**CLASSIFICATION OF DISASTER-RELATED TWEETS WITH FEW-SHOT LEARNING USING GENERATIVE PRE-TRAINED TRANSFORMER**

**BY DIVYAPRAKASH DHURANDHAR**

**A thesis submitted to the School of Graduate Studies**

**Rutgers, The State University of New Jersey - New Brunswick**

**in partial fulfillment of the requirements for the degree of**

**Master of Science**

**in**

**Electrical and Computer Engineering.**

**Written under the direction of Professor Maria Striki and approved by**

**New Brunswick, New Jersey**

**March 2022.**

**ABSTRACT OF THE THESIS**

**CLASSIFICATION OF DISASTER-RELATED TWEETS WITH FEW-SHOT LEARNING USING GENERATIVE PRE-TRAINED TRANSFORMER**

**by Divyaprakash Dhurandhar**

**Thesis Director: Maria Striki**

Informative tweets are a valuable resource for disaster management, where timely relevant information is critical. The main idea behind the use of machine learning in disaster management is to automate the process of detecting relevant information in real-time. In this research, we aim to delineate a framework of tweet classification for disaster management.

**Acknowledgments**

Foremost, I would like to express my appreciation and gratitude toward Professor Maria Striki for her guidance and patience shown as my thesis advisor. Her dedication and keen interest to help students had been solely responsible for developing an interest in this work. Her prompt suggestions, kindness, enthusiasm, and dynamism have enabled me to finish my thesis with utmost satisfaction.

I want to show my gratitude to – and – for taking their time out from their busy schedules and serving on my thesis defense committee.

I would also like to thank my local guardians Kaushal and Priya Verma, for their love, support, and making my stay in the US extremely comfortable. Special thanks go to my parents for their unconditional love, support, and encouragement throughout my entire academic career.

Contents

**1.**

[**Introduction** 5](#_Toc82620634)

[**1.1** **Motivation** 5](#_Toc82620635)

[**1.2** **Objective** 8](#_Toc82620636)

[**1.3 Overview** 9](#_Toc82620637)

[**Chapter 3** 11](#_Toc82620638)

[**GPT-3 API** 11](#_Toc82620639)

[**3.1 GPT-3 API Overview** 11](#_Toc82620640)

[**3.2 API Completion Endpoint** 11](#_Toc82620641)

[**3.3 API Search Endpoint** 13](#_Toc82620642)

[**3.4** **Prompt Design** 14](#_Toc82620643)

[**Chapter 4** 15](#_Toc82620644)

[**Experiments** 15](#_Toc82620645)

[**4.1 Semantic Search for Disaster-Type Classification** 15](#_Toc82620646)

[**4.2 Binary Classification with Few-Shot Learning** 16](#_Toc82620647)

[**4.3 Complex Multi-Class Batch Classification with Few-Shot Learning** 19](#_Toc82620648)

[**4.4 Validation** 23](#_Toc82620649)

[**Chapter 5** 24](#_Toc82620650)

[**Conclusion** 24](#_Toc82620651)

[**Chapter 6** 24](#_Toc82620652)

[**Future Work** 24](#_Toc82620653)

[**References** 26](#_Toc82620654)

**Chapter 1**

# **Introduction**

## **Motivation**

Social media upsurge over the past 15 years has marked a shift in the potential of how information is collected and dispersed during natural disasters [1]. The Federal Emergency Management Agency (FEMA) identifies social media as an essential component of future disaster management [2]. Social networks such as Twitter offer a wealth of information and are an active communication channel during emergency events such as disasters and natural hazards. It is challenging to understand tweet messages for a machine learning-based system since they are short (maximum 280 characters) and informal, especially in disasters when identifying timely relevant information is critical. It is not always clear whether a person's words announce a disaster (see figure 1).



Figure 1: Non-Disaster-related tweet. Source [13]

In Figure 1, the author uses the word "ABLAZE" to describe the sky but only metaphorically. Such metaphorical reference is immediately apparent to a person, especially with the visual aid. A machine, on the other hand, has a more challenging time understanding it. As a result, a massive corpus of work has been done to identify disaster-related tweets.

In this sector, the binary categorization of disaster-related tweets is a commonly used classification criterion. Tweets are divided into two categories, namely on-topic or off-topic. Another classification criterion is informativeness-based. The usual labels are: affected individuals, infrastructure and utility damage, caution and advice, donation and volunteering, sympathy and emotional support, other useful information, not related or not informative. This labeling could offer us more insights into the local situation when a disaster occurs.

Existing studies select tweets that are relevant to a specific disaster using either learning-based or matching-based approaches. The learning-based approach builds a model from a set of labeled tweets and uses the model to predict another set of data (e.g., [3], [4], [5], [6]). The matching-based technique identifies a collection of keywords and hashtags relevant to a certain disaster and searches for tweets containing those terms. (e.g., [7], [8], [9], [10], [11]). A foundation that guarantees the success of all these tasks is an effective filtering technique that could filter out noisy information carried by the data stream and separate those messages containing rich information on disasters.

However, current learning-based and matching-based filtering techniques pose several challenges. For learning-based approaches, the accuracy of the trained model highly depends on the quality and size of the training dataset. Moreover, training datasets in the existing studies are often small because having large labeled data for training is demanding. Furthermore, the models used in these systems are trained for a specific task such as binary classification of disaster-related tweets[], extraction of keywords on disasters[], sentiment analysis[], and classification of tweets based on disaster type[]. Therefore, these machine learning systems are task-specific and can only be applied for the above-mentioned tasks, one at a time.

The recent advances in deep learning and statistical model architecture have made it possible to transfer knowledge across tasks and domains. Specifically, the pre-training procedure effectively improves the performance of neural networks on multiple tasks and datasets and is widely used in the field of NLP. However, existing work on pre-training does not consider the scenarios where the source of training data is irrelevant or noisy. A pre-trained model with a poor source of training data can result in poor performance of downstream tasks. Moreover, the pre-trained model is often specific to one task, meaning that the model is trained to transform input representations of one task into output representations consistent with the training data of that task. Although task-agnostic in architecture, this method still requires fine-tuning with thousands of examples. Thus, the pre-trained model may not process input representations of other tasks, resulting in poor generalization. By contrast, the human ability to learn new tasks is not dependent on the source of training data, and we can usually perform a new language task after only a few examples, something which current NLP systems still struggle to do. A pre-training procedure that is task agnostic, generalizes across domains, and can be applied to different kinds of training data is of interest.

More recently, researchers have discovered that scaling up language models increases task-agnostic, few-shot performance (10-100 training instances), sometimes even approaching competition with past state-of-the-art fine-tuning techniques[]. Specifically, GPT-3 (Generative Pre-trained Transformer-3), a deep learning-based autoregressive language model that generates human-like writing. GPT-3 has 175 billion parameters, ten times the number of prior non-sparse language models. GPT-3 achieves strong performance on many NLP datasets, including translation, question-answering, and several tasks that require on-the-fly reasoning or domain adaptation, such as unscrambling words and using a novel word in a sentence, or performing 3-digit arithmetic. For all tasks, GPT-3 can function without any gradient updates or fine-tuning and can achieve high accuracy with only a few examples (few-shot learning) specified purely via text interaction with the model. More so, GPT-3 can generate samples of news articles that human evaluators have difficulty distinguishing from articles written by humans.

Keeping in mind the capabilities mentioned above of GPT-3, it is worthwhile to investigate whether GPT-3 can be employed for binary and multi-class classification tasks. The idea here is to apply the GPT-3 model to tasks specific to disaster management, such as classifying tweets as disaster-related or not, multi-class classification for informativeness-based disaster response, and classifying the type of disaster as an effective filtering technique for crisis management.

## **Objective**

The objective of this thesis is to demonstrate binary classification of on-topic/off-topic tweets related to disasters and complex multi-class classification for informativeness-based disaster response, in real-time, purely via text interaction using a powerful language model called GPT-3. Secondly, we want to blueprint a software tool for identifying, filtering, and categorizing disaster-related tweets to optimize crisis informatics using this powerful technology. This thesis focuses on the versatility of large-scale language models for classification and highlights their superiority over pre-existing, task-specific classification solutions. The design of the software tool is based on a set of requirements, and we evaluate the design using a set of quantitative and qualitative metrics.

First, to show a successful software design, it is necessary to understand the components of the application that motivate the need for such innovation. Next, it is essential to understand the limitations and landscape of pre-existing software tools. Lastly, it is necessary to know how computational resources and similar applications will evolve in the future and how this evolution will impact the framework's design. We implement our design using a powerful neural network language generation A.I called GPT-3. We back up our design decisions with scientific validation based on tweets from multiple natural disasters, including the 2014 floods in India, Hurricane Sandy in 2012, Earthquakes in California, the MERS epidemic, forest fires, and landslides, all of which were hand-classified by volunteers and paid individuals to show that the software tool meets the requirements.

## **1.3 Overview**

The motivation and objective for this thesis are given in Chapter 1. We discuss the impact and key components of crisis informatics using social media as an information extraction resource. We discuss the current landscape of software systems that support disaster response and highlight the software barriers that limit scientific progress.

In Chapter 2, we provide background information necessary to appreciate this thesis. First, we introduce and motivate data preprocessing approaches to perform Natural Language Processing (NLP), specifically data preprocessing of tweets for machine learning classifier algorithms. Next, we provide an overview of the existing approaches to classifying disaster-related tweets and highlight the computational challenges of existing methods. We also discuss the specific composition of the application that motivates the design of the framework and experiments that evaluate the requirements of the software. Finally, we introduce the concept of language models and discuss the evolution of large-scale deep learning for NLP and their impact over the years, specifically the Transformer architecture. We describe the design and implementation of the GPT-3 API. In this chapter, we discuss the use of the Completion endpoints used for this project and showcase their use with the help of examples. Furthermore, we discuss prompt design and tools required to fine tune GPT-3 models and how to apply it to complex natural language tasks.

Chapter 3 presents and discusses experiments performed to demonstrate on-topic/off-topic classification of disaster-related tweets and complex multi-class classification for an informativeness-based approach to disaster response. We present evidence to appreciate the accuracy of our experiments and show that the project meets the required specifications.

In Chapter 4, we outline the key conclusions of this thesis with a discussion of the impact of this project. We discuss how well GPT-3 satisfies the set of requirements we have identified in Chapter 3 and how well we achieved the objectives of this thesis. Finally, we highlight the possible near-term directions for the development of this project.

# **Chapter 2**

# **Background**

## **2.1 Literature review**

Social media is getting increasingly crucial during a crisis, mainly due to the widespread usage of mobile devices. Social media allows communicating the current situation to other (affected) people or emergency agencies through mobile phones or emergency lines. As a result, numerous studies have been conducted in the last few years concentrating on various elements of social media in crisis management, highlighting its ever-increasing relevance in this field Tweets posted during disasters have been known to provide information that aids situational awareness [14], and [15] has a current overview of research for evaluating social media in disaster response.Existing studies focus on extracting disaster-related information from socially-generated content during natural disasters, from which actionable data can be disseminated to disaster relief workers [16].

The earliest efforts consist of matching-based approaches where social media content is matched with disaster-related information and relevant keywords and hashtags and analyzed and assessed for usefulness and completeness. The previous studies in this method have a problem in that they usually utilize a restricted number of preset hashtags, such as combining disaster name/type with the name of the afflicted location (e.g., #napaearthquake) or the official name of the catastrophe (e.g., #hurricanesandy). However, a basic method may overlook numerous relevant hashtags created and used by people, such as #3amearthquake, #staysafenapa, and #fearoftheearthquake, which utilize different wording. Also, many such hashtags are misspelled, such as #eathquake, #eartquake, #earrhquake, which the existing simple solution cannot detect. The performance of the matching-based approach relies on the completeness of the used keywords and hashtags.

When compared to the matching-based method, the learning-based method tends to incorporate more tweets into relevant sets. The amount and quality of training datasets have a significant impact on the accuracy of learning-based approaches. Usually, the tweets have to undergo extensive data pre-processing before model training.

Diagram, schematic

Description automatically generated Diagram

Description automatically generated

Figure 1: Matching-based method[] Figure 2: Learning-based method[]

A number of systems have been developed to classify, extract, and summarize [21] crisis-relevant information from social media; for a detailed survey see [7].

Cameron, et al., describe a platform for emergency situation awareness [2]. They classify interesting tweets using an SVM classifier.

Verma, et al., use Naive Bayes and MaxEnt classifiers to find situational awareness tweets from several crises [25].

Imran, et al., implemented AIDR to classify a Twitter data stream during crises [8]. They use a random forest classifier in an offline setting. After receiving every mini-batch of 50 training examples, they replace the older model with a new one.

In [10], the authors show the performance of a number of non-neural network classifiers trained on labeled data from past crisis events. However, they do not use DNNs in their comparison.

# **Chapter 3**

# **Tools and Technologies**

## **3.1 GPT-3 Design and Implementation**

GPT-3 is a deep neural network that predicts the next word in a phrase using the attention mechanism. The architecture of GPT-3 is made up of two primary parts: an encoder and a decoder. The encoder takes the previous word in the sentence as input and converts it to a vector representation, which is then passed through an attention mechanism to determine the next word prediction. The decoder accepts the previous word as well as its vector representation as inputs and produces a probability distribution over all possible words given those inputs.

Diagram

Description automatically generated

Figure: Transformer based architecture

Table

Description automatically generated

It has been trained on a corpus of over 1 billion words and is capable of producing text with character-level precision.

Table

Description automatically generated

Key facts about GPT-3:

* Models: GPT-3 has eight different models with sizes ranging from 125 million to 175 billion parameters.
* Model Size: The largest GPT-3 model has 175 billion parameter. This is 470 times bigger than the largest BERT model (375 million parameters)
* Architecture: GPT-3 is an Autoregressive model and follows a decoder only architecture. It is trained using next word prediction objective
* Learning: GPT-3 learns through Few Shots and there is no Gradient updates while learning
* Training Data Needed: GPT-3 needs less training data. It can learn from very less data and this enables its application on domains having less data

GPT-3's performance is on par with the best language models for text generation, which is significantly better than previous GPT models. Microsoft's Turing NLG model can generate text at character level accuracy on a test set of Wikipedia articles, but requires an enormous amount of training data to do so. OpenAI claims that GPT-3 can achieve this level of performance without any additional training data after its initial pre-training period. In addition, GPT-3 is capable of generating longer sentences and paragraphs than earlier models such as Google's BERT and Stanford NLP's Transformer.

## **2.3 GPT-3 API Overview**

GPT-3 is accessible via OpenAI’s API. There are three main reasons to release an API instead of open-sourcing the models. First, commercializing the technology helps pay for ongoing AI research, safety, and policy efforts.

Second, many of the models that underpin the API are huge, necessitating a great deal of knowledge to design and deploy and being quite costly to run. This makes it difficult for anyone other than more giant corporations to reap the benefits of the underlying technology. Smaller enterprises and organizations may now use robust AI systems thanks to the API.

Third, the API approach makes it easier to respond to technological misuse. Because it is difficult to forecast the models' downstream use cases, it is fundamentally safer to expose them via an API and gradually extend access rather than an open-source model with no way to change access if it turns out to have negative applications.

## **2.4 API Completion Endpoint**

The API provides endpoints to interact with the models. We can interact with the API through HTTP requests from any language. For this project, we will use the officially supported Python bindings.

The completions endpoint is at the center of the API. It provides a simple text-in, text-out interface to the models that are highly flexible and powerful. We input some text as a prompt, and the model will generate a text completion that attempts to match whatever context or pattern you gave it. For example, if we provide the API the prompt, “As Descartes said, I think, therefore,” it will return the completion “ I am” with high probability. Listing 1.1 illustrates an example request with the completion endpoint. We first initiate an HTTP connection by authenticating using a private key. We select davinci as our engine. Next, we pass the prompt “Once upon a time ” and wait for a response. Listing 1.2 illustrates a response to our prompt, with the completion “ there was a girl who”



Listing 3.1: Example request to the Completion API endpoint.

****

Listing 1.2: Response to request made in Listing 1.1.

## **Prompt Design**

The GPT-3 models can do everything from generating original stories to performing complex text analysis. Because they can do so many things, we have to be explicit in showing what we want. Showing, not just telling, is often the secret to an excellent prompt. There are two basic guidelines for creating prompts: (1) Show and tell; We have to be clear in what we want through instructions, examples, or a combination of the two. If we want the model to rank a list of items in alphabetical order or to classify a paragraph by sentiment, we have to show it that's what we want. (2) Provide quality data; If we're trying to build a classifier or get the model to follow a pattern, we have to make sure that there are enough examples. Secondly, we can proofread our examples — the model is usually intelligent enough to see through basic spelling mistakes and give you a response. Still, it also might assume this is intentional, and it can affect the answer.

## **Fine-tuning**

Fine-tuning lets you get more out of the models available through the API by providing:

1. Higher quality results than prompt design
2. Ability to train on more examples than can fit in a prompt
3. Token savings due to shorter prompts
4. Lower latency requests

GPT-3 has been pre-trained on a vast amount of text from the open internet. When given a prompt with just a few examples, it can often intuit what task you are trying to perform and generate a plausible completion. This is often called "few-shot learning."

Fine-tuning improves on few-shot learning by training on many more examples than can fit in the prompt, letting you achieve better results on a wide number of tasks. **Once a model has been fine-tuned, you won't need to provide examples in the prompt anymore.** This saves costs and enables lower-latency requests.

At a high level, fine-tuning involves the following steps:

1. Prepare and upload training data
2. Train a new fine-tuned model
3. Use your fine-tuned model

Training data is how you teach GPT-3 what you'd like it to say.

Your data must be a [JSONL](https://jsonlines.org/) document, where each line is a prompt-completion pair corresponding to a training example. You can use our [CLI data preparation tool](https://beta.openai.com/docs/guides/fine-tuning/cli-data-preparation-tool) to easily convert your data into the right file format.

{"prompt": "<prompt text>", "completion": "<ideal generated text>"}

{"prompt": "<prompt text>", "completion": "<ideal generated text>"}

{"prompt": "<prompt text>", "completion": "<ideal generated text>"}

...

# **Chapter 3**

# **Experiments**

In this section, we will assess the classification performance of Gpt-3 on multiple real world crisis datasets. First we perform qualitative assessment using Davinci model and demonstrate few-shot learning capabilities of the model. We will evaluate the model in real-time using the completion api endpoint. Next, we perform quantitative assessment for classification tasks by fine tuning ada, babbage and curie models and evaluate the performance using evaluation metrics as described in section[].

## **3.1 Qualitative Assessment-Binary Classification with Few-Shot Learning**

This experiment uses the GPT-3 Davinci model’s completion API endpoint for on-topic/off-topic disaster categorization. The tweets used for this experiment are hand-classified by volunteers and fall either into the general category or tweets announcing a disaster. The tweet dataset used in this experiment was created by the company figure-eight and shared initially on their 'Data For Everyone' website [12].

Specifically, GPT-3 was shown only seven examples of disaster-related and general tweets purely via text interaction, following a specific pattern. The experiment proceeds as follows: (1) We first design a prompt for binary classification. (2) We then test our classifier model.

Figure 2.1 shows the prompt design to make the classifier model for on-topic/off-topic classification. Figure 2.2 shows the use of the completion API endpoint to classify and test subsequent tweets.

**This is a tweet classifier.**

**Text:** Forest fire near La Ronge Sask. Canada.

**Category:** Disaster

###

**Text:** All residents asked to 'shelter in place' are being notified by officers.

No other evacuation or shelter in place orders are expected.

**Category:** Disaster

###

**Text:** What's up man?

**Category:** General

###

**Text:** What a wonderful day!

**Category:** General

###

**Text:** "'The man who can drive himself further once the effort gets painful

is the man who will win.' Roger Bannister"

**Category:** General

###

**Text:** @NorwayMFA #Bahrain police had previously died in a road

accident they were not killed by explosion

**Category:** Disaster

###

**Text:** #WisdomWed BONUS - 5 Minute Daily Habits that could really

improve your life. How many do you already do? #lifehacks

**Category:** General

###



Figure 4.1: Prompt design for on-topic/off-topic disaster-related tweet classification.

Prompt (in blue)

Text: That moment when you get on a scary roller coaster and the guy

behind you is just screaming bloody murder ?????? #silverwood #aftershock

Category: General

###

Text: Horrible Accident Man Died In Wings of Airplane (29-07-2015)

Category: Disaster

###

Text: Experts in France begin examining airplane debris found on Reunion

Island French air accident experts on WednesdayÂ‰Ã›\_

Category: Disaster

###

Text: What's the police or ambulance number in Lesotho? Any body know?

Category: General

###

Text: @ACarewornHeart Have a good un fella sorry I won't be there to

get annihilated with you :(

Category: General

###

Text: Cop pulls drunk driver to safety SECONDS before his car is hit by train.

Category: ~~General~~ Disaster

###

Text: @smallforestelf Umm because a gun stopped the gunman with who

was carrying a bomb!

Category: ~~General~~ Disaster

###

Text: The Catastrophic Effects of Hiroshima and Nagasaki Atomic Bombings

Still Being Felt Today http://t.co/1kRPz3j1EU

Category: Disaster

###

Prediction (in green)

Stop Sequence

Predicted General. Corrected to Disaster.

Model learns to predict “Category: Disaster/General”

Figure 2.2: Testing on-topic/off-topic disaster-related tweets classifier using text interaction with GPT-3 via the Completion API endpoint.

The experiment serves as a good measure for qualitative analysis of the classifiers built for this project. Although qualitative analysis is proof of scientific validation, it is not complete without a quantitative analysis of the underlying classifier models. Calculating the accuracy, precision, and f1-scores requires much more data for validation and fine-tuning of the models which is presented in the next section. But we can get a calculated intuition from the vast corpus of research made on GPT-3’s performance on multiple natural language datasets and benchmarks. The following graph shows in-context learning performance on a simple task requiring the model to remove random symbols from a word, both with and without a natural language task description as stated in GPT-3’s original research paper[]. The steeper “in-context learning curves” for large models demonstrate improved ability to learn a task from contextual information. Similar behavior is observed across a wide range of tasks including binary classification as preseted in this thesis.

Chart, line chart

Description automatically generated

Figure: Larger models make increasingly efficient use of in-context information

## 3.2 Quantitative Assessment- Fine tuning models for binary classification

In this experiment, we use the dataset[] for binary classification of on-topic/off-topic disaster related tweets. The dataset needs to be prepared for the model training and for that we use the openai data preparation toolkit as mentioned in section[]. The dataset is split into training and validation sets and converted into a JSONL file. The data preparation toolkit separates 1000 examples out of the training set for model validation.

The models are trained in openai servers for n = 4 epochs, learning rate = 0.01 and batch size = 16. The models are evaluated using a set of classification metrics. For classification metrics, we use accuracy, precision, recall, AUROC, AUPRC and F1-score. The metrics are based on a classification threshold of 0.5 (i.e. when the probability is > 0.5, an example is classified as belonging to the positive class.)

These evaluations assume that the text labels for classes tokenize down to a single token, as described in table[]. If these conditions do not hold, the numbers will likely be wrong. We also present a plot of training and validation loss to show that the model does not overfit the training data. Furthermore, we plot training sequence accuracies to gain further insights about the models.

|  |  |  |  |
| --- | --- | --- | --- |
| **Labels** | **tokens** | **Modified label to single token** | **# tweets** |
| Relevant | 2 | relevant | 4673 |
| Not relevant | 2 | not | 6187 |
| **Grand Total** | | | **10860** |
|  |  |  |  |

Table: Relevance classification dataset[].

Chart

Description automatically generatedChart

Description automatically generated

Figure: Comparison of Ada and Babbage model losses plotted against step count where 1 step = 16 examples. Plot shows validation loss minimize after 1 epoch for both models.

Chart, line chart

Description automatically generatedTimeline

Description automatically generated with medium confidence

1. (b)

Figure: Scatter plots for training sequence accuracies plotted against elapsed examples for model Ada. Plot (a) shows model reaches 80% accuracy after only 100 instances.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | Accuracy | Precision | Recall | AUROC | AUPRC | F1-Score |
| Ada | 0.838 | 0.852 | 0.782 | 0.895 | 0.906 | 0.815 |
| Babbage | 0.838 | 0.861 | 0.773 | 0.902 | 0.909 | 0.815 |

Table: Model evaluation metrics for binary classification of on-topic/off topic tweets related to crisis.

## 3.3 Quantitative Assessment- Fine tuning models for multi-class classification

For this experiment, we use the dataset[] for multi-class classification. We will use the dataset preparation toolkit as mentioned in section[]. The dataset consists of tweets from multiple real world crisis. The table[] shows the lables and their respective counts. The dataset is split into training and validation sets. The validation set consists of 1000 examples mutually exclusive from the training set. The models are trainded for n = 4 epocs, learning rate = 0.01 and batch size = 16. We use accuracy and weighted F1-score as an evaluation metric and compare the model performance with previous research performed on the same dataset as shown in table[].

|  |
| --- |
| **Crisis name** |
| Hurricane Irma |
| Hurricane Harvey |
| Hurricane Maria |
| California Wildfires |
| Mexico Earthquake |
| Iraq-Iran Earthquake |
| Sri Lanka Floods |

Table: Disaster types

|  |  |  |  |
| --- | --- | --- | --- |
| **Labels** | **tokens** | **Modified label to single token** | **# tweets** |
| Affected individuals | 4 | affected | 472 |
| Infrastructure and utility damage | 5 | damage | 1210 |
| Injured or dead people | 5 | people | 486 |
| Missing or found people | 4 | missing | 40 |
| Not humanitarian | 2 | na | 4549 |
| Other relevant information | 3 | other | 5954 |
| Rescue volunteering or donation effort | 7 | give | 3293 |
| Vehicle damage | 3 | car | 54 |
| **Grand Total** | | | **16,058** |

Table: Labels and counts



Figure: Training and validation losses for multiclass classification tasks. No significant improvement after ~2 epochs.

|  |  |  |
| --- | --- | --- |
| Model | Accuracy | Weighted F1-Score |
| Bert (Ma et al.) | 67% | 64% |
| CNN (Imran et al.) | 70.4% | 67.7% |
| **Babbage** | **71.6%** | **71.3%** |
| **Curie** | **71.4%** | **71.1%** |

Table: Model performance comparison on similar datasets.

# **Chapter 5**

# **Conclusion**

Artificial Intelligence is a double edged sword that could be used for good or evil. Which is why we must be ever more vigilant and thoughtful in our efforts to ensure that AI and the internet can be used for good. And that we do not become the catalyst for the next generation of technological disasters. Specifically, this thesis makes the following contributions: (1) discusses the landscape of methods and approaches used for classification of disaster-related tweets and their computational challenges (2) provides the design of a software tool (GPT-3) which addresses both challenges of scalability and adaptivity (3) characterizes the performance of GPT-3 using a set of experiments performed on real world crisis datasets (4) highlights the possible controversial and negative use cases of a language model

The classification models presented in this thesis are available for use in openAI’s servers in real-time and are made available to the disaster response teams via the completion API endpoint.

# **Chapter 6**

# **Future Work**

The most beneficial task in this project would be developing aFurthermore, it is crucial to identify scenarios and applications that span other classification-based approaches besides disaster management. Also, it is essential to provide a clear guideline on maximizing benefits arising from adaptive execution, less from an application perspective and more from an execution perspective. In addition to quantitative experiments presented in Chapter 4, there are a number of scenarios that we have not covered such as performance of gpt-3 in understanding tweets from multiple languages.

Another direction for future work is to build an end-to-end software tool for disaster management. The pretexts used for classification in Chapter 4 can create an online real-time disaster management system. The figure below shows the software component diagram, which consists of two modules. The data sorting module can use the semantic search functionality of GPT-3 to separate tweets into the respective disaster types. The classification module could use the pretexts used in Chapter 4 to classify tweets. The software system could then be deployed, which could aid the disaster response teams to conduct their duties during times of crisis more efficiently.

Diagram

Description automatically generated

There are also many other areas for future work that may be of interest to others exploring the use of GPT-3 for disaster informatics. Some of these ideas include: Creating a robust system for determining the severity of a disaster event Expanding the database to include other types of natural disasters Creating a visual representation of the performance of GPT-3

# **References**

[1] Christian Reuter and Marc-Andre Kaufhold. "Fifteen years of social media in emergencies: A retrospective review and future directions for crisis Informatics". In: Journal of Contingencies and Crisis Management 26.1 (Mar. 2018), pp. 41–57. DOI: 10 . 1111 / 1468 - 5973 . 12196. URL: http : //doi.wiley.com/10.1111/1468-5973.12196.

[2] FEMA. Situational awareness in mass emergency: A behavioral and linguistic analysis of microblogged communications. 2012.

[3] S. Zhang and S. Vucetic. Semi-supervised discovery of informative tweets during the emerging disasters. arXiv preprint arXiv:1610.03750, 2016

[4] D. T. Nguyen, S. Joty, M. Imran, H. Sajjad, and P. Mitra. Applications of online deep learning for crisis response using social media information. arXiv preprint arXiv:1610.01030, 2016.

[5] A. Olteanu, S. Vieweg, and C. Castillo. What to expect when the unexpected happens: Social media communications across crises. In Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, pages 994–1009. ACM, 2015.

[6] F. Morstatter, J. Pfeffer, H. Liu, and K. M. Carley. Is the sample good enough? comparing data from twitter's streaming api with twitter's firehose. arXiv preprint arXiv:1306.5204, 2013.

[7] D. D. Vu, H. To, W.-Y. Shin, and C. Shahabi. Geosocialbound: an efficient framework for estimating social poi boundaries using spatio–textual information. In Proceedings of the Third International ACM SIGMOD Workshop on Managing and Mining Enriched Geo-Spatial Data, page 3. ACM, 2016.

[8] A. Cobo, D. Parra, and J. Nav ´on. Identifying relevant messages in a twitterbased citizen channel for natural disaster situations. In Proceedings of the 24th International Conference on World Wide Web, pages 1189–1194. ACM, 2015

[9] M. Imran, C. Castillo, J. Lucas, P. Meier, and S. Vieweg. AIDR: Artificial intelligence for disaster response. In Proceedings of the 23rd International Conference on World Wide Web, pages 159–162. ACM, 2014.

[10] A. Olteanu, C. Castillo, F. Diaz, and S. Vieweg. Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In ICWSM, 2014.

[11] Z. Ashktorab, C. Brown, M. Nandi, and A. Culotta. Tweedr: Mining twitter to inform disaster response. Proc. of ISCRAM, 2014.

[12] Appen AI Resource Center | Open-Source Datasets. (2021, January 7). <https://appen.com/resources/datasets/>

[13] <https://twitter.com/AnyOtherAnnaK/status/629195955506708480>