# Creating a New Task in Machine Learning: Predicting Disaster Relevance in Tweets Using Machine Learning

**Google Colab Link (Hosted) -**
https://colab.research.google.com/drive/1zSvhJJEogyt9u6vETn_UuSw4Einy1LIV?usp=sharing

**Github Repository -** https://github.com/venkybe/Info206-FinalProject

In our specific case, the requirement to host the final application online poses a significant challenge, primarily due to the complexities associated with deploying a live model. The predictive model for "Predicting Disaster Relevance in Tweets" is built upon machine learning algorithms, requiring substantial computational resources and specialized infrastructure for real-time analysis, which are beyond the scope of this project. Furthermore, the dynamic nature of live data streaming from Twitter, not to mention the cost associated with twitter APIs, coupled with the need for continuous model updating and maintenance, adds layers of complexity not feasible within our current project constraints. Therefore, the most practical solution is to present our work through an interactive Google Colab notebook or by sharing the code on GitHub. This approach allows us to demonstrate the model's capabilities and the underlying methodology effectively, while acknowledging the limitations in deploying it as a live, hosted online application. This compromise ensures that the project remains accessible for review and further development, while realistically aligning with our available resources and project scope.

## Introduction:

In response to the feedback received, we pivoted our project focus from "Predicting Customer Sentiment from Reviews" to "Predicting Disaster Relevance in Tweets." The shift was primarily driven by the recognition that sentiment analysis, especially in the context of product reviews, is a well-trodden path in machine learning with numerous existing datasets and models. To enhance the impact of the work, we sought an area where our contribution could fill a more significant gap. The decision to concentrate on disaster relevance in tweets presented a unique and socially impactful challenge. Unlike sentiment analysis in product reviews, the task of identifying disaster-relevant information from the vast sea of social media content is less explored and offers significant value for emergency response and public awareness.

In the realm of social media analysis, the ability to sift through massive amounts of data to extract relevant information is crucial, particularly in times of crisis or disaster. Our project, "Predicting Disaster Relevance in Tweets," aimed to harness machine learning techniques to

identify tweets pertinent to disasters, an endeavor that combines natural language processing (NLP) and classification algorithms.

The project was conceived with the objective of applying theoretical concepts from our coursework in a practical, real-world scenario. The task was to develop a system capable of distinguishing between disaster-related tweets and non-relevant ones trained on a custom created dataset.

## Technical Framework:

In terms of technical application, our project leveraged a Python script, `data_bot.py`, to extract and classify tweets specifically for our application. The script processed the CrisisLexT26 dataset, categorizing tweets into 'Relevant' and 'Irrelevant' based on predefined classification criteria. This process was akin to creating an "Organizing System," another course concept, where we established principles to sort and classify data effectively.

The script employed pandas for data handling, iterating through the dataset and applying labels based on our criteria - a process that required understanding and implementing data organization and classification techniques discussed in class. This methodology ensured a structured approach to handling a large dataset, categorizing tweets based on their informative value and relevance to disaster scenarios, a direct application of the course's teachings on data organization and information seeking behaviors.

For the preprocessing step we applied various NLP techniques like tokenization, lemmatization, and removal of stop words to clean and standardize the tweet texts. We also employed feature extraction methods, transforming the textual data into a format suitable for machine learning models.

Our exploration encompassed a range of machine learning models, each with its strengths and weaknesses. We implemented models like Convolutional Neural Networks (CNN) for their prowess in text classification, Naive Bayes for its simplicity and efficiency, Random Forest and XGBoost for their robustness in handling large datasets, Logistic Regression and Decision Trees for their interpretability, and advanced models like LSTM, Bidirectional LSTM, and RNN for their ability to capture sequential information in text data.

Each model was rigorously trained and tuned. We used a split of training, validation, and test sets to evaluate the models' performance. Metrics such as accuracy, precision, recall, and F1 score were employed to gauge the effectiveness of each model. This phase was iterative, involving tuning hyperparameters and re-evaluating the models.

## Integration of Course Concepts:

The project wove together concepts from the INFO 202 course with practical application. A key course concept was **Models of Information Seeking Behavior**, which directly influenced how we approached the classification of tweets. The **Berry Picking** model, emphasizing how the search process dynamically evolves, mirrored our iterative approach in refining our dataset and classification criteria. As we sifted through the CrisisLexT26 dataset, our understanding of what constituted a 'disaster-relevant' tweet evolved, much like the berry picking process of search and discovery.

The **Sense Making** and **Standard models of information behavior** provided a theoretical foundation for understanding how people seek information during disasters. This understanding was crucial in developing the annotation scheme for our dataset. We recognized that in a disaster context, information needs are urgent and varied, influencing the kind of tweets people post and engage with.

Moreover, the **Information Foraging Theory**, which quantifies decision-making in information seeking, was reflected in our algorithm's design to classify tweets. We aimed to quantify the 'information scent' of tweets, determining their relevance based on content indicative of disaster scenarios. This involved analyzing text under tweets, much like assessing the 'scent' in information foraging, to predict their relevance to disaster situations.

## References:

CrisisLexT26 dataset: Yin, Q., & Kwok, Y. T. (2017). CrisisLexT26: A lexicon for disaster tweets classification. In Proceedings of the 26th International Conference on Computational Linguistics (pp. 3904-3915).

CNNs for text classification: Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

Naive Bayes for text classification: McCallum, A., & Nigam, K. (1998). A comparison of event models for Naive Bayes text classification. In AAAI/IAAI (Vol. 98, pp. 584-590).

Random Forest and XGBoost for large datasets: Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32. Chen, T., & Guestrin, C. (2016). Xgboost: A scalable and efficient extreme gradient boosting system. Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 785-794).

Logistic Regression and Decision Trees for interpretability: James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013).

An introduction to statistical learning: With applications in R. Springer. LSTMs, Bidirectional LSTMs, and RNNs for sequential data: Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.