# Research Summary Document
# Hatebase - AI for Hate Speech Monitoring

The Sentinel Project

30 September 2022

**1.0 - Statement of Purpose** **3**
1.1 - Research Questions 3

**2.0 - Introduction to Hatebase** **4**
2.1 - Origins and Structures 4
2.2 - Technical Elements 4

**3.0 - Data Analysis Process** **5**
3.1 - Data Cleaning 5

**4.0 - Machine Learning Process** **6**
4.1 - Introduction and Explanation 6
4.2 - Breakdown of Processes 6
4.2.1 - Convolutional Neural Networks versus Recurrent Neural Networks 6
4.2.2 - Supervised, semi-supervised, and unsupervised learning models 7
4.2.3 - TensorFlow, Keras, and GloVe 7
4.3 - Walkthrough of Code, Discovery, Analysis 8
4.3.1 - Introduction 8
4.3.2 - Code 9
4.3.3 - Discovery and Analysis 12
4.3.4 - Section Summary 13

**5.0 - Findings** **13**
5.1 - Advantages 14
5.1.1 - Information Processing and Automation 14
5.1.2 - Cross-Compatible Developments 14
5.2 - Disadvantages 15
5.2.1 - Opacity of ML Models and Decisions 15
5.2.2 - High Technical Barrier to Entry 16
5.2.3 - Over-Reliance on Technical Solutions 16
5.2.4 - Maintenance and Retraining Requirements 16
5.2.5 - Fundamental Shortcomings 16
5.2.6 - Large Data Needs 17
5.2.7 - Regionalization, Localization, and Language 18

**6.0 - Summary** **19**

## 1.0 - Statement of Purpose

Hate speech is widely acknowledged as a precipitator of violence, particularly against vulnerable minorities, and a contributor to societal polarization which can lead to instability and conflict escalation. The impact of hate speech on fragile states has risen in recent years as a result of increased internet connectivity, enabling hate speech to be disseminated through social media and contributing on a large scale to persecution, armed conflict, and genocide in various countries. The sheer volume of hate speech constantly circulating online exceeds the capabilities of human moderators, thus resulting in the need for increasingly effective automation. The pervasiveness of online hate speech also presents an opportunity since these large volumes of data are potentially useful as indicators of spiraling instability and introduce the possibility of early intervention. The purpose of this research is to provide a technical explanation of the processes involved while still attempting to make the material digestible for an audience with limited or no technical experience in the subject.

We have selected the most accessible methods and tools to demonstrate the broadest possible utility with an emphasis on existing tools rather than building entirely from the ground up. Additionally, we will reference publicly available code repositories that we have produced during the course of the project.

### 1.1 - Research Questions

The primary research question being addressed here is the following:

> *How can automated hate speech monitoring be improved to reduce the need for human moderation and, specifically, in what ways can artificial intelligence and natural language processing be most efficiently employed for this purpose?*

In order to approach this question, we developed a machine learning algorithm that replaces the existing Hatebase software for hate speech monitoring. This new software will be based on machine learning algorithms which can be trained with existing data. This training will allow the model to learn complex linguistic features and parse difficult contextual information. The training will also enable the software to better parse new data and remove the need for a human to code all possible linguistic rules in order to generate a reasonably accurate output. Beyond that, the training model is able to detect subtle data trends which may be nearly invisible to human observation, though this does come with the complication of making the generated outcomes more difficult for a human moderator to re-trace and evaluate.

# 2.0 - Introduction to Hatebase

## 2.1 - Origins and Structures

In April 2013, the Sentinel Project launched Hatebase, a software platform which combines automation and crowdsourcing to monitor online hate speech. Individuals are able to report vocabulary with malicious intent that they have encountered along with descriptions including the targeted population and locations where the term has been used. Hatebase then uses this data to independently identify sightings of online hate speech and is now monitoring over 2,300 hate speech terms in more than 90 languages across over 175 countries. The software was designed to help government agencies and local NGOs monitor regional, time-delimited incidents of hate speech for the purposes of (a) efficiently triaging resources to geographically disparate areas and (b) acting as a potential early warning indicator of violence. Hatebase has accrued a large dataset of multilingual vocabulary and incident sightings, has been used by hundreds of public and private entities working on a variety of projects, and has been used to support research on hate speech at Harvard University, MIT, the University of Waterloo, and several other notable institutions.

Hatebase has acquired over 2,000,000 data points, resulting in approximately 1,200 units of vocabulary and approximately 650,000 sightings on social media platforms.[1]

Hatebase was developed to be “the world's largest online repository of structured, multilingual, usage-based hate speech” and it adheres to the following principles:

1. Hate is defined by intent, not by vocabulary. In other words, (a) innocuous vocabulary can be hateful, and (b) hateful language can be discussed clinically and without intent to shock, offend, or harm.

2. The problem of hate speech is not solved by censorship. This does not mean that individuals and governments should behave irresponsibly in volatile situations. However, the perils of censorship and curtailed individual liberties rival those of hate speech.

## 2.2 - Technical Elements

Before undertaking this research project, collection of our automated sightings drew largely from Twitter because (a) it is a large open dataset and (b) we are able to creatively geocode about 40% of all tweets we retrieve. There are two modules that currently process our tweets: HateBot interacts with the Twitter API and retrieves tweets which appear to contain our hate speech vocabulary (cycling through one term every ten minutes). HateBrain then takes over and parses those tweets, applying some rudimentary language rules to determine if the context of the hate

[1] A unit of vocabulary consists of one root word; a sighting is one social media post containing a vocabulary unit and the sentiment of the post indicates the vocabulary is being used malignantly rather than clinical usage, cordially or self-referentially.

speech is correct and, if it can make a reasonably definitive decision (which it can about 40% of the time), it saves or deletes the data point depending on whether it meets this threshold.

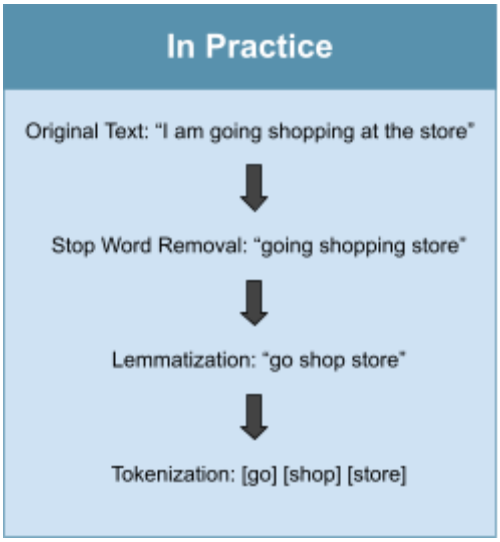
This system is called a *rule-based algorithm* because the parameters to parse the data must essentially be hardcoded before any processing can take place. Furthermore, it is not able to learn from ongoing inputs and can only be modified through direct human intervention to revise the ruleset. In addition, it is important to note that this process is capturing hate speech data points based solely on context, not on intent. For this reason, the rule-based algorithm is unable to adequately collect data in accordance with the first principle listed above, and is instead only able to infer intent indirectly.

The software development part of this research project aimed to develop an enhanced hate speech detection algorithm by utilizing machine learning models which adapt to datasets and are able to discover subtle patterns within large datasets which may be imperceptible to human moderators. These algorithms are also able to incorporate sentiment analysis to better situate text within specific contexts for added clarity when making a determination.[2]

## 3.0 - Data Analysis Process

### 3.1 - Data Cleaning

**Stop words -** In almost every language there are numerous words or terms that are not strictly necessary for the comprehension of a sentence. These words are often very common but generic, and are not important for understanding an idea being conveyed through language. Examples of English stop words include “the”, “is”, “a”, and “so”. Because these words are not essential to comprehension, most data processing removes these terms before any analysis takes place.




**Lemmatization -** Due to the complexity of language, many words are modified versions of a root or base term. In linguistics, this base term is called a *lemma*.[3] Lemmatization is the process of grouping together words with a shared lemma so that they can be analyzed as an individual unit. An example of lemmatization would be determining the grouping of words like “multiplication”, “multiplier”, and “multiply” to

[2] Sentiment analysis is the identification, extraction and quantification of subjective information within human expression. This is most common in the evaluation of spoken or textual statements.
[3] The canonical, dictionary, or citation form of a set of words.

the base lemma of “multiple”. For more advanced language interpretation this step may be reduced in order to maintain the nuance of each vocabulary item.

**Tokenization -** Computers do not inherently recognize words and language and so they must be broken down into elements that a computer can understand. *Tokenizing* text involves separating it into distinct components such as words, word roots, and characters. These values are stored in computer memory as pieces of information unique to that component. A computer can more easily apply calculations and analysis to these components than to larger sets of data.

Though the sentence has been drastically simplified, the essential meaning is maintained and stored in a format that is easier for a computer to analyze. This simplified and consistently formatted dataset can now be passed to machine learning algorithms for processing.

# 4.0 - Machine Learning Process

## 4.1 - Introduction and Explanation

Artificial intelligence (AI) is the broad subject of computer emulation of human thought and intellectual functioning. Machine learning refers to the processes and methodologies that facilitate this emulation.

Artificial neural networks (ANNs) are the primary focus of this research. ANNs are based on the principle of biological neural networks which form connections which permit learning and understanding. Organized as layers of artificial neurons, data is processed and assigned a weight and significance of association and these neurons essentially remember this link. This process is repeated across the breadth of a dataset and through repeated layers of association. The output of this neural network is a complex model which has developed extensive connections between the many data points. In this way it can be understood that the neural network has begun to grasp the rules within the data without those rules having to be explicitly defined.

## 4.2 - Breakdown of Processes

### 4.2.1 - Convolutional Neural Networks versus Recurrent Neural Networks

Convolutional neural networks (CNNs) work by processing a data point within a larger dataset and condensing a set of data points into a digestible pool of its more important elements. The neural network will then essentially build up an understanding of the dataset through each iteration, gaining a more detailed comprehension of the data with each iteration. This process allows for the neural network to understand inputs on a very basic level but with a high degree of accuracy. The shortcoming is that it learns primarily from those external inputs but does not have the inherent capability to reassess data within the model itself. If we think about how

humans learn we can broadly categorize this learning into two main types - classification and understanding. CNNs excel at classification through external inputs, such as identifying an object, but they fail to understand a lot of important context. CNNs may be able to classify an image as containing a fire but still struggle to recognize the context of the fire such as whether it is near or far, or whether it represents a danger.

This is where recurrent neural networks (RNNs) factor in. Unlike CNNs which pool layers of the most relevant data and refine these layers to build up a model, RNNs cycle through data and generate a map of this data which is then tested against the next cycle of analysis. In this way an RNN generates a sort of memory which allows for a more nuanced understanding of data. Temporal and contextual changes in data inform the neural network of subtle variations which permit it a depth of comprehension which allows additional insights. To extend the fire example above, where CNNs can classify the image of a fire properly, RNNs are capable of identifying the fire and also understanding the context in which the fire appears, which is significantly more useful.

#### 4.2.2 - Supervised, semi-supervised, and unsupervised learning models

In machine learning, models must be developed using existing datasets. There are several approaches to building these models which vary depending on the available data and the objective of the application.

**Supervised learning** involves feeding a machine learning algorithm with clearly defined data points that have been classified and labeled. The algorithm is then trained by building a model around this labeled data. Through the training process, the algorithm is able to establish what characteristics of the data contribute to it being labeled as it was and can therefore take new data points and interpret them within this structure.

**Semi-supervised learning** follows a similar process but not all the data is labeled. This is often because there is a large dataset but only a few points will have been labeled, often due to circumstances which challenge the ability to clearly label all data points. Semi-supervised learning therefore acts to generate rules for labeling the existing unlabeled data as well as interpreting the appropriate labeling and context of future data.

Finally, **unsupervised learning** involves using a dataset which is entirely devoid of labeling and classification. The purpose of this method is not to precisely classify new data but rather to find subtle connections, measure the strength of associations, and discover meaningful insights in datasets which may not be apparent to or easily observed by human examination.

#### 4.2.3 - TensorFlow, Keras, and GloVe

**TensorFlow 2** is an open-source software library developed to formalize deep learning and neural network tasks. Its primary value is bundling a large suite of tools and functions into a

single package to streamline the development of machine learning algorithms. It is possible for researchers and developers to quickly and easily build machine learning models without having to develop the core tools themselves. TensorFlow 2 also integrates with a wide variety of programming languages and coding environments which make it greatly accessible to a broad range of users.

**Keras** is a software library that allows researchers to interact with the TensorFlow 2 library through the Python programming language. The Keras library and Python language provide an excellent platform for rapid development of neural network systems in coherent and human-readable[4] formats. It is within this library that the majority of the code base and parameters are set for machine learning model building.

**GloVe** is a pre-trained word vector which serves to guide the machine learning algorithm in understanding the relationship between language tokens.[5] This matrix of 6 billion language tokens covering a lexicon of 400,000 separate English words enables researchers and practitioners to develop machine learning algorithms without having to build their own massive data vectors first. It is on top of this dataset that custom datasets with application specific terminology can be modeled.

## 4.3 - Walkthrough of Code, Discovery, Analysis

### 4.3.1 - Introduction

For this application, we have generated a custom dataset consisting of 600,000 total words representing approximately 45,000 social media posts. Half of these posts have been labeled as hate speech and the other half have been rated as normal speech. Given the task it is designed around, there is an inherent shortcoming with this dataset since it is relatively small. This is a fundamental challenge with regards to model training when larger datasets can assure a higher degree of accuracy but such datasets often require their own significant effort to collect, classify, and process. However, as this research project is meant to demonstrate the applicability of AI in monitoring for hate speech the dataset is sufficient to exhibit the core concepts behind the principle.

[4] "Human-readable" refers to an encoding of data or information that can be read by humans in a coherent way. This generally involves language parsing that is similar to human grammar and syntax.

[5] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation.

#### 4.3.2 - Code

The Github repository containing the code produced by the project can be accessed here: https://github.com/thesentinelproject/Hatewise

Before examining the structure of the machine learning model itself, we must first clarify a few terms which arise within it:

> **Long short-term memory (LSTM)** - A highly flexible recurrent neural network used for this research application. Broadly speaking, it operates by processing data sequences and adjusting weights and biases between data points over consecutive training periods by using self-referential operations.[6] This action prevents the recurrent data cycle from drifting uncontrollably due to statistical anomalies.
>
> **Adaptive moment estimation (ADAM)** - An optimization algorithm that efficiently calculates neural network weights and biases which allows for the processing of large datasets with relative accuracy over many training cycles and with acceptably minimal computational requirements.
>
> **Categorical cross-entropy** - A function used to measure the difference between two probabilistic distributions within a data series. In this project, categorical cross entropy determines the gulf between training iterations in order to prioritize statistically significant associations between data points.
>
> **Softmax** - A process that calculates values generated by categorical cross-entropy and advances large values while discarding statistically less significant values. This loss rate is the portion of data which is considered not useful to further training within that cycle.

The algorithm works in three stages. The first stage is to clean the dataset by standardizing formatting, removing stop words, tokenizing the data, and loading the categorized data into a machine-readable format.

```
def load_data(num_words, sequence_length, test_size = 0.50, oov_token = None):
   # Read data from dataset
   dataterm = []
   with open("dataset.txt", encoding = “utf-8”) as f:
       for term in f:
           term = term.strip()
           dataterm.append(term)
```

[6] Self-referential operations not only analyze new data but continually analyze previous versions of analysis from the same data model. In human learning this would be akin to periodically reviewing current knowledge and using this to better understand new concepts as they are learned.

```
    labels = []
    with open("dataset_labels.txt") as f:
        for label in f:
            label = label.strip()
            labels.append(label)

    # Text processing (tokenize words, remove stopwords)
    tokenizer = Tokenizer(num_words = num_words, oov_token = oov_token)
    tokenizer.fit_on_texts(dataterm)
    x = tokenizer.texts_to_sequences(dataterm)

    x, y = np.array(x), np.array(labels)

    x = pad_sequences(x, maxlen = sequence_length)
    y = to_categorical(y)

    # Divide data into training and testing subsets
    x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = test_size,
    random_state = 1)

    data = {"x_train": x_train, "x_test": x_test, "y_train": y_train, "y_test": y_test,
    "tokenizer": tokenizer,
            "int2label": {0: "normal speech", 1: "hateful speech"},
            "label2int": {"normal speech": 0, "hateful speech": 1}}

    return data
```

The second step is to create and run the training model using the LSTM neural network. As it runs, the recurrent neural network compares all data points and assigns weights and biases between them. The model compares data points within a training cycle and between different training cycles in order to develop an understanding of patterns within the larger dataset.

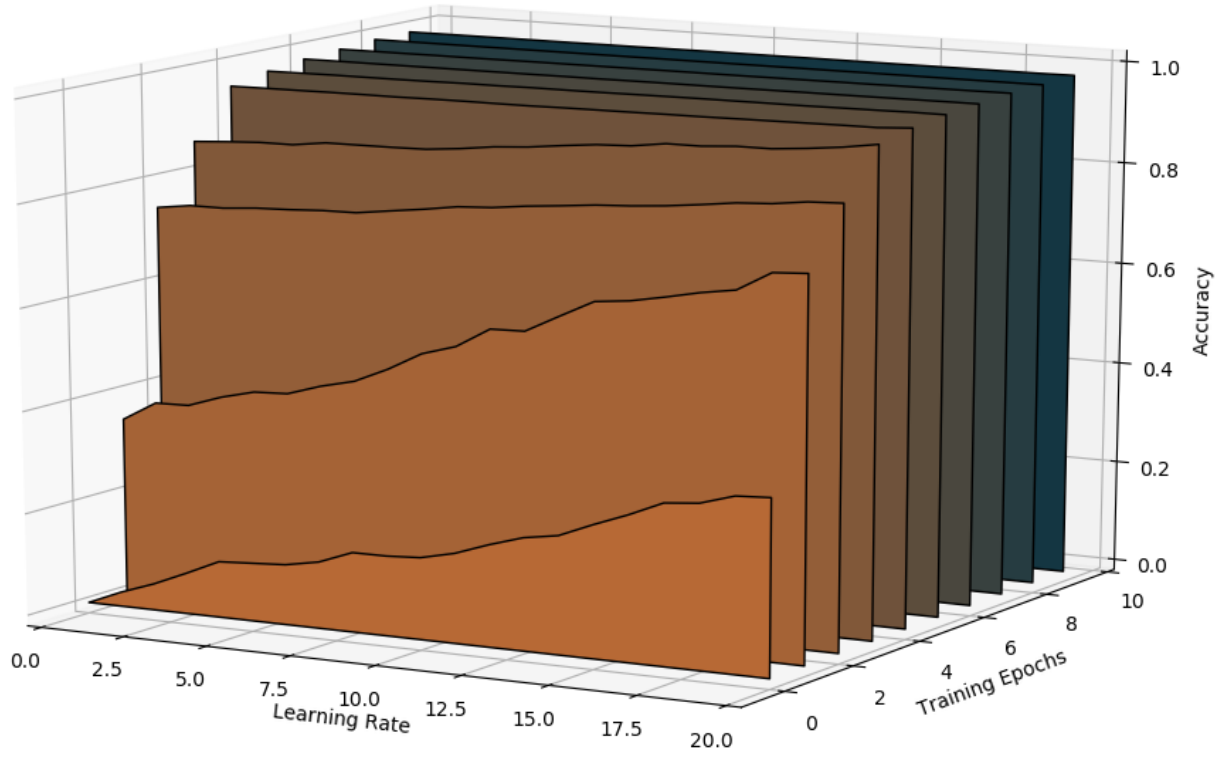
```
def create_model(word_index, units = 128, n_layers = 1, cell=LSTM, bidirectional=False,
                 embedding_size = 100, sequence_length = 100, dropout = 0.3,
                 loss = "categorical_crossentropy", optimizer = "adam",
                 output_length = 2):

    embedding_matrix = get_embedding_vectors(word_index, embedding_size)
    model = Sequential()
    # Created new embedded layer
    model.add(Embedding(len(word_index) + 1,
              embedding_size,
              weights = [embedding_matrix],
              trainable = False,
              input_length = sequence_length))

    for i in range(n_layers):
        if i == n_layers - 1:
            # Add final layer
```

```
                model.add(cell(units, return_sequences = False))
            model.add(Dropout(dropout))

        model.add(Dense(output_length, activation = "softmax"))
        model.compile(optimizer = optimizer, loss = loss, metrics = ["accuracy"])
        return model
```




*Model learning rate accuracy over successive training epochs*

The third step involves testing the newly trained model by introducing data points which were not part of the training data and gauging its prediction accuracy. The model applies the new data to the trained model and generates a value output by calculating all the tokenized parameters. In addition to making a broad classification it also provides a confidence estimation which quantifies the certainty with which it made the prediction.

```
def predictions(text_data):
    sequence = data["tokenizer"].texts_to_sequences([text_data])
    sequence = pad_sequences(sequence, maxlen = SEQUENCE_LENGTH)

    # Process prediction model output
    prediction = model.predict(sequence)[0]
    return data["int2label"][np.argmax(prediction)], prediction

text = "sample text"
prediction = predictions(text)
```

```
classification = prediction[0]
if classification > 0.5:
    conf = round(prediction[1][1] * 100, 2)
else: conf = round(prediction[1][0] * 100, 2)
log.info(f"This has been rated as {classification} with {conf}% confidence.")
```

**Output:** “This has been rated as normal speech with 88.2% confidence.”

### 4.3.3 - Discovery and Analysis

With the provided model it is possible to begin analyzing the accuracy and efficiency of the algorithm, as well as discover any potential shortcomings or error states.

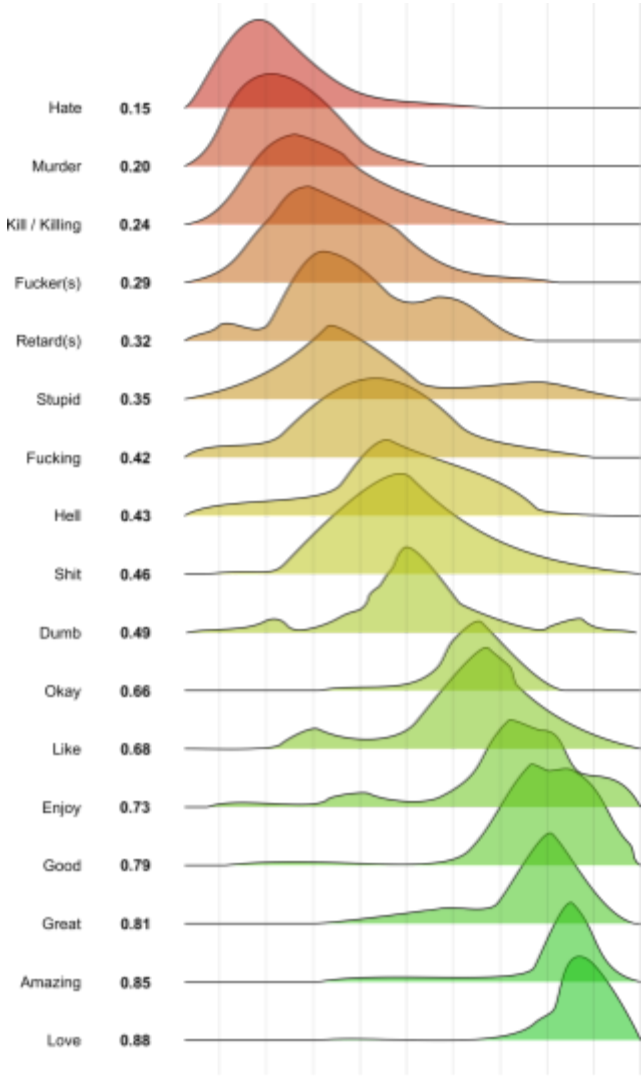



*Figure 1 - Sentiment reference indexing*

The first observation is that the model is generally more accurate with a long text string data input. A fully formed sentence of greater than 100 characters and proper spelling provides the best accuracy rate while messages shorter than 20 characters and comprising less than three words struggles to accurately predict sentiment and context. Though this is understandable and even human comprehension of such short comments may also be limited, this challenge means that the model has more difficulty detecting brief and terse hate speech (which is very common, especially on social media platforms) and may misclassify normal speech or miss hate speech.

There are some notable anomalies which are likely the result of an insufficient dataset, which itself is a difficulty facing machine learning efforts to detect hate speech. An obvious gap discovered while testing the model was that there were issues in being able to distinguish between singular and plural occurrences of the same noun.[7] This anomaly is more common the less extreme the term is which indicates that this may be the result of difficulty with context where a term has an inoffensive or reclaimed meaning as well as a highly vitriolic meaning. Here the

[7] For example, the singular use of a noun generally should not cause a large statistical deviation in rating and confidence estimations and yet some instances of these were observed. The increase of occurrence of this anomaly with the decrease in severity of terminology indicates a blind spot in the learning model with regards to benign speech.

lack of a sufficiently robust dataset means that the model has not had enough breadth of data points to understand the links between vocabulary and the various contexts it is used in. Importantly, tracing the source of this anomaly is incredibly difficult, because the machine learning model uses stochastic methods to compare data and therefore the source of this shortcoming lies within an enormous and nearly incomprehensible dataset of value associations between vocabulary.[8] It is not a simple matter of tracing where the model went wrong and therefore only educated guesses can reasonably be made as to the cause of the anomaly.

Lastly, the model is able to interpret a great deal of context even with a limited dataset but experiences functional losses when new data inputs do not match the content, style, or format of existing training data. This is to say that if the model is trained on a dataset containing regionalized and modern colloquial language it will return sufficiently accurate assessments. If new data inputs differ markedly, such as, for example, with long form text using drastically different regional linguistic characteristics, the model can only account for this to a small degree and only to the extent that the different input data may overlap with the existing model training. Beyond that, the model will need to be retrained to account for these differences and this requires that work be done to identify potential shortcomings in the machine learning model before such regional differentiation can be undertaken.

#### 4.3.4 - Section Summary

Ultimately these models demonstrate that with a modest dataset they are able to estimate context and evaluate content for occurrences of hate speech but that there are known limitations. In addition, there are unknown limitations which are the result of the opaque training processes involved with generating the models themselves.

Nonetheless, the models themselves exhibit a reasonably high probability of accuracy with input data that is of a similar format to the training dataset and can serve as useful filtering tools for monitoring large datasets. The data needs required to create a truly robust and flexible monitoring tool are beyond the scope of this research and indeed, beyond the current reach of many practitioners who may wish to apply such technology.

## 5.0 - Findings

This research project has produced several interesting findings that require further contemplation. Though there are distinct benefits of machine learning in hate speech monitoring, they do come with several significant shortcomings and must be weighed against the notable disadvantages.

[8] Stochastic modeling relies on random probability distributions to analyze and weigh elements, in contrast to progressive modeling which builds proficiency in a linear manner.

## 5.1 - Advantages

### 5.1.1 - Information Processing and Automation

One of the most obvious advantages to machine learning algorithms in humanitarian and development applications is the ability to process large amounts of data before human involvement or moderation is necessary. These models, once properly trained, can be utilized to review, analyze, and prioritize enormous volumes of data for which human involvement would be highly inefficient. Because of its wide variety of applications, from textual analysis to image examination, machine learning is a tool which can amplify human efforts and refine appropriate responses.

Two significant caveats must be included in the discussion of automating information processing. The first is that an algorithm must be properly trained on the dataset and tuned to acceptable margins of error. It may also require periodic retraining if the nature of the initial training dataset becomes too dissimilar to the target dataset. Failure to calibrate the machine learning models risks generating wildly inaccurate outputs.

The second caveat is that machine learning models for information processing and automation cannot be viewed as a substitute for human involvement. Indeed, such tools find ideal applications where their capabilities exceed human abilities (such as pure data volume processing) or where human involvement can be shown to be more error prone. They are best thought of as a filtering tool or for broad data analysis rather than as a substitute for human involvement entirely.

Lastly, even the most advanced machine learning algorithms available currently do not show sufficient capability to be trusted in mission critical roles. The tools available to most potential users in the humanitarian and development fields fall far below the cutting edge and must be viewed even more cautiously. Vital data pertaining to immediate risk, the provisioning of services or support, health, and security, among other tasks, should not be reliant on machine learning systems except in secondary support roles.

### 5.1.2 - Cross-Compatible Developments

Another clear benefit of machine learning in the humanitarian and development fields is that advances made in ML/AI within other fields does have cross-compatibility and can be applied to new contexts. Because these developments are not solely dependent upon specific data types the innovations found within the broader machine learning discipline can be utilized elsewhere. This means that these advances do not necessarily need to be supported by humanitarian and development funding which reduces potential obligations of funding organizations.

## 5.2 - Disadvantages

### 5.2.1 - Opacity of ML Models and Decisions

One of the core principles of human organization is the ability to interpret structures and concepts. From very basic ideas like how a hierarchy works to intensely complicated scientific theories, human organization (and by extension, comprehension) relies on the ability to understand and explain these concepts in a coherent way. Plainly speaking, humans struggle to accept things that are unintuitive. This is where a key shortcoming of machine learning comes into play.

Machine learning is built largely on the idea of stochastic learning, which means it develops a model of relationships through random probability distributions. Essentially, it will process large amounts of data with no specific guidance and generate weighted probabilities of the associations between different data points. It does not learn as a human might, by progressing through degrees of comprehension before reaching a level of proficiency. Because this learning process is so fundamentally different from human learning, and because the associations and weights assigned to these associations are generated en masse and with an opaque path, its outputs are equally difficult for humans to retrace and understand the nature of.

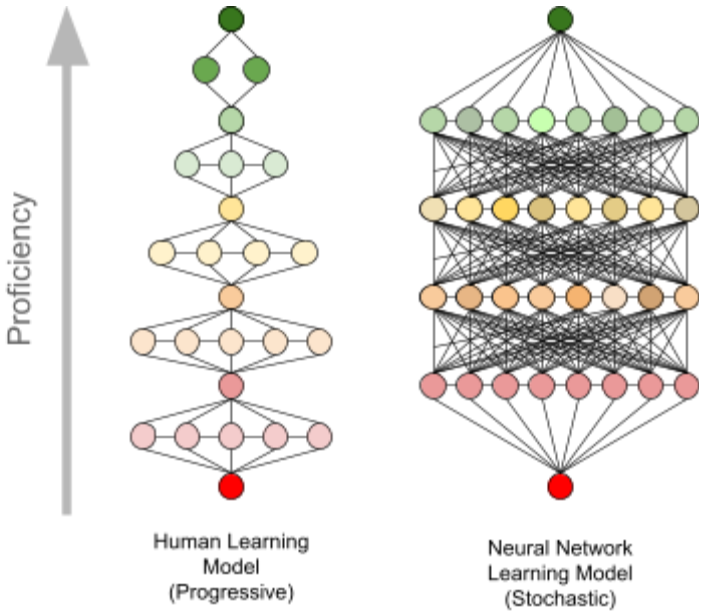



Figure 2 - Complexity of progressive and stochastic models

As such, machine learning models exhibit significant opacity when it comes to understanding why they have reached a specific determination. If a particularly serious error in output is detected, it is difficult or impossible to isolate precisely where the error occurred within the model with the aim of remedying it.

For this reason, machine learning models have often been referred to as “black boxes” which generate outputs that cannot be fundamentally substantiated. One must ask then what is the ultimate value of a process which produces data which is not essentially trustworthy.

### 5.2.2 - High Technical Barrier to Entry

The platforms, tools, and software used for this research project were selected with ease and accessibility in mind. Cutting edge AI programs and machine learning initiatives therein are the realm of large and well-resourced governments, corporations, and educational institutions. For more modest efforts, the fruits of those groundbreaking projects have thankfully generated a plethora of tools which make the subject more accessible.

However, even with the benefit of a vastly simplified toolset the technical requirements for undertaking this kind of work are steep and out of reach for a large majority of organizations which may want to investigate machine learning and AI.

Any interest in utilizing machine learning for development and humanitarian purposes must take these requirements into account since the benefits may be quickly swamped by the many obligations necessary to facilitate such an initiative.

### 5.2.3 - Over-Reliance on Technical Solutions

As noted above, machine learning algorithms currently available are not proficient enough to replace human action for many tasks. Pairing this with the tendency in the humanitarian and development field to focus too heavily on the novelty of new technical concepts, there is a significant risk of a *tool* such as machine learning being implemented as a *solution* to a problem. AI and machine learning are best thought of as one element of a larger body of knowledge but the allure of these cutting edge notions may overpower the wiser application of them.

### 5.2.4 - Maintenance and Retraining Requirements

Like many other systems, even machine learning realistically requires consistent management. This not only necessitates the retention of staff capable of handling these tasks; it also requires maintenance of the systems themselves. As machine learning models are based on datasets, if the training dataset becomes detached from the target dataset by some manner of *genetic drift* the model can become deprecated. In the context of this research project, the machine learning model developed to detect hate speech will become increasingly unsuitable as language itself shifts.

### 5.2.5 - Fundamental Shortcomings

Even state-of-the-art AI systems operate on a low *intellectual* level and can display fundamental myopias as a result. They excel at interpreting large volumes of data, detecting trends that may be too insignificant to be identified by human perception, and developing weighted associations between disparate data points. However, at no level do these systems intrinsically *understand* data to an extent approaching human comprehension. The machine learning model developed

for this project can, with reasonable accuracy, determine whether a specific block of text is hateful or not but it does not do so as the result of a comprehensive understanding of the human nature, language, and culture from which they originate. It has accomplished this uncanny feat by quantifying a large dataset of previous examples and drawing an untold number of links between data points. It does not articulate why some text is hateful and other text is not, just that the target text meets a threshold that matches a pattern in the training data. Though this can in some ways mimic parts of human learning, it is a far cry from matching it.

For this reason, there remains a foundational shortcoming in existing machine learning models which cannot equal human functions. For applications such as essential language interpretation, this is an area in which humans still excel (and even then, often with some difficulty) and cannot be reasonably removed from the process. At best, current technology enables AI applications in conducting high-level surveys of large datasets to identify items for further human investigation.

### 5.2.6 - Large Data Needs

In order to train machine learning models there is a necessity to supply enormous amounts of quantifiable data and ideally data which has been clearly categorized. Machine learning algorithms attempt to parse this data by drawing associations between all of the points within it. Though there are ways to integrate sparser datasets into machine learning algorithms, a general rule is that more data allows for the possibility (but not a guarantee) of a better model.

As an example, if you are provided with a graph containing a single data point it is virtually impossible to extrapolate the next data point. If you are given two data points then you may have more data but extrapolation is still extremely difficult. As you add more data points to the graph it becomes easier to detect statistically meaningful trends within the larger dataset.

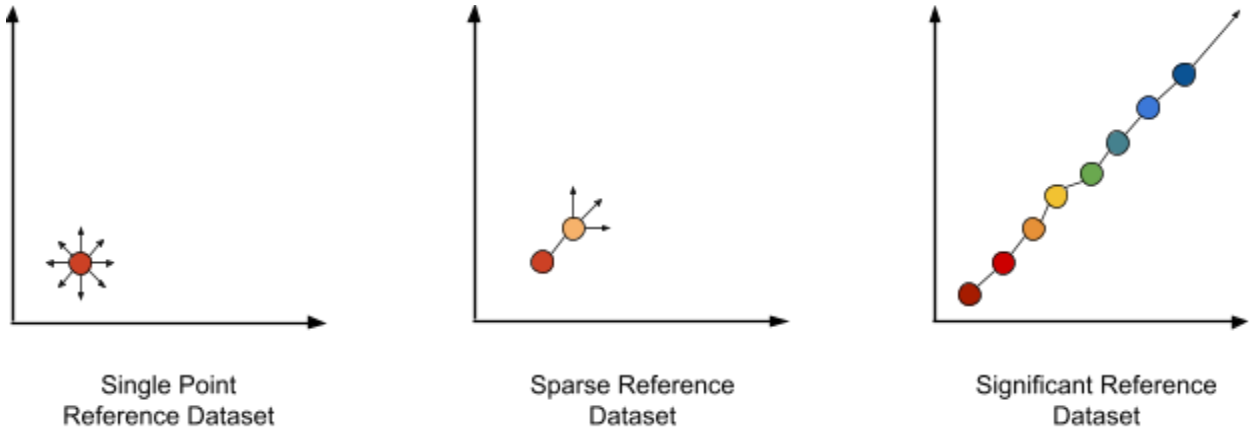



Figure 3 - Extrapolation ability from data points

Within the humanitarian and development field there is often a significant shortage of data compared to many other fields, owing largely to the nature of the work being done. Scientific and technological endeavours generate data as a matter of course where more pragmatic fields do so as a secondary function. Though this is starting to change with the increasing digitization of humanitarian and development work, the dearth of meaningful data presents a significant disadvantage.

### 5.2.7 - Regionalization, Localization, and Language

Another fundamental data consideration that must be taken into account relates to how language interacts with culture. This can be as narrow as the influence of technological platforms and as broad as the structure of language itself.

Most obviously, a machine learning to detect patterns in language must be trained on a dataset consisting of that language. For that reason, it is necessary to compile a sufficiently large dataset which is reflective not only of the language itself but also extensive examples of how certain trends such as hate speech are formulated within the language. For this, human involvement with expertise on the language is necessary in order to parse the breadth of data to select for statistically significant criteria. This can be somewhat automated in terms of collection but it still requires the human touch before data training can even begin.

There are far more subtle ways in which language requires contextualization. Regional distinctions in language may be significant enough to result in less accurate models even though the base language is the same. Furthermore, distinct cultural, linguistic, and socio-economic factors influence language to a point that is detectable by humans but not necessarily by existing machine learning tools. The ambiguity that is introduced when such variables are not taken into account can leave invisible weaknesses in the model which are not easily traced and corrected.

Lastly, the methods by which ideas are disseminated also shape language. Written language may differ significantly from spoken language and there can be technical or arbitrary limitations placed on language depending on the platform being used to communicate. For example, the short message service (SMS) that has become a ubiquitous part of modern telecommunications had established character restrictions which resulted in an abbreviated form of communication on those platforms in order to convey ideas that fit within the parameters of that platform. Even the manner by which individuals are presented may have an influence on the content they generate; an anonymous user may be inclined to convey ideas differently - or convey entirely different ideas - than if that individual were identifiable.

It is therefore crucial that these considerations be examined and taken into consideration when developing not just automation tools but the data which underpins their creation, efficacy, and output.

## 6.0 - Summary

This research has shown that automation of hate speech monitoring can be improved to reduce the need for human moderation. We have demonstrated that by using more readily accessible machine learning tools it is possible to develop an algorithm that does not require a prohibitively high level of technical expertise while still having consistent accuracy in detecting hate speech. The most efficient method of employing AI for hate speech monitoring is as a high-level detection tool rather than a central element of content moderation. This is because the inherent myopia of machine learning models can result in significant shortcomings in overall evaluation.

Where AI does excel is in pure information processing and automation since a machine learning algorithm is able to analyze enormous amounts of data and detect subtle patterns which may escape human observation. Another indirect advantage is that broader developments within the AI field and machine learning subtopic can be applied to hate speech monitoring even if the advances are not directly related. This is because AI is fundamentally about working with data associations and progress made in one field may still generate meaningful applications in other fields.

Caution must be exercised when looking to apply AI to hate speech monitoring too intensely. The outputs of machine learning models are notoriously difficult to re-trace and anomalous results are common. They require large datasets to train properly and a great deal of attention must be paid to properly building the training dataset so that it reasonably reflects the sort of data one might expect to be monitoring. This includes having not just linguistic support but also data that reflects the nuances of expression and localization.

It is important to understand that AI and machine learning, while new and exciting fields of study, are still largely experimental in practice. Recent developments and the rapid increase of computing power have allowed for enormous advances in complexity, accuracy, and data processing but they exhibit fundamental shortcomings and require consistent maintenance and retraining. Though tool sets have been developed which significantly lower the barrier to entry, the technical skills required to meaningfully implement machine learning systems in the humanitarian and development field - especially with respect to hate speech monitoring - still remains prohibitively high for most organizations that might wish to apply it.

For this reason, organizations that are able to overcome the technical limitations would be best served by using AI as a high-level filter to survey a large dataset for instances of hate speech rather than as any kind of substitute for human moderation.