Arab Twitter Reactions to the Russo-Ukrainian War with Semi-Supervised Learning Techniques

Omar Essam[1], Omar A. Bakr[1], Mervat T. Abassy[1], Ali Taha[1], and Walid Gomaa[2,1]

[1]Faculty of Engineering, Alexandria University, Alexandria, Egypt
[2]Egypt Japan University of Science and Technology, Alexandria, Egypt
es-omar.elshobki2025@alexu.edu.eg
es-omar.bakr2025@alexu.edu.eg
es-mervattamer2025@alexu.edu.eg
es-ali.taha2025@alexu.edu.eg
walid.gomaa@ejust.edu.eg

***Abstract*—The scarcity of labeled datasets poses significant challenges in Natural Language Processing (NLP), particularly for sentiment and partiality analysis of Arabic tweets regarding the Russo-Ukrainian War. Semi-supervised learning (SSL) addresses this issue by leveraging both labeled and unlabeled data. This paper investigates SSL approaches in Arabic NLP by analyzing sentiment and partiality in collected tweets. We trained four models each with a different approach : (1) a supervised model, (2) a model initialized with pretraining, (3) a model trained with consistency regularization, and (4) a model combining pretraining and consistency regularization, we limited our dataset to 200,000 labeled to compare the effectiveness of the chosen SSL methods with previous work done on a 500,000 tweet dataset. Then we built further over the best model obtained from the previous ones by training it model on 1.3 million labeled tweet, to acheive a state-of-the-art performance 98.9% test accuracy. Our findings demonstrate that SSL can reduce the dependence on labeled data while still achieving competitive performance. This study underscores SSL's potential to enhance Arabic NLP applications, providing more efficient and scalable solutions.**

***Index Terms*—Semi-supervised learning, Natural Language Processing, Sentiment analysis, Partiality analysis, Arabic language processing, Twitter data analysis, Russo-Ukrainian War**

I. Introduction

The Russo-Ukrainian War stands as a pivotal geopolitical event, creating significant interest within the Arab world. This study aims to analyze Arab reactions to the conflict through the lens of semi-supervised learning, inspired by a previously published paper utilizing the same dataset[1]. Building upon prior work, we explore various semi-supervised techniques tailored for natural language processing (NLP), analyze the dataset, and incorporate improved lexicons. Our objective is to assess sentiment, emotions, and biases conveyed through Arabic tweets, comparing the effectiveness of semi-supervised learning approaches against traditional supervised methodologies. By analyzing the original dataset comprising over 2 million Arabic tweets, we aim to uncover diverse perspectives from different Arab nations.

Partiality analysis on social media offers immediate and unfiltered insights into public partiality, facilitating a deeper understanding of societal responses. Unlike conventional media, social media provides a grassroots-level panorama of public sentiment, crucial for policymakers, researchers, and the broader populace to comprehend the war's implications on international relations and public morale within the Arab domain. Additionally, this study focuses on leveraging semi-supervised learning methodologies tailored for NLP within the context of sentiment and partiality analysis.

In this study, we not only compare semi-supervised learning techniques against conventional supervised paradigms but also conduct a comparative analysis among different semi-supervised learning methods and build upon the best obtained model to achieve better performance than previous work. Furthermore, we emphasize the preprocessing of Arabic data, which is essential to ensure accurate and reliable sentiment and partiality analysis. Through this comprehensive analysis, this paper aims to underscore the potential advantages of semi-supervised learning in enhancing partiality analysis models' accuracy and robustness in Arabic NLP applications even with less data.

II. Related Work

This paper draws inspiration from previous work [1], yet introduces distinct contributions to the field. Our primary contribution lies in the exploration of various training methodologies designed to effectively utilize minimal labeled data alongside vast amounts of unlabeled data.By utilizing only half the dataset size used in [1], we achieved similar performance. The previous study reported a test accuracy of 95.07% using 449,000 pseudo-labeled tweets. In contrast, our study attained a test accuracy of 93.85% with merely 200,000 labeled tweets. This discrepancy underscores the efficacy of the advanced training techniques employed in our study.

A second significant contribution is the enhancement of the lexicon utilized in [1] with the help of Aravec[4] to facilitate more accurate pseudo-labeling and subsequently improve the training process. By refining the lexicon, we aimed to augment

the quality of the training data and thereby enhance model performance.

Our third significant contribution was the development of a superior model compared to the one proposed in [1]. Leveraging an enhanced lexicon and incorporating advanced semi-supervised learning (SSL) methods, we remarkably achieved a test accuracy of 98.9%, surpassing the previous benchmark and setting a new standard.

Our fourth and final contribution involves the execution of extensive sentiment and partiality analyses on the acquired dataset. This comprehensive analysis provided valuable insights into the temporal and geographical variations in sentiment and partiality within the tweet data, contributing to a deeper understanding of the underlying trends and dynamics.

Together, these contributions advance the existing body of knowledge in the field of tweet classification and semi-supervised learning, offering valuable insights and methodologies for future research endeavors.

III. Data Collection and Analysis

A. Data Collection

Tweets were collected using the Twitter API from February 23, 2022, to January 31, 2023, utilizing a set of 40 keywords associated with the war. The dataset comprises over 2 million tweets, with 220,000 tweets designated for the comparative analysis of the methods, sentiment, and emotional analysis. The collected tweets represent diverse opinions from various Arab countries, facilitating a comprehensive understanding of public sentiment. Carefully selected keywords covered multiple facets of the war, including military operations, humanitarian concerns, and political responses.

The number of hashtags in each country is presented in Figure 1, while Figure 2 illustrates the number of tweets in each country and Figures 3, 5, 4 illustrates the amount of tweets every month from the top 3 countries

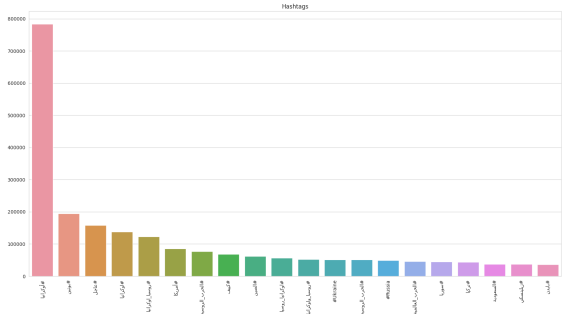

Fig. 1. Number of Hashtags in tweets.

B. Data Analysis

1) Sentiment Analysis: We utilized three pre-trained models for sentiment analysis:

- **Mazajak**[5]: A robust model for Arabic text sentiment analysis, fine-tuned on a large Arabic corpus.

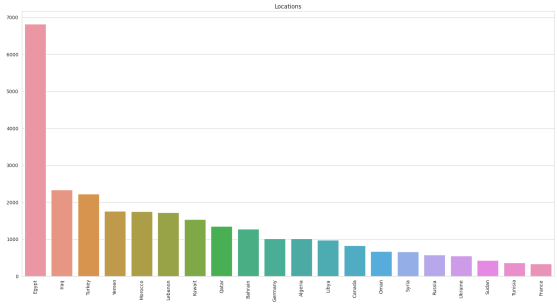

Fig. 2. Number of tweets in each country.

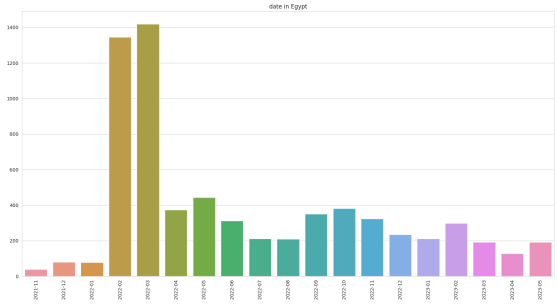

Fig. 3. Egyptian tweets timeline.

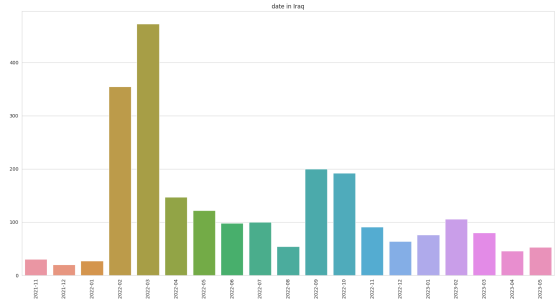

Fig. 4. Iraqi tweets timeline.

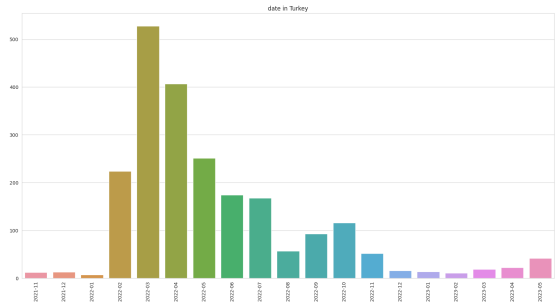

Fig. 5. Turkish tweets timeline.

- **CaMeL-Tools**[6]: A suite of tools tailored for Arabic language processing, including sentiment analysis, part-of-speech tagging, and named entity recognition.
- **Fine-tuned AraBERT**[7]: An Arabic BERT model fine-tuned on HARD arabic dataset to improve accuracy. AraBERT has shown state-of-the-art performance in various Arabic NLP tasks.

These models were chosen for their effectiveness in handling Arabic text and have been validated in various research studies. By employing multiple models, we ensure a comprehensive analysis and mitigate the biases of individual models.

To predict the tweet's sentiments a vote was cast between the three models mentioned above then labeling each tweet either positive, negative or neutral sentiment

Most tweets exhibited negative sentiment. Mixed feelings were common, with a significant number of tweets showing either joy or anger. Negative sentiment was often associated with reports of violence and casualties, while positive sentiment, though less frequent, was linked to acts of solidarity and support for Ukraine.

The sentiment analysis revealed that the initial reaction to the war was predominantly negative, reflecting the shock and disapproval of the aggression. Over time, sentiments evolved, with fluctuations corresponding to major events and developments in the conflict. The overall sentiment distribution per country is shown in Figure 9 and the sentiment changes over time is shown in Figure 10

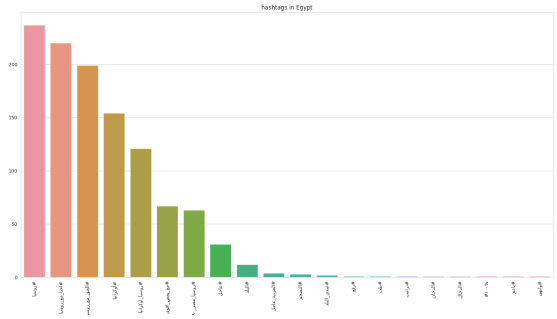

Fig. 6. Distribution of hashtags in Egypt.

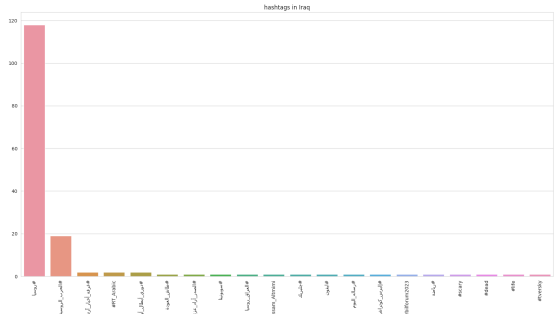

Fig. 7. Distribution of hashtags in Iraq.

2) Partiality Analysis: Partiality analysis aimed to identify biases in the tweets. We found that most tweets were neutral, but a significant number showed bias towards one of the parties involved. This analysis helps to understand the influences and potential propaganda effects on public opinion. By examining partiality, we can identify the presence of bias and its sources. This is essential for understanding the factors shaping public opinion and the role of media and political narratives in influencing perceptions. It also helps to identify areas where more balanced and objective information is needed. The partiality levels per country are illustrated in Figure 8.

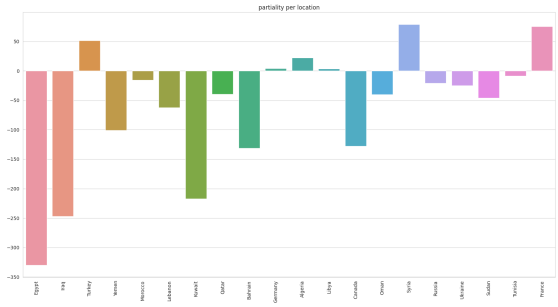

Fig. 8. Partiality numbers per Country.

3) Geographical Analysis: Geo-tagged tweets were analyzed to identify regional variations in sentiment and opinions within the Arab world. This analysis helps to understand how different countries and regions perceive the conflict and the factors influencing their viewpoints. We mapped the sentiments to specific locations using geocoding techniques and visualized the results on a geographical map. Geographical analysis is critical in this study as it provides insights into the local contexts and socio-political factors that influence public opinion. The differences in sentiment across countries can reveal underlying geopolitical and cultural dynamics. The sentiment varied significantly across different Arab countries, with some regions showing more support for Ukraine initially and others displaying a more neutral or Pro-Russia stance as the conflict progressed. This variation can be attributed to political affiliations, economic ties, and media influences in different countries. Therefore, we created statistics for the number of hashtags in Egypt, Iraq and Turkey as shown in 6, 7 and **??**

4) Temporal Analysis: The data was segmented by time to observe how sentiments and opinions evolved throughout the duration of the conflict. We conducted a time-series analysis to track the changes in sentiment and emotions in response to key events during the war, such as major battles, international diplomatic actions, and humanitarian crises.

Temporal analysis allows us to capture the dynamic nature of public opinion. By correlating sentiment changes with specific events, we can understand the triggers and drivers of public reaction. This information is valuable for predicting future trends and formulating strategic responses.

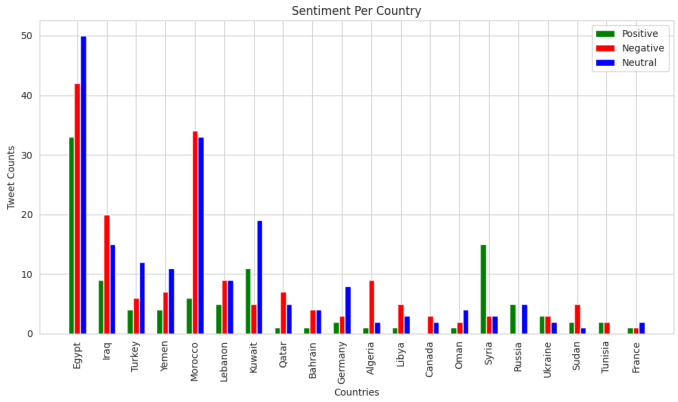

Fig. 9. Tweet sentiments per country.

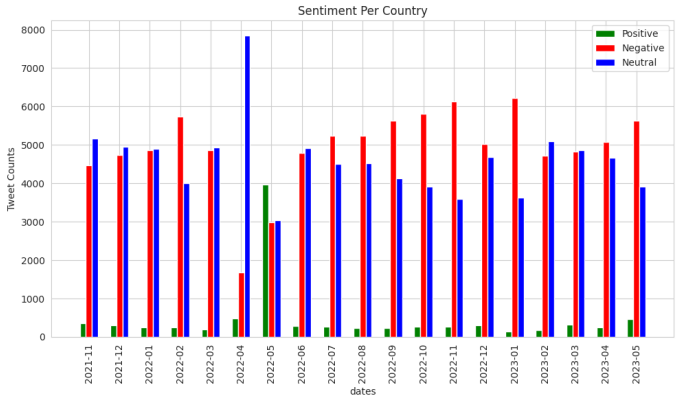

Fig. 10. Tweet sentiments over time.

Initially, there was strong support for Ukraine, likely due to the initial Russian offensive and the humanitarian crisis. Over time, sentiment shifted, with increasing empathy towards Russia, particularly as the conflict became protracted and economic sanctions impacted global perceptions. This temporal shift highlights the dynamic nature of public opinion and the influence of ongoing events.

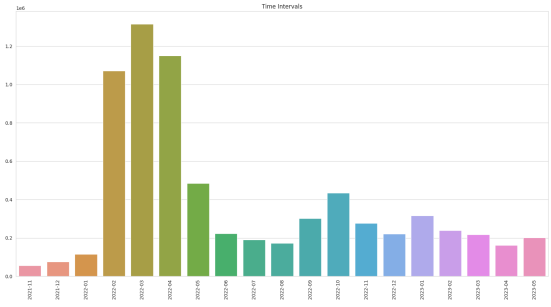

Fig. 11. Time Intervals

IV. Data Pre-processing and Labelling

A. Preprocessing

To ensure data cleanliness and uniformity, several preprocessing steps were applied to the tweet dataset. Duplicate entries, retweets, and non-Arabic content were removed to eliminate redundancy and maintain relevance. Following this, tokenization and normalization techniques were employed to prepare the data for analysis.

Tokenization involved segmenting the text into individual tokens while excluding extraneous elements such as URLs, hashtags, mentions, and non-Arabic characters. This step aimed to streamline the text for further processing and analysis.

Normalization procedures were then implemented to standardize the textual content. Variations in spelling and diacritics were addressed by converting the text to a standardized form, thereby facilitating consistency and improving the quality of subsequent analyses.

Together, these preprocessing steps ensured the integrity and uniformity of the tweet dataset, laying the groundwork for robust and insightful analyses.

Table I shows the preprocessing steps and their effects:

TABLE I
Preprocessing Steps on Tweets

Preprocessing Step	**Description**
Removing usernames	Any word starting with "@" is removed (e.g., @moe123).
Removing links	Any text starting with "www." or "http" is removed.
Removing emojis	Emojis are removed using the emoji's Unicode.
Removing hashtags	Octothorpe, underscores, and hyphens (-) are removed.
Removing non-Arabic characters	Non-Arabic characters and words are removed.
Removing punctuation marks	Punctuation marks are removed.

B. Labeling

To label the tweets, we employed the lexicon referenced in [1]. This lexicon was further expanded using AraVec[4], an Arabic Word2Vec embedding model, which assisted in identifying words similar to those in the original lexicon. We then manually selected the appropriate terms from these suggestions. Each n-gram in the lexicon is assigned a score, where positive and negative scores represent a pro-ukraine and pro-russia respectively. The occurrence of the n-gram would add its score to the total score of the tweet. After acquiring a score for each tweet in the dataset we removed tweets with zero n-gram occurence, leaving us with 1.3 million tweets from the original dataset.We limited this amount to 200,000 , chosen at random, which was a small enough fraction to allow us show the effectiveness of the SSL approaches in incorporating small amount of labeled data to build a model while still producing a competitive performance. After going manually through a small fraction of the dataset, We came to a

conclusion that tweets with absolute score below 5 were still difficult to claim as a supporter to either parties. Therefore we decided to set threshold of -5 to 5 for "neutral" tweets, where below -5 is classified "pro-russia" and above 5 is "pro-ukraine". Then we split the data into training set, 200,000 tweets, and testing set, 50,000 tweets. The we used the 1.3 million labeled tweets for the improved model further in the training process.

C. Text Representation

Each token is represented with an embedding of a dimension 100 determined by the embedding model Aravec[4]. the specific variation Aravec used in this study is "Twitter-Skipgram 100", a pretrained embedder trained on 66,900,000 tweets with a vocab of 331,679 arabic word.

Despite the extensive vocabulary coverage provided by the Aravec model, we encountered several frequently used words in our dataset that were absent from its vocabulary. To address this limitation, we extended the vocabulary of the Aravec embedder by incorporating these missing words. Subsequently, we manually assigned embeddings to these new words, ensuring their inclusion in the embedding space.

V. Training Methods

In this section, we construct a comparison between Four models : (1) Supervised learning with, (2) Pretraining , (3) Cross view regularization, and (4) a Combination of Pretraining and Cross view methods, Then we use the best of them and trained it on the whole dataset to produce a state-of-the-art performance .

A. Supervised training

We utilized the previously described pseudo-labeled dataset to train a two-layer Bidirectional LSTM model with an input of an embedding vector of size 100. The first and second LSTM layers consist of 64 and 128 units, respectively. These are followed by a fully connected layer comprising 20 neurons, which is connected to a softmax output layer with 3 neurons to perform the final predictions.

The model was trained on the original training set and 200,000 labeled tweets over 9 epochs. This training process resulted in a training accuracy of 98% and a test accuracy of 92.1%.

B. Pretraining

The pretraining method is implemented using a bidirectional sequence-to-sequence (seq2seq) autoencoder model without attention mechanisms. The primary objective is to train the autoencoder such that the target sentence is identical to the input sentence.

The autoencoder comprises two main components: An Encoder, converts the input sequence into a vector, And a Decoder, converts this vector back into the target/input sentence. During the training process, the Encoder will learn to efficiently reduce the dimension of the input sequence and represent it in a vector that contains enough information for the Decoder to be able to extract the input sentence back from the vector.

Furthermore, we initialize our tweet classification model with the pretrained Encoder, to have a better starting point for the model which can help in faster convergence and better overall accuracy.

We pretrained the autoencoder on 500,000 unlabeled tweets for 20 epochs. then we used the pretrained Encoder for text classification training on 200,000 pseudo-labeled tweet for partiality classification for 9 epochs.

The results of this approach was slightly better than the conventional supervised model. Where the pretrained model resulted in 98.1% ,93.8% trainig and test accuracy respectively. On the other hand the un-initialized method reached 96%,92.1% after 9 epochs of training under strictly similar environment. Keeping in mind that the only difference, between both approaches, was the initialization of the Encoder. Hence the 1.7% difference in performance portrays the perks of the pretrainig method and its ability to employ unlabeled data to enhance performance of the model.

C. Cross view

This method designed to enhance the robustness and generalization capabilities of our model. This technique involves creating multiple "views" or variations of the input data by strategically modifying the original text sequences, thereby encouraging the model to produce consistent predictions across these varied inputs.

1) Generating Multiple Views: For each input text sequence in the unlabeled dataset, we generate multiple variants by randomly dropping words. Specifically, we create four different views of each sequence:

1) The original sequence.
2) Three additional sequences where a randomly selected word in the original sequence is replaced with a predefined drop symbol.

2) Model Predictions: The model is then tasked with making predictions for each of these views. By processing these multiple views of the same underlying input, the model is exposed to a form of augmented data that introduces variability while preserving the essential information.

3) Consistency Loss Calculation: To ensure that the model's predictions remain stable across the different views, we compute a consistency loss. This loss is calculated by measuring the average absolute differences between the model's predictions for the original sequence and each of the modified sequences. Mathematically, the consistency loss L_c can be expressed as:

$$L_c = \frac{1}{3} \sum_{i=1}^{3} \|f(x) - f(x_i)\| \tag{1}$$

where $f(x)$ represents the model's prediction for the original sequence, and $f(x_i)$ represents the predictions for the modified sequences. The consistency loss encourages the model to learn representations that are invariant to the minor perturbations introduced by the word drops.

4) Incorporating Consistency Loss: The total loss L used for training is a combination of the supervised loss L_s on labeled data and the consistency loss L_c:

$$L = L_s + \lambda L_c \tag{2}$$

where λ is a hyperparameter that balances the influence of the consistency loss relative to the supervised loss.

The results of cross-view regularization method showed improvement over the conventional method, however it didn't outperform the pretraining method. It produced a training accuracy of only 96.1% and a test accuracy of 93.6% after 9 epochs.

D. Pretraining and Cross view

In this study, we implemented a training method that combines pretraining with cross-view regularization. This approach leverages the strengths of both techniques to enhance the model's performance. Initially, the model undergoes a pretraining phase using a large dataset of 500,000 unlabeled tweets. Following this, the model is further trained with cross-view regularization on a dataset comprising 200,000 labeled tweets and 4,000 unlabeled tweets over 9 epochs. This combined method aims to improve the model's generalization and robustness.

The Results of this method was not quite as expected. Despite the strengths of this approach, it produced only 94.6% training accuracy and 93.75% after 5 epochs and its performance started declining. Due to lack of computational power we couldn't perform cross view regularization on more than 4,000 tweets nor more than 200,000 labeled tweets.

E. Improved model

We proceeded training the pretraining model, the best performance, using the full 1.3 million labeled tweet. We split the data into 1.1 million for training , 100,000 for validation , and 100,000 for testing. It produced 92% training accuracy and 98.9% test accuracy.

VI. Conclusion

In this study, We explored four different training methods for tweet classification: one supervised learning approach (normal supervised training) and three semi-supervised learning approaches (pretraining, cross-view regularization, and a combination of pretraining and cross-view regularization), and used the best method out of them to proceeded building upon it. Our objective was to compare the effectiveness of these methods in improving model performance.

The normal supervised training method served as our baseline. Among the semi-supervised approaches, the pretraining method demonstrated superior performance, achieving a test accuracy of 93.85%. This result highlights the effectiveness of leveraging large amounts of unlabeled data to enhance model generalization.

Surprisingly, the combined method, which integrates pretraining with cross-view regularization, did not perform as well as anticipated. Despite its theoretical strengths and increased complexity, it resulted in a slightly lower test accuracy of 93.75%. This indicates that the additional regularization did not contribute as expected and suggests potential areas for further optimization and investigation.

Thus we used the pretrained model and proceeded the training process with the full dataset to acheive 98.90% test accuracy.

Throughout the study, we encountered significant challenges due to limited computational resources. This constraint particularly affected the results of the combination method, as we were unable to perform cross-view regularization on more than 4,000 unlabeled tweets or use more than 200,000 labeled tweets, which definitely affected the performance of both the cross view and the combined method. The lack of computational power also made it difficult to efficiently preprocess, train, and analyze the gathered data, which likely impacted our findings.

Our main contributions include comparing several training paradigms to analyze their performance, improving the lexicon presented in [1] with the help of AraVec[4], and performing sentiment and partiality analysis on the data over time and geographically. These contributions provide a comprehensive evaluation of different training strategies and their impact on model performance.

These findings underscore the potential of pretraining in semi-supervised learning contexts and invite further exploration into the optimal integration of multiple training strategies. Future work could focus on overcoming computational limitations to fully exploit the benefits of cross-view regularization and improve the robustness of the combined method.

References

[1] Moayadeldin Tamer, Mohamed A. Khamis , Abdallah Yahia , SeifALdin Khaled , Abdelrahman Ashraf and Walid Gomaa, Arab reactions towards Russo-Ukrainian war.

[2] Andrew Dai, Quoc Le. 2015, Semi-supervised Sequence Learning .

[3] Kevin Clark, Minh-Thang Luong, Christopher D. Man- ning, and Quoc Le. 2018, Semi-supervised se- quence modeling with cross-view training.

[4] Abu Bakr Soliman, Kareem Eissa, Samhaa R. El-Beltagy, AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP.

[5] I. A. Farha and W. Magdy, "Mazajak: An Online Arabic Sentiment Analyser," in Proceedings of the Fourth Arabic Natural Language Processing Workshop, pp. 192–198, 2019.

[6] Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash. 2020. CAMeL Tools: An Open Source Python Toolkit for Arabic Natural Language Processing. In Proceedings of the Twelfth Language Resources and Evaluation Conference, pages 7022–7032, Marseille, France. European Language Resources Association.

[7] Antoun, Wissam and Baly, Fady and Hajj, Hazem AraBERT: Transformer-based Model for Arabic Language Understanding