BRIDGEi2i's

Automatic Headline And Sentiment Generator

Preliminary Report

Problem Statement:

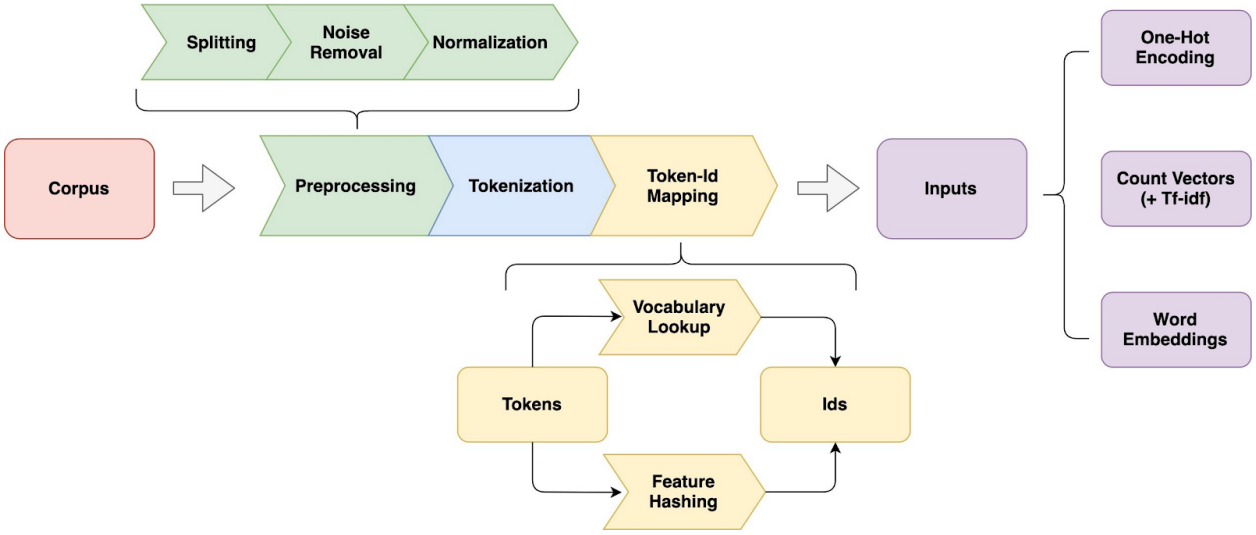
Develop an intelligent system that could first identify the theme of tweets and articles.If the theme is mobile technology then it should identify the sentiments against a brand (at tweet/paragraph level).We would need a one sentence headline of max 20 words for articles which follow the mobile technology theme.

Proposed Solution:

The problem can be divided into separate modules

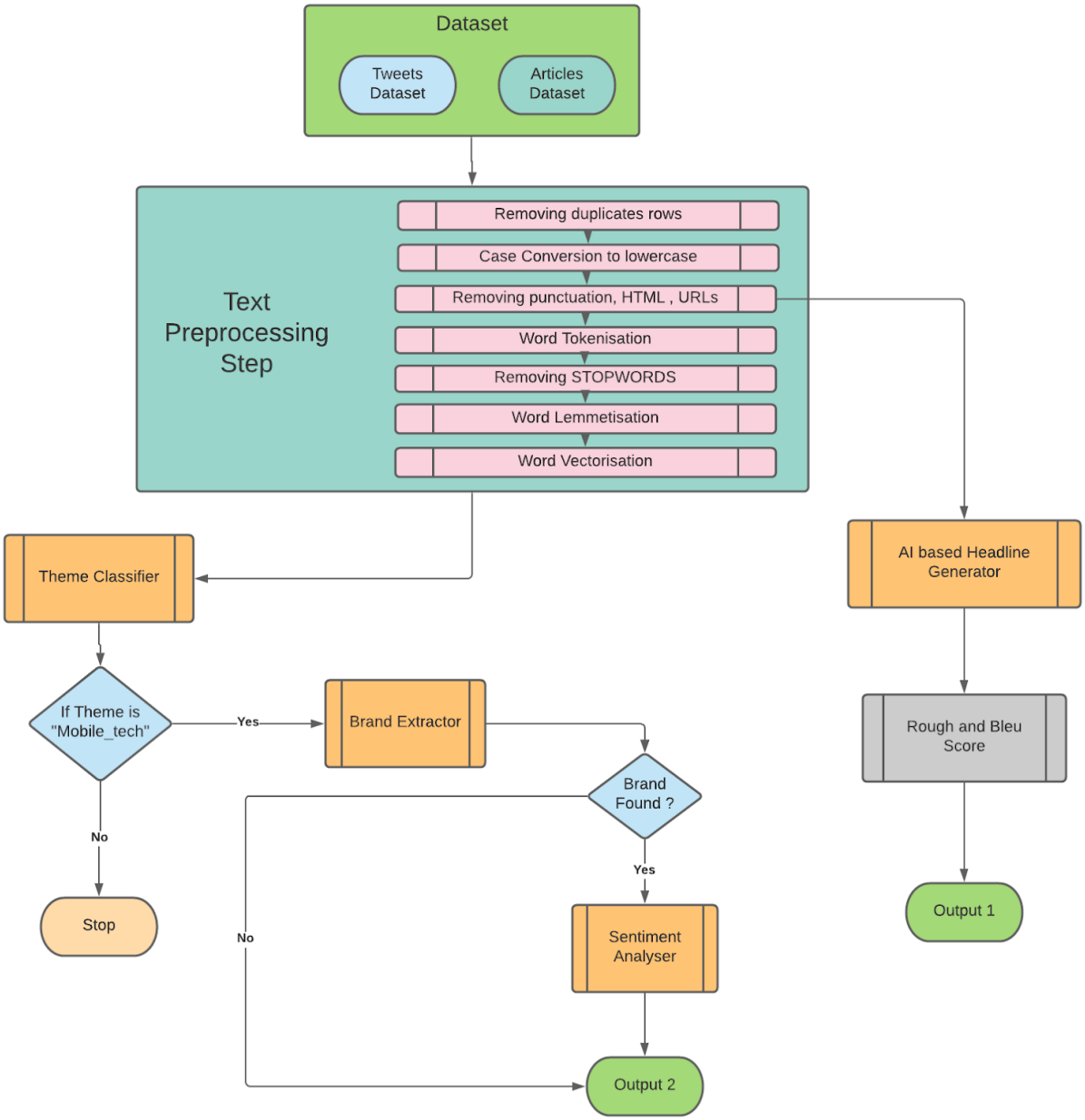
1: Preprocessing Step:

- Text Cleaning - Removed punctuation, urls, emojis , etc
- Tokenization
- Word Lemmatization
- Removed Stop Words
- Vectorisation using Tf-Idf



2: Text Classifier
3: Brand Extractor
4: Sentiment Analyser
5: Headline Generator

Data Flow Diagram:



Text Classification Approach

After completing the preprocessing for the text classification, we split our dataset in the ratio of 70:30 .
Then we used the Term Frequency-Inverse Document Frequency approach for vectorization. It gave numerical weights to every word.

Term frequency summarized how often a name appears within a document. Inverse Document Frequency down scales words that seem a lot across documents.
The main advantage of using this approach is that it doesn't get biased with word frequencies. TF-IDF are actually assigning word frequency scores that try to highlight the most interesting words. We used various approaches like CountVectorizer, word2vec and One-hot encoding but this approach of vectorization gave the best results.

For predicting the outcome, we used machine learning algorithm SVM- Support Vector Machine.

Advantages of using support vector machine:-

1. It is very effective with high dimensional data.
2. We have gone through many research papers for identifying which ML model performs best for text classification and found out that SVM performs better.
3. SVM also works well with emojis data as well.
4. SVM doesn't need much training data to start providing accurate results and since the dataset provided to us is small we proceeded with SVM.

Mobile Brands Extraction Approach

For this we have used nltk library to tokenize the cleaned text , after this we have a csv file which consists of the names of brands and using spacy we have implemented a model similar to lookup table.
But there is always a chance that some brands are not given in the lookup list so in order to cope with this scalability issue we have used POS (Parts of Speech) Tagging using spacy to extract the organisation name. There is a tag “ORG” in Spacy which identifies companies and Organizations.

Sentiment Analysis Approach

The processed data is run through Flair. Flair gives the polarity and subjectivity of the tweets. But, we need to do it for hindi text too, so we used google translate API to translate the hindi text to english and then apply Flair to it. Now we need a brand-specific sentiment. The approach to that is

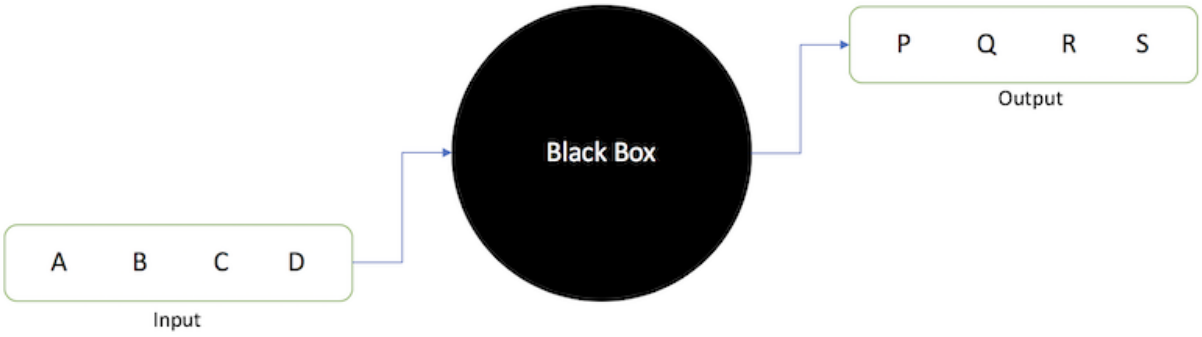
1. We extracted the brands from all the tweets using a simple lookup table algorithm.
2. We have filtered out the adjectives used for the brand that are closer to them.
3. Then have performed sentiment analysis on the filtered adjectives and mark them as positive or negative for the filtered out brands.

What is Flair?

Flair is a powerful NLP library. Flair allows you to apply state-of-the-art natural language processing (NLP) models to text, such as named entity recognition (NER), part-of-speech tagging (PoS), special support for biomedical data, sense disambiguation and classification, with support for a rapidly growing number of languages.

Headline Generator:

Sequence2Sequence Model



1. The Seq2Seq model takes a sequence of objects (words, letters, time series, etc) and outputs another sequence of objects.
2. The contents of the 'black box' of Sequence to sequence models rely on encoder-decoder architecture – a combination of layered RNNs that are arranged in way that allows them to perform the tasks of encoding a word sequence and then passing that encoded sequence to a decoder network to produce an output

Training Strategy for headline generation

1. The hypothesis that much of an article's content is present simply in its first x-words and that the remaining text is just further exposition of the already-outlined topic(s).
2. Trimming is done for long articles so as to contain max 80 tokens(words).
3. For the longer articles, extensive removal of rare words is carried out.
4. For the trimmed articles, a less strict removal of rare words is carried out and training is performed.

Performance Metrics:

The overall accuracy came out to be of >80% efficient for theme classification.
The brands are also identified accurately , In the sentiment analysis we achieved a accuracy of 70-80 % . The headline Generator model are giving somewhat similar results (however have not tested it.)

Scalability:

The method which we have implemented can be scaled up for other languages as well. Since we are using Google translate for converting text to english , our solution will work for other languages as well apart from Hindi, English and Hinglish.

What makes our model different from Others?

1. We used emojis in the tweets as a feature in our sentiment analysis model to improve the accuracy.
2. We have incorporated Brands names as separate features which are extracted from text to train our machine learning model.
3. For the headline generation model we have also used Google AI.