Project E2: Kaggle: Natural Language Processing With Disaster Tweets
Team: Robyn Tomson, Otto-Cristofer Vanasaun
Link to repo: https://github.com/Otto-Cristofer-Vanasaun/IDS-NLP-disaster-tweets

Task 2: Business understanding

Identifying your business goals

Background

- Organizations such as emergency response teams and governmental bodies,need accurate, timely information to prioritize resources effectively. Identifying tweets that are about real disaster events can enhance decision-making and improve resource allocation. Currently, social media does not play a big role in recognising disasters in real-time. However, the popularity of social media and the ease of creating posts and interacting online makes it a very effective way of communication if used properly. Therefore, we think that a social media platform like Twitter could provide potentially good information about natural disasters and emergencies, and prove to be helpful.

Business Goals

- **Real-Time Monitoring:** Enable users to filter relevant disaster-related tweets from non-relevant ones quickly.
- **Actionable Insight:** Provide insights into disaster type(s). If a tweet is classified as a disaster, what type of disasters should the project recognise and display? Provide probabilities of the tweet being about a real disaster.

Business Success Criteria

- **Reliability:** Achieve high precision and recall in classifying disaster-related tweets. We value recall the most, then precision, since those two are factors in the F-measure value used for the Kaggle competition. Recall is the most important, since false positives are cheaper than false negatives in real life applications of this model.
- **Speed:** Ensure the system processes new data in near real-time.

Assessing your situation

Inventory of Resources:

- **Data Resources:** Training dataset with 7600 tweets and test dataset with 3200 tweets from the Kaggle competition page.
- **Human Resources:** Robyn Tomson and Otto-Cristofer Vanasaun
- **Computing Resources:** Google Colab Cloud-based systems.

Requirements, assumptions, and constraints:

- **Requirements:**
  - NLP pipeline for data preprocessing, model training, and deployment, so that the model can also work with new tweets that it is tested on (for the demo).
- **Assumptions:**
  - Language in tweets matches the language the model is trained on (English).
- **Constraints:**
  - Limited labeled data.

Risks and contingencies

- **Risk:** Noise in tweets (e.g. metaphorical meanings, sarcasm, user mentions, emojis).
  - **Mitigation:** Include domain-specific lexicons and develop advanced preprocessing techniques.
- **Risk:** False negatives in decision making
  - **Mitigation:** If negative and positive have the same probability, choose positive, because for this model's applications in real life, false positives are much cheaper than false negatives.
- **Risk:** False positives in decision making
  - **Mitigation:** Make it a priority to ensure the model has good precision.

Terminology

- **Disaster:** Any event that could be called a disaster or emergency in real life.
- **Disaster Tweet:** A tweet containing information related to a disaster.
- **False Positive:** A non-disaster tweet incorrectly classified as disaster-related.
- **False Negative:** A disaster tweet incorrectly classified as non-disaster-related.
- **Label/y-value:** Whether or not a tweet is classified as describing a real disaster or not.

Costs and benefits

- **Costs:** Computing resources, person-hours and model maintenance.
- **Benefits:** Enhanced decision-making, improved disaster response efficiency, and potentially saving lives.

Defining Your Data-Mining Goals

Data-Mining Goals

- **Classification Task:** Develop a model to classify tweets as disaster-related or not based on the tweets location, full text and keywords present in the text.
- **Metadata Extraction:** Extract additional information like disaster type or additional classification when the tweet is predicted as a disaster tweet.

Data-Mining Success Criteria

- **Performance Metrics:** Achieve an F1-score of at least 0.8 on the test dataset.
- **Deployment Readiness:** The model should be able to classify new tweets in under a second per tweet.
- **Validation:** Kaggle submission, manual testing with writing our own test tweets.

Task 3: Data understanding

Gathering Data

- **Outline Data Requirements**
    - The data cannot be unbiased to ensure that the model remains as reliable as possible.
    - Identify necessary data attributes:
        - Target variable (“label”).
        - Text, keyword and location variables.
- **Verify Data Availability**
    - Identify data sources: Kaggle competition page, Twitter.
    - Assess access permissions and constraints: we can assume we have permission to access and use the data provided in Kaggle for this task, since this is the intended use of that data.
    - In case we need more data: Twitter (nowadays X) is still a big social media platform, we can gather more data if absolutely needed as a substitute/addition to our current training data.
- **Define Selection Criteria**
    - Are all of the tweets relevant to our project? Are there any unrelated tweets? (e.g. a tweet not even mentioning any important words that could be understood to describe a disaster).
    - Decide on exclusions: Data anomalies, unrelated segments, or incomplete tweets. Can we still use parts of them? Are they disaster tweets? What to do if there is a lot of missing data from either class of tweets? E.g. keep in mind the chance that if, for example, one of the training variables does not matter, we can include the data that was before regarded as “incomplete” and excluded from the training data.
    - Is grammatical correctness a problem for the project? For example, are there many instances of a word, that would otherwise imply a disaster, spelled incorrectly? If yes, then we need to account for typos in data.

Describing Data

- Summarize the dataset structure and contents:
    - Number of rows and columns.
    - Types of data (numeric, categorical, temporal).

Exploring Data

- Visualize key features:
    - Univariate analysis: Histograms, box plots, or bar charts for individual attributes.
    - Bivariate analysis: Correlation heatmaps, scatterplots, and pair plots for attribute relationships.
    - Time-series trends of disasters, if possible.
- Identify patterns, trends, and initial hypotheses:
    - Is there any correlation between the location and label of the tweet? (are there more true (or false) disaster tweets from any specific region?)
    - Is the location relevant? (e.g. if the tweets are all in one region of the world, would the model work anywhere else?)

Verifying Data Quality

- Assess completeness:
    - Count missing values for each column and decide if we need to use over- or undersampling to remove bias.
- Check data consistency:
    - Ensure correct formats (e.g., dates are uniformly formatted, consistent units for measurement).
    - Identify outliers that may represent errors.
- Validate accuracy:
    - Compare with known standards or public data of actual disasters if available.
- Evaluate data bias:
    - Examine if the dataset disproportionately represents certain populations or locations.
- Document issues:
    - Record incomplete records, anomalies, and potential cleaning needs.

Task 4: Planning the project

List of tasks (at least 5) (specify how many hours each member will contribute to each task)

Task	**Description**	Time allocation (Robyn)	Time allocation (Cristofer)
Project Initialization	Define goals, gather initial requirements, assign responsibilities, and set up project management tools.	2h	2h
Data Understanding	Look at the training data set and the Kaggle competition page, and	1h	1h

	understand the format of the data.		
Data Preparation	Clean, preprocess, and transform data - remove unnecessary punctuation, words, find and deal with null-values.	3h	3h
Modeling	Train and evaluate machine learning models, fine-tune parameters, and compare performance metrics.	10h	10h
Deployment and final report	Connect a user-friendly environment to the product (e.g. a webpage) for the demo at the poster session. Make a report with the most important information.	8h	8h
Poster	Design the poster: choose colours, fonts, decide on the composition and content of the poster (what and where to write/display), perform data visualisation.	6h	6h

Methods and Tools

Project Initialization

- **Methods**: Requirement gathering, goal setting.
- **Tools**: Google Workspace (Docs, Sheets).
- **Comments**: Emphasize collaboration and ensure clear communication of objectives.

Data Understanding

- **Methods**: Exploratory Data Analysis, data visualization, statistical profiling.
- **Tools**: Python (Pandas, Matplotlib, Seaborn…), Jupyter Notebooks.
- **Comments**: Focus on understanding correlations, trends, and missing data patterns.

Data Preparation

- **Methods**: Data cleaning (e.g., filling missing values), feature engineering, data normalization.
- **Tools**: Python (Scikit-learn, NumPy).
- **Comments**: Ensure reproducibility of the cleaning steps through detailed documentation.

Modeling

- **Methods**: Supervised learning algorithms, cross-validation, and hyperparameter tuning.
- **Tools**: Python, MLFlow for tracking experiments.
- **Comments**: Document the performance of all models clearly and select based on both metrics and interpretability.

Deployment, Reporting and Poster

- **Methods**: Model deployment, webpage creation, and detailed reporting.
- **Tools**: HTML, CSS, JavaScript, GitHub Pages (GPages) for deployment, Google Slides/PowerPoint for presentations.
- **Comments**: Ensure the deployed system is user-friendly and reports are correct, reliable and coherent.