SemanticCuetSync at ArAIEval Shared Task: Detecting Propagandistic Spans with Persuasion Techniques Identification using Pre-trained Transformers

Symom Hossain Shohan, Md. Sajjad Hossain, Ashraful Islam Paran, Shawly Ahsan, Jawad Hossain and **Mohammed Moshiul Hoque**
Department of Computer Science and Engineering
Chittagong University of Engineering and Technology
{u1904048,u1904031,u1904029,u1704057,u1704039}@student.cuet.ac.bd
moshiul_240@cuet.ac.bd

Abstract

Detecting propagandistic spans and identifying persuasion techniques are crucial for promoting informed decision-making, safeguarding democratic processes, and fostering a media environment characterized by integrity and transparency. Various machine learning (Logistic Regression, Random Forest, and Multinomial Naive Bayes), deep learning (CNN, CNN+LSTM, CNN+BiLSTM), and transformer-based (AraBERTv2, AraBERT-NER, CamelBERT, BERT-Base-Arabic) models were exploited to perform the task. The evaluation results indicate that CamelBERT achieved the highest micro-F1 score (24.09%), outperforming CNN+LSTM and AraBERTv2. The study found that most models struggle to detect propagandistic spans when multiple spans are present within the same article. Overall, the model's performance secured a 6^{th} place ranking in the ArAIEval Shared Task-1.

1 Introduction

Detecting propagandistic spans with persuasion techniques identification involves pinpointing particular text sections characterized by the deliberate use of persuasive language and tactics to shape opinions, attitudes, or actions. This involves analyzing linguistic cues, rhetorical strategies, and contextual elements to identify instances of propaganda and the specific persuasion techniques employed. It is difficult for a naive user to determine if the information is factually correct or if it is just another propaganda technique. There have been many studies done on detecting fake news (Nguyen et al., 2020), biased (Baly et al., 2020), hyperpartisan (Potthast et al., 2017), and propagandistic news and articles (Da San Martino et al., 2019). The influence of fake news on Twitter (Bovet and Makse, 2019) and other online media (Vosoughi et al., 2018) can be dangerous. Deliberately spreading fake news about a particular topic (Nakov et al., 2021a,b) is also a propaganda technique. Social media users tend to believe anything with more views without any factual basis (Guo et al., 2020).

Propaganda detection (Pastor et al., 2024) is a multidisciplinary field that draws upon techniques from NLP and identifies elements indicative of propaganda. There are diverse propaganda techniques (Jones, 2024), and classifying them is a challenge. Span detection (Li et al., 2022; Martino et al., 2020) in propaganda techniques is another big challenge. Propaganda aims to influence the audience to advance a specific agenda. Detecting the exact span where it occurs is tricky and arguably more difficult than finding false information in a news article. Few studies have focused on propaganda technique detection (Othman, 2023; Refaee et al., 2022) and span detection (Attieh and Hassan, 2022) in Arabic.

By applying propagandistic textual spans detection with persuasion techniques identification across various domains, individuals and organizations can better understand and counteract the influence of propaganda, ultimately fostering more informed, ethical, and resilient societies. The contributions of this work are illustrated as follows:

- Proposed a fine-tuned transformer model to classify various propaganda techniques and detect the span of propaganda techniques.
- Investigated several ML, DL, and transformer-based models for detecting propagandistic techniques and performed in-depth error analysis, offering valuable insights into detecting propagandistic span and persuasion techniques.

2 Related Work

Propaganda detection has been the subject of numerous research in recent years. However, the majority of them were primarily concerned with English. Li et al. (2019) proposed a model to detect propaganda at the sentence level using logistic

Proceedings of The Second Arabic Natural Language Processing Conference, pages 518–523

regression, which achieved an f_1-score of 0.66. Several DL, ML, and transformer-based techniques were proposed by Gupta et al. (2019) to detect propaganda. The ensemble technique yielded the highest f_1-score of 0.669 out of all of them. Both of these studies approached propaganda detection as a binary classification task.

Propaganda detection in Arabic has seen substantial advancements in recent years. Hasanain et al. (2023b) provided an overview of the ArAIEval shared task, presented at the first ArabicNLP 2023 conference, focusing on persuasion technique identification and disinformation detection in both binary and multitask settings. Alam et al. (2022) organized a shared task specifically on propaganda detection in Arabic Tweets, addressing a wide range of topics. This shared task included two objectives: identifying the type of propaganda technique employed and determining the specific text spans where these techniques were utilized. Refaee et al. (2022) used pre-trained BERT model to detect propaganda in Arabic tweets. Their model achieved an f_1-score of 0.602. In contrast to the earlier study, Samir et al. (2022) details how to detect propaganda and its validity span. GPT-4 was used in experiments by Hasanain et al. (2024a) to identify and locate the propaganda span on multiple languages' news datasets. They found that GPT-4's performance is low for this sequence tagging and multilabel task, especially in low-resourced languages. Hasanain et al. (2023a) proposed using LLM for propaganda span annotation. This work uses a transformer-based model to address the downstream task of detecting propagandistic techniques from Arabic tweets and paragraphs extracted from news articles.

3 Task and Dataset Description

To address the phenomena of detecting propaganistic techniques, shared task organizers[1] provided a unimodal (text) dataset. The dataset (Hasanain et al., 2024b) contains Arabic tweets and paragraphs for propagandistic technique detection. The dataset covers two genres: tweets and paragraphs extracted from news articles. The dataset covers a total of 23 different classes representing various propaganda techniques. Table 1 illustrates the distribution of train, dev, and test sets for tweets and paragraphs, where T_W denotes total words.

The train, dev, and test sets consisted of 6997, 921, and 1046 Arabic tweets and paragraphs extracted from news articles. Task-1 (Hasanain et al., 2024b) focused on detecting and classifying the propaganda techniques used in the text and the exact span(s) where each technique appears. Table 2 illustrates a few examples of text, technique, and the start and end of the span in the dataset. The "start" and "end" columns indicate which part of the text contains propaganda.

Type	Train	Dev	Test	T_W
Tweet	995	249	260	1504
Paragraph	6002	672	786	7460
Total	6997	921	1046	8964

Table 1: Dataset statistics for Task-1.

Text	Technique	Start	End
حرب (War)	Loaded-Language	11	14
عبيد الطغاه (Slaves of tyrants)	Name-Calling-Labeling	78	89
شبهة (suspicion)	Doubt	28	32
الإصرار (Determination)	Loaded-Language	52	59

Table 2: Task-1 sample with text, technique, and start and end of span.

4 System Development

Figure 1 illustrates the schematic process of classifying propaganda techniques and detecting textual spans. We used a tokenizer to tokenize each text. We added unique tokens *[CLS]* at the beginning and *[SEP]* at the end and assigned a -100 label to both so that the loss function ignores them. Then, we used the BIO tagging scheme (Ramshaw and Marcus, 1999; Dai et al., 2019), where the B-tag indicates the beginning of an entity, the I-tag indicates a token contained inside the same entity, and the O-tag denotes that no entity is contained within the token.

4.1 Classifiers

We explored several ML, DL, and transformer-based models for this task.

- **ML-based Models:** Logistic Regression (LR), Random Forest (RF), and Multinomial Naive Bayes (MNB) were used for this task.

[1] https://araieval.gitlab.io/task1/

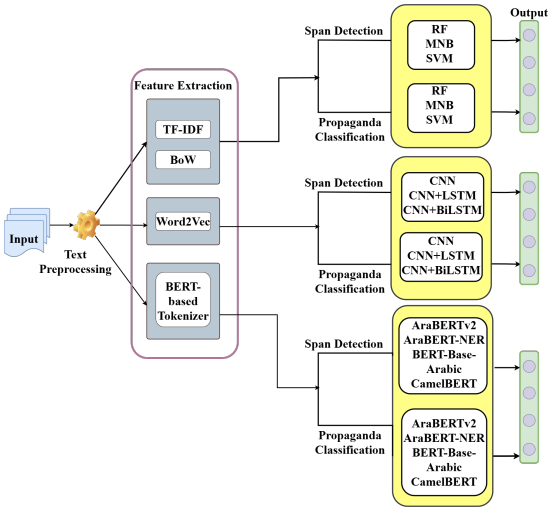

Figure 1: Schematic process for Propagandastic Technique Detection.

Classifier	Parameters	Value
RF	max-depth	20
	n-estimator	500
MNB	alpha	1.0
	fit-prior	False
LR	solver	lbfgs
	max_iter	20000

Table 3: Parameters used for ML models

We employed Bag-of-Words (BoW) method for feature extraction using 'CountVectorizer' from Scikit-learn[2]. Besides we also used TF-IDF from Scikit-learn for extracting textual features. A maximum depth of 20 was employed with 500 estimators and the '*max_features*' was set to 'sqrt' in the RF classifier. For MNB, we set the alpha value at 0.1 and fit before False. Lastly, the LR is used with the 'lbfgs' solver and max iter set to 20000. Table 3 shows the several ML parameters used for the models.

- **DL-based Models:** The CNN model uses an embedding layer with an output dimension of 256. It comprises one Conv1D layer with 512 filters and a GlobalMaxPooling layer for downsampling. To prevent overfitting, non-linearity is introduced by the dense layer with L2 regularisation. The output layer allows for multi-class classification with its 23 units and softmax activation. We also set loss to 'SparseCategoricalCrossentropy', optimizer to 'adam' and metrics to 'accuracy'.

  CNN+LSTM: This method utilized one embedding layer with output-dim=256, one Conv1D layer, one MaxPooling1D layer, three LSTM layers, and three dense layers with a dropout layer. The number of units for the Conv1D layer was 512, whereas the LSTM layers had 512, 256, and 128. The unit size for the first dense layer was 256 with 'tanh' activation; the second dense layer had 128 units with 'relu' activation; and the third dense layer had 23 units for classification with 'softmax' activation. The loss function utilized in this case was 'SparseCategoricalCrossentropy' with the 'Adam' optimizer.

  CNN+BiLSTM: This method utilized one Conv1D layer, one MaxPooling1D layer, three Bidirectional LSTM layers, and three dense layers with a dropout layer. The number of units for the Conv1D layer was 512, whereas the LSTM layers had 512, 256, and 128. The unit size for the first dense layer was 256 with 'relu' activation; the second dense layer had 128 units with 'relu' activation; and the third dense layer had 23 units for classification with 'softmax' activation. The loss function utilized in this case was 'SparseCategoricalCrossentropy' with the 'Adam' optimizer.

Models	LR	WD	BS	EP
AraBERTv2	1e-5	0.001	8	5
AraBERT-NER	1e-5	0	32	3
BERT-Base-Arabic	5e-5	0.001	8	5
CamelBERT	5e-5	0.001	8	6

Table 4: Hyperparameters for transformer-based models, where LR, WD, BS, and EP denote learning rate, weight decay, batch size, and epochs, respectively.

- **Transformer-based Models:** BERT (Bidirectional Encoder Representations from Transformers), are developed to extract contextualized word representations from unlabeled texts (Devlin et al., 2018). It utilizes the transformer architecture's encoder representation approach, initially introduced by Vaswani et al., 2017. Transformer library of `Huggingface`[3] includes numerous pre-trained models for text processing. In order to determine the propaganda techniques,

[2]https://scikit-learn.org

[3]https://huggingface.co/docs/transformers/en/index

we employed CamelBERT (Inoue et al., 2021), AraBERTv2 (Antoun et al., 2020), BERT-Base-Arabic (Safaya et al., 2020) and AraBERT-NER. Table 4 illustrates the hyperparameters used in these models. This study presents a framework that leverages CamelBERT for embedding model generation and token classification. Initially, the text was tokenized using CamelBERT's tokenizer. The tokenization process transforms each token into a dense vector representation through the embedding layer of the model. This approach enables the extraction of intricate semantic and syntactic features. In addition, a specialized classification head was incorporated into CamelBERT for token classification. This process ensures accurate and context-aware token-level predictions.

5 Results

Table 5 shows the evaluation results on the test set. Results reveal that LR achieved the most elevated micro F1-score (11.76%) among the ML approaches, surpassing RF (11.54%) and MNB (11.31%). CNN+LSTM and CNN+BiLSTM with word embeddings have the same micro F1-score of 11.54%, which is approximately 0.22% lower than the best ML approach. On the other hand, CNN achieved a micro F1 score of 11.26%, slightly slower than the other two approaches. The transformer-based approach performs better on the test dataset. The best-performing transformer model beats the best-performing ML and DL models. CamelBERT, with a micro F1 score of 24.09, is the best performing, while the other three, AraBERTv2, BERT-Base-Arabic, and AraBERT-NER, scored 23.84, 20.31, and 7.7, respectively. The CamelBERT model has produced embeddings that highlight domain-specific features relevant to propaganda. Its tokenizer is superior in handling Arabic morphology and syntax, resulting in more effective tokenization of complex Arabic structures. Consequently, CamelBERT outperforms other models in terms of performance.

ML Models				
Classifier	**P**	**R**	**MF1**	**mF1**
LR	**6.51**	**60.37**	**0.47**	**11.76**
RF	6.36	61.76	0.23	11.54
MNB	6.27	57.25	0.57	11.31
DL Models				
Classifier	**P**	**R**	**MF1**	**mF1**
CNN	6.21	60.33	0.32	11.26
CNN+LSTM	6.36	61.76	0.23	11.54
CNN+BiLSTM	**6.36**	**61.76**	**0.23**	**11.54**
Transformers				
Classifier	**P**	**R**	**MF1**	**mF1**
AraBERTv2	34.81	18.12	14.43	23.84
CamelBERT	**20.54**	**29.11**	**12.93**	**24.09**
AraBERT-NER	5.42	13.68	0.23	7.76
BERT-Base-Arabic	23.20	18.06	10.28	20.31

Table 5: Performance of the employed models on the test set. This table displays the Precision (P), Recall (R), macro F1 (MF1), and micro F1 (mF1) scores.

5.1 Error Analysis

The error analysis with the confusion matrix showed that the employed models can detect Loaded-Language, but the classification of other classes is imperfect. Here, 1680 of the Loaded-Language are correctly classified, and 1067 are incorrectly classified. On the other hand, only 50 of the Name-Calling-Labelling is correctly classified, and the other 254 are incorrectly classified. It is evident that the dataset is significantly unbalanced. 14 out of the 23 classes have no entries in the test set. Furthermore, the Loaded-Language class is overly represented in the test set. The Name-Calling-Labelling class is frequently misclassified as Loaded-Language. Due to these imbalances, the model faces difficulties interpreting the dataset, leading to poor performance on the test set.

Table 6 compares the predicted outcomes and the actual labels. In the first and third samples, the model predicted the propaganda technique accurately but failed to detect the span correctly. In the second example, the model incorrectly predicted the propaganda technique and the span. The test set presented poses significant challenges due to the texts' extensive length and various propaganda techniques within a single text.

6 Discussion

Transformer-based models are capable of identifying various propaganda techniques within the exact text. However, they need to detect the spans consistently and accurately. This inconsistency contributed to the test set's less-than-optimal perfor-

Value	Actual	Predicted
Technique	Loaded-Language	Loaded-Language
Text	حادث (Accident)	الا (No)
Start	96	0
End	100	4
Technique	Name-Calling-Labeling	Questioning-the-Reputation
Text	هجوم تخريبي (Destructive Attack)	اوك (Okay)
Start	192	229
End	203	232
Technique	Loaded-Language	Loaded-Language
Text	هجوم تخريبي (Destructive Attack)	هجوم (Attack)
Start	192	192
End	203	195

Table 6: Few predictions by CamelBERT.

mance. The error analysis findings suggest that most models struggle to detect propaganda span when there are multiple ones in the same article. Most of the errors were introduced in the model span detection phase. Moreover, detecting the span of the propaganda involves the model understanding not only the context but also the tone, intent, and bias. Though the pre-trained BERT-based models can understand contextual elements, they need help in other aspects.

7 Conclusion

This study investigates the capabilities of transformer-based models in thoroughly analyzing propaganda span detection and identifying the persuasive techniques in the Arabic dataset. Among all employed ML, DL, and transformer-based models, CamelBERT showed notable performance with the highest micro-F1 score of 24.09. Further advancements could be achieved by expanding the dataset and investigating large language models in the task.

References

Firoj Alam, Hamdy Mubarak, Wajdi Zaghouani, Giovanni Da San Martino, and Preslav Nakov. 2022. Overview of the WANLP 2022 shared task on propaganda detection in Arabic. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 108–118, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. Arabert: Transformer-based model for arabic language understanding. *arXiv preprint arXiv:2003.00104*.

Joseph Attieh and Fadi Hassan. 2022. Pythoneers at wanlp 2022 shared task: Monolingual arabert for arabic propaganda detection and span extraction. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 534–540.

Ramy Baly, Giovanni Da San Martino, James Glass, and Preslav Nakov. 2020. We can detect your bias: Predicting the political ideology of news articles. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4982–4991.

Alexandre Bovet and Hernán A Makse. 2019. Influence of fake news in twitter during the 2016 us presidential election. *Nature communications*, 10(1):7.

Giovanni Da San Martino, Yu Seunghak, Alberto Barrón-Cedeno, Rostislav Petrov, Preslav Nakov, et al. 2019. Fine-grained analysis of propaganda in news article. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 5636–5646. Association for Computational Linguistics.

Zhenjin Dai, Xutao Wang, Pin Ni, Yuming Li, Gangmin Li, and Xuming Bai. 2019. Named entity recognition using bert bilstm crf for chinese electronic health records. In *2019 12th international congress on image and signal processing, biomedical engineering and informatics (cisp-bmei)*, pages 1–5. IEEE.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Lei Guo, Jacob A. Rohde, and H Denis Wu. 2020. Who is responsible for twitter's echo chamber problem? evidence from 2016 us election networks. *Information, Communication & Society*, 23(2):234–251.

Pankaj Gupta, Khushbu Saxena, Usama Yaseen, Thomas Runkler, and Hinrich Schütze. 2019. Neural architectures for fine-grained propaganda detection in news. *arXiv preprint arXiv:1909.06162*.

Maram Hasanain, Fatema Ahmed, and Firoj Alam. 2023a. Large language models for propaganda span annotation. *arXiv preprint arXiv:2311.09812*.

Maram Hasanain, Fatema Ahmed, and Firoj Alam. 2024a. Can gpt-4 identify propaganda? annotation and detection of propaganda spans in news articles. In *Proceedings of the 2024 Joint International Conference On Computational Linguistics, Language Resources And Evaluation*, LREC-COLING 2024, Torino, Italy.

Maram Hasanain, Firoj Alam, Hamdy Mubarak, Samir Abdaljalil, Wajdi Zaghouani, Preslav Nakov, Giovanni Da San Martino, and Abed Alhakim Freihat. 2023b. Araieval shared task: Persuasion techniques and disinformation detection in arabic text. In *Proceedings of the First Arabic Natural Language Processing Conference (ArabicNLP 2023)*, Singapore. Association for Computational Linguistics.

Maram Hasanain, Md. Arid Hasan, Fatema Ahmed, Reem Suwaileh, Md. Rafiul Biswas, Wajdi Zaghouani, and Firoj Alam. 2024b. Araieval shared task: Propagandistic techniques detection in unimodal and multimodal arabic content. In *Proceedings of the Second Arabic Natural Language Processing Conference (ArabicNLP 2024)*, Bangkok. Association for Computational Linguistics.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in arabic pre-trained language models. *arXiv preprint arXiv:2103.06678*.

Daniel Gordon Jones. 2024. Detecting propaganda in news articles using large language models. *Eng OA*, 2(1):01–12.

Jinfen Li, Zhihao Ye, and Lu Xiao. 2019. Detection of propaganda using logistic regression. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 119–124.

Wei Li, Shiqian Li, Chenhao Liu, Longfei Lu, Ziyu Shi, and Shiping Wen. 2022. Span identification and technique classification of propaganda in news articles. *Complex & Intelligent Systems*, 8(5):3603–3612.

G Martino, Alberto Barrón-Cedeno, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. 2020. Semeval-2020 task 11: Detection of propaganda techniques in news articles. *arXiv preprint arXiv:2009.02696*.

Preslav Nakov, Firoj Alam, Shaden Shaar, Giovanni Da San Martino, and Yifan Zhang. 2021a. Covid-19 in bulgarian social media: Factuality, harmfulness, propaganda, and framing. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 997–1009.

Preslav Nakov, Firoj Alam, Shaden Shaar, Giovanni Da San Martino, and Yifan Zhang. 2021b. A second pandemic? analysis of fake news about covid-19 vaccines in qatar. *arXiv preprint arXiv:2109.11372*.

Van-Hoang Nguyen, Kazunari Sugiyama, Preslav Nakov, and Min-Yen Kan. 2020. Fang: Leveraging social context for fake news detection using graph representation. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 1165–1174.

Ghazal Ayyoub Othman. 2023. *Detecting Propaganda Techniques in Arabic Tweets*. Ph.D. thesis, Hamad Bin Khalifa University (Qatar).

Martial Pastor, Nelleke Oostdijk, and Martha Larson. 2024. The contribution of coherence relations to understanding paratactic forms of communication in social media comment sections. In *JADT 2024: 17th International Conference on Statistical Analysis of Textual Data*.

Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2017. A stylometric inquiry into hyperpartisan and fake news. *arXiv preprint arXiv:1702.05638*.

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Eshrag Ali Refaee, Basem Ahmed, and Motaz Saad. 2022. Arabem at wanlp 2022 shared task: Propaganda detection in arabic tweets. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 524–528.

Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. Kuisail at semeval-2020 task 12: Bert-cnn for offensive speech identification in social media. *arXiv preprint arXiv:2007.13184*.

Ahmed Samir, Abu Bakr Soliman, Mohamed Ibrahim, Laila Hesham, and Samhaa R El-Beltagy. 2022. Ngu_cnlp at wanlp 2022 shared task: Propaganda detection in arabic. In *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 545–550.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *science*, 359(6380):1146–1151.