Arabic stance detection of COVID-19 vaccination using transformer-based approaches: a comparison study

Reema Khaled AlRowais
Department of Computer Sciences, King Saud University, Riyadh, Saudi Arabia, and
Duaa Alsaeed
King Saud University, Riyadh, Saudi Arabia

Received 8 January 2023
Revised 3 April 2023
30 May 2023
25 July 2023
8 September 2023
Accepted 19 September 2023

Abstract

Purpose – Automatically extracting stance information from natural language texts is a significant research problem with various applications, particularly after the recent explosion of data on the internet via platforms like social media sites. Stance detection system helps determine whether the author agree, against or has a neutral opinion with the given target. Most of the research in stance detection focuses on the English language, while few research was conducted on the Arabic language.

Design/methodology/approach – This paper aimed to address stance detection on Arabic tweets by building and comparing different stance detection models using four transformers, namely: Araelectra, MARBERT, AraBERT and Qarib. Using different weights for these transformers, the authors performed extensive experiments fine-tuning the task of stance detection Arabic tweets with the four different transformers.

Findings – The results showed that the AraBERT model learned better than the other three models with a 70% F1 score followed by the Qarib model with a 68% F1 score.

Research limitations/implications – A limitation of this study is the imbalanced dataset and the limited availability of annotated datasets of SD in Arabic.

Originality/value – Provide comprehensive overview of the current resources for stance detection in the literature, including datasets and machine learning methods used. Therefore, the authors examined the models to analyze and comprehend the obtained findings in order to make recommendations for the best performance models for the stance detection task.

Keywords Stance detection, Natural language processing, Arabic transformers

Paper type Research paper

1. Introduction

Nowadays, social networking sites on the Internet have become known as the new social media, witnessing a dynamic movement of development, and spread. People can communicate their feelings and stances on social media in various ways. Thus, this prompted researchers to analyze posts on social media to gain insights. Twitter is one of the most popular social media platforms, allowing users to publish whatever they want. Since

This research project was supported by a grant from the "Research Center of College of Computer and Information Sciences", Deanship of Scientific Research, King Saud University.

Arab Gulf Journal of Scientific Research
Vol. 42 No. 4, 2024
pp. 1319-1339
Emerald Publishing Limited
e-ISSN: 2536-0051
p-ISSN: 1985-9899
DOI 10.1108/AGJSR-01-2023-0001

people can easily share their locations, opinions and feelings within a small text, Twitter-based research has become more realistic and available.

Given the significance that social media plays in modern culture and its power over various sectors of society, the field of stance detection (SD) is unquestionably essential. Its purpose is to collect valuable data for decision making by individuals, businesses, or even governments. Given the volume of data uploaded on social networking platforms, mining information from them is crucial. Thus, we can use this feature to know people's opinions or stances on specific issues (Al-Ghadir, Azmi, & Hussain, 2021).

Natural language processing (NLP) is the ability of machines to understand human language. A subfield of NLP, SD, is one of the most important research topics in NLP for automatic analysis of content (Al-Ghadir *et al.*, 2021), [3] and is a classification of the author's position for a specific target into one of three categories including in favor, against or neutral [3]. In addition, SD is a recent NLP topic and a significant amount of research is in the English language. In contrast, in Arabic NLP, it is still unexplored and in its early stages. Since Arabic SD is in its infancy, it presents significant research problems with several practical implications. Thus, this provides a motivation for conducting Arabic SD research.

The aim of this paper is to investigate and compare the use of different transformers in SD to analyze the stances of Arabic society toward COVID-19 vaccines through the social networking site Twitter. People's positions can help the government visualize how people cope with the situation.

In the current research trend, transformer models are being used for different NLP tasks, including tackling SDs. However, these research methodologies are readily available for languages such as English and Chinese. One can see that there is a minimal amount of research on Arabic SD and a general lack of Arabic resources. Therefore, this project will fill this research gap by tackling the SD problem in Arabic by leveraging various transformer models. To achieve this purpose, we have chosen two different variants of the transformer models, namely Araelectra (Antoun, Baly, & Hajj, 2021), MARBERT (Abdul-Mageed, Elmadany, & Nagoudi, 2021), AraBERT (Antoun, Baly, & Hajj, 2021) and Qarib (Chowdhury *et al.*, 2020). All these models are pretrained on a huge text corpus. This project will retrain those transformers on the specific problem of SD using the dataset previously introduced in a related work (Mubarak, Hassan, Chowdhury, & Alam, 2022), where it was used with two transformers, namely AraBERT and Qarib. However, our objective extended beyond simple comparison. We aimed to leverage the latest transformers, such as Araelectra and MARBERT, to train and develop new stance detection models.

In addition to exploring newer transformers, we retrained and developed new models using the transformers (AraBERT and Qarib) as in the related work (Mubarak *et al.*, 2022). Our comprehensive experiments involved fine-tuning the task of stance detection on Arabic tweets using these four different transformers. The primary goal was to assess their performance and compare our results to the models developed in the related work.

The remaining sections of this paper are organized as follows: section 2 gives a background in the subject matter including SD, transformer-based models. Section 3 presents related literature in the field. Proposed method is presented in section 4, while section 5 discusses experimental results. Finally, section 6 concludes this paper with main findings and future work.

2. Background

This section of the paper provides some background information needed to understand the underlying concepts of NLP and SD. First, it provides the theoretical background on SD, followed by an overview of transformer-based models.

2.1 Stance detection (SD)

The position or standing of a person, object, concept or opinion is referred to as stance (Jannati, Mahendra, Wardhana, & Adriani, 2018). Positions can be favor, against or neutral, where "favor" means directly or indirectly supporting someone or something, disagreeing or criticizing something or someone against the target; "against" means directly or indirectly rejecting or criticizing someone or something by supporting something or someone opposed to the target; and "neutral" means being in a neutral posture or failing to effectively communicate one's perspective within the paragraph (Jannati *et al.*, 2018).

The SD in NLP is to determine whether the author favors, is against, or has no opinion on a specific event or topic. It is commonly thought of as a subproblem of sentiment analysis (SA), and it seeks to determine the author's stance toward a target. The computational treatment of sentiments and opinions in texts is commonly referred to as SA. This problem is commonly equated to detecting a text producer's sentiment polarity, and a classification result, such as positive, negative or neutral, is expected from the SA technique.

The primary difference between SA and SD problems are that the former is concerned with sentiment without a specific aim, whereas the latter is concerned with a specific target (Küçük & Can, 2020). Also, within the same text, the sentiment and stance for the target may not be matched at all; that is, the text's polarity may be positive while the stance is against, and vice versa, such as "we live in a sad world when wanting equality makes you a troll" (Küçük & Can, 2020).

The SD is widely recognized to have a various practical applications, such as opinion detection, emotion recognition, sarcasm detection, fake news detection and claim validation (Padnekar, Kumar, & Deepak, 2020). It has many uses in the fields of public opinion, politics and marketing. SD is particularly interesting in the field of social media analytics, as it can aid in determining the positions of many users, possibly millions, on various problems (Darwish, Stefanov, Aupetit, & Nakov, 2020). It has also has been used in a variety of studies as a way to connect language forms and social identities to better understand the backgrounds of people who have a polarized stance (ALDayel & Magdy, 2021). However, humans are perfectly capable of determining the correct stance, while ML models typically fall short (Schiller, Daxenberger, & Gurevych, 2021) because of the various dataset sizes and ML models that have only been trained on a single dataset tend to underperform when applied to new domains.

2.2 Transformer-based models

Artificial intelligence (AI), the computational theory of learning, introduced ML, which investigates the analysis and development of algorithms that learn from raw data, train the system and generate predictions based on this train data (Chauhan & Singh, 2018). In 1959, ML was defined by Arthur Samuel, a pioneer in the field of ML, as a "field of study that gives computers the ability to learn without being explicitly programmed" (Chauhan & Singh, 2018). It focuses on how data and algorithms can be used to imitate humans in learning, analysis, decision making and improving accuracy (Chauhan & Singh, 2018).

Conventional ML algorithms and techniques aim to train a system using a training set to produce a trained model (Chauhan & Singh, 2018). However, depending on the data used with the algorithms, the learning process for these algorithms can be supervised, unsupervised or semi-supervised.

Despite being widely used, conventional ML techniques are limited in their ability to perform analysis in data in its natural form. These methods need a high level of understanding and experience; for example, feature selection demand careful engineering (Chauhan & Singh, 2018).

Deep learning (DL), on the other hand, is an advanced ML approach for teaching computers to automatically extract, analyze and understand relevant information from raw data. The outcomes of DL are far superior to those of the conventional ML (Chauhan & Singh, 2018).

DL has been hailed for not requiring as much manual feature engineering as traditional techniques; thus, they not only outperform traditional ML, but they also require less human work, making their adoption easier (Magnini, Lavelli, & Magnolini, 2020). On the other hand, DL algorithms require a massive amount of data to train a network, unlike conventional ML (Chauhan & Singh, 2018).

DL algorithms use multiple processing layers to learn hierarchical data representations and have shown excellent results in numerous fields. In the past few years, NLP has been increasingly focusing on the use of new DL approaches, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) (Young, Hazarika, Poria, & Cambria, 2018). Although CNNs and RNNs are widely used, the primary disadvantage of these DL models is that they require a large number of labeled instances for training, which are expensive to build and maintain (Kalyan, Rajasekharan, & Sangeetha, 2021). Fortunately, one solution that addresses this very problem is transfer learning. This is an ML method in which we reuse a previously trained model as the foundation for a new model on a new task. Simply put, a model trained on one task is repurposed on a second related task as an optimization that leads to faster progress when modeling the second task (Alyafeai, AlShaibani, & Ahmad, 2020). Figure 1 illustrates the basic transfer learning pipeline. As shown, the data needed for building the source model are very large; however, when the knowledge is transferred and the pretrained source model is used to develop a new task-specific model, the data required are far smaller.

Transformers are a type of transfer learning, and transformer-based models are pretrained models, which means a large dataset of labeled instances for training is not needed and the trained model will be fine-tuned to the new data of a specific domain (Daniel Jurafsky, n.d.; Alyafeai *et al.*, 2020).

Transformers are a new model of DL that uses complicated neural networks. It primarily makes use of the self-attention process to extract intrinsic characteristics and has a big future for AI applications (Vaswani *et al.*, 2017; Han *et al.*, 2021).

Antoun, Baly *et al.* (2021) proposed AraBERT, the first transformer-based language model for Arabic. AraBERT was evaluated for different tasks in Arabic NLP and achieved state-of-

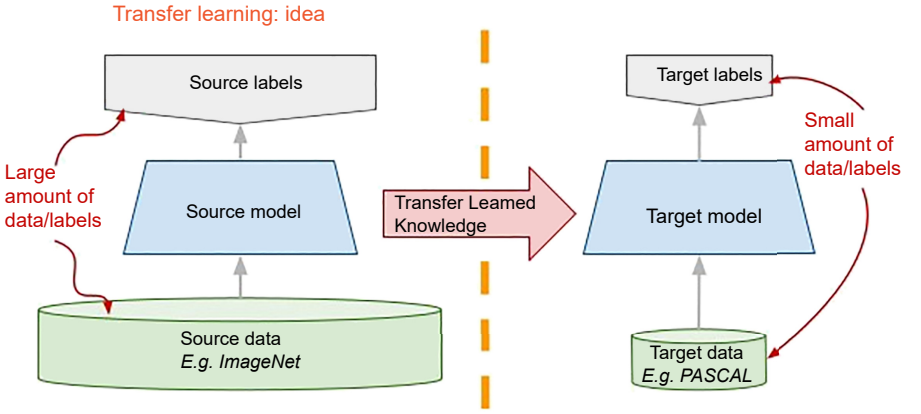


Figure 1.
Transfer learning outline

Source(s): Author's own work; Kalyan *et al.* (2021)

the-art performance on SA, question answering and named-entity recognition. ARBERT and MARBERT, provided by Abdul-Abdul-Mageed *et al.* (2021), assist transfer learning on modern standard Arabic (MSA) and Arabic dialects, and it has been pretrained on massive datasets; ARERT was trained on 61 GB of MSA, and MARBERT was trained on 28 GB of dialectal Arabic using social media. QARiB provided by Chowdhury *et al.* (2020) was evaluated for Arabic text categorization and trained on the Arabic GigaWord corpus, Abulkhair Arabic Corpus, OpenSubtitles and 50 million tweets. The topologies, sizes and types of training data used by these models differ (Abu Farha & Magdy, 2021).

3. Related work

This section gives a review of recent works related to SD in Twitter data. From our review, we found that most of the work in SD can be categorized according to the approach into three approaches: rule-based ML (RBML), DL algorithms and transformer-based methods.

Several studies have been conducted using RBML for SD. Tun and Hninn Myint (2019) proposed a method for detecting stance in Twitter, as a two-phase approach for stance classification. In the first phase, the Naive Bayes classifier was used to classify the tweet to determine whether the stance was neutral or non-neutral. In the second phase, the decision tree (DT) classifier was used to classify whether the non-neutral tweets were favored or against. Moreover, the method was tested on the dataset "SemEval 2016 Task A, Task B" and 4,870 tweets. In addition, the majority class and SVM-ngrams-comb were employed to compare with the proposed approach, which shows the proposed approach a good performance with appropriate F1 score 68.6 in task A and F1 score 59.2 in task B of measure in compared to baseline approaches, which was higher F1 score achieved is 68.98 in task A and 29.72 in task B (Tun & Hninn Myint, 2019).

To detect the stance of people's tweets, Aldayel and Magdy (2019) employed additional features about the users' utilized features, such as their topic postings, the networks they followed, the websites they frequently visited and the stuff they enjoyed. To classify the stance of tweets, the researchers used an SVM model with a linear kernel. However, a macro-average of the F1 scores for the "against" and "favor" classes was calculated, with the F1 scores for the "none" class omitted. On the SemEval-2016 Task 6 dataset, the method showed promising results with an F1 score of 72.49% compared to the baseline, which achieved an F1 score of 68.98%. Another study (Al-Ghadir *et al.*, 2021) was conducted by Santosh *et al.* using the same dataset. The experiment examined different classifiers, which are SVM, k nearest neighbor (K-NN), weighted K-NN (WKNN) and class-based K-NN (CKNN) to evaluate the SD. WKNN was the most potent classifier with an F1 score of 76.45%.

Darwish *et al.* (2020) developed an unsupervised framework for determining Twitter users' opinions on sensitive topics. Their method employed dimensionality reduction by using the Uniform Manifold Approximation and Projection (UMAP) algorithm to project people into a low-dimensional space, followed by mean shift for clustering to core users who represent the various stances. They compared their framework to previous methods, which were based on semi-supervised or supervised categorization. The results showed that using UMAP with more than 98% accuracy, these setups were able to identify groupings of users according to the main stance on difficult topics. They found that their framework offers some key advantages: first, the method creates clusters without requiring users to label; second, they do not require domain- or topic-level knowledge to conduct the labeling.

Kovacs, Cotfas, Delcea, and Florescu (2023) analyze the tweets about COVID-19 vaccination using the SVM model to identify the gender of the author, the result showed 85% classification accuracy. Also, they detected the stance and showed that RoBERTa was the most effective classifier accuracy 93.64%.

On the other hand, new approaches relying on DL were applied in the SD field. The fake news challenge (FNC) benchmark dataset has been used in several studies for SD. Padnekar *et al.* (2020) presented a stance prediction architecture based on bidirectional long short-term memory (Bi-LSTM) and autoencoder. The system was tested using the FNC-1 corpus, which has a training set of 50,000 data sets and a test set of 25,000 with a stance label. As a result, their proposed method showed 94% classification accuracy. Another study (Santosh, Bansal, & Saha, 2019) used the same dataset provided by the FNC and aimed to demonstrate the effectiveness of the Siamese adaptation of LSTM networks for SD. The result showed a high FNC score and accuracy with an FNC score of 0.85 compared with baseline, which was the highest achieved FNC score of 0.82. Also, Mohtarami *et al.* (2018) used the same dataset with different classifiers, namely SVM, (K-NN), WKNN and class-based K-NN (CKNN) to assess position detection. WKNN was the most effective classifier F1 score of 76.45%.

Sobhani, Inkpen, and Zhu (2017) presented a multitarget stance dataset for each instance. They proposed a bidirectional RNN-based attentive encoder decoder to capture the interdependence between stance labels for numerous targets. For example, a model built using this approach should be able to classify a tweet in terms of Clinton and Trump at the same time. While the framework allows for more than two targets, it is still limited to a limited number of targets.

Liviu-Adrian *et al.* (2021) and Ahmed *et al.* (2023) investigated the opinion dynamics surrounding the COVID-19 vaccine by performing sentiment analysis on various vaccine-related tweets.

Recently, most of the research in the field of NLP in general and in SD in particular focuses on a transformer approach. Schiller *et al.* (2021) presented a feature vector for multidataset learning (MDL) based on the BERT architecture. Multitask and MDL can improve the accuracy and resilience of SD, according to research. The findings indicate that transfer learning and multi-task learning can also help enhance performance. Another (Lin, Wu, Chou, Lin, & Kao, 2020) was conducted by Mohtarami *et al.* using BERT to detect the stance in FNC-1 and gave an accuracy of 88.7%. Ghosh, Singhania, Singh, Rudra, and Ghosh (2019) examined SD approaches on two datasets (SemEval 2016, and multiperspective consumer health query (MPCHI)) and discovered that the BERT pretrained model outperforms existing approaches for SD with a performance F1- score of 0.751 on the SemEval dataset and an F1-score of 0.756 on MPCHI. Müller, Salathé, and Kummervold (2020) proposed a model COVID-Twitter-BERT (CT-BERT) that was pretrained on a large corpus of COVID-19-related Twitter messages. Therefore, one of the datasets referred to the stance of maternal vaccines. The results show that CT-BERT has a higher performance, with an average F1 score of 0.833, compared to BERT-LARGE (Devlin, Chang, Lee, & Toutanova, 2019) with an average F1 score of 0.802.

From reviewed studies, one can say that most of the work in SD is on the English language with minimal work on other natural languages. One of the studies on SD in other languages is a study for the Italian language conducted by Kayalvizhi, Thenmozhi, and Chandrabose (2021). In this study, BERT was applied to transformer to detect authors' stance in their tweets. The evaluation of their model gave an average F1 score of 47.07, and the BERT model outperformed the encoder–decoder model, which gave an average F1 score of 0.4473.

Alhindi, Alabdulkarim, Alshehri, Abdul-Mageed, and Nakov (2021) presented a new Arabic SD dataset (AraStance), which contained 4,063 claim-articles from various domains and Arab countries. They investigated a variety of BERT-based models that were pretrained on Arabic or multilingual data, then fine-tuned and applied it to their dataset. The best model had a macro F1 score of 78% and an accuracy of 85%.

Table 1 summarizes the above reviewed studies on SD. To conclude, the majority of research on SD is in English language, and there is a gap in in the Arabic language. A review of the literature has also shown that there are many developments in the field of NLP and on

Source	Natural language	Data type and name	Techniques	Compared with	Result
ML approaches					
Tun and Hninn Myint (2019) 2019	English	Twitter (SemEval2016 Task A and Task B)	Naive Bayes classifier and decision tree classifier	SVM-ngrams F1 score 68.98 in Task A Majority Class F1 score 29.72 in Task B	F1 score 68.6 in Task A F1 score 59.2 in Task B
Aldayel and Magdy (2019) 2019		Twitter (SemEval2016 Task 6)	SVM model with a linear kernel	linear SVM model F1 score of 68.98% (Mohammad, Kiritchenko, Sobhani, Zhu, & Cherry, 2016)	F-measure of 72.49%
Al-Ghadir *et al.* (2021) 2021			WKNN, CKNN and SVM	–	WKNN was the most powerful classifier. F-score of 76.45%
Darwish *et al.* (2020) 2020	English and Turkish	2 type DS 1-label dataset "Kavanaugh" (English), "Trump" (English) and "Erdogan" (Turkish) 2-Unlabeled "Collected tweets on six polarizing topics in the USA"	Unsupervised approach UMAP, mean shift	supervised approach SVM fasttext (Joulin, Grave, Bojanowski, & Mikolov, 2016) Precision 86.0%	precision 99.1% cluster purity
Ahmed *et al.* (2023)	English	"COVID-19 All Vaccines Tweets"	Extra tree classifier (ETC)	TF-IDF, BoW, Word2Vec	ETC outperform BoW with 92% accuracy
DL approaches					
Sobhani *et al.* (2017) 2017	English	collected tweets related to the 2016 US election. Selected four presidential aspirants: "Donald Trump," "Hillary Clinton" "Ted Cruz" and "Bernie Sanders" as targets	deep RNNs (Seq2Seq)	SVM (Pedregosa *et al.*, 2011) F-macro of 52.05	F-macro of 54.81

(*continued*)

Table 1.
A summary of stance detection studies

Source	Natural language	Data type and name	Techniques	Compared with	Result
Padnekar *et al.* (2020) 2020	English	"Fake News Challenge (FNC)"	BiLSTM	–	Accuracy 94%
Santosh *et al.* (2019) 2020			Siamese adaptation of LSTM networks	Baseline (MLP-6) FNC score 0.819	FNC score of 0.85
Mohtarami *et al.* (2018) 2021			CNN + LSTM (sMemNN)	Baseline CNN LSTM macro-F1 of 40.33	Macro-F1 of 56.75
Transformer approaches					
Schiller *et al.* (2021) 2021	English	combined datasets from different domains (ibmcs-semeval2019t7-semeval2016t6-fnc1- snipes- scd-perspectrum- iac1-arc- argmin)	BERT	–	Transfer learning and multi-dataset learning can improve the performance
Alhindi *et al.* (2021) 2021	Arabic	Arabic Stance Detection dataset (AraStance) of 4,063 claim it covers false and true claims from multiple domains (e.g. politics, sports, health) and several Arab countries	BERT	–	Accuracy of 85% and a Macro F1 score of 78%
Lin *et al.* (2020) 2020	English	"FAKE NEWS CHALLENGE" STAGE 1(FNC-1)	BERT	CNN + LSTM LSTM + CNN Macro-F1 of 40.33 (Mohtarami *et al.*, 2018)	Macro-F1 of 75.96
Kayalvizhi *et al.* (2021) 2020	Italian	Italian tweets about the Sardines movement	BERT	encoder-decoder model F1 score of 0.4473	F1-average score of 0.47
Ghosh *et al.* (2019) 2020	English	SemEval 2016 and MPCHI	BERT	CNN (*Pkudblab at SemEval-2016 Task 6, n.d.*) F1 score of 0.690 (semval dataset)	F1 score of 0.75
Müller *et al.* (2020) 2020	English	Maternal Vaccine Stance (MVS)	CT-BERT	BERT-LARGE (Devlin *et al.*, 2019) Mean F1 -score of 0.802	Mean F1 score of 0.833

Table 1.

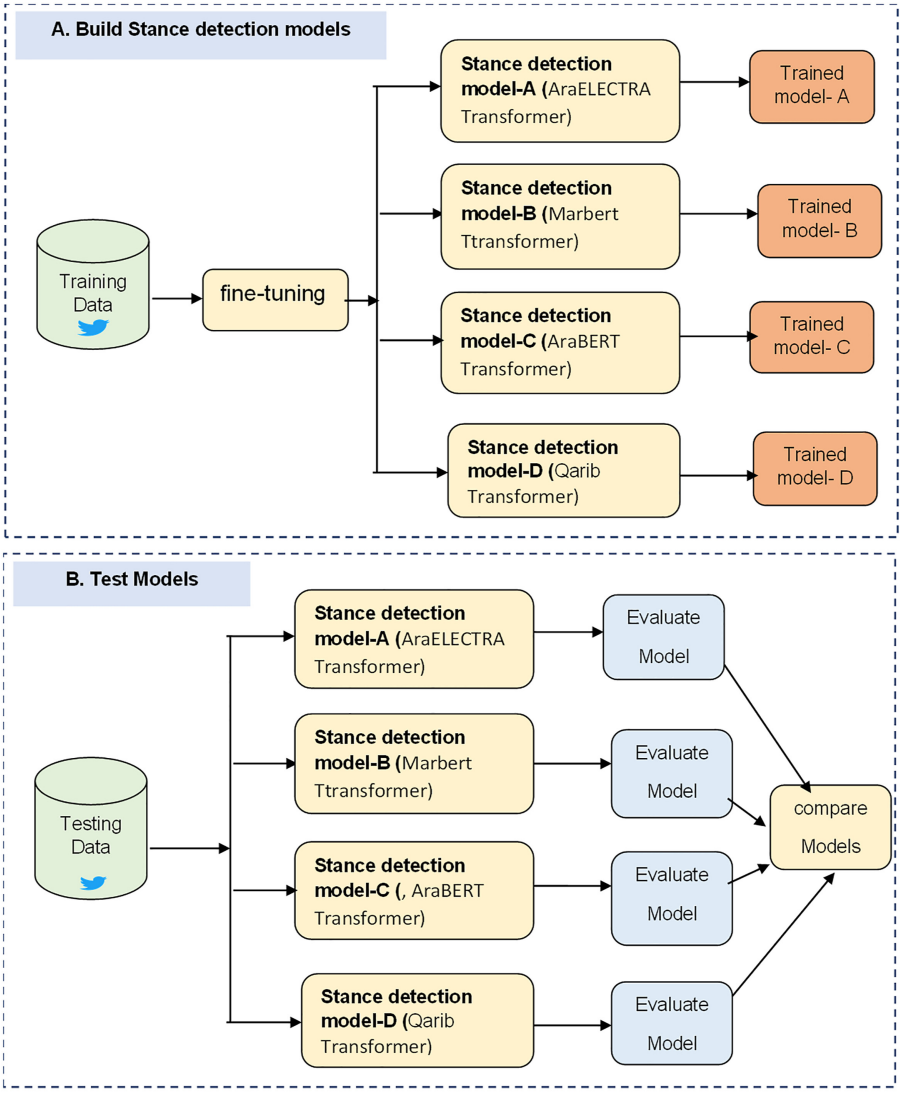
Source(s): Authors' own work

the use of transfer learning. This encouraged us to investigate the use of transformers for SD in Arabic.

4. Proposed method

This section discusses the proposed method and shows the workflow of experiments. First, we will perform some preprocessing operations on the dataset. Second, we chose some

pretrained transformer-based models and retrain them on SD problem using an Arabic dataset for SD. Since the style data used to train these pretrained models is from a different application and for a different task, it is necessary to fine-tune these models for our own domain and target task using domain related data, and in our case, we need to fine-tune the model for SD using COVID-19 vaccination Twitter data. Third, we will use a set of evaluation metrics to evaluate the performance of all retrained transformer-based models. Finally, we will analyze and compare the models' performance. The framework adopted in this study is illustrated in Figure 2. Next, we will discuss in brief the different parts of this framework.



Source(s): Author's own work

Figure 2.
Framework of stance detection

4.1 Dataset

In this study, the models will be trained on the manually annotated Arabic tweet dataset for the COVID-19 vaccination "ArCovidVac" (Mubarak *et al.*, 2022). The dataset size is 10k tweets, which covers many Arab regions. The stance was annotated using the following labels: pro-vaccination (positive), neutral and anti-vaccination (negative). To collect the tweets, they used the twarc [1] search API and the following keywords: تطعيم,لقاح,مطعوم between January 5 and February 3, 2021. They used the Appen [2] crowdsourcing platform for manual annotation and used the standard evaluation (Alshaabi *et al.*, 2021) to improve the quality of the annotation, such as to participate in the annotation activity, each annotator required to complete at least 70% of the tweets. Moreover, they calculated the annotation agreement using Cohen's kappa coefficient and discovered a score of 0.82, indicating high annotation quality (Mubarak *et al.*, 2022). Figure 3 shows distribution of the dataset.

In the experiment setting, the regular training used the 80:20 splitting rule, and a 5-fold validation with an ensemble approach is used for more reliable results.

4.2 Data preprocessing

The collected texts from the web are unstructured. The data included irrelevant data such as English letters, special characters, punctuation and typographical errors format such as Hamza-Alif (أ) and bare Alif (ا). They must be converted to a machine-readable format using data cleaning and preprocessing techniques.

The following are the data cleaning and preprocessing steps that are usually applied to Arabic language corpus:

(1) Remove numbers, all English letters and newline.

(2) Removes all words containing underscore, hashtag sign and @ sign from the corpus.

(3) Remove special characters including symbols and emojis.

(4) Remove repeating character such as "هههههه. "

(5) Arabic normalization, which is the unification of some characters that have many forms, demonstrated in Figure 4.

Label	Number of tweets	Example
Positive	7,968	خادم الحرمين الشريفين، الملك سلمان، يتلقى الجرعة الأولى من لقاح كورونا. Custodian of the Two Holy Mosques, King Salman, receives the first dose of the Corona vaccine.
Negative	638	زعيم ديني يهودي شهير: أخذ لقاح كورونا سيحولكم إلى مثليين! Famous Jewish Religious Leader Taking the Corona Vaccine Will Turn You Gay
Neutral/Unclear	1,396	أخذ لقاح فيروس كورونا ليس شرطاً للسماح بالسفر إلى الخارج.:الصحة Health: Taking the Corona virus vaccine is not a requirement to be allowed to travel abroad.

Figure 3.
Distribution of the dataset

Source(s): Author's own work

4.3 Fine-tuning transformer models

As mentioned earlier, this project aims to explore the use of transformers in Arabic SD. Thus, we will perform extensive experiments on retraining several pre-trained transformer models for SD problems using COVID-19 vaccination Twitter data. These transformers are as follows:

(1) Efficiently Learning an Encoder that Classifies Token Replacements Accurately (ELECTRA) transformer was selected as it proved its effectiveness in Arabic SA and named-entity recognition. The training dataset used with it was 77 GB and was mostly made up of news articles.

(2) The second type of transformer to be used is MARBERT, which has been examined for a variety of NLP applications such as SA, classification and dialect identification. MARBERT was trained on the Arabic Twitter dataset of size 1B, which uses both MSA and dialectal Arabic.

(3) AraBERT is an Arabic-specific BERT provided by Antoun Baly *et al.* (2021). However, AraBERT used about 24 GB of data and 64k vocab size. It has been used in various Arabic NLP tasks, such as SA, named-entity recognition and question answering.

(4) Qarib was provided by Chowdhury *et al.* (2020). This model was trained using a variety of data sources, including news articles and tweets.

4.4 Fine-tuning process

Since the pretrained models were trained on data from a different domain and for a different NLP task, one needs to retrain the model for the specific task using problem domain-related data. This retraining involves fine-tuning the model. Fine-tuning is a supervised learning method in which the weights from pretrained models are used as the first weights for a new model being trained on a task (Imran & Amin, 2021). This technique not only expedites training but also produces a state-of-the-art model for a number of different of NLP tasks (Imran & Amin, 2021). It is an extremely effective training strategy that uses a pretrained model and trains it on a dataset relevant to your task. Fine-tuning is a supervised learning procedure in which the weights of a previously trained model are utilized as the beginning weights for a newly learned model on a similar task. This technique not only expedites training but also produces a state-of-the-art model for a wide range of NLP tasks (Rifat & Imran, 2021).

In this project, to retrain a pretrained model in a new task—as in the case with transformers—we will need to fine-tune the training parameters to get the best performance for each model.

Training parameters: When training a model, hyperparameter tuning uses the processing infrastructure to evaluate multiple hyperparameter configurations. It can provide us with

After normalization	letter
ي	ى
و	ؤ
ي	ئ
ه	ة
ك	گ
ا	إأآ

Source(s): Author's own work

Figure 4.
Arabic text normalization

optimal hyperparameter values, thereby increasing the prediction accuracy of our model. Hyperparameters are critical for improving model performance in NLP. The main parameters are as follows:

(1) *Batch size:* The number of instances (up to 128 token sequences) in each mini-batch. We try batch sizes of 128 since our hardware has enough memory (Izsak, Berchansky, & Levy, 2021).

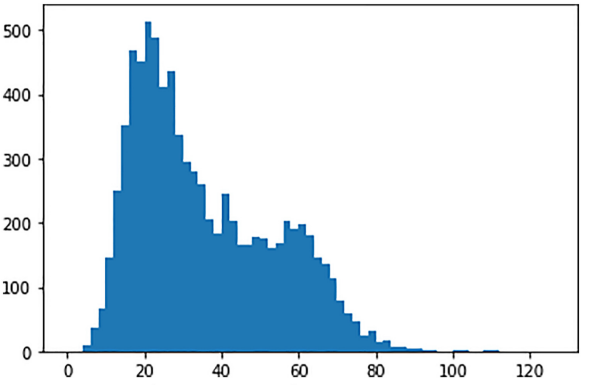
(2) *Learning rate:* Starting with 2e–5, the learning rate will gradually increase until it heats up to the peak learning rate before it begins to decline. We tested a range of learning rates between 2e–5 and 10e–5.

(3) *Epoch:* This is the number of times the entire dataset must be passed. Since our disk has limited memory, we tested 2–4 epochs.

(4) *Sequence length:* Another important parameter is the input sequence length must also be determined. Figure 5 presents the frequency distribution of the text sequence length. The average length of a sequence is 20, and the highest length is 120. Therefore, in order to preserve the data as much as possible, we selected the largest sequence length available in the data (120).

To determine the optimal number of epochs, we manually tuned the hyperparameters on the suggested values and set a fixed number of four epochs while observing the validation loss.

In our fine-tuning process, we followed the fine-tuning strategy recommended by Clark, Luong, Le, and Manning (2020) and Mubarak *et al.* (2022) and we used the same training parameters for each of the models. Table 2 shows a list of the different values for each parameter.



Source(s): Author's own work

Figure 5.
Sequence length distribution

Parameter	Value
Learning rate	2e–5–10e–5
Batch size	128
Number of epochs	2–4
Sequence length	120

Source(s): Authors' own work

Table 2.
Parameter values

4.5 Evaluation

Based on our literature review, the commonly used measures for evaluating the performance of the AI model in NLP in general and for SD models in particular, are accuracy, precision, recall and F1 score. Thus, we will use these metrics to evaluate and compare the performance of different models. Accordingly, we selected the best-performing model to use in our dashboard. These evaluation metrics depend on counting the number of the following factors.

(1) True positive (TP): Correctly classify an observation as positive

(2) False positive (FP): Wrongly classify a negative observation as positive

(3) True negative (TN): Correctly classify an observation as negative

(4) False negative (FN): Wrongly classify a positive observation as negative

Table 3 shows the description and equation of each metric (Daniel Jurafsky, n.d.).

5. Experiments and results

In this section, we describe the experimental setup in detail, utilizing four different models. Thus, we investigated at a large variety of hyperparameters and analyze the effect of each hyperparameter on model performance by synchronizing the learning rate schedule with each epoch for each model.

As planned, we trained the four transformers (Araelectra, Marabert, AraBERT and Qarib). For fine-tuning the training parameters, we tested all different settings for each of the four transformers using different values for learning rate and number of epochs, as shown in Table 4, and the setting of batch size was 120. To determine the optimal number of epochs, we manually tuned the hyperparameters on the suggested values and set a fixed number of four epochs while observing the validation loss.

In the following subsections, we discuss the performance of the developed models using each of the four transformers.

5.1 Results of Araelectra

The stance-detection model developed with Araelectra was tested with different parameter settings; the results are shown in Table 4. The performance of the model was enhanced after 4 epochs compared to 2 and 3 epochs. Within the four epochs, the peak performance was attained when the learning rate was (9e–5). Therefore, with these parameters, the model attained an accuracy of (83%) and an F1 score of (66%). However, the worst performance with Araelectra was in two epochs and the learning rate (2e–5) in which the accuracy was 80% and F1 score was 33%.

Metric	Description	Equation
Accuracy (AC)	The proportion of correctly classified observations	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision (P)	The ratio of observations correctly classified as positive to all observations classified as positive	$\frac{TP}{TP+FP}$
Recall (R)	The ratio of observations correctly classified as positive to all actual positive observations	$\frac{TP}{TP+FN}$
F1 score (F1)	It is the harmonic mean (average) of precision and recall	$\frac{2PR}{P+R}$

Source(s): Authors' own work

Table 3. Evaluation metrics

Learning rate	Epochs	Accuracy	F1	Recall	Precision
2e–5	2	80%	33%	35%	51%
3e–5	2	80%	40%	40%	43%
5e–5	2	81%	54%	50%	69%
2e–5	3	80%	43%	44%	44%
3e–5	3	81%	54%	51%	66%
5e–5	3	82%	60%	57%	66%
2e–5	4	81%	52%	50%	68%
3e–5	4	82%	60%	57%	65%
5e–5	4	83%	64%	61%	67%
8e–5	4	83%	65%	63%	68%
9e–5	4	83%	66%	65%	67%
10e–5	4	83%	64%	62%	67%

Source(s): Authors' own work

Table 4. The performance of Araelectra

5.1.1 Results of Marabert. Table 5 presents the performance of the Marabert model with different parameters. As with Araelectra, the performance improved after 4 epochs. The best performance was attained at the learning rate of (5e–5); the accuracy achieved was 84% and the F1 score was 67%. Araelectra's worst performance was at a learning rate of (2e–5) and two epochs.

5.1.2 Results of Qarib. Qarib is another transformer that was used in Mubarak *et al.* (2022) to develop an SD model. The authors in Mubarak *et al.* (2022) reported the best result as accuracy of 81% and an F1 score of 63% with three epochs and a learning rate of 8e–5. As with AraBERT, we tested Qarib with several parameter settings, and the results are shown in Table 6. A similar effect to that seen with AraBERT was observed in Qarib. The performance was enhanced with the increase in number of epochs and change in learning rate. The best results achieved by Qarib were an accuracy of 84% and an F1 score of 68%) with four epochs and a learning rate of (8e–5).

5.2 Results of Arabert

As discussed earlier in the literature review chapter, the AraBERT transformer has been used to develop a SD model (Mubarak *et al.*, 2022). In their experiments, they achieved an accuracy of 82.2% and an F1 score of 62.2% at a learning rate of 8e–5 and 3 epochs. To compare the performance of different transformers with AraBERT and to investigate the effect of

Learning rate	Epochs	Accuracy	F1	Recall	Precision
2e–5	2	83%	53%	50%	75%
3e–5	2	84%	60%	56%	69%
5e–5	2	84%	63%	60%	68%
2e–5	3	84%	63%	59%	69%
3e–5	3	84%	64%	61%	68%
5e–5	3	84%	66%	65%	68%
2e–5	4	84%	63%	61%	67%
3e–5	4	84%	66%	64%	68%
5e–5	4	84%	67%	65%	68%
6e–5	4	84%	66%	64%	68%
8e–5	4	84%	65%	62%	68%

Source(s): Authors' own work

Table 5. The performance of Marabert

Learning rate	Epochs	Accuracy	F1	Recall	Precision
2e–5	2	83%	54%	49%	70%
3e–5	2	83%	59%	55%	68%
5e–5	2	83%	64%	60%	70%
2e–5	3	83%	61%	57%	69%
3e–5	3	84%	66%	63%	70%
5e–5	3	84%	67%	65%	70%
2e–5	4	84%	65%	61%	69%
3e–5	4	84%	66%	64%	70%
5e–5	4	84%	68%	65%	70%
8e–5	4	84%	68%	67%	68%

Source(s): Authors' own work

Table 6. The performance of Qarib

changing learning parameters, we decided to test AraBERT again on the same data set but with different parameters, such as different values of learning rate and epochs. In addition, a sequence length of 120 was used, which was not reported in their work. The different performance results achieved in different settings are reported in Table 7. As shown, the increase in the number of epochs improved the performance of our model compared to what was achieved in Mubarak *et al.* (2022) with two epochs. The best performance achieved was an accuracy of 85% and F1 score of 70% at a learning rate of 8e–5 and 4 epochs.

5.3 Comparison between different transformers

To compare the performance of all transformers tested in this work, we compared the best achieved performance for each transformer, Table 8 and Figure 6 show those results. The performance of all transformers based on accuracy ranged between 83% and 85%, showing small differences However, the F1 measure shows a wider range of differences, as it ranged from 66% to 70%. Taking all evaluation metrics into consideration: accuracy, F1, precision and recall, we found that AraBERT outperformed other transformers in all measures with a setting of 8e–5 learning rate and four epochs. Conversely, Araelectra had the worst performance in all metrics of all models.

Learning rate	Epochs	Accuracy	F1	Recall	Precision
2e–5	2	82%	55%	50%	68%
3e–5	2	83%	60%	55%	68%
5e–5	2	84%	64%	61%	70%
2e–5	3	83%	62%	59%	68%
3e–5	3	84%	66%	64%	70%
5e–5	3	85%	68%	66%	71%
2e–5	4	84%	66%	64%	70%
3e–5	4	85%	68%	67%	70%
5e–5	4	85%	69%	68%	71%
8e–5	4	85%	70%	68%	72%
9e–5	4	85%	70%	69%	71%
10e–5	4	85%	70%	69%	71%
11e–5	4	85%	70%	69%	70%
12e–5	4	84%	69%	69%	69%

Source(s): Authors' own work

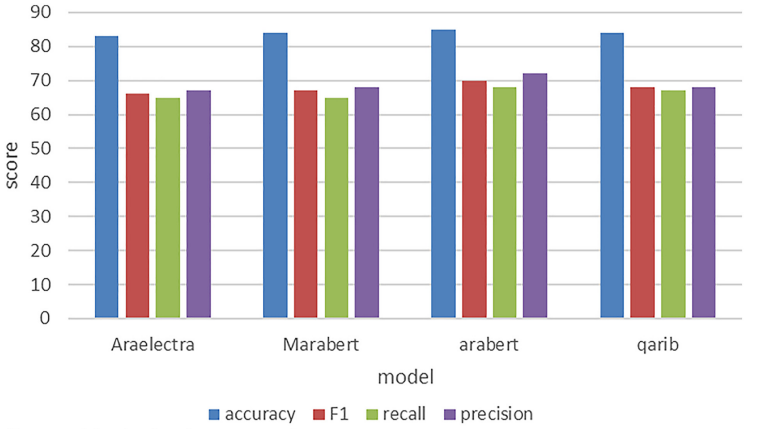
Table 7. The performance of Arabert

As discussed earlier, the performance of different transformers was affected by the change in the number of epochs. Figure 7 shows the different F1 scores achieved with different epochs. As clearly seen, all models (Araelectra, Marabert, AraBERT and Qarib) reached their peak at four epochs.

Model	Learning rate	Epochs	Accuracy	F1	Recall	Precision
Our results						
Araelectra	9e–5	4	83%	66%	65%	67%
Marabert	5e–5	4	84%	67%	65%	68%
AraBERT	8e–5	4	85%	70%	68%	72%
Qarib	8e–5	4	84%	68%	67%	68%
Results reported in (Mubarak *et al.*, 2022)						
AraBERT (Mubarak *et al.*, 2022)	8e–5	3	82%	62%	62%	61%
Qarib (Mubarak *et al.*, 2022)	8e–5	3	81%	62%	65%	64%

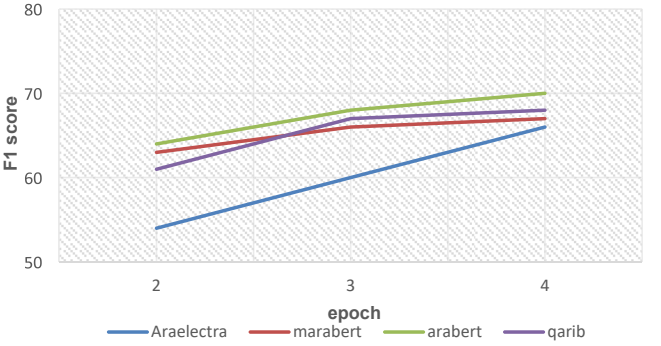
Source(s): Authors' own work

Table 8.
Results for different models



Source(s): Author's own work

Figure 6.
The performance for different model



Source(s): Author's own work

Figure 7.
F1 over the number of epochs

We also compared the performance of Qarib and AraBERT in our work with the results reported in Mubarak *et al.* (2022). As seen in Table 8, both AraBERT and Qarib have enhanced results compared to what was achieved in (Mubarak *et al.*, 2022) in terms of accuracy, precision, recall and F1.

The change in training parameters was proven to enhance performance. Figure 8 shows a comparison between the results in both transformers achieved in our work and those reported in Mubarak *et al.* (2022). In terms of accuracy, AraBERT improved from 82% to 85% and increased in F1 from 62% to 70%. A similar observation was made on Qarib; the accuracy increased to 81% from 84% in Mubarak *et al.* (2022) and F1 to 62% from 68%.

6. Limitations

This study has limitations posed by the imbalanced dataset and its potential impact on model performance. As part of our future work, we plan to address this limitation by undertaking a new study to construct our own dataset for Arabic stance detection.

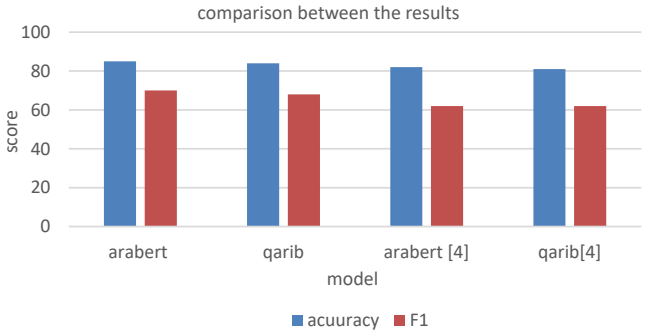
7. Conclusion and future work

In this study, we aimed at investigating and comparing the performance of transformer-based models in Arabic stance detection. We developed transformer-based models for Arabic SD, which was applied as a case study on Covid-19 vaccination using Arabic Twitter data, and the ArCovidVac AraBERT (Mubarak *et al.*, 2022) dataset.

In our extensive experiment, we tested four Arabic transformers (Araelectra, Marabert, AraBERT and Qarib). As for stance classification performance, our AraBERT-based model outperformed other tested transformers and represents a high-performance classifier with an F1 score of 70%. It also outperformed the model developed by Mubarak *et al.* (2022), in which the F1 score was 82%. The worst-performing model was based on Araelectra, with an F1 score of 66%.

Research in Arabic SD research faces a significant challenge due to the lack of Arabic stance detection datasets, which limited options for alternative sources of data. And in our research here, we encountered this challenge and to overcome this challenge, we turned to a dataset previously introduced in Mubarak *et al.* (2022) as a starting point for our research. And this was also a good option as our aim was to compare the use of transformers in Arabic SD with results found in study by Mubarak *et al.* (2022) that introduced this dataset.

While the dataset helped us establish a foundation for our work, we recognized the need for improvements in the methodology to address the issues related to dataset imbalance.



Source(s): Author's own work; Mubarak *et al.* (2022)

Figure 8.
Comparison between our results and study

As part of our future work, we acknowledge the importance of addressing the problem of imbalanced data directly. We intend to explore techniques specifically designed to tackle the class imbalance issue in Arabic SD datasets and compare the results to our results found in this study.

In addition, another challenge is the small dataset size is small, and these issues may affect the re-training and learning process. Therefore, a future extension of this work would be to experiment with datasets and explore the effect of dataset size on the performance of different transformers. To accomplish this, we need to build a new larger corpus for SD from Twitter Arabic data. And we have already started by collecting a new dataset of tweets to construct a corpus with balanced distribution across the three stance labels, unlike the case with ArCovidVac dataset AraBERT (Mubarak *et al.*, 2022). Part of the process of creating a corpus is data annotation. For this purpose, three Arabic native speakers were recruited to annotate the collected tweets. Annotation guidelines were given to the annotators to explain the labels and assure accurate labeling. The adopted guidelines are similar to those proposed by Alhindi *et al.* (2021).

Another research direction to expand and enhance this work is testing the models on other case studies not related to COVID-19 using Twitter data on general or other trending topics in the social media.

Notes

1. https://twarc-project.readthedocs.io/en/latest/
2. https://appen.com/solutions/crowd-management/

Reference

Abdul-Mageed, M., Elmadany, A., & Nagoudi, E. M. B. (2021). Arbert & MARBERT: Deep bidirectional transformers for Arabic, ArXiv:2101.01785 [Cs]. Available from: http://arxiv.org/abs/2101.01785

Abu Farha, I., & Magdy, W. (2021). Benchmarking transformer-based language models for Arabic sentiment and sarcasm detection. *Proceedings of the Sixth Arabic Natural Language Processing Workshop* (pp. 21–31). Available from: https://aclanthology.org/2021.wanlp-1.3

Ahmed, S., Khan, D. M., Sadiq, S., Umer, M., Shahzad, F., Mahmood, K., . . . Ashraf, I. (2023). Temporal analysis and opinion dynamics of COVID-19 vaccination tweets using diverse feature engineering techniques. *PeerJ Computer Science*, *9*, e1190. doi: 10.7717/peerj-cs.1190.

Al-Ghadir, A. I., Azmi, A. M., & Hussain, A. (2021). A novel approach to stance detection in social media tweets by fusing ranked lists and sentiments. *Information Fusion*, *67*, 29–40. doi: 10.1016/j.inffus.2020.10.003.

Aldayel, A., & Magdy, W. (2019). Your stance is exposed! Analysing possible factors for stance detection on social media. *Proceedings of the ACM on Human-Computer Interaction*, *3*(CSCW), 205:1–205:20. doi: 10.1145/3359307.

ALDayel, A., & Magdy, W. (2021). Stance detection on social media: State of the art and trends. *Information Processing and Management*, *58*(4), 102597. doi: 10.1016/j.ipm.2021.102597.

Alhindi, T., Alabdulkarim, A., Alshehri, A., Abdul-Mageed, M., & Nakov, P. (2021). AraStance: A multi-country and multi-domain dataset of Arabic stance detection for fact checking, ArXiv: 2104.13559 [Cs]. Available from: http://arxiv.org/abs/2104.13559

Alshaabi, T., Dewhurst, D. R., Minot, J. R., Arnold, M. V., Adams, J. L., Danforth, C. M., & Dodds, P. S. (2021). The growing amplification of social media: Measuring temporal and social contagion dynamics for over 150 languages on Twitter for 2009-2020. *EPJ Data Science*, *10*(1), 15. doi: 10.1140/epjds/s13688-021-00271-0.

Alyafeai, Z., AlShaibani, M. S., & Ahmad, I. (2020). A survey on transfer learning in natural language processing, ArXiv:2007.04239 [Cs, Stat]. Available from: http://arxiv.org/abs/2007.04239

Antoun, W., Baly, F., & Hajj, H. (2021). AraBERT: Transformer-based model for Arabic language understanding, ArXiv:2003.00104 [Cs]. Available from: http://arxiv.org/abs/2003.00104

Antoun, W., Baly, F., & Hajj, H. (2021). AraELECTRA: Pre-Training text discriminators for Arabic language understanding, ArXiv:2012.15516 [Cs]. Available from: http://arxiv.org/abs/2012.15516

Chauhan, N. K., & Singh, K. (2018). A review on conventional machine learning vs deep learning. In *2018 International Conference on Computing, Power and Communication Technologies (GUCON)* (pp. 347–352). doi: 10.1109/GUCON.2018.8675097.

Chowdhury, S. A., Abdelali, A., Darwish, K., Soon-Gyo, J., Salminen, J., & Jansen, B. J. (2020). Improving Arabic text categorization using transformer training diversification. *Proceedings of the Fifth Arabic Natural Language Processing Workshop* (pp. 226–236). Available from: https://aclanthology.org/2020.wanlp-1.21

Clark, K., Luong, M.-T., Le, Q. V., & Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. *ArXiv Preprint ArXiv:2003.10555*.

Daniel Jurafsky, J. H. M. (n.d.). Speech and language processing. Available from: https://web.stanford.edu/~jurafsky/slp3/ (accessed 11 December 2021).

Darwish, K., Stefanov, P., Aupetit, M., & Nakov, P. (2020). Unsupervised user stance detection on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media* (Vol. 14, pp. 141–152). Available from: https://ojs.aaai.org/index.php/ICWSM/article/view/7286

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-Training of deep bidirectional transformers for language understanding, ArXiv:1810.04805 [Cs]. Available from: http://arxiv.org/abs/1810.04805

Ghosh, S., Singhania, P., Singh, S., Rudra, K., & Ghosh, S. (2019). Stance detection in web and social media: A comparative study, ArXiv:2007.05976 [Cs], *11696*, 75–87. doi: 10.1007/978-3-030-28577-7_4.

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., … Tao, D. (2021). A survey on vision transformer, ArXiv:2012.12556 [Cs]. Available from: http://arxiv.org/abs/2012.12556.

Imran, A. A., & Amin, M. N. (2021). Deep bangla authorship attribution using transformer models. In D. Mohaisen, & R. Jin (Eds.), *Computational Data and Social Networks* (pp. 118–128). Springer International Publishing. doi: 10.1007/978-3-030-91434-9_11.

Izsak, P., Berchansky, M., & Levy, O. (2021). How to train bert with an academic budget. *ArXiv Preprint ArXiv:2104.07705*.

Jannati, R., Mahendra, R., Wardhana, C. W., & Adriani, M. (2018). Stance classification towards political figures on blog writing. In *2018 International Conference on Asian Language Processing (IALP)* (pp. 96–101). doi: 10.1109/IALP.2018.8629144.

Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification, ArXiv:1607.01759 [Cs]. Available from: http://arxiv.org/abs/1607.01759

Kalyan, K. S., Rajasekharan, A., & Sangeetha, S. (2021). Ammus: A survey of transformer-based Pretrained models in natural language processing, ArXiv:2108.05542 [Cs]. Available from: http://arxiv.org/abs/2108.05542

Kayalvizhi, S., Thenmozhi, D., & Chandrabose, A. (2021). SSN_NLP@SardiStance: Stance detection from Italian tweets using RNN and transformers. In V. Basile, D. Croce, L. C. Passaro, & M. Maro (Eds.), *EVALITA Evaluation of NLP and Speech Tools for Italian—December 17th, 2020: Proceedings of the Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian Final Workshop* (pp. 220–223). Accademia University Press. Available from: http://books.openedition.org/aaccademia/7207

Kovacs, E.-R., Cotfas, L.-A., Delcea, C., & Florescu, M.-S. (2023). 1000 Days of COVID-19: A gender-based long-term investigation into attitudes with regards to vaccination. *IEEE Access*, *11*, 25351–25371.

Küçük, D., & Can, F. (2020). Stance detection: A survey. *ACM Computing Surveys*, *53*(1), 12:1–12:37. doi: 10.1145/3369026.

Lin, S.-X., Wu, B.-Y., Chou, T.-H., Lin, Y.-J., & Kao, H.-Y. (2020). Bidirectional perspective with topic information for stance detection. *2020 International Conference on Pervasive Artificial Intelligence (ICPAI)* (pp. 1–8). doi: 10.1109/ICPAI51961.2020.00009.

Liviu-Adrian, C., Delcea, C., Roxin, I., Ioanăş, C., Gherai, D. S., & Tajariol, F. (2021). The longest month: Analyzing COVID-19 vaccination opinions dynamics from tweets in the month following the first vaccine announcement. *Ieee Access*, *9*, 33203–33223.

Magnini, B., Lavelli, A., & Magnolini, S. (2020). Comparing machine learning and deep learning approaches on NLP tasks for the Italian language. *Proceedings of the 12th Language Resources and Evaluation Conference* (pp. 2110–2119). Available from: https://aclanthology.org/2020.lrec-1.259

Mohammad, S., Kiritchenko, S., Sobhani, P., Zhu, X., & Cherry, C. (2016). SemEval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)* (pp. 31–41). doi: 10.18653/v1/S16-1003.

Mohtarami, M., Baly, R., Glass, J., Nakov, P., Marquez, L., & Moschitti, A. (2018). Automatic stance detection using end-to-end memory networks. Available from: https://arxiv.org/abs/1804.07581v1

Müller, M., Salathé, M., & Kummervold, P. E. (2020). COVID-twitter-BERT: A natural language processing model to analyse COVID-19 content on twitter, ArXiv:2005.07503 [Cs]. Available from: http://arxiv.org/abs/2005.07503

Mubarak, H., Hassan, S., Chowdhury, S. A., & Alam, F. (2022). ArCovidVac: Analyzing Arabic tweets about COVID-19 vaccination. *ArXiv Preprint ArXiv:2201.06496*.

Padnekar, S. M., Kumar, G. S., & Deepak, P. (2020). BiLSTM-autoencoder architecture for stance prediction. In *2020 International Conference on Data Science and Engineering (ICDSE)* (pp. 1–5). doi: 10.1109/ICDSE50459.2020.9310133.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, *12*, 2825–2830.

pkudblab at SemEval-2016 Task 6: A Specific Convolutional Neural Network System for Effective Stance Detection (n.d). OpenAIRE - explore, Available from: https://explore.openaire.eu/search/publication?pid=10.18653%2Fv1%2Fs16-1062 (accessed 1 May 2022).

Rifat, M. R. I., & Imran, A. A. (2021). Incorporating transformer models for sentiment analysis and news classification in Khmer. In D. Mohaisen, & R. Jin (Eds.), *Computational Data and Social Networks* (pp. 106–117). Springer International Publishing. doi: 10.1007/978-3-030-91434-9_10.

Santosh, T. Y. S. S., Bansal, S., & Saha, A. (2019). Can Siamese Networks help in stance detection?. *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data* (pp. 306–309). doi: 10.1145/3297001.3297047.

Schiller, B., Daxenberger, J., & Gurevych, I. (2021). Stance detection benchmark: How robust is your stance detection?. *KI - Künstliche Intelligenz*. doi: 10.1007/s13218-021-00714-w.

Sobhani, P., Inkpen, D., & Zhu, X. (2017). A dataset for multi-target stance detection. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, (Vol. 2), Short Papers, 551–557. Available from: https://aclanthology.org/E17-2088

Tun, Y. M., & Hninn Myint, P. (2019). A two-phase approach for stance classification in twitter using name entity recognition and term frequency feature. In *2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS)* (pp. 77–81). doi: 10.1109/ICIS46139.2019.8940282.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems* (Vol. 30). Available from: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent trends in deep learning based natural language processing [review article]. *IEEE Computational Intelligence Magazine*, *13*(3), 55–75. doi: 10.1109/MCI.2018.2840738.

Corresponding author
Reema Khaled AlRowais can be contacted at: rkalrowais@ksu.edu.sa