# **Contradictory, My Dear Watson**

| Team Member | SJSU ID | Email ID |
| --- | --- | --- |
| B. Shanmukh Krishna | 016005743 | shanmukhshivasai.boddu@sjsu.edu |
| CH. Roopa Sree | 016005795 | satyasairoopasree.chiluvuri@sjsu.edu |
| D. Ravi Kiran | 015513121 | ravikiran.dendukuri@sjsu.edu |
| Kartikeya Jain | 015973997 | kartikeya.jain@sjsu.edu |

**Introduction**

Machine learning models tackle question answering, text extraction, sentence generation and many other complex tasks. Natural language processing comes into the picture for processing and analyzing the large amount of natural language data. Natural language Inferencing (NLI) is a common NLP task that entails finding the relationship between two sentences given.

As the world becomes more global and companies expand their consumer base, firms are required to work with different languages to gather the sentiment of their target demographic. To perform this task using human workforce can be an expensive and time consuming process [2]. As a result, the utility of machine learning algorithms for multilingual sentiment analysis is a dire need for many corporate organizations.

Recently, one text mining research has been assessing and identifying sentiments for multilingual data. The project “Contradictory, My Dear Watson” is based on sentiment analysis. It is the process of analyzing a premise and a hypothesis and determining the relationship between the two sentences. The goal of this project is to label 2 sentences as either entailment, neutral or contradictory.

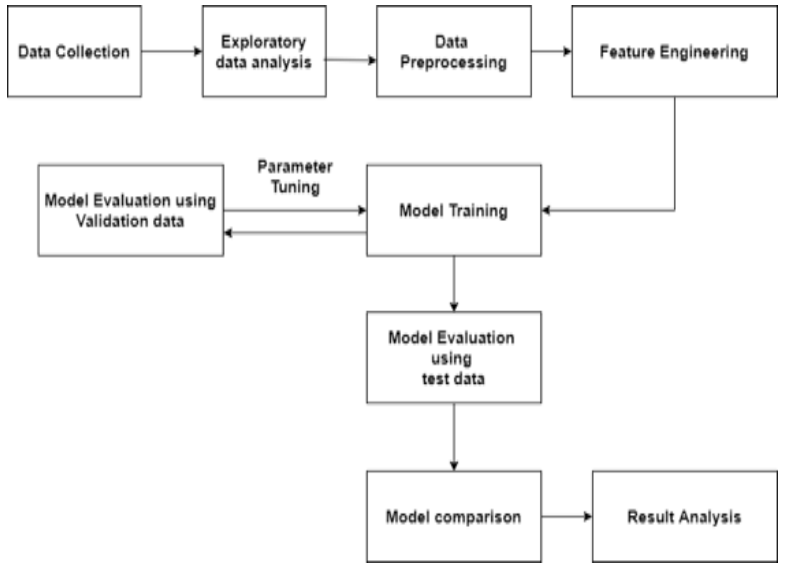
**Literature review/related work**

As mentioned in the introduction, our problem statement is to find the relationship between the two columns which are Premise and Hypothesis, in the dataset and assign them a label. So, it is clear that we need a NLP algorithm to solve this problem.

The basic step in NLP is to vectorize the text data. The vectorizers that we tried to implement in the process of solving this problem are listed below:

1. **CountVectorizer:** This mainly takes the text document as input and parses it in such a way that we get the frequency of every word that is present in the document. One additional feature in CountVectorizer is that it neglects single letter words like **a, I**.
2. **TFIDFVectorizer**: This is the extended version of the CountVectorizer. The frequency of the word is calculated with respect to the occurrence of the word in other documents of the document corpus.
3. **Word2Vec**: This is another vectorizer which is more efficient than the above two vectorizers. Word2Vec converts each word into a vector and using these vectors the similarity can also be found between two words.
4. **BERT**: Currently, the state-of-the-art algorithm for NLP tasks, BERT which is the abbreviated form of Bidirectional Encoder Representations from Transformers. This is developed by researchers at Google. The main reason for BERT becoming the State-of-the-art algorithm is its Bidirectional training of the Transformer. Before BERT the process for NLP is to process the text sequentially. The introduction of Bidirectional training has proved to train the model more efficiently.

**Project Flow Diagram:**

****

**Dataset Description:**

The Train and Test data are imported from inputs of the competition. Below are the links to the datasets.

| **Train.csv** | [**https://www.kaggle.com/competitions/contradictory-my-dear-watson/data?select=train.csv**](https://www.kaggle.com/competitions/contradictory-my-dear-watson/data?select=train.csv) | 2.77 MB |
| --- | --- | --- |
| **Test.csv** | [**https://www.kaggle.com/competitions/contradictory-my-dear-watson/data?select=test.csv**](https://www.kaggle.com/competitions/contradictory-my-dear-watson/data?select=test.csv) | 1.18 MB |

The above datasets consists of the following columns:

1. Id: This column contains a unique identity for every sentence pair.

2. Premise: The First Sentence of the Pair

3. Hypothesis: The Second Sentence of the Pair

4. Lang\_abv: Two-letter abbreviation of the language of the sentence pair

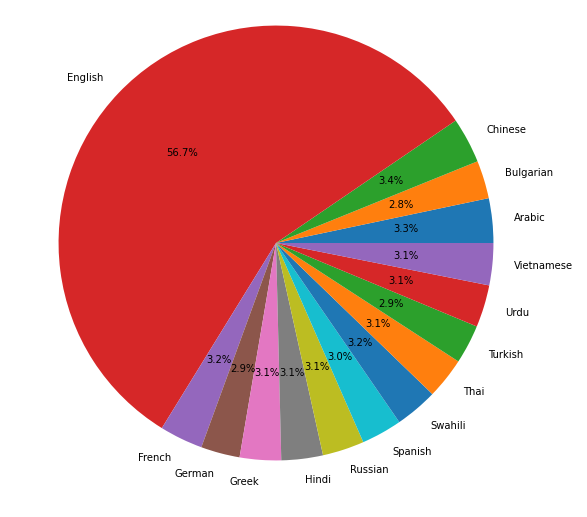
5. Language: Language of the sentence pair

6. Label: The Target column in train.csv, it will be 0 for entailment, 1 for neutral, 2 for contradiction based on the relationship between the premise and hypothesis.

**Exploratory Data Analysis**

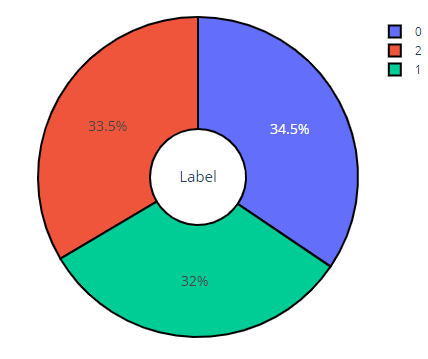
**Data Distribution:**

1. **Language Distribution:**



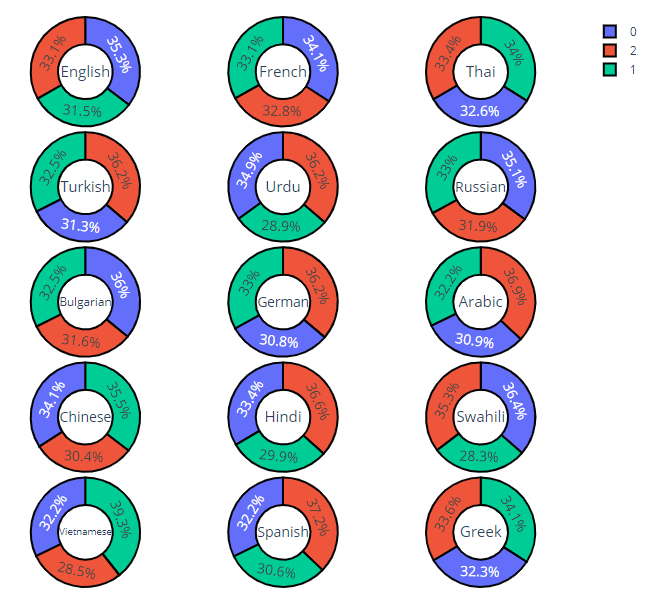
The above pie chart depicts the distribution of the languages in the dataset. As you can see that the dataset is biased towards the English language which occupies more than half of the dataset. Apart from the English language we have other languages as you can see in the pie chart.

1. **Label Distribution (Entire Train Dataset):**



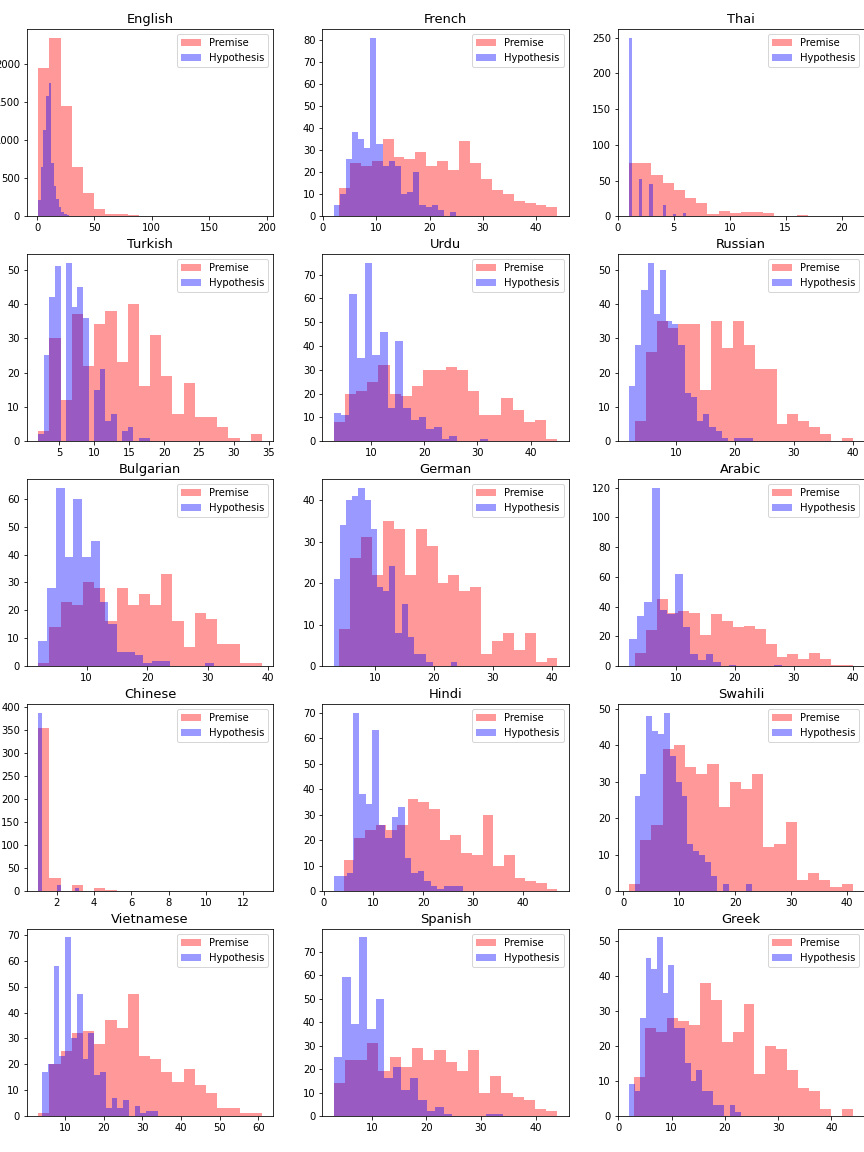
The above pie chart shows the percentage of records with each label. The percentage of distribution is similar but itBert is not completely balanced.

1. **Label Distribution (Language Specific):**

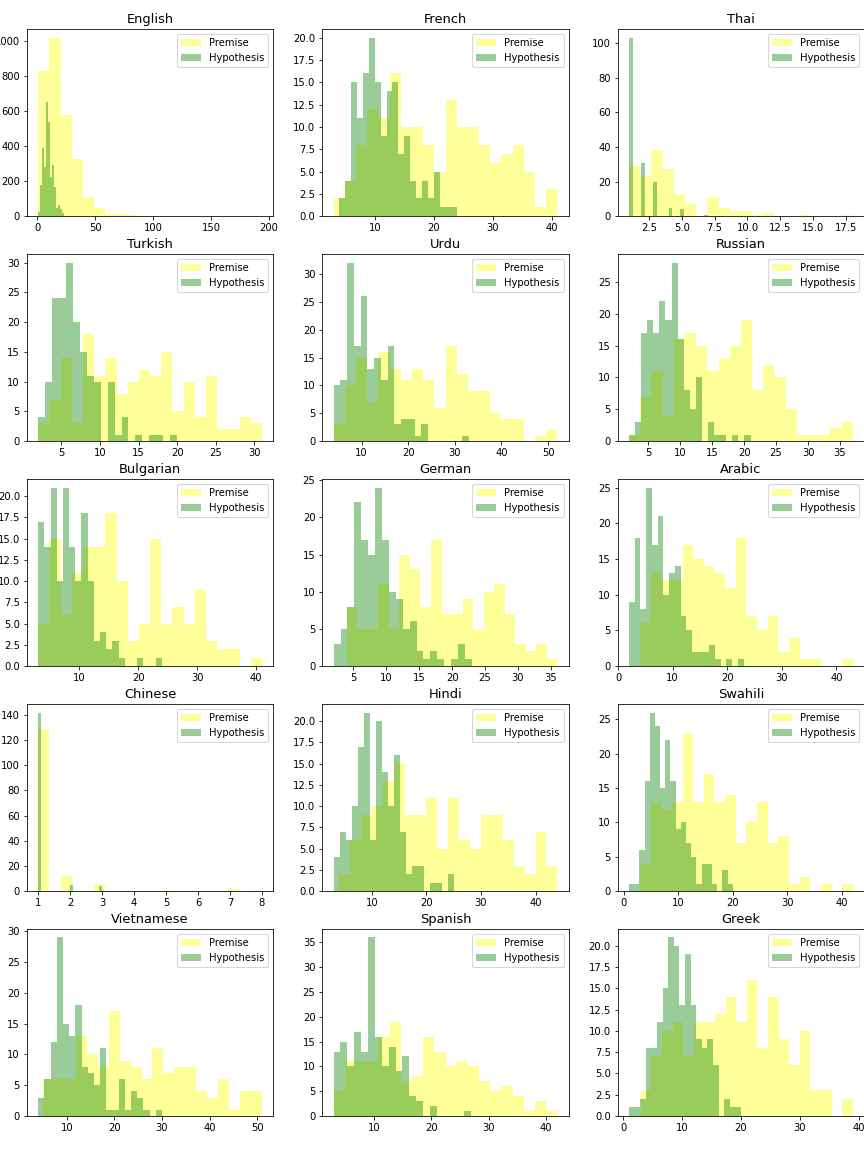


The above series of pie charts represents the label distribution with respect to every language present in the dataset. As you can see, the distribution is not balanced.

1. **Train: Premise & Hypothesis Word Count**



1. **Test: Premise & Hypothesis Word Count**



The word counts of premise and hypothesis for each language are shown in the two figures above. One intriguing discovery is that the premise contains more words than the hypothesis in practically all languages. Because the maximum length of a phrase before encoding is around 60 characters, the sequence length during the encoding phase should be greater.

**Data Preprocessing(Shanmukh):**

To feed any data

The traditional models that we tried to implement to solve this problem are listed below:

1. WordVectorizer

// write about the diff datasets that we loaded

**Model Selection (Karthikeya):**

//Describe about disadvantages of Lstm

//And summarize why we took bert

/\* sample for reference

Started with Naive Bayes as Baseline as it is a standard baseline for text classification. Considering the High dimensional nature of vectorized data which might increase the probability of finding a linearly separable hyperplane we considered Logistic Regression. Based on the kaggle scores we could understand we needed a more complex model to get to top 50 as a natural progression from LSTM we considered the BERT model as BERT can be considered an evolution to LSTM in many aspects BERT is superior as it operates on the entire context unlike only sequence in LSTM and its parallelizable and relatively faster to LSTM too.

/\*

**Problem Formulation and model selection (Karthikeya):**

//Bert

//Bert Architecture

**Model Evaluation:**

We explored a variety of variants of Bert and executed different variants such as Roberta, Albert and DistilBert.

**Roberta:**

It is an enhanced version of Bert. Which means it has the most Training Parameters out of all the variants. Roberta has few differences with Bert where it doesn’t have Next Sentence Predictor and it is trained on Large Mini Batches. Roberta is trained on a Multilingual Dataset. It has a base and large model for the multilingual model.

**Albert:**

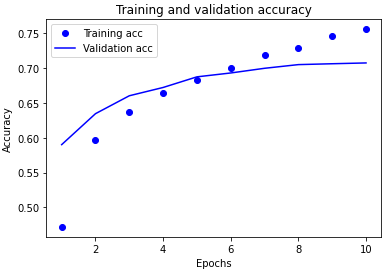
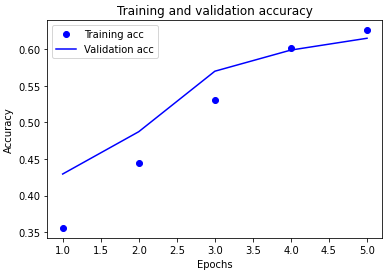
It is a lite version of Bert.It has the least training parameters out of all the variants.Albert has Sentence ordering Prediction (SOP). It is not trained on a multilingual dataset.

**Distillbert:**

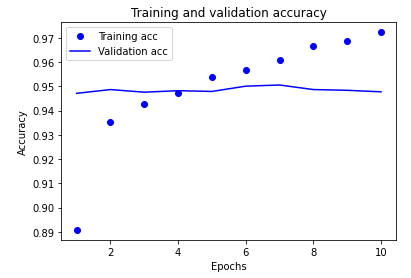
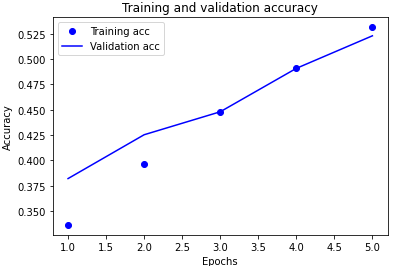
It is a distilled version of Bert - faster, cheaper and lighter.It has 66 Million Training Parameters.Using Teacher - Student Framework.It trained on Multilingual Dataset. It doesn’t have a large model, only a base model.

**Results:**

One parameter for assessing classification models is accuracy. Informally, accuracy refers to the percentage of correct predictions made by our model.The accuracy curve charts below show how well the models we tried performed:

** **

Bert DistilBert

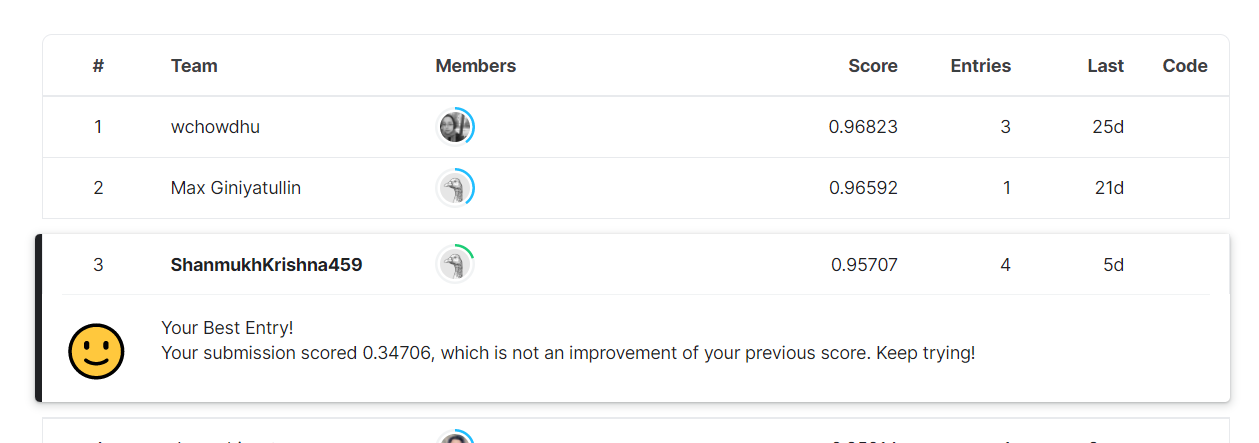
Roberta Albert

We assessed the performance of several models by submitting them to the Kaggle competition, and we found that Roberta, DistilBert, and BERT fared better for label creation, as shown in the table below:



**Kaggle Standings**

The picture below shows our kaggle rating. Out of 50 total entries, we are in the top three. For the Contradictory, My Dear Watson kaggle competition, we received the top score of 0.95.

****

**Conclusion:**

This endeavor provided an excellent learning opportunity. Through sentiment analysis, we learnt how to integrate numerous natural language processing approaches. Overall, working on this Kaggle tournament has been an extremely educational experience. We were able to figure out why certain strategies didn't do well in multilingual data analysis and why some models did.

**Future Scope:**

1. With hyperparameter tweaking, use more deep learning algorithms to analyze more accurate findings.
2. Recreate the project using a distributed system design such as Spark.

**Github Repository for Project:**

**References:**

* Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, & Illia Polosukhin. (2017). Attention Is All You Need.
* Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, & Alexander M. Rush (2020). Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (pp. 38–45). Association for Computational Linguistics.
* Contradictory, my dear Watson. Kaggle. (n.d.). Retrieved May 12, 2022, from <https://www.kaggle.com/competitions/contradictory-my-dear-watson/overview>
* Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP).
* Conneau, A., Rinott, R., Lample, G., Williams, A., Bowman, S., Schwenk, H., & Stoyanov, V. (2018). XNLI: Evaluating Cross-lingual Sentence Representations. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics.
* Williams, A., Nangia, N., & Bowman, S. (2018). A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (pp. 1112–1122). Association for Computational Linguistics.