Oppositional Thinking Analysis: Conspiracy Theories vs Critical Thinking Narratives

Notebook for PAN at CLEF 2024

Prabavathy Balasundaram[1,†], Karthikeyan Swaminathan[1,*,†], Oviasree Sampath[1,†] and Pradeep Km[1,†]

[1]*Department of CSE, SSN College of Engineering, Rajiv Gandhi Salai, Chennai, Tamil Nadu, India*

Abstract

Conspiracy theories [1] are complex narratives that attempt to explain the ultimate causes of significant events as cover plots orchestrated by secret, powerful, and malicious groups, whereas critical thinking on the other hand is the process of objectively analyzing and evaluating information to form a reasoned judgment and putting them forth for the public view. Identifying conspiracy theories using Natural Language Processing (NLP) models is challenging because it is hard to tell them apart from critical thinking. Mislabeling critical messages as conspiratorial can push curious individuals towards conspiracy communities, and hence it is highly important to be accurate in such classifications. The task involves distinguishing between two types of oppositional narratives: (1) conspiracy narratives, which suggest secret plots by powerful, malicious groups, and (2) critical thinking narratives, which question major decisions without implying a conspiracy. To achieve subtask 1, a pre-trained BERT classifier is employed to differentiate between the two classes using a sigmoid activation function. The model for subtask 2 is a pretrained BERT-based sequence classifier fine-tuned for multi-label classification, which enables span-level classification of oppositional narratives. This working note paper presents the results of the Kaprov team at the Oppositional thinking analysis: Conspiracy theories vs critical thinking narratives [2] of PAN at CLEF 2024 [3],which includes two subtasks.

Keywords

BERT, Multi-label classification, Conspiracy Theories (CTs), Tokenizer

1. Introduction

In the realm of Natural Language Processing, the computational detection and analysis of conspiracy theories (CTs) within textual data has gained significant momentum [4]. CTs are elaborate narratives attributing significant events to covert actions by powerful clandestine groups, contrasting with critical thinking, which challenges mainstream beliefs without endorsing conspiracies. Differentiating between these is crucial, as mislabeling opposing views as conspiratorial may sway individuals towards extreme viewpoints. Current research predominantly focuses on binary classification tasks aimed at accurately distinguishing between conspiratorial and critical texts. Existing methodologies for distinguishing between conspiratorial and critical texts typically involve leveraging advanced natural language processing techniques and machine learning models. Some common approaches include:

- Feature-based Classification: Using algorithms like SVMs (Support Vector Machines) [5] or logistic regression, which analyze word frequencies, n-grams, and syntax to classify texts.
- Graph-based Methods: Representing texts as graphs, where nodes represent entities (e.g., words or phrases) and edges represent relationships (e.g., co-occurrence). Graph-based methods can capture structural patterns and semantic relationships indicative of conspiratorial or critical narratives.

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

*Corresponding author.

†These authors contributed equally.

prabavathyb@ssn.edu.in (P. Balasundaram); karthikeyan2210394@ssn.edu.in (K. Swaminathan); oviasree2210386@ssn.edu.in (O. Sampath); pradeep2210432@ssn.edu.in (P. Km)

- Sentiment Analysis: Analyzing the sentiment expressed in texts can provide insights into whether the text is promoting conspiratorial beliefs (e.g., distrust, fear) or engaging in critical discourse (e.g., skepticism, questioning).

Subtask 2 focuses on token-level classification within oppositional narratives, distinguishing between conspiracy theories and critical thinking. It aims to identify specific text segments—goals, effects, agents, facilitators, objectives, and negative effects—using advanced NLP techniques. This approach enhances understanding of nuanced narrative elements for effective content moderation and societal discourse analysis. The approach includes:

- Topic Modeling: Techniques like Latent Dirichlet Allocation (LDA) [6] or Non-Negative Matrix Factorization (NMF) [7] can uncover latent topics within texts, revealing prevalent themes or ideologies associated with conspiratorial or critical narratives. By identifying dominant topics, these methods contribute to understanding the discourse's thematic focus and distinguishing between different narrative types.
- Contextual Embeddings: Utilizing pre-trained contextual embeddings like BERT (Bidirectional Encoder Representations from Transformers) or GPT (Generative Pre-trained Transformer) can capture nuanced contextual information within texts, enabling models to discern subtle linguistic cues indicative of conspiratorial versus critical narratives. These embeddings enhance model performance by integrating rich contextual understanding into classification tasks.

The two tasks discussed in this paper and their successful implementation collectively advance the field by enabling automated detection and analysis of conspiratorial narratives, facilitating nuanced understanding and effective management of such discourse in various domains, and thereby create stable and peaceful platforms for discussion on public health issues.

2. Task and Dataset Description

There are two tasks that have been worked upon, the first involves distinguishing between critical and conspiracy texts, while the second focuses on detecting elements within oppositional narratives. The dataset contains Telegram messages in English and Spanish. Both the tasks are performed exclusively in English.

Subtask 1 involves classifying texts into two categories: (1) messages that critically question public health decisions without promoting conspiracy theories, and (2) messages that attribute pandemic or health decisions to secret, influential conspiracies. Each text in the dataset is labeled as either CONSPIRACY or CRITICAL. Evaluation of model performance is done based on Matthews Correlation Coefficient (MCC) [8], with a baseline established by a BERT classifier [9].

Subtask 2 involves a token-level classification challenge where the goal is to identify specific text segments that represent essential elements in oppositional narratives. Each text data in the input dataset contains span texts along with their starting and ending positions and the type of oppositional narrative the span text belongs to out of: AGENT, FACILITATOR, VICTIM, CAMPAIGNER, OBJECTIVE, and NEGATIVE EFFECT. The performance of models is evaluated using the macro-averaged span-F1 score, which assesses overall accuracy across all span categories.

3. Data Pre-Processing

This section outlines the process of preparing data for the two tasks.

3.1. Subtask 1 : Distinguishing between critical and conspiracy texts

In the data pre-processing stage for subtask 1, the dataset is initially split into two subsets: one for critical messages and another for conspiracy messages . Each subset is filtered based on the "category" column

values. Exploratory Data Analysis (EDA) [10] begins with a count plot to visualize the distribution of categories ("CRITICAL" and "CONSPIRACY") [Fig. 1]. This provides an initial understanding of the dataset's class distribution. Following EDA, data cleansing involves checking for missing values. Addressing any missing data ensures the dataset is ready for subsequent steps such as tokenization, feature extraction, and model training for binary classification.

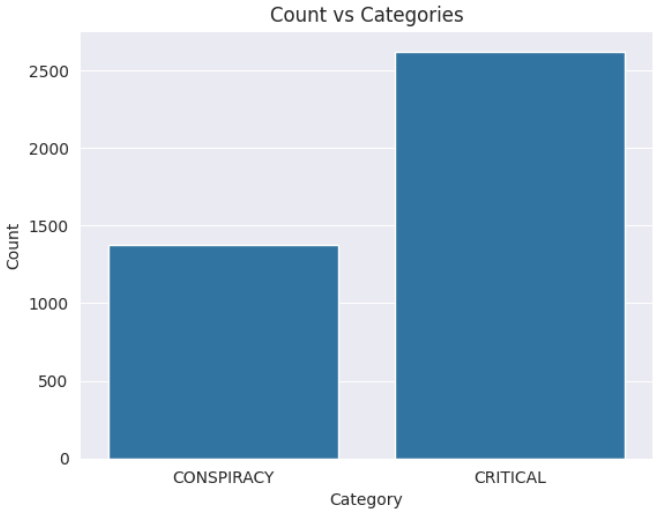

Figure 1: Visualization of category distribution in the dataset.

3.2. Subtask 2 : Detecting elements of the oppositional narratives

Subtask 2 pre-processing starts with extracting annotations from each JSON (JavaScript Object Notation) entry, gathering crucial details about the relevant text spans and their corresponding categories. This step prepared the dataset for subsequent pre-processing, ensuring its alignment with the machine learning pipeline. Post annotation extraction, the Hugging Face AutoTokenizer [11] tailored for BERT models was employed to tokenize the dataset. Tokenization converted raw text sequences into numerical token IDs suitable for ingestion by the BERT-based model. To meet BERT's input specifications, a truncation strategy was applied to handle sequences exceeding the model's maximum input length. This approach maintained consistency in sequence lengths across the dataset, optimizing computational efficiency during training and evaluation phases.

4. Methodologies Used

4.1. Tiny BERT Text Classifier

The *Tiny BERT Text Classifier model* [12] is a variant of BERT optimized for English text classification tasks, specifically focusing on the SST (Stanford Sentiment Treebank)-2 dataset [13] for sentiment analysis. Built on transformer architecture, this model enables bidirectional understanding of language nuances, enhancing accuracy in classifying sentences as either critical or conspiracy in nature. This capability is crucial for distinguishing between texts that question public health decisions (critical) and those that attribute them to malevolent conspiracies (conspiracy). By leveraging bidirectional context, these models can capture subtle linguistic cues that differentiate between these two types of narratives effectively.

4.2. Enhanced Multi-label BERT Classifier

Methodologies of subtask 2 typically involve initial dataset preparation by sourcing annotated text spans and categorizing them for training, validation, and test sets to ensure unbiased model evaluation. Utilizing tools like AutoTokenizer from Hugging Face's Transformers library [11], raw text sequences are tokenized into numerical token IDs, with strategies like truncation and padding managing sequence lengths. Model selection focuses on transformer-based architectures pretrained on extensive text corpora, fine-tuned for span-level classification using transfer learning techniques. Training optimizes model parameters with Adam optimizer and Binary Cross-Entropy loss [14], while evaluation metrics such as span-level F1-score, precision, recall, and micro-averaged F1-score assess model performance.

5. Implementation

To implement subtask 1, the dataset is structured into a format where each text sample is categorized either as "CRITICAL" or "CONSPIRACY". The BertClassifier model from keras-nlp.models is then employed with specific configurations for binary classification. Pre-trained weights are loaded, and a sigmoid activation function is utilized to facilitate binary output. The model is trained on the training data to distinguish between critical viewpoints and conspiracy theories regarding public health decisions. Evaluation is performed on the test set to assess the model's capability in accurately classifying these texts. This approach leverages the capabilities of BERT for semantic understanding, thereby supporting the task's objective of discerning between critical analyses and conspiratorial narratives in the domain of public health.

The BERT-Based Multi-Label Text Classifier was implemented in Python using the bert-base-uncased model architecture from Hugging Face's Transformers library. The dataset, sourced from JSON files, contained annotated text spans (span text) categorized into specific classes (category). After partitioning the dataset into training (70%), validation (10%), and test (20%) sets, annotations were extracted to prepare the data for tokenization. The Hugging Face AutoTokenizer [11] was employed to tokenize the text sequences into numerical token IDs, with a truncation strategy applied to handle sequences longer than BERT's maximum input length. The model was fine-tuned for multi-label classification, optimizing with the Adam optimizer and Binary Cross-Entropy loss function [14] over multiple epochs. Evaluation on the validation set involved monitoring metrics such as accuracy, precision, recall, and F1-score to ensure model performance. Finally, the trained model and tokenizer were saved for deployment, emphasizing reproducibility and scalability in future applications. The BERT-Based Multi-Label Text Classifier was implemented in Python using the bert-base-uncased model architecture from Hugging Face's Transformers library. The dataset, sourced from JSON files, contained annotated text spans (span text) categorized into specific classes (category). After partitioning the dataset into training (70%), validation (10%), and test (20%) sets, annotations were extracted to prepare the data for tokenization. The Hugging Face AutoTokenizer was employed to tokenize the text sequences into numerical token IDs, with a truncation strategy applied to handle sequences longer than BERT's maximum input length. The model was fine-tuned for multi-label classification, optimizing with the Adam optimizer and Binary Cross-Entropy loss function over multiple epochs. Evaluation on the validation set involved monitoring metrics such as accuracy, precision, recall, and F1-score to ensure model performance. Finally, the trained model and tokenizer were saved for deployment, emphasizing reproducibility and scalability in future applications.

6. Results and Analysis

Based on the provided results for subtask 1 and subask 2 in English, the performance of the BERT-Based Multi-Label Text Classifier was evaluated. For subtask 1 [Table 1], focusing on conspiracy and critical categorization, the model achieved an F1-macro score of 0.3700 and 0.8255, respectively, indicating moderate performance in identifying critical texts compared to conspiracy-related ones . In

subtask 2 [Table 2], which evaluated span-level F1-score and micro-averaged F1, the model attained scores of 0.0150 and 0.0600, respectively, suggesting challenges in precise span-level predictions. The implementation utilized Python with the bert-base-uncased model from Hugging Face's Transformers library, leveraging AutoTokenizer for tokenization and fine-tuning with Adam optimizer and Binary Cross-Entropy loss. The results underscore the model's effectiveness in critical text classification but highlight areas for improvement in span-level prediction accuracy.

Table 1
Subtask 1 : Distinguishing between critical and conspiracy texts.

Metric	Value
MCC	0.3700
F1-MACRO	0.6240
F1-CONSPIRACY	0.4224
F1-CRITICAL	0.8255

Table 2
SUubtask 2 : Detecting elements of the oppositional narratives.

Metric	Value
span-F1	0.0150
span-P	0.0261
span-R	0.0165
micro-span-F1	0.0600

7. Conclusion

The task Oppositional Thinking Analysis: Conspiracy vs Critical tackles the challenge of distinguishing conspiratorial from critical narratives in oppositional texts, especially regarding COVID-19. Conspiracy theories, often depicting events as manipulated by secretive, powerful groups, are complex and hard to separate from genuine critical thinking. The competition aims to enhance understanding and automatic detection of these narratives, crucial for content moderation on social media. Differentiating conspiratorial messages from critical ones is essential, as mislabeling can push individuals toward conspiracy communities. This task involved developing sophisticated NLP models to discern these nuances for accurate classification and better content management. The approach included preprocessing steps like text cleaning and feature extraction using TF-IDF (Term Frequency-Inverse Document Frequency) [15] and word embeddings. Both traditional machine learning algorithms, such as logistic regression and support vector machines, and advanced deep learning models, like LSTM (Long Short-Term Memory) and BERT, were used. Evaluations with metrics such as accuracy, precision, recall, and F1-score showed deep learning models, especially BERT, outperformed traditional ones. Cross-validation ensured robustness and mitigated overfitting. The methodologies from this competition promise to improve automatic detection of conspiratorial versus critical narratives, aiding effective content moderation on digital platforms.

References

[1] K. M. Douglas, R. M. Sutton, What are conspiracy theories? a definitional approach to their correlates, consequences, and communication, Annual review of psychology 74 (2023) 271–298.

[2] D. Korenčić, B. Chulvi, X. Bonet Casals, M. Taulé, P. Rosso, F. Rangel, Overview of the oppositional thinking analysis pan task at clef 2024, in: G. Faggioli, N. Ferro, P. Galuvakova, A. G. S. de Herrera (Eds.), Working Notes of CLEF 2024 – Conference and Labs of the Evaluation Forum, Springer, 2024.

[3] A. A. Ayele, N. Babakov, J. Bevendorff, X. B. Casals, B. Chulvi, D. Dementieva, A. Elnagar, D. Freitag, M. Fröbe, D. Korenčić, M. Mayerl, D. Moskovskiy, A. Mukherjee, A. Panchenko, M. Potthast, F. Rangel, N. Rizwan, P. Rosso, F. Schneider, A. Smirnova, E. Stamatatos, B. Stein, M. Taulé, D. Ustalov, X. Wang, M. Wiegmann, S. M. Yimam, E. Zangerle, Overview of pan 2024: Multi-author writing style analysis, multilingual text detoxification, oppositional thinking analysis, and generative ai authorship verification - condensed lab overview, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fifteenth International Conference of the CLEF Association CLEF-2024, 2024.

[4] M. Ghasemizade, A computational journey through conspiracy theories: A genealogical approach (2024).

[5] S. Suthaharan, S. Suthaharan, Support vector machine, Machine learning models and algorithms for big data classification: thinking with examples for effective learning (2016) 207–235.

[6] D. M. Blei, A. Y. Ng, M. I. Jordan, Latent Dirichlet allocation, Journal of Machine Learning Research 3 (2003) 993–1022.

[7] N. Lopes, B. Ribeiro, N. Lopes, B. Ribeiro, Non-negative matrix factorization (nmf), Machine Learning for Adaptive Many-Core Machines-A Practical Approach (2015) 127–154.

[8] D. Chicco, N. Tötsch, G. Jurman, The matthews correlation coefficient (mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation, BioData mining 14 (2021) 1–22.

[9] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[10] E. Camizuli, E. J. Carranza, Exploratory data analysis (eda), The encyclopedia of archaeological sciences (2018) 1–7.

[11] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al., Huggingface's transformers: State-of-the-art natural language processing, arXiv preprint arXiv:1910.03771 (2019).

[12] X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, Q. Liu, Tinybert: Distilling bert for natural language understanding, arXiv preprint arXiv:1909.10351 (2019).

[13] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the 2013 conference on empirical methods in natural language processing, 2013, pp. 1631–1642.

[14] M. R. Rezaei-Dastjerdehei, A. Mijani, E. Fatemizadeh, Addressing imbalance in multi-label classification using weighted cross entropy loss function, in: 2020 27th National and 5th International Iranian Conference on Biomedical Engineering (ICBME), 2020, pp. 333–338. doi:10.1109/ICBME51989.2020.9319440.

[15] L. Havrlant, V. Kreinovich, A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation), International Journal of General Systems 46 (2017) 27–36.