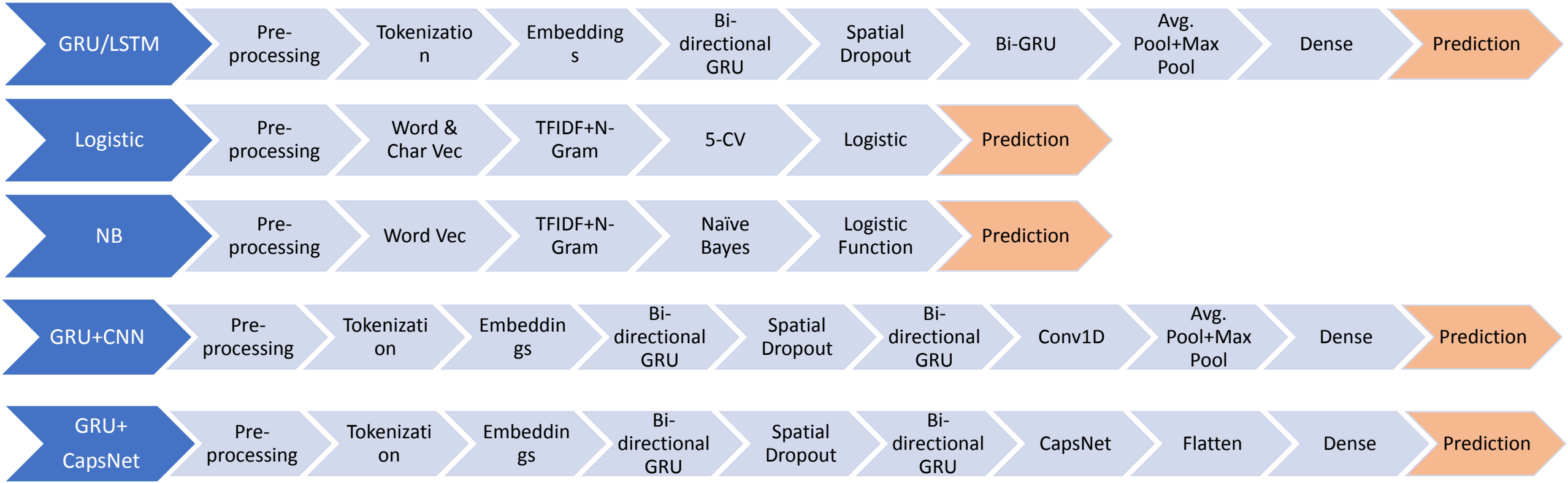
Kaggle Toxic Comment Identification Challenge

Pre-Trained Embeddings used:

1. Glove
2. Fasttext
3. Emoji2vec

Source: Emoji2Vec, Glove, FastText

Kaggle Toxic Comment Identification Challenge



1. Pre-processing is specific to each model and varies from model to model
2. All predictions are 5 fold Cross Validated

Source: CapsNet, CNN, LSTM,NB, FTRL

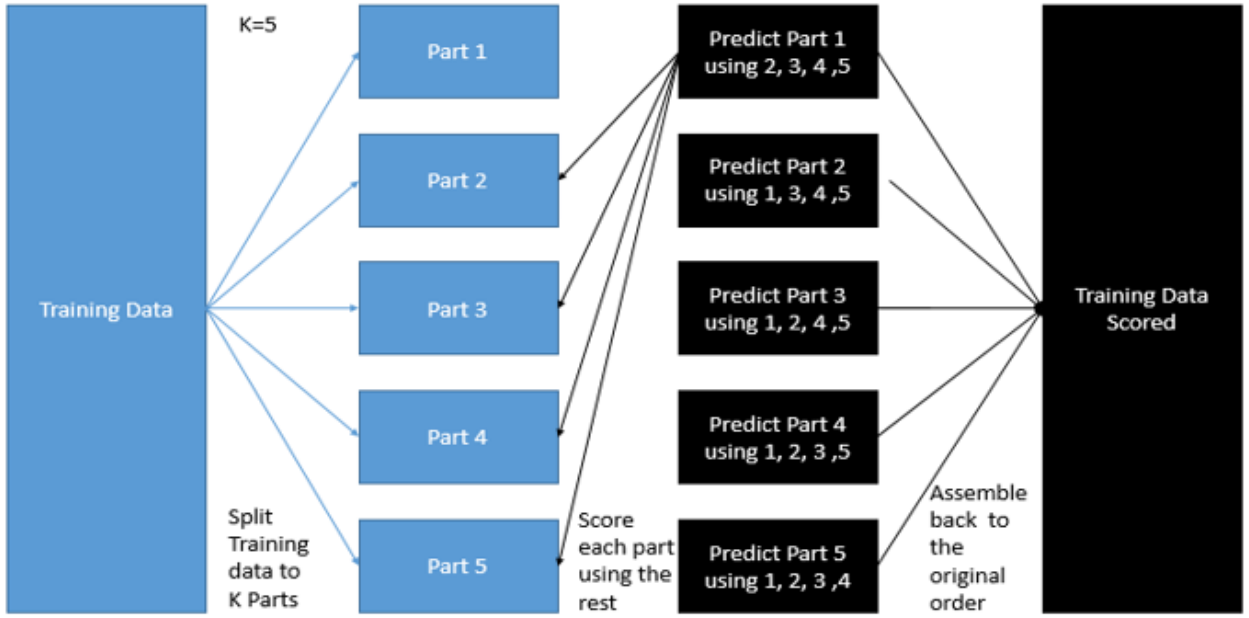
Kaggle Toxic Comment Identification Challenge

Final Submission Approach:

My final submission was based on stacknet as below:

1. 5 fold OOF predictions from Level 1 models as per slide 6
2. Concatenate all predictions for train and test
3. Level 2 models: RandomForests, XGB, LightGBM, FM
4. Meta model: LightGBM & RandomForest
5. Blend multiple iterations of above

StackNet Model Flow consisting of OOF preds as Training data



Source: StackNet

Kaggle Toxic Comment Identification Challenge

Other innovative Approaches:

1. Data Augmentation using Language conversion: Eg: EN → DE → EN for 4 other languages such as DE, FR, ES, CN
2. Test Time Augmentation using above and averaging multiple predictions

Algorithmic Transparency

discussion led by: Suvayu

24th March 2018

Why?

- As ML models have become more complex, our ability to understand them have gone down.
- Sometimes it's necessary due to:
  - unknown biases in data (e.g. societal features like race, gender),
  - could be a legal requirement (e.g. financial or healthcare services),
  - precondition for third-party auditing, or transparency reports.

White box transparency

AI detectives in Science: need deep understanding of the problem, as well as the algorithm. Some techniques:

- *counter-factual* probes: varying the input and observing the change in output to identify "substructures",
- indirectly use black box algorithms to build transparent models (e.g. GAMs)
- embracing black box algorithms completely, and use them on data with explanations augmented, and eventually provide explanations along with output (e.g. the Frogger example in the article).
- etc ...

Algorithmic Transparency via QII[1]

Datta, Sen, & Zick

TLDR? You could skip to section VII, and read just the empirical study.

Quantitative Input Influence (QII) are a class of measures that laydown a transparency framework

- treats the ML algorithm as a black box
- needs complete control over the data (input features/outputs), and the training procedure.

Three primary goals:

1. formalise a class of *transparency reports* that allows us to answer a various *transparency queries*,
2. find measures that can help us distinguish b/w merely *correlated* and truly influential inputs, and
3. *joint influence measures* identify input sets that are influential (but not individually).

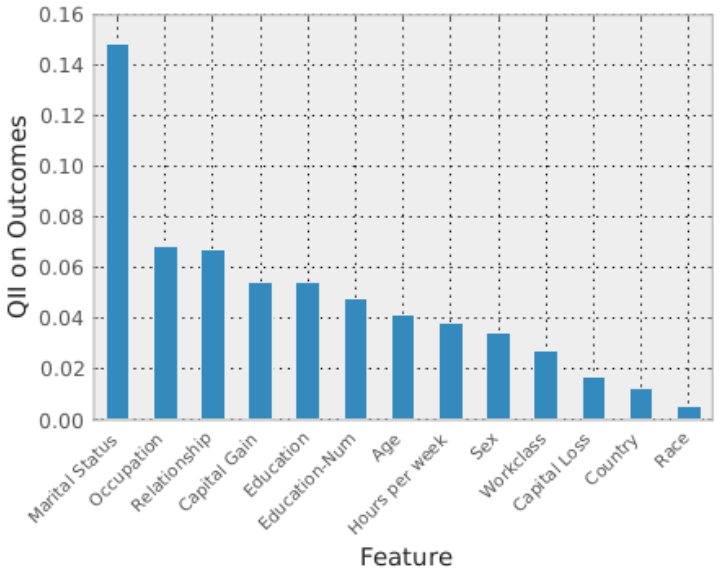
[1] DOI: 10.1109/SP.2016.42

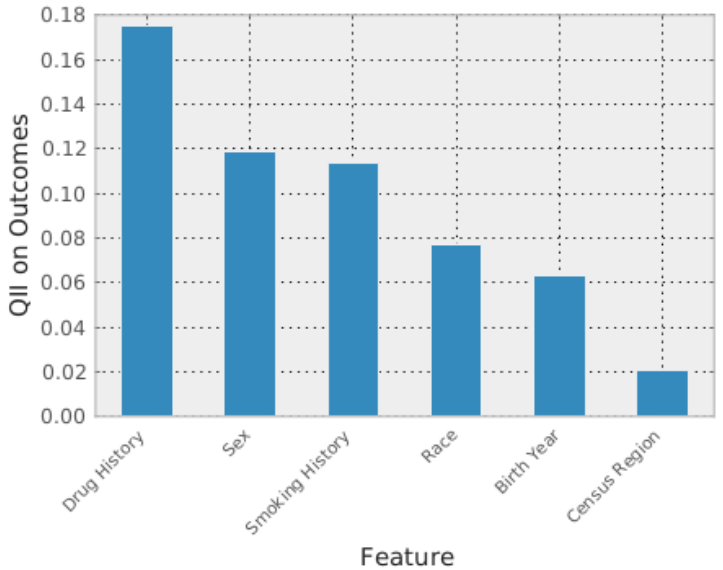
Example datasets

1. adult: subset of US census data with demographic information, and socio-economic factors; use to predict income.
2. arrests: a survey spanning more than a decade where the respondents joined the study in their teens. The features used from this dataset: age, gender, race, region, history of drug use, history of smoking, history of arrests.

Examples

Goal (2)



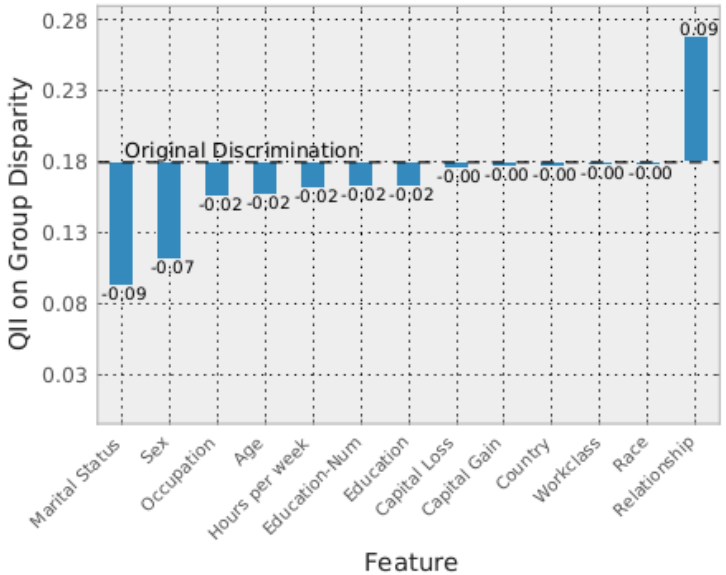


- apply an *intervention* on a correlated input
- replace the input with a random input derived from an unrelated but still consistent with the original distribution.

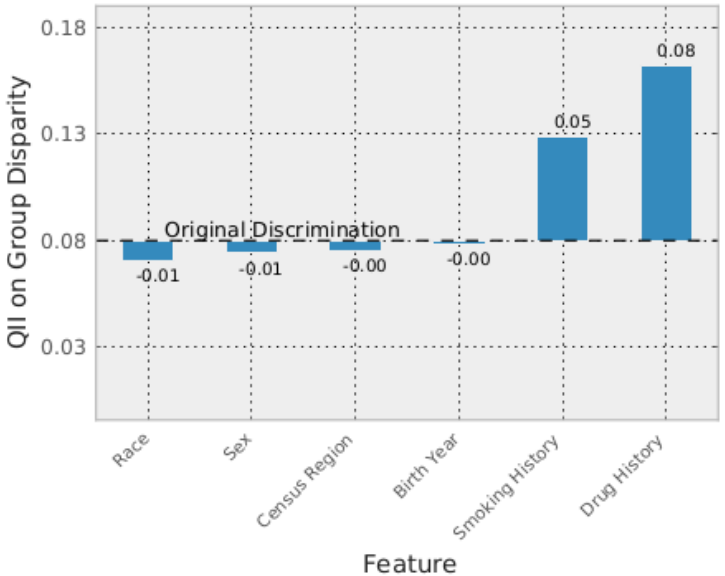
Examples

Goal (3)

(a) QII of inputs on Outcomes for the `adult` dataset



(b) QII of inputs on Outcomes for the `arrests` dataset



group disparity, a measure that tests the sensitivity of the outcome to an input within a set. (left) disparity introduced by considering gender in the adult dataset, (right) disparity introduced by considering race in the arrests dataset.

Remarks on *QII* measures

- very powerful way to probe black box ML algorithms
- probably only appropriate for classification problems (not sure)
- try to replicate the empirical studies yourself!