UTB-NLP at SemEval-2023 Task 3: Weirdness, Lexical Features for Detecting Categorical Framings, and Persuasion in Online News

Juan Cuadrado and **Elizabeth Martinez** and **Anderson Morillo** and **Daniel Peña**
Kevin Sossa and **Juan Carlos Martinez-Santos** and **Edwin Puertas**
Universidad Tenologica de Bolivar, Cartagena Colombia
epuerta@utb.edu.co

Abstract

Nowadays, persuasive messages are increasingly frequent in social networks, which generates particular concern in several communities, given that persuasion seeks to guide others towards adopting ideas, attitudes, or actions that they consider to be beneficial to themselves. The efficient detection of news genre categories, detection of framing, and detection of persuasion techniques require several scientific disciplines, such as computational linguistics and sociology. Here we illustrate how we use lexical features given a news article to determine whether it is an opinion piece, aims to report factual news, or is satire. This paper presents a novel strategy for communication based on Lexical Weirdness. The results are part of our participation in Sub-Tasks 1 and 2 in SemEval 2023 Task 3.

1 Introduction

Frame identification is a critical task in natural language processing (NLP) that recognizes the frames or semantic structures used in an article or text. In the SemEval 2023 challenge context, we propose the task of frame detection in news articles. In this case, we selected the English language. This task is essential because it allows a better understanding of the perspective and tone of news articles and other media. In addition, it can help detect bias and improve accuracy in news and opinion article classification, which is relevant for applications such as news monitoring, sentiment analysis, and text classification. In particular, the frame detection challenge in which this paper seeks to promote the development of new NLP techniques and models for frame identification in complex and realistic texts and builds on the dataset and evaluation presented in the challenge overview article.

The approach presented in this paper is Lexical Weirdness. This approach seeks a more appropriate lexicon for the domain as we generated these lexicons focusing on having a more appropriate vocabulary. The first thing that we did was to create a corpus associated with each of the Framing categories. Then, we scrapped Wikipedia-related articles.

Subsequently, we extracted the nominal unigrams of these articles. After acquiring the unigrams for each category, we calculated the relevance of words within the context by comparing the lexicon of each context with the lexicon generated by Google. Finally, we selected the words with the highest occurrence, which had the most significant relevance in the context, to become the specific lexicon for each domain we implemented in a bag of words model.

Finally, the frequency of words per category was obtained and combined with the frequencies of unigrams using the Weirdness technique to generate a lexicon. From this lexicon constructed in the previous phases, we tested different classification techniques to obtain the prediction of frames in news articles.

The rest of the paper is structured as follows. First, we introduce the related work in Section 2. Then, Section 3 presents the details of the proposed strategy. In Section 4 and Section 5, we discuss the setup and analysis of the experiment of results. We conclude in Section 6 with remarks and future work.

2 Background

2.1 Sub-Task 1

The system receives news articles as input in the task setup, already filtered from any HTML or special characters. Then, the articles undergo feature extraction based on linguistics, and the extraction of noun phrases, vectorized using Term Frequency - Inverse Document Frequency (TF-IDF) (Kim et al., 2019). Additionally, we oversample the data using

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1551–1557

SMOTE (Camacho et al., 2022).

For the news genre categorization, we only used the English dataset, which consisted of 433 labeled elements for training and 83 unlabeled aspects for development. In addition, two articles focus on the classification of satire news vs. reportage. The other focuses on the distinction between opinion and reportage, which were reliable signals to identify satire news and reportage.

(Hassan et al., 2020) uses various feature extractions such as The Absurdity Feature (Abs), Humor (Hum), text Processing and Feature Weighting, Grammar (Gram), Negative effect (Neg), and Punctuation (Pun) to identify satire within the text. The system's basis proposes text vectorization with then TF-IDF.

(Yang et al., 2017) proposes a system that uses both paragraph-level and document-level linguistic features to identify satire. The attributes for the document level include Psycholinguistic, Writing Stylistic, Readability, and Structural characteristics that distinguish how a journalist writes conservatively versus a satirical text that intentionally uses a humorous and aggressive message to entertain.

(Alhindi et al., 2020) proposes the classification between a report and an opinion and suggests feature extractions such as Linguistic, Document-level Contextualized Embeddings, and Argumentation.

All these articles have in common the extraction of linguistic features, which focuses on the document level. Therefore, this study focuses on this feature and uses the vectorization of the noun phrases with TF-IDF as a basis for the components.

Compared to these articles, we propose implementing linguistic feature extraction for a multi-class classification system (single label) for reportage, option, and satire classes. In addition, we present the extraction of noun phrases.

2.2 Sub-Task 2

We designed the system to receive news articles as input in the task setup, previously filtered from any HTML or special characters. We did feature extraction based on linguistics and the extraction of noun phrases, which are then vectorized using Term Frequency - Inverse Document Frequency (TF-IDF) (Kim et al., 2019). To ensure balanced training data, we also employ the SMOTE oversampling technique (Camacho et al., 2022).

For news genre categorization, we used only the English dataset consisting of 433 labeled elements for training and 83 unlabeled aspects for development. Two articles focus on the classification of satire news versus reportage. In contrast, the other focuses on distinguishing between opinion and reportage, which are reliable signals to identify satire news and reportage.

In their work, (Hassan et al., 2020) uses various feature extractions such as The Absurdity Feature (Abs), Humor (Hum), text Processing and Feature Weighting, Grammar (Gram), Negative effect (Neg), and Punctuation (Pun) to identify satire within the text, and the system's basis proposes text vectorization using TF-IDF.

Meanwhile, (Yang et al., 2017) proposes a system that uses both paragraph-level and document-level linguistic features to identify satire. The attributes for the document level include Psycholinguistic, Writing Stylistic, Readability, and Structural characteristics that distinguish how a journalist writes conservatively versus a satirical text that intentionally uses a humorous and aggressive message to entertain. On the other hand, (Alhindi et al., 2020) proposes a classification between a report and an opinion, suggesting feature extractions such as Linguistic, Document-level Contextualized Embeddings, and Argumentation.

All of the mentioned articles have the extraction of linguistic features in common, which focuses on the document level. Therefore, this study aims to focus on this feature and utilize the vectorization of the noun phrases with TF-IDF as a basis for the components.

Compared to the existing literature, we propose the implementation of linguistic feature extraction for a multi-class classification system (single label) for reportage, opinion, and satire classes. Moreover, we also present the extraction of noun phrases as a critical feature for the classification task.

3 System Overview

This section outlines the predictive models utilized in our proposed solutions for Task 3- Sub-Tasks 1 and 2 of SemEval 2023. This task involves detecting the genre, framing, and persuasion techniques in online news across multiple languages.

Using a four-step method, we employed the models proposed for solving Sub-Tasks 1 and 2 to identify elements contributing to text persuasiveness, including genre and framing of opinion, report, or satire.

The first step involves pre-processing the

data(Puertas et al., 2019), the second step uses feature extraction(Puertas and Martinez-Santos, 2021) that we regularized in the third step (Puertas et al., 2021). Then the regularized data is submitted through a classifying voting system evaluated using F1-score macro and F1-score micro for Sub-Tasks 1 and 2, respectively.

Figure 1 shows the proposed pipeline for both Sub-Task. This pipeline comprises the following stages: pre-processing, feature extractions, regularization, classifiers, and evaluation. This section outlines the predictive models utilized in our proposed solutions for Task 3, Sub-Tasks 1 and 2 of SemEval 2023, which involves detecting the genre, framing, and persuasion techniques in online news across multiple languages.

To identify elements contributing to text persuasiveness, including genre and framing of opinion, report, or satire, we employed the models proposed for solving Sub-Tasks 1 and 2 using a four-step method. The first step involves pre-processing the data (Puertas et al., 2019), followed by feature extraction (Puertas and Martinez-Santos, 2021), which is then regularized in the third step (Puertas et al., 2021). Finally, we submitted the regularized data through a classifying voting system evaluated using F1-score macro and F1-score micro for Sub-Tasks 1 and 2, respectively.

Figure 1 illustrates the proposed pipeline for both Sub-Tasks, which includes pre-processing, feature extraction, regularization, classifiers, and evaluation.

3.1 Data Description

The dataset has three labeled classes: opinion, report, and satire, each indicating the genre of the news article. In addition, the framing detection Sub-Task 3 requires additional labels for the framing techniques used in the article, such as appeal to authority, fear, or urgency.

The dataset has been pre-processed to remove any HTML or special characters and is available in a balanced form, with an equal number of articles for each class. The development set contains 83 articles, and the test set contains 100 articles in the same format as the training set.

The dataset also includes metadata for each article, such as the date of publication, source, and language, which we can use for additional analysis.

3.2 Pre-processing

In the pre-processing stage, we applied a series of techniques using the Natural Language Toolkit (NLTK) library to eliminate any noise or distortion that could hinder our models' accuracy.

First, the text was converted to **lowercase**, ensuring that words with uppercase and lowercase letters were identical. Next, we removed **unwanted characters** such as URLs, mentions, retweets, and non-alphabetic characters to ensure that only meaningful text was analyzed.

Word tokenization was then employed to split the text into individual words, allowing for more granular text analysis. Additionally, **empty words**, also known as stop words, were removed. These words, such as "the," "and," and "in," have little additional meaning and can be safely removed without affecting the overall analysis.

Finally, we used **lemmatization** to convert words to their base form, which helped to reduce variability and ensure the model's accuracy.

The above techniques have proven crucial in effectively pre-processing text for analysis in natural language processing tasks. By removing unwanted characters and empty words, the analysis can focus on the most relevant aspects of the text, leading to more accurate results.

The application of lemmatization also reduces the complexity of the text, making it easier to analyze and interpret. Overall, this pre-processing stage helps to ensure that the analysis is conducted on the most pertinent aspects of the text, thereby producing reliable results.

3.3 Feature Extraction

In this phase, we describe the feature extraction approach implemented for each Sub-Task.

3.3.1 Sub-Task 1

To identify linguistic features contributing to text persuasiveness, we utilized techniques outlined in the literature, such as average sentence and token length, character count, normalized frequencies of negation and negation-suffix, and the ratio of ending character per sentence, as demonstrated in (Yang et al., 2017). We implemented these techniques using the Natural Language Toolkit (NLTK) (Arora, 2020). In addition, we leveraged sentiment and polarity analysis through the text blob library (Hazarika et al., 2020).

To ensure we don-t lost critical information during pre-processing, we employed several tech-

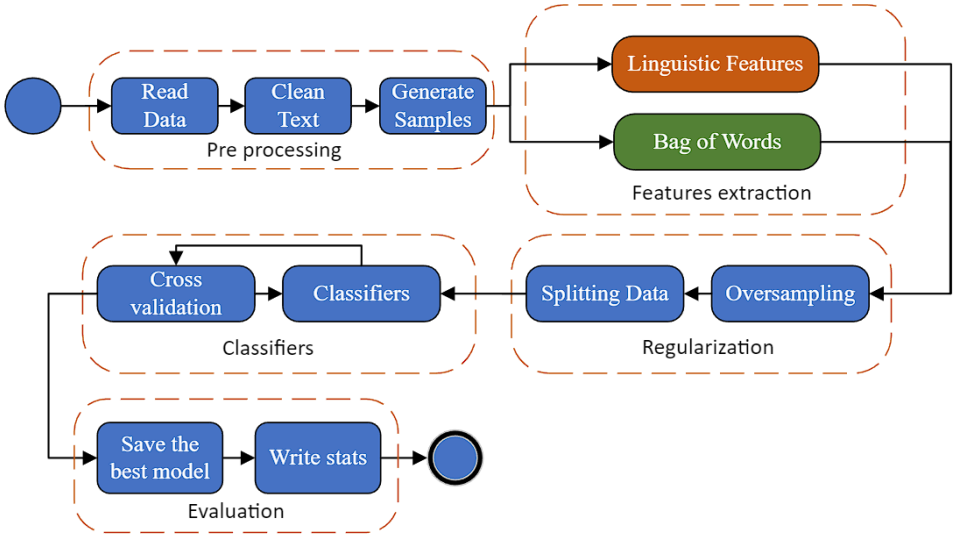

Figure 1: System General Pipeline

niques, including conversion to lowercase, removal of special characters like "/n," replacement of contradictions, removal of stop words, and lemmatization of text. We also identified noun phrases using text blob and vectorized them using Term Frequency-Inverse Document Frequency (TF-IDF) before merging them with the extracted features

3.3.2 Sub-Task 2

We conducted the feature extraction process in two stages: data collection and text processing. In the data collection stage, we employed web scraping techniques to gather relevant texts related to the categories provided in the challenge. We obtained links for each category from Wikipedia and utilized the Beautifulsoup library (Patel and Patel, 2020) to extract relevant text from these links to obtain the required texts.

The next stage involved text pre-processing to clean up the extracted text. This process included removing special characters and common words and tokenizing the English text using the Natural Language Toolkit (NLTK) library (Bird, 2006). Furthermore, we utilized the Weirdness technique to compare the frequency of the words obtained with that of the English unigrams, which helped generate a lexicon. Finally, the developed lexicon was utilized in the Bag-of-Words (BOW) technique to represent the data and classify the text frames.

3.4 Regularization

The regularization process employed in our proposed model involves two crucial steps: class balancing and data splitting into training and validation sets. In order to balance the classes, we utilized the Synthetic Minority Over-sampling Technique (SMOTE), which generates synthetic data points for the minority class to create a balanced dataset. For the data splitting process, we adopted the K-Fold StratifiedShuffleSplit method introduced by Sandoval et al. (2020), which partitions the data into training and validation sets in a way that preserves the distribution of classes in both sets. It helps to prevent the model from overfitting to the training data and ensures its generalizability to new data. The model is trained on the training set and then evaluated on the validation set to assess its performance.

3.5 Classifiers

We have introduced a voting mechanism to enhance the classification performance of our proposed model. We hypothesized that combining multiple classifiers would produce a more accurate final verdict. At the configuration stage, we utilized the Python lazy classifier to identify the top three classifiers for each category based on the majority gain criteria. In this approach, each classifier casts a vote for a predicted class, and the final prediction is determined based on the majority of votes. This method helps to improve the robustness and accuracy of the classification model. Therefore, we evaluated the system's performance using the best classifiers for each category as determined by the voting process. This approach ensures the classification model is optimized for each category and

can generalize well to new data.

3.6 Evaluation

The test dataset was read and pre-processed during the evaluation stage by generating samples from each article. Subsequently, features were extracted based on the Sub-Tasks and passed through a voting system that utilized the best classifiers for each category. To identify the best model, we performed cross-validation in all the models. We evaluated the system's performance and saved the results. Finally, we performed a multi-label classification using the best model. Using a voting system helped improve the accuracy of the final verdict by combining the predictions of multiple classifiers.

4 Experimental setup

4.1 Sub-Task 1

Due to the limited number of satire articles in our data set, we pre-processed the data. Then we applied the Synthetic Minority Over-sampling Technique (SMOTE) to increase the number of examples. Initially, there were 432 articles, with ten being satirical, while the remaining 382 were opinion articles, and 41 were reporting articles. After applying SMOTE, we generated 1146 articles, with 389 articles for each type. Finally, we used the train-test split method to divide the articles into three sets: 115 articles for development, 1028 for training, and 54 for testing.

Table 1: Summary of the libraries implemented

Library	Version
Pandas	1.3.3
NumPy	1.19.5
Matplotlib	3.4.3
Seaborn	0.11.2
Scikit-learn	1.0
NLTK	3.6.5
Lazypredict	0.2.9
TQDM	4.62.3
Imbalanced-learn	0.8.1
Keras	2.6.0
LightGBM	3.3.2
Imblearn	0.0
Spacy	3.4.4
Negspacy	1.0.3
textblob	0.17.1

4.2 Sub-Task 2

In order to ensure a rigorous and replicable evaluation of our AI model, we employed a database containing plain-text news and web articles, each stored in a TXT file along with the article's title, if any. To partition the data into training, development, and testing sets, we strategically divided the dataset into 433 items (76%) for the training set, 83 items (15%) for the development set, and 54 items (9%) for the testing set. During the training phase, the model was trained on the training set to identify patterns and relationships in the data and optimize its parameters for improved accuracy.

We used the development set to fine-tune the model's hyperparameters and evaluate its accuracy before the final testing phase. Finally, we used the test set to objectively evaluate the model's effectiveness in terms of accuracy and compare its performance with existing models. It is noteworthy that the three datasets were kept separate throughout the training and evaluation process to ensure an objective and accurate evaluation of our model.

To facilitate the replication of our experiments, we provide a detailed list of the tools and libraries used in this work in Table 1, including their version numbers and URLs.

5 Results

5.1 Sub-Task 1

The performance of the news genre categorization model could have been better compared to the baseline. It is evident from the F1 macro differential of 4.5% and F1 micro differential of 3.7%. Furthermore, the model's rank was low at 21 out of 23, lagging behind the baseline by five spots, as depicted in Table 2.

Table 2: Ranking of results in news Genre Categorisation

Lang	F1 micro	F1 macro
EN	0.57407	0.24314

The inadequacy of data was one of the primary reasons behind the non-representative results presented in Table 2. Only ten articles were available for satire and 41 for reporting. To tackle this, we resorted to oversampling and chose the top-performing models using LazyPredict.

Table 3 summarizes the model's performance after oversampling and using the development data.

Table 3: Summary of result using the development data.

Model performance	F1	Acc
Opinion	98	96
Reporting	99	100
Satire	100	100

Again, the results indicate a significant improvement in the model's performance, as evidenced by high F1 and accuracy scores for all three classes.

The model tended to extract more information from opinion articles, leading to more predictions for this class. However, oversampling with satire and reporting caused the models to overfit the data. To enhance the model's learning feedback and enable it to extract more characteristics, we require further investigation to acquire more satire and reporting articles.

5.2 Sub-Task 2

We evaluated the performance of the framing detection models using both the training and test data sets. We present the results in Table 4. Our model was ranked 19th on the scale of the best results.

Table 4: Ranking of results in framing detection classification

Lang	F1 micro	F1 macro
EN	0.34112	0.30908

In addition, Table 5 summarizes the performance of the voting system for each category in terms of F1 and accuracy scores. The highest-performing category was Cultural Identity, with an F1 score of 94.38 and an accuracy score of 94.38. In contrast, the lowest-performing category was Crime and Punishment, with an F1 score of 67.72 and an accuracy score of 67.72. These results suggest that the voting system is adequate for detecting framing in specific categories, and further improvements may be needed to enhance its performance in other categories.

6 Conclusion

In conclusion, the research conducted in this paper focused on the prediction of topic frames in news articles. The proposed methodology involved creating a vocabulary of topic frames and using different classification techniques, which showed the potential to improve the system's performance.

Table 5: Summary of results in framing detection classification

Voting Systems by Categories	F1	Acc
External regulation and reputation	84.55	84.55
Morality	73.39	73.39
Cultural identity	94.38	94.38
Security and defense	63.66	63.66
Quality of life	88.55	88.55
Policy prescription and evaluation	82.93	82.93
Economic	90.69	90.69
Fairness and equality	87.42	87.42
Crime and punishment	67.72	67.72
Public opinion	93.71	93.71
. Health and safety	92.37	92.37
Political	72.37	72.37
Legality Constitutionality and jurisprudence	68.17	68.17
Capacity and resources	92.52	92.52

However, as the results have shown, the unbalanced and limited training data and the use of a linguistic classification model limited the system's performance. To overcome these limitations and enhance the system's accuracy and robustness, we require more diverse and balanced data and explore other natural language processing models and techniques, such as deep learning.

Furthermore, additional linguistic features, such as noun phrases, can be used to improve model performance. Finally, the results indicated that the system tended to extract more information from opinion articles, resulting in more predictions for this class. We could address this by exploring more balanced data and feature extraction techniques.

Overall, this work provides a solid foundation for future advances and developments in topic frame prediction in news articles. Furthermore, it emphasizes the importance of acquiring more diverse and balanced data and exploring different natural language processing models and techniques to improve system performance.

Acknowledgments

To the SemEval contest, sponsored by the SIGLEX Special Interest Group on the Lexicon of the Association for Computational Linguistics. To the mas-

ter's degree scholarship program in engineering at the Universidad Tecnológica de Bolivar (UTB) in Cartagena, Colombia.

References

Tariq Alhindi, Smaranda Muresan, and Daniel Preotiuc-Pietro. 2020. Fact vs. opinion: the role of argumentation features in news classification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6139–6149, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Gaurav Arora. 2020. iNLTK: Natural language toolkit for indic languages. In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*. Association for Computational Linguistics.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Luís Camacho, Georgios Douzas, and Fernando Bacao. 2022. Geometric smote for regression. *Expert Systems with Applications*, 193:116387.

Isyaku Hassan, Mohd Nazri Latiff Azmi, and Akibu Mahmoud Abdullahi. 2020. Evaluating the spread of fake news and its detection. techniques on social networking sites. *Romanian Journal of Communication and Public Relations*, 22(1):111–125.

Ditiman Hazarika, Gopal Konwar, Shuvam Deb, and Dibya Jyoti Bora. 2020. Sentiment analysis on twitter by using textblob for natural language processing. *ICRMAT*, 24:63–67.

Donghwa Kim, Deokseong Seo, Suhyoun Cho, and Pilsung Kang. 2019. Multi-co-training for document classification using various document representations: Tf–idf, lda, and doc2vec. *Information Sciences*, 477:15–29.

Jay M Patel and Jay M Patel. 2020. Web scraping in python using beautiful soup library. *Getting Structured Data from the Internet: Running Web Crawlers/Scrapers on a Big Data Production Scale*, pages 31–84.

Edwin Puertas and Juan Carlos Martinez-Santos. 2021. Phonetic detection for hate speech spreaders on twitter notebook for pan at clef 2021. *CEUR Workshop Proceedings*.

Edwin Puertas, Luis Gabriel Moreno-Sandoval, Flor Miriam Plaza-del Arco, Jorge Andres Alvarado-Valencia, Alexandra Pomares-Quimbaya, and L Alfonso. 2019. Bots and gender profiling on twitter using sociolinguistic features. *CLEF (Working Notes)*, pages 1–8.

Edwin Puertas, Luis Gabriel Moreno-Sandoval, Javier Redondo, Jorge Andres Alvarado-Valencia, and Alexandra Pomares-Quimbaya. 2021. Detection of sociolinguistic features in digital social networks for the detection of communities. *Cognitive Computation*, 13:518–537.

Fan Yang, Arjun Mukherjee, and Eduard Dragut. 2017. Satirical news detection and analysis using attention mechanism and linguistic features. *arXiv preprint arXiv:1709.01189*.