2023 Fall CSE431 Team2 Colloquial Expressions Detection Using Machine and Deep Learning Techniques

1st Mahir Faisal Chowdhury
dept. of Computer Science and Engineering(CSE)
Brac University
Dhaka, Bangladesh
mahir.faisal.chowdhury@g.bracu.ac.bd

***Abstract*—Colloquial expressions are the sentences containing slang and Abbreviations. For the natural language processing classifying these are very complex processing work. To detect the colloquialism from the sentence we need to do data processing to reduce the unimportant data from the dataset and for processing I have used methods like tokenization, stop word removal, special characters and link removal, lemmatization and got the processed text data on which with the label data then by performing vectorization used to train the models like naive bayes, SVM, random forest, and BERT.**

***Index Terms*—Colloquial expressions, Lemmatization, Bert, Naive bayes, SVM, Random forest, Deep learning, Machine learning**

I. Introduction

The rich fabric of daily spoken language is shaped in large part by colloquial terms. A feeling of comfort and familiarity permeates discussions when people employ these terms, which are defined by their deviation from formal standards. Colloquial terms bridge language gaps between groups and regions, bringing individuals together via common understanding and subtle cultural cues. The intriguing thing about colloquial language is how it changes and adapts all the time to reflect society's shifting environment. Colloquialisms are an intriguing branch of language research due to the wide variety of terms used and the ways in which they are shaped by local languages and cultural influences. As an example, the phrase ”gonna,” which is a contraction of ”going to,” shows how efficient informal language may be. In addition to preserving syllables, it captures the essence of informality and immediacy. Just like ”wanna” means ”want to” in English, ”They're” is an abbreviation meaning ”they are” illustrating the language's ability to condense without compromising the depth of expression. The origins of these slang terms are often in oral traditions that have been maintained over the years and help to shape a common cultural identity. For people who share a language, it may be a bonding experience that brings them closer together. The ever-evolving linguistic landscape is shaped by colloquial terms, which highlight the ever-changing dynamics of communication and how we express ourselves. Understanding colloquial terms enhances our capacity to have genuine and relevant discussions, which in turn fosters relationships across groups and generations, as we traverse the different linguistic landscape.

Slang is a way of speaking that is unique to certain communities, subcultures, or social groupings. As a means of collective identification and cohesion, it often arises. Slang is defined by its quick rate of change, uniqueness, and informality. Some examples include ”lit” meaning thrilling or cool, ”bae” meaning before everybody else, and ”on fleek” meaning flawlessly styled.

When you want to abbreviate a word or phrase, you may use an acronym, which is just a word or phrase with no letters or numbers. For reasons of space, time, or clarity, people often utilize abbreviations. Common and universal ones are ”USA” for the United States of America and ”LOL” for ”laugh out loud,” both of which are more casual and tailored to internet or texting conversation. The clear tapestry of common speech is enhanced by both forms; slang places an emphasis on inventiveness and shared experiences, while abbreviations prioritize shortness and ease. When they're in the same room, they liven up routine discussions. Colloquial words detecting is a complex work with nlp and dataset availability is rare and the dataset those are available seems to be small.

When it comes to text categorization using high-dimensional datasets, the Naïve Bayes algorithm is a supervised learning approach which is very successful since it depends on Bayes' theorem.

SVM seeks the hyperplane that divides classes most effectively. It entails optimizing the margin for linear SVM. The support vector machine (SVM) determines the classes by locating the hyperplane with the largest margin between them. It works well in three-dimensional environments. The work of Leo Breiman with Adele Cutler's Random Forest is an extensively used machine learning technique that merges the results of several decision trees into one. Since it can manage regression as well as classification problems, its versatility and user-friendliness have contributed to its widespread usage. Learn about the inner workings of the random forest algorithm, its unique features, and its practical applications in this article.

The bidirectional encoder representations from Transformers, or BERT for short, is an attention-based language model that uses a transformer architecture. Feedforward neural networks, layer normalization, and attention scores are its building blocks. BERT is an exceptional pre-trained model for NLP tasks since it uses bidirectional context to interpret natural language. It is a potent tool in the domain of NLP due to its novel architecture, which enables it to grasp complex connections and contextual information. For categorical data detection in terms of machine learning model naive bayes is the best model to classify text data where random forest works better for mixed type of data. When compared to more conventional models, BERT performs better on sequential and unstructured data, such as text. For natural language processing (NLP) tasks, it excels because it captures complex connections and contextual information. The more the training data is complex the more BERT or deep learning techniques learn better and give better accuracy. After training the processed data, the accuracy that machine learning models showed is Naive Bayes (80%), SVM (80%), and Random Forest (79%). And on the other hand, by training the same processed data Deep learning model, BERT given 98.56% of accuracy which is a huge disparity.

II. Literature Review:

In this paper [1], the researchers claim that Their aim is to improve translations by contextual embeddings. As idiomatic expressions have limited datasets and the modern models were never used on this topic before. The contributions of the research include introducing innovative approaches, developing an evaluation strategy, applying contextual embeddings, identifying challenges and limitations, and providing implications for future research in the domain of idiomatic expression translation. They have used two methods to achieve these goals. They have first translated idiomatic sentences from English to German and then they paraphrase those translated sentences. And on those parts they have done the dataset splitting for better accuracy by noise reduction. For idioms detection and translation from one language to another they have used two methods to achieve these goals. They have first translated idiomatic sentences from English to German and then they paraphrase those translated sentences. They have used different versions of T5 model to their custom and available datasets to the accuracy of the translation task.

Paper [2] explains, The authors state in their paper that non-compositional expression is a more complex process then general language processing for natural language processing and there is not much research on this topic because of less available dataset and low learning accuracy. They created a framework which combines Contrastive Learning and Curriculum Learning and it was designed in a way so that it gets better results from limited datasets than larger ones. Moreover, they used six datasets and used two types of splitting techniques like random and typebase to check the models efficiency. The model provides better accuracy than baseline models in every parameter. The model finds it challenging to transfer from the task of idioms usage recognition to metaphor detection. This model can measure difficulty level and schedule the tasks accordingly from easy to hard.

In paper [3], finding and identifying slang in phrases written in natural language is the main topic of this study. In order to automatically recognize and categorize slang use, the authors provide a methodology based on deep neural network methods. The difficulties that slang presents to natural language systems and its pervasiveness in informal speech motivate this study. Using a mix of conditional random fields, multilayer perceptrons, and bidirectional recurrent architecture of networks of neurons, the authors put their models through their paces. Their goal is to differentiate slang from formal language by looking at a wide range of linguistic components. Word syntactic changes and syntactic categories are among the characteristics. When it comes to slang token recognition, the top models have an F1-score of 0.50 and a F1 score of 0.80 percent for sentence-level detection. This work's contribution is a methodology framework that is not dependent on dictionary creation as much. By using these models, one may automate the process of slang phrase detection and location identification in sentences. Automatically discovering slang use is the main focus of learning features. This study has some caveats, such as its narrow emphasis on slang recognition and identification and its heavy dependence on linguistic factors. There may be more facets of slang that were not considered for this research, and the models could miss certain subtleties in slang use. Ultimately, the article lays forth a methodology for automated slang recognition and detection that relies on deep learning methods. When it comes to recognizing slang in everyday speech, the models work well. The goal of this study is to help NLP systems better comprehend and deal with slang.

From this paper [4] we can see, the idea behind this method is to evaluate aspects that are driven by language and aim to capture key idiom characteristics. The purpose of this study is to examine the feasibility of using pre-trained models to improve idiomaticity recognition systems by collecting these properties. An idiomaticity-based feature model, sentence embeddings, and BERT fine-tuning are all part of the system's multi-model architecture. Using the training data, the BERT algorithm is fine-tuned, and then several model versions are evaluated. We utilize HuggingFace's distiluse-base-multilingual-cased-v15 model to get sentence embeddings. On top of the phrase embeddings, a classifier is trained using logistic regression. With the use of HuggingFace pipelines including pre-trained models for sentiment analysis, back translation, and lexical replacement, idiomaticity characteristics are retrieved. Both the BERT and the feature models contribute to the final categorization. A variety of languages make up the SemEval-2022 objective dataset, which is used to measure the system's performance. When idiomaticity elements are included, the results are better than the baseline model. Among the twenty entries for the job, the system comes in at number fifteen. The paper's error analysis, which includes feature contribution visualizations, aids in comprehending the model's strengths and shortcomings. Due to the limited amount of instances accessible for each idiom, the technique does not adequately reflect the qualities of expressions, such as structuralism and substitutability. Additionally, the authors point out that BERT is known to be more confident in its conclusions, therefore the logarithmic regression and BERT models' values may not be exactly comparable. Using phrase embeddings, idiomaticity features, and BERT fine-tuning, the article proposes a feature-based method for idiomaticity recognition in many languages. With these enhancements, the system outperforms the baseline and reaches a competitive level in the SemEval-2022 challenge. Findings from the error analysis show where the model excels and where it needs work, paving the way for further investigation and refinement.

III. Methodologies:

A. Scrapping and Dataset:

As we are working with colloquial words and available text datasets are very less and to get better results and to teach the model in an efficient way we need a large dataset. For getting a larger dataset I have scraped the tweets from twitter with setting keywords as colloquial words, abbreviations and slang words using twitter's scraping library. We have collected datasets of 7614 tweets for the training purpose and 3264 tweets for testing purposes. After scrapping, collected data transferred to csv files of train and test.

B. Data Preprocessing:

For training text data and fetching better results from the text data, preprocessing the data is a must thing to do. After preprocessing the data the unnecessary text filters out and when we train our model, it will not get any garbage data which will make it learn more efficiently. After collecting the data, when I watched the data of tweets, there I could see abbreviations, slang words which are examples of colloquial expressions but there are also emojis, http links, stopwords. To clean those data I have used tokenization, stopwords removal, special characters removal, emojis removal by converting them to text, and lemmatization by using NLTK, pandas, and re which gives us the basic form of every word.

C. Machine learning:

The processed text we got after cleaning used to train the model like naive bayes, support vector machine and random forest and get the accuracy of those models which they fetch by training those datasets. Moreover, before training processed data and target data, vectorization needs to be done with those data. In this process, the data we vectorize makes the data into vectors which means the data becomes numeric and every model trains this numeric data and gets the accuracy based on this vectorized data. Used TF-IDF vectorizer from scikit-learn library.

Naive Bayes is a model which gives the best performance for the categorical data. Text categorization and spam filtering are two areas where the probabilistic machine learning method Naive Bayes has found widespread use. In order to make predictions, it uses probabilities for several classes, in accordance with Bayes' theorem. The assumption of feature independence given the class label simplifies probability calculations and gives rise to the algorithm's "naive" trait. Several varieties of Naive Bayes exist, each designed to handle a particular kind of data. Some examples are Multinomial, the Gaussian, and Probabilistic Naive Bayes. Its efficacy in managing high-dimensional datasets, computing economy, and simplicity are some of its benefits. But there are certain caveats, such as being sensitive to unimportant traits and assuming that features are independent. Exhibiting its adaptability and practicality, common use cases include text categorization, medical diagnosis, and consumer segmentation. From vectorizing the processed text and target data I have trained the model with the naive bayes model and collected the accuracy.

When it comes to classification and regression, a reliable supervised machine learning approach is Support Vector Machine (SVM). It finds the best possible margin between classes by locating the hyperplane in a three-dimensional space. The kernel method for non-linear bounds, support vectors, and big margins for enhanced generalization are key components. When working with high-dimensional spaces, SVM excels because it is adaptable, uses a variety of kernel functions well, and is less likely to overfit. On the other hand, it may be computationally demanding and noise-sensitive. Demonstrating SVM's adaptability and efficacy in several disciplines, its applications span from bioinformatics to picture and text categorization. To get the accuracy with SVM, use the previous vectorized processed data with data of target and train that data with the SVM model.

Many classification and regression problems make use of Random Forest, an ensemble learning technique. During training, it builds a forest of decision trees and then returns the mean (regression) or mode (classification) of those trees. A different subset of the initial training data and characteristics is used to construct every single tree in the forest. This unpredictability aids in enhancing generalizability and decreasing overfitting. Random Forest can manage big datasets with high dimensionality, and it is well-known for its accuracy and resilience. Though Random forest is good for mixed data, trained the dataset with this model also with those vectorized forms of the data and fetched the accuracy. Used scikit-learn library to train the dataset with the above mentioned models.

D. Deep Learning:

BERT's ability to efficiently collect bidirectional context is derived from its foundation in the transformer architecture. In order to comprehend the connections between words, it employs attention processes to zero down on certain portions of the input sequence.

BERT has learned word contextualized embeddings via extensive pre-training on big corpora.It learns the meaning of words and their contexts via bidirectional word prediction during pre-training. Words used in a more casual setting may have different meanings depending on the surrounding words and phrases. When it comes to identifying slang terms, BERT shines because of its capacity to grasp both context and semantics. It is possible to fine-tune BERT for certain tasks, such as colloquial word identification, after pre-training. The model is fine-tuned so that it fits the specifics of the job or subject at hand. Word meaning is determined by context, according to BERT's contextual embeddings. This helps a lot with activities where the whole context determines how a word is understood. Colloquialisms, informal phrases, and slang may be detected and understood by BERT in a colloquial phrase detection challenge. BERT excels at tasks involving colloquial terms because its rich contextualized representations allow it to capture the nuances of language. Despite its strengths, BERT may struggle to deal with domain-specific slang or phrases that were not in the previous training data because they were out-of-vocabulary. This is the reason BERT gives better accuracy for colloquial types of data than any other machine learning model. To train our data I have used a specific BERT model which has 12 layers with 768 hidden units and 12 attention heads. For compiling the model we have used Binary cross-entropy loss and BinaryAccuracy metrics to classify this binary task which is Colloquial or Non-colloquial sentence. For optimizing the model we have used adamw optimizer and used learning rate to control the step size during weight updates which helps to optimize the model. While creating these models I have changed the batch size and seeds and got the optimal result from batch size 16 with 42 seeds. And trained the model for 20 epochs.

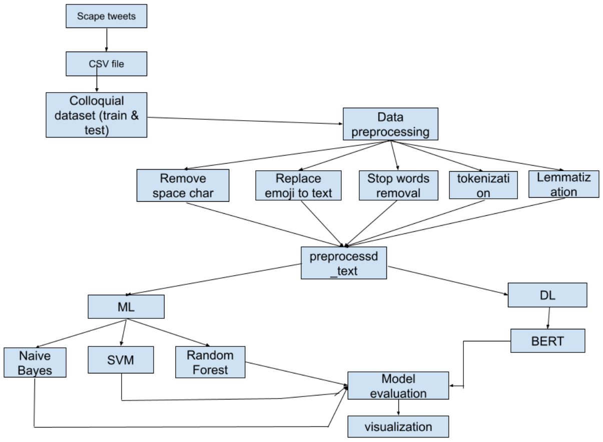

Fig. 1. Data Flow of Methodology

E. Model evaluation:

The metrics like accuracy, F1 score, recall, precision and Auc-roc are the metrics which are used for model evaluation and we have used these metrics to evaluate our machine learning models. For the evaluation purpose I used scikit-learn library's confusion matrix. The percentage of incidents that are accurately labeled is called precision. Accuracy is the percentage of occurrences that were really predicted as positive out of all the cases. The fraction of real positives relative to the total number of positive occurrences are Recall. A balanced measure of accuracy and recall, the F1 score strikes a balance between the two. A helpful metric for binary classification, it measures the area according to the receiver operating feature curve. The numbers of correct, incorrect, and misleading results are shown in a table called a confusion matrix. It gives you the rundown on how well the model worked.

F. Model Visualization:

Interpreting models, troubleshooting them, and communicating with them are all greatly facilitated by visualization. By doing so, stakeholders, academics, and developers may better comprehend the behavior of the model and make educated judgments on its use. For the visualization purpose we have used libraries like seaborn, matplotlib, itertools and scikit-learn.

IV. Result Analysis:

In this project the training dataset has 7614 tweets and the testing dataset has 3264 tweets after cleaning both the dataset was cleaned using lemmatization, special characters, links, stopword removal and tokenization and got preprocessed text for the datasets. Using these preprocessed text and target column of training dataset where 0 means non-colloquial and 1 means colloquial expressions and after splitting them to train and validation parts, vectorized them to train with the machine learning models. Where naive bayes gave me accuracy of 80 percent. Moreover, from the naive bayes confusion matrix we can see that values of precision, recall, f1-score for non-colloquial parts are 0.79, 0.89 and 0.83 and for colloquial parts 0.82, 0.67 and 0.74 respectively. After using the SVM model on the dataset we get the accuracy of 80 percent. Additionally, from the SVM confusion matrix we get that the values of precision, recall, f1-score for non-colloquial parts are 0.78, 0.90, 0.84 and for colloquial parts are 0.83, 0.66, 0.74 respectively. On the training dataset after using the random forest model, the accuracy is 0.79. In addition, from the random forests classification report we see that the values of precision, recall, f1-score for non-colloquial parts are 0.79, 0.85, 0.82 and for colloquial parts are 0.77, 0.69, 0.73 respectively. So we see that accuracy wise naive bayes and SVM both give the same accuracy of 80 percent and random forest lacks with 79 percent.

TABLE I
Accuracy rate

Result Analysis	
Machine Learning Models	Accuracy
Naive Bayes	80%
SVM	80%
Random Forest	79%

The BERT model has 12 layers with 768 hidden units and 12 attention heads and for compiling used BinaryCrossEntropy and BinaryAccuracy metrics, 'adamw' as the optimizer and used learning rate to control the step size. After training the model for 20 epochs, it got 98.57% accuracy (last epoch accuracy). The peak Binary Accuracy during these 20 epochs was 98.58% which we got at 19th epoch.

1st epoch accuracy was 72.4 percent and the loss at that time was 0.5726. Though the 19th epoch has the highest accuracy, the loss was lowest on the last epoch, which was 0.0297.

Deep Learning Model	Accuracy
BERTs	98.6%

After running the dataset with these machine learning models (Naive Bayes, SVM and Random Forest) and deep learning model, we can see that Deep learning model (BERT) performs far better than machine learning models for categorical data like colloquial expressions detection.

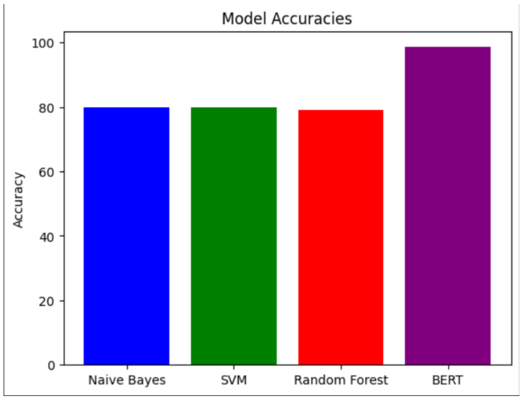

Fig. 2. Model Accuracies

V. CONCLUSION:

Colloquial expressions have limited available datasets. After collecting the data and training with models like Naive Bayes, SVM, and Random Forest we got the accuracy of 80%, 80% and 79% respectively. And the Deep learning model BERT has trained the same data and provided an accuracy of 98.6%. So, for categorical data like colloquial expressions Deep learning models give much better accuracy than machine learning models.

REFERENCES

[1] Lukas Santing, Sijstermans, R., G. Anerdi, Jeuris, P., Marijn ten Thij, & Riza Batista-Navarro. (2022). Food for Thought: How can we exploit contextual embeddings in the translation of idiomatic expressions? https://doi.org/10.18653/v1/2022.flp-1.14

[2] Zhou, J., Zeng, Z., & Bhat, S. (2023). CLCL: Non-compositional Expression Detection with Contrastive Learning and Curriculum Learning. https://doi.org/10.18653/v1/2023.acl-long.43

[3] Pei, Z., Sun, Z., & Xu, Y. (2019). Slang detection and identification (pp. 881–889). https://aclanthology.org/K19-1082.pdf

[4] Itkonen, S., Tiedemann, J., & Creutz, M. (2022, July 1). Helsinki-NLP at SemEval-2022 Task 2: A Feature-Based Approach to Multilingual Idiomaticity Detection (G. Emerson, N. Schluter, G. Stanovsky, R. Kumar, A. Palmer, N. Schneider, S. Singh, & S. Ratan, Eds.). ACLWeb; Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.semeval-1.14