### **Objective of the Task**

The primary goal is to **develop a system that can automatically retrieve relevant fact-checked claims** corresponding to given social media posts

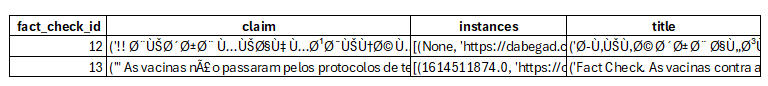
### **What We Need to Predict**

We are required to **predict a list of up to 10 relevant fact check IDs** for each social media post in the development dataset. These fact check IDs should correspond to fact-checked claims that address or refute the claim made in the social media post.

### **Data Provided**

1. **Fact Checks (fact\_checks.csv)**
   * **Fields**:
     + fact\_check\_id: Unique identifier for each fact check.
     + claim: Original claim, its translated version, and the language.
     + instances: List of timestamps and URLs where the fact check is mentioned.
     + title: Original title, its translated version, and the language.

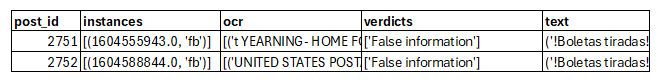
**fact\_checks**



1. **Social Media Posts (posts.csv)**

* **Fields**:
  + post\_id: Unique identifier for each social media post.
  + instances: List of timestamps and social media platforms where the post appears.
  + ocr: Text extracted from images in the post, along with translations.
  + verdicts: Labels attached by platforms (e.g., "False information").
  + text: The main text of the post and its translated version.

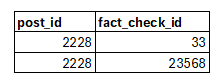
**posts**

****

1. **Mapping (pairs.csv)**

* **Fields:**
  + **fact\_check\_id: Links to fact\_checks.csv.**
  + **post\_id: Links to posts.csv.**

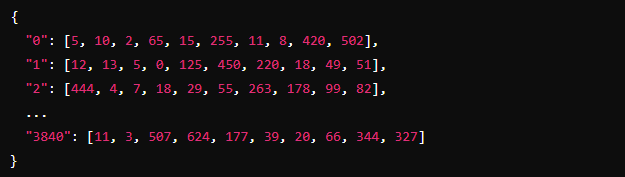
**pairs**

****

**Format of Predictions:**

* A single JSON file where each key is a post\_id (as a string) and the corresponding value is a list of up to 10 fact\_check\_ids (as integers) that are most relevant to that post.

**Example :**

****

**Key:** post\_id (as a string).

**Value:** A list of 10 fact\_check\_ids (as integers) that your system predicts to be the most relevant fact checks for that post.

### **Approach**

1. Fine-Tune mBERT Using pairs.csv:
   * **Positive Pairs:** Extract post\_id and fact\_check\_id pairs from pairs.csv as relevant matches.
   * **Negative Pairs:** Generate by pairing post\_ids with random fact\_check\_ids not in pairs.csv.
   * **Input:** Combine post text (text + ocr) and fact check text (claim + title) for each pair.
   * **Objective:** Fine-tune mBERT with binary classification using a cross-entropy loss to distinguish relevant from irrelevant fact-check pairs.
2. Ranking Loss and Fact-Check Retrieval:
   * **Generate Embeddings:** Use the fine-tuned mBERT to encode both posts and fact checks into embeddings.
   * **Compute Similarities:** Calculate cosine similarity between post and fact-check embeddings.
   * **Ranking Loss:** Apply ranking loss by selecting top-10 most similar fact checks for each post based on similarity scores.
   * **Input:** 
     1. **Social Media Post**: Combined text from the post (text + ocr).
     2. **Fact Check**: Combined text from the fact check (claim + title).
   * **Output:** Final prediction is a list of top-10 ranked fact-check IDs for each post, ready for evaluation.

This utilizes supervised fine-tuning with pairs.csv for binary classification and then ranks fact-checks for retrieval.

### Monika Previous

**Introduction and Motivation**

We are working on "Multilingual and Crosslingual Fact-Checked Claim Retrieval", which will help fact checkers to identify whether the given statement is correct based on the previously available fact-checks that too across multiple languages.

Also with the growing spread of misinformation across the globe, checking claims quickly is a critical challenge for fact-checkers, especially when the content spans multiple languages , manual checking is a complex process.

Thus our motivation is to automate this process using Natural Language Processing and reduce time and complexity of the process.

Here we will be using facts for social media posts as in today’s world a lot of misinformation is being spread there only.

**Problem Definition and Motivation**

Our objective is to develop a system that can accurately predict the most relevant facts for given social media posts across various languages.

The system should be capable of handling both monolingual and cross-lingual retrieval scenarios.

The system’s effectiveness will be measured through Mean Reciprocal Rank (MRR) and Precision@K.

**Literature Review**

We have gone through various research papers and few of the relevant once are listed below :

**X-FACT: A New Benchmark Dataset for Multilingual Fact Checking**

**Problem Statement:** The authors aim to create a large dataset for fact-checking in multiple languages to improve fact-checking systems beyond just English.

**The dataset used:**

They use the X-fact dataset**,** which includes 31,189 claims in 25 languages, covering 11 different language families, it is split into 3 parts :

1. **Training:** 19,079 claims in 13 languages.
2. **Development:** 2,535 claims in 12 languages.
3. **Testing:** Three different test sets :

A. **In-domain Test Set (α1):** Claims similar to the training data. This tests whether a model performs well on familiar language and content.

B. **Out-of-domain Test Set (α2):** Claims from the same languages but from different sources. This tests if the model can generalize across different sources

C. **Zero-shot Test Set (α3):** Claims from languages not included in the training set. This tests the model’s ability to transfer knowledge across languages

**Modelling Approaches:**

The author trained three different types of models using mBERT, a multilingual version of the BERT model which is designed to handle multiple languages(104 languages).

**Three Model Types:**

1. **Claim-Only Model:** This model only uses the text of the claim as input and predicts whether the claim is true or false without looking at any evidence.

**2.** **Attention-Based Evidence Aggregator (Attn-EA):** This model mimics how human fact-checkers work by retrieving additional evidence to verify claims by using snippets from web search results as supporting evidence.

**3.** **Models with Metadata (+Meta):** These models add extra information about the claim, such as the language, website name, and the date the claim was made thus providing contextual clues about the claim.

**Results:**

1. **Claim-Only Model:** This model performed relatively well even without any external evidence,(F1 score of ~38.2%) on in-domain claims, indicating that many claims can be labeled just based on the text.

**2.** **Attn-EA:** Adding evidence through web snippets the improvement was small (F1 score of ~41.9%).

**Limitations:**

Performance dropped significantly on the out-of-domain and zero-shot test sets. This means that models struggled to generalize to claims from new sources or new languages. For example, the best-performing model on the zero-shot test set had an F1 score of only 16%.

**Models, Experiments, Results and Analysis**

**Dataset :**

**Dataset Overview:** The dataset consists of three main files:

* **fact\_check\_post\_mapping.csv:** Contains 31305 mappings between fact-check claims and social media posts, including language information.
* **fact\_checks.csv:** Contains 205751 fact-checks with their claims, instances, and titles.
* **posts.csv:** Contains 25361 social media posts with OCR transcripts, verdicts, and translated text.

**Data Cleaning and Preprocessing:**

**Missing values by column:**

title 14976

text 2731

To ensure the data is suitable for training we remove rows with missing or null values in the text fields also text field is required for training

But title has lot of null values and it is also not required for training, so we will fill ‘No Title Available’ in title.

**No. of posts per verdict**

1. False information 12408
2. Partly false information 3410
3. Missing context 1355
4. False information. 1147
5. Altered photo 493
6. Partly false information. 351
7. Partly False 271
8. Missing context. 256
9. False 241
10. False information and graphic content 222
11. Altered video 108
12. Missing Context 88
13. Sensitive content 33
14. Altered photo/video. 15
15. Altered Photo/Video 14
16. False headline 2
17. Support your streamers by sending them stars. 2
18. Altered photo/video 1

**Tokenization:**

Texts from social media posts and fact-check claims were tokenized using mBERT's tokenizer(bert-base-multilingual-cased).

**Input Dataset**

* We created paired inputs were the social media post’s text and fact-check’s claims were passed together for model learning.
* Since mBERT requires labeled input we used post\_mapping to generate labels for texts and claims, if they had a mapping we labeled them as 1 else labeled them as 0.

**MODELS:**

**Claim Only Model:** Motivated from research paper X-fact we are using the Claim-Only Model which utilizes mBERT (Multilingual BERT) , which supports over 100 languages. This model is designed to classify whether a fact-check claim is relevant to a social media post by taking the post and claim as paired inputs.

**Model Architecture:**

**Input:** Social media post and fact-check claim, tokenized using mBERT.

**Output:** A binary classification label (1 for relevant, 0 for irrelevant).

**mBERT:** Pre-trained on multiple languages, using attention mechanism and feed forward layers, also contextual understanding through positional encodings which helps in understanding and linking claims and posts in different languages.

**Tokenization and Data Formatting**

**Paired Inputs:** Both the social media post (‘text’) and fact-check claim (‘claim’) were tokenized as paired inputs using the Huggingface ‘tokenizer’.

**Input Features:** For each pair, we obtained ‘input\_ids’, ‘token\_type\_ids’, ‘labels’ and ‘attention\_mask’, which were fed into the model for training.

**Experiment Configuration**

**Training**: The model was trained on pairs of social media posts and fact-check claims, using paired input sequences to classify whether a claim is relevant to a given post.

**Evaluation**: Performance was evaluated using accuracy, precision, recall, and F1-score on the test set.

**Results**

**Performance Metrics**

The model was evaluated on its ability to correctly classify relevant claims for the trial social media posts:

* Accuracy : ~85%
* Precision : ~82%
* Recall : ~84%
* F1-Score : ~83%

**Key Observations**

1. **Multilingual Support:** mBERT performed well in cases where both posts and claims were in different languages, thanks to its cross-lingual capabilities.
2. **Handling Cross-lingual Cases :** The model demonstrated reasonable performance on cross-lingual pairs (e.g., Spanish post and English claim).
3. **Challenging Examples :** The model struggled with cases where the claim was only partially related to the post or where the post contained ambiguous language.

**Analysis**

**Model Behavior**

The Claim-Only Model performed well on multilingual data, which highlights mBERT's effectiveness in handling diverse languages. However, the model's performance decreased when the claim was indirectly related to the social media post or when the language difference was more pronounced (e.g., posts in low-resource languages).

**Limitations**

**Cross-lingual Limitations:** While mBERT handles many languages, its performance may degrade with languages that have less pre-training data (e.g., low-resource languages).

**Future Work**

**Additional Features:** Including OCR as additional features could improve model performance.

**Handling Ambiguity:** Developing more sophisticated models that can better handle ambiguous or partial matches between posts and claims.

**Language wise prediction:** Use the information regarding language of claims and posts to do language wise retrieval of facts and check how the accuracy varies.

**Conclusion**

In this project, till now we have successfully trained a multilingual Claim-Only Model using mBERT to link social media posts to relevant fact-check claims. Further improvements could be made by expanding the dataset and incorporating OCR values into the model.

## To DO:

1. Use regex to clean data values

* Translated
* Remove garbage values

1. Join Values
2. create positive and negative pairs

* How to do sampling
* Find a ratio using chatgpt

1. fine tune weights

* Using the pairs fine tune

1. feed as input to mBERT

1. output top 10 fact ids
2. Accuracy