Semi-supervised learning and bidirectional decoding for effective grammar correction in low-resource scenarios (#85217)

First submission

Guidance from your Editor

Please submit by **26 May 2023** for the benefit of the authors (and your token reward) .

Structure and Criteria
Please read the 'Structure and Criteria' page for general guidance.

Raw data check
Review the raw data.

Image check
Check that figures and images have not been inappropriately manipulated.

If this article is published your review will be made public. You can choose whether to sign your review. If uploading a PDF please remove any identifiable information (if you want to remain anonymous).

Files
Download and review all files from the materials page.

16 Figure file(s)
2 Latex file(s)
1 Other file(s)

Structure and Criteria

Structure your review

The review form is divided into 5 sections. Please consider these when composing your review:

1. **BASIC REPORTING**
2. **EXPERIMENTAL DESIGN**
3. **VALIDITY OF THE FINDINGS**
4. General comments
5. Confidential notes to the editor

You can also annotate this PDF and upload it as part of your review

When ready submit online.

Editorial Criteria

Use these criteria points to structure your review. The full detailed editorial criteria is on your guidance page.

BASIC REPORTING

- Clear, unambiguous, professional English language used throughout.
- Intro & background to show context. Literature well referenced & relevant.
- Structure conforms to PeerJ standards, discipline norm, or improved for clarity.
- Figures are relevant, high quality, well labelled & described.
- Raw data supplied (see PeerJ policy).

EXPERIMENTAL DESIGN

- Original primary research within Scope of the journal.
- Research question well defined, relevant & meaningful. It is stated how the research fills an identified knowledge gap.
- Rigorous investigation performed to a high technical & ethical standard.
- Methods described with sufficient detail & information to replicate.

VALIDITY OF THE FINDINGS

- Impact and novelty not assessed. *Meaningful* replication encouraged where rationale & benefit to literature is clearly stated.
- All underlying data have been provided; they are robust, statistically sound, & controlled.
- Conclusions are well stated, linked to original research question & limited to supporting results.

Standout reviewing tips

The best reviewers use these techniques

Tip	*Example*
Support criticisms with evidence from the text or from other sources	*Smith et al (J of Methodology, 2005, V3, pp 123) have shown that the analysis you use in Lines 241-250 is not the most appropriate for this situation. Please explain why you used this method.*
Give specific suggestions on how to improve the manuscript	*Your introduction needs more detail. I suggest that you improve the description at lines 57- 86 to provide more justification for your study (specifically, you should expand upon the knowledge gap being filled).*
Comment on language and grammar issues	*The English language should be improved to ensure that an international audience can clearly understand your text. Some examples where the language could be improved include lines 23, 77, 121, 128 – the current phrasing makes comprehension difficult. I suggest you have a colleague who is proficient in English and familiar with the subject matter review your manuscript, or contact a professional editing service.*
Organize by importance of the issues, and number your points	*1. Your most important issue* *2. The next most important item* *3. …* *4. The least important points*
Please provide constructive criticism, and avoid personal opinions	*I thank you for providing the raw data, however your supplemental files need more descriptive metadata identifiers to be useful to future readers. Although your results are compelling, the data analysis should be improved in the following ways: AA, BB, CC*
Comment on strengths (as well as weaknesses) of the manuscript	*I commend the authors for their extensive data set, compiled over many years of detailed fieldwork. In addition, the manuscript is clearly written in professional, unambiguous language. If there is a weakness, it is in the statistical analysis (as I have noted above) which should be improved upon before Acceptance.*

Semi-supervised learning and bidirectional decoding for effective grammar correction in low-resource scenarios

Zeinab Mahmoud [1], **Chunlin Li** [1], **Marco Zappatore** [2], **Aiman Osman** [Corresp., 3], **Ali Alfatemi** [4], **Ashraf Osman Ibrahim** [5], **Abdelzahir Abdelmaboud** [6]

[1] School of Computer Science, Wuhan University of Technology, Wuhan, Hubei, China
[2] Department of Engineering for Innovation, University of Salento, Lecce, Lecce, Italy
[3] School of Software Engineering, South China University of Technology, Guangzhou, China
[4] Computer Science, Graduate School of Arts and Sciences (GSAS), Fordham University, New York, United States
[5] Advanced Machine Intelligence Research Group, Universiti Malaysia Sabah, Kota Kinabalu 88400, Malaysia
[6] Department of Information Systems, King Khaled University, Muhayel Aseer, Saudi Arabia

Corresponding Author: Aiman Osman
Email address: seaiman@mail.scut.edu.cn

The correction of grammatical errors in natural language processing is a crucial task as it aims to enhance the accuracy of written language. However, developing a grammatical error correction (GEC) framework for low-resource languages presents significant challenges due to the lack of available training data. This paper proposes a novel GEC framework for low-resource languages, using Arabic as a case study. To generate more training data, we propose a semi-supervised confusion method called the Equal Distribution of Synthetic Errors (EDSE), which generates a wide range of parallel training data. The EDSE method generates a wide range of parallel training data. Additionally, this paper addresses two limitations of the classical seq2seq GEC model, which are unbalanced outputs due to the unidirectional decoder and exposure bias during inference. To overcome these limitations, we apply a Knowledge Distillation technique from neural machine translation. This method utilizes two decoders, a forward decoder right-to-left and a backward decoder left-to-right, and measures their agreement using Kullback-Leibler divergence as a regularization term. The experimental results on two benchmarks demonstrate that the proposed framework outperforms the Transformer baseline and two popular bidirectional decoding techniques. Furthermore, the proposed framework reported the highest F1 score, and generating synthetic data using the equal distribution technique for syntactic errors resulted in a significant improvement in performance. These findings demonstrate the effectiveness of the proposed framework for improving grammatical error correction for low-resource languages, particularly for the Arabic language.

Semi-supervised Learning and Bidirectional Decoding for Effective Grammar Correction in Low-Resource Scenarios

immediate

ABSTRACT

The correction of grammatical errors in natural language processing is a crucial task as it aims to enhance the accuracy of written language. However, developing a grammatical error correction (GEC) framework for low-resource languages presents significant challenges due to the lack of available training data. This paper proposes a novel GEC framework for low-resource languages, using Arabic as a case study. To generate more training data, we propose a semi-supervised confusion method called the Equal Distribution of Synthetic Errors (EDSE), which generates a wide range of parallel training data. The EDSE method generates a wide range of parallel training data. Additionally, this paper addresses two limitations of the classical seq2seq GEC model, which are unbalanced outputs due to the unidirectional decoder and exposure bias during inference. To overcome these limitations, we apply a Knowledge Distillation technique from neural machine translation. This method utilizes two decoders, a forward decoder right-to-left and a backward decoder left-to-right, and measures their agreement using Kullback-Leibler divergence as a regularization term. The experimental results on two benchmarks demonstrate that the proposed framework outperforms the Transformer baseline and two popular bidirectional decoding techniques. Furthermore, the proposed framework reported the highest F1 score, and generating synthetic data using the equal distribution technique for syntactic errors resulted in a significant improvement in performance. These findings demonstrate the effectiveness of the proposed framework for improving grammatical error correction for low-resource languages, particularly for the Arabic language.

INTRODUCTION

Automatic correction of grammatical errors is one of the most common NLP tasks in research and industry and it has seen rapid development with the advancement of deep learning techniques. Recent deep neural network approaches are essentially an encoder-decoder architecture Solyman et al. (2022). In GEC neural-based systems, the encoder receives the source, which is an ungrammatical sentence and maps it into an intermediate hidden vector that encodes all the source information. The decoder takes the hidden vector to generate the output correction word by word.

The major challenge of GEC is that required massive parallel training data are not available for languages such as Slovenian, Albanian, and Arabic language (the so-called low resource languages). The classical form of seq2seq GEC often uses a unidirectional decoder that suffers from unbalanced outputs Solyman et al. (2022), which leads the system to generate corrections with good prefixes and bad suffixes. The effects of this problem vary depending on the model structure and the length of the input sequence. However, the autoregressive structure of deep neural network approaches in GEC has a limitation during inference when the previous target word is unavailable; consequently, the model depends on itself and generates a new word that may be out of context, thus generating the so-called exposure bias problem Solyman et al. (2022). The incorrect words generated during inference lead to weakness in the prediction of the next word and result in unsatisfactory correction results. Previous studies such as Yuan et al. (2019) sought to use a complementary decoder (R2L) to rerank the n-best list of the L2R decoder, but still the same decoder suffers from an exposure bias problem which leads to bad prefixes corrections.

The current research direction is aimed at lessening the discrepancy that exists between the training and inference stages to increase robustness while feeding erroneous previous predictions to overcome this issue. For instance, a Type-Driven Multi-Turn Corrections approach was proposed by He et al. (2016), which involves constructing multiple training instances from each original instance during training. Zhang et al. (2018) proposed a two-stage decoding neural translation model in the inference, that is time-consuming. Another notable work in Zhang et al. (2019), proposed a regularization method during training to increase the agreement between two decoders (L2R and R2L); however, it complicates the training phase because of dynamic sampling and requires more training time and computation resources. To tackle the drawback associated with previous studies, the current work introduces a semi-supervised confusion method that widens synthetic training data. Furthermore, an Arabic grammatical error correction (AraGEC) model was proposed, based on bidirectional knowledge distillation with a regularization method inspired by NMT, as proposed by Zhang et al. (2022), which aims to improve the agreement between the two decoders of forward (R2L) and backward (L2R) into a joint framework. This forces both decoders to act as helper systems for each other and to integrate their advantages to address the exposure bias problem and generate corrections as output with good prefixes and suffixes. The notable outcomes of this work are outlined below:

- A semi-supervised method is proposed to overcome the shortage of parallel training data in AraGEC by generating synthetic training data.
- AraGEC model is proposed based on Transformer-base equipped with a bidirectional knowledge distillation method to address the exposure bias problem typically experienced in automatic GEC systems.
- Experimental results on two benchmarks demonstrate that our model outperforms the current most powerful bidirectional decoding methods as well as previous AraGEC systems.

This paper is structured as follows. Section 2 describes the related works. The proposed confusion method and the GEC framework are presented in Section 3. Section 4 examines the experimental details, whereas Section 5 reports our evaluation results and analysis. Finally, conclusions are given in Section 6. The code, trained models, and data files are available online[1].

RELATED WORK

Automatic detection and correction of grammatical and other related errors are one of the most popular tasks in NLP, as the interest in it began in the late 1970s with the advent of electronic computing. Rule-based systems were the earliest applications adopted to that end, which use a simple knowledge base that contained all the grammar rules of the relevant language Simmons (1978). In the 1990s, there was a significant development in the field of computational linguistics that led to the use of n-gram language models to measure the probability of characters and words in a contiguous sequence from a given sample of text Brown et al. (1992). Recently, GEC can be considered a machine translation task, which translates text with errors (interpreted as the source language) into error-free text. GEC-based SMT is a phrase-based system that optimizes the conditional probability of finding the correct sentence Y given the input sentence X, among all possible corrections Junczys-Dowmunt and Grundkiewicz (2016). Due to the increases in computer processing capabilities and the availability of massive training data, GEC-based NMT systems demonstrated the ability to outperform the previous and more traditional GEC techniques thanks to their new approach that allows to correct texts using a set of hidden layers in the form of seq2seq models such as Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), or Transformer Solyman et al. (2022).

English and Chinese languages have received so far research attention thanks to extensive resources that include parallel text corpora usable as training data, pre-trained models, and open access GEC systems. For instance, GPT-3 is a per-trained language model with 175 billion machine learning parameters focused on generating natural human language text and achieved significant performance in English GEC Brown et al. (2020). Google AI introduced a mega language model named Pathways that has a capacity of 540 billion parameters and achieved human-like performance in multi-NLP tasks including GEC Chowdhery et al. (2022).

[1] https://github.com/Zainabobied/SLBDEGC

However, the main challenge of low-resource languages[2] such as Italian, French, and Arabic is the lack of such resources. Ge et al. (2018) propose to correct texts in an iterative routing process named *fluency boost learning* based on CNN, and achieved a remarkable improvement in the accuracy and fluency of GEC systems. Acheampong and Tian (2021) introduced a notable GEC system based on cascading learning strategies that reduced the need for massive training data for neural-based GEC systems. Wan et al. (2020) proposed the only work in GEC that used data augmentation to increase the diversity of training examples by editing the latent representations of grammatical sentences. Grundkiewicz et al. (2019) employed a spell-checker to synthesize parallel training data from an out-of-domain monolingual corpus used to train a multi-head attention network. Zhao et al. (2019) suggested a copy-augmented approach for Transformer-based Indonesian GEC systems. This method enhances accuracy by incorporating correct or unaltered words from the source text into the target text. Sun et al. (2022) proposed a generic and language-independent strategy for multilingual GEC systems that can be used for other low-resource languages benefiting from available resources (e.g., parallel translation data between English and the other language, and pre-trained cross-lingual language models). Hagiwara and Mita (2020) introduced GitHub Typo Corpus, a large-scale multilingual GEC training data for 15 languages. Náplava and Straka (2019) introduced a synthetic multilingual GEC training data used to train Transformer, which achieved significant improvements in Czech, German, and Russian.

AraGEC is receiving more attention after successfully shared tasks in 2014 and 2015 Mohit et al. (2014); Rozovskaya et al. (2015). Despite the early attention; however, AraGEC suffers from a lack of training data, since the only annotated Arabic training data consist of 20430 examples. Rozovskaya et al. (2014), introduced a hybrid AraGEC system made of rule-based and machine-learning approaches. Nawar (2015) proposed a GEC system that utilized word patterns and rule-based statistics to detect and correct grammatical errors. Sina (2017) employed seq2seq RNN and the attention mechanism in AraGEC. Madi and Al-Khalifa (2020) employed LSTM, BiLSTM, and SimpleRNN baselines used to detect errors, that outperform the commercial Arabic Grammar Checker (Microsoft Word 2007), and also introduced their own training data. Watson et al. (2018) utilize seq2seq Bidirectional recurrent neural networks (BRNN) and FasTest word embedding to obtain more linguistic information in GEC. Solyman et al. (2019) proposed a convolutional AraGEC model, which was extended in Solyman et al. (2021), a GEC framework comprising a classical confusion method and CNN seq2seq model equipped with an attention mechanism. Pajak and Pajak (2022) tuned a set of pre-trained multilingual models such as mBART, mT5, or xProphetNet for GEC in seven different languages, including Arabic and reported encouraging results.

As it can be inferred from the in-domain literature overview provided so far, the existing systems for low-resource scenarios predominantly use spell-confusion methods to generate synthetic data that almost lacks diversity, thus leading to limited training patterns and, consequently, limiting significantly the true application potential of those systems. Therefore, an extended effort is needed to introduce more efficient approaches capable of addressing the lack of training data and the exposure bias problem.

METHODOLOGY

System Overview

In this section, we introduce the proposed GEC framework in detail, formulate the hypotheses, and strive to avoid ambiguity. Initially, a novel approach was proposed to construct reliable synthetic parallel training data for GEC. Furthermore, we introduce a knowledge distillation with bidirectional decoding for AraGEC based on Transformer. This technique was proposed by Zhang et al. (2022) in NMT, and we have successfully integrated it into our model.

Noise method

Despite the widespread use of Arabic on the Internet, there is still a lack of freely available training data for NLP applications such as semantic analysis Baghdadi et al. (2022), text classi-

[2]low-resource languages in the NLP are those that have insufficient data available for training automatic GEC systems.

fications Masri and Al-Jabi (2023), and automatic grammar correction. Qatar Arabic Language Bank[3] (QALB) is the only available annotated parallel data for GEC: it consists of 20430 examples, which is not enough to train GEC neural-based systems effectively. Furthermore, building extensive parallel training data for GEC is expensive, time-consuming, and requires appropriate tools. To this end, numerous methods have been proposed such as back-translation Kiyono et al. (2020) and misspelling confusion sets Grundkiewicz et al. (2019) to overcome the lack of training data. However, these techniques are unreliable to construct high-quality training data containing the most common grammatical errors (training patterns) and cannot control the types of errors, rate, and distribution. Therefore, one of the main challenges in this study is to build a massive synthetic training set that contains most types of errors in the Arabic language.

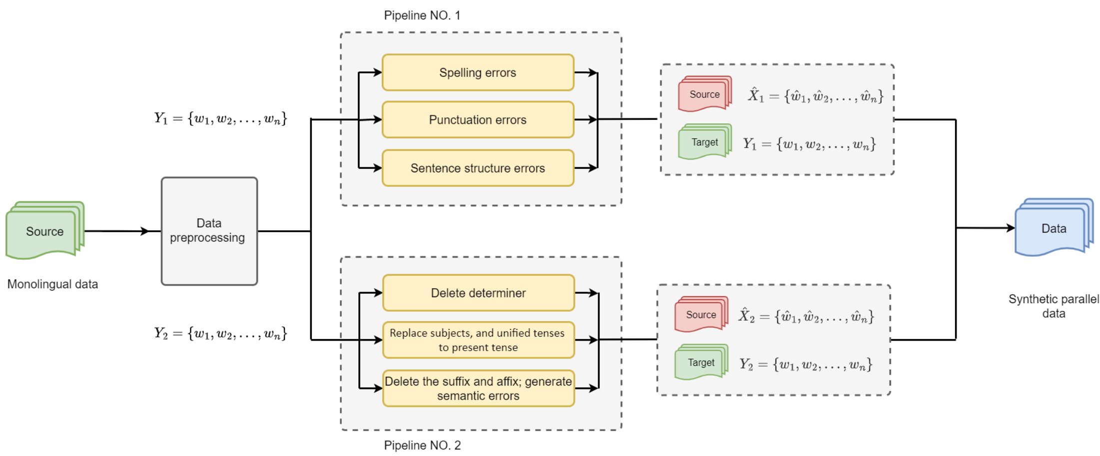


Figure 1. Architecture of the Equal Distribution of Synthetic Errors (EDSE) approach is made of two synthetic pipelines that have the same probability of error generations, green refers to the original data, red is the synthetic data (erroneous), and blue is the parallel training data.

The seed of our synthetic data was a monolingual corpus namely CC100-Arabic, created by Conneau et al. (2020) from Facebook AI. The data was collected during January-December 2018 Commoncrawl snapshots from the CC-Net repository, and the total data size was 5.4 GB organized into a single text document. The CC100 arabic corpus was selected because it is freely available and it is the most extensive monolingual Arabic corpus. In addition, it contains various topics such as education, history, economy, law, health, stories, cooking recipes, and sport. Besides, it is well-formatted and free from grammatical errors and dialectal words. Several steps of data prepossessing were initially applied over the given corpus, such as removing the duplicate paragraphs and spaces between lines. We decided to use 25 million examples in different lengths, between 10 to 100 words. Then, data was normalized from diacritical marks, non-UTF8 encoding, links, and mentions, and we kept punctuation, numbers, and Arabic stop words.

Recently, the performance of GEC systems was improved thanks to monolingual data, which was used during training to provide more training patterns; this depends on the size and quality of the synthetic data Grundkiewicz et al. (2019). This paper proposes a semi-supervised method for generating massive synthetic data that contains most types of grammatical errors in Arabic. In order to cover all types of errors in AraGEC, two pipelines were applied; hence the type of errors was grouped into two groups: group one includes spelling errors, sentence structure, and punctuation errors; while group two includes syntax and semantic errors. This makes it easy to control the rate and distribution of each type of error.

The proposed method has two key parameters: N refers to the number of words to be processed and has initial value between 0 and 1, we set the value of N during training to 0.1; T is the total number of words in each input sentence. Let us begin with pipeline number one: generating spelling errors starts by tokenizing the input sentence and then we choose a

[3] http://nlp.qatar.cmu.edu/qalb/

random word to delete a character or add more characters. Furthermore, injects punctuation errors from a given list or removes existing punctuation. To cause sentence structure errors, we transform the input sentence into a PoS tagging format, followed by one of two operations: (1) swapping two of the sentence components such as subject, object, or verb; (2) removing one of the sentence structures.

The second pipeline of the proposed method contains the most complex error types such as syntactic and semantic errors. Initially, each input sentence was transformed into PoS format and followed by one of the listed operations: (1) delete a determiner; (2) replace the subject with another word from the corpus vocabulary to cause a verb-subject disagreement; (3) use the PoS tags to unify the tense in the present tense format and ignore the future and past tense to cause tense verb errors; (4) delete the suffix and affix to cause a morphological error and inconsistency in the sentence; (5) replacing a random word in the sentence with a word from the data to causes a semantic error, which confuses the reader and affects the sentence context. The proposed method is named Equal Distribution of Syntactic Errors (EDSE), Figure 1 shows the architecture of EDSE.

Bidirectional decoding

The proposed AraGEC framework uses forward and backward decoders in the decoding structure. The decoder that moves in a forward direction utilizes a mask matrix that is in the form of an upper triangular shape, which sees the information on the right of y_t, and named it R2L decoder. The backward decoder in the regular language model perceives the sequences from left to right and is named the L2R decoder. Furthermore, a lower triangular mask matrix was used in the L2R decoder. Both given decoders are utilized to detect and correct the next token from $(t+1$ to T) or (1 to $t-1)$ given the source X and the target Y as the following equations.

$$logP(y|\mathbf{X};\overleftarrow{\theta}) = \prod_{n=1}^{N} P(y_t|y_{t+1:T},\mathbf{X};\overleftarrow{\theta}), \tag{1}$$

$$logP(y|\mathbf{X};\overrightarrow{\theta}) = \prod_{n=1}^{N} P(y_t|y_{1:t-1},\mathbf{X};\overrightarrow{\theta}), \tag{2}$$

The literature of the previous work in AraGEC demonstrates that the R2L performs better than the L2R decoder as described by Solyman et al. (2022); hence, in this work the backward decoder (R2L) will be the student and the forward decoder (L2R) represent the teacher. R2L decoder learns dependencies of the output sequences from right to left, whereas the L2R learns the dependencies of the output sequences from left to right, and this is the relative future information of the R2L. Thereon, the output of both decoders (R2L - L2R) which is the probability destitution of words in each position that can be represented as complementary information of two decoding sides. This makes the model force the probability distribution of P_{R2L} and P_{L2R} to support each other during training to generate future information, as shown in the following equation.

$$P_{R2L}(y_t = w|y_{1:t-1},\mathbf{X}) \sim P_{L2R}(y_t = w|y_{t+1:T},\mathbf{X}) \tag{3}$$

where w is the given token from the training vocabulary, and t refers the t_{th} position of the output corrected sequence. However, these decoders cannot improve equally and cannot fulfill Equation (4) if optimized separately using the standard MLE. The L2R decoder cannot learn the global coherence from R2L and this will lead to unsatisfactory corrections. To this end, a Knowledge Distillation method was proposed to improve both decoders during training process and the transferred information learning across R2L and L2R decoders. Furthermore, the L2R decoder will not be used during inference so as to not affect decoding speed as compared to the conventional GEC models that used the L2R model during inference.

Knowledge Distillation

The main objective of the proposed knowledge Distillation method is to incorporate the learning information from the backward decoder to the forward decoder, which uses L2R decoder as a teacher that has future knowledge (hidden states) of R2L decoder. This approach utilizes the logits and the teacher's final layer hidden states model for increased versatility and effectiveness. Furthermore, since the student and teacher models will learn during training at the same time, so we called this method Bidirectional Knowledge Distillation Grammatical Error Correction (BKDGEC), which encompasses hidden state-based distillation and logit-based as depicted in Figure 2.

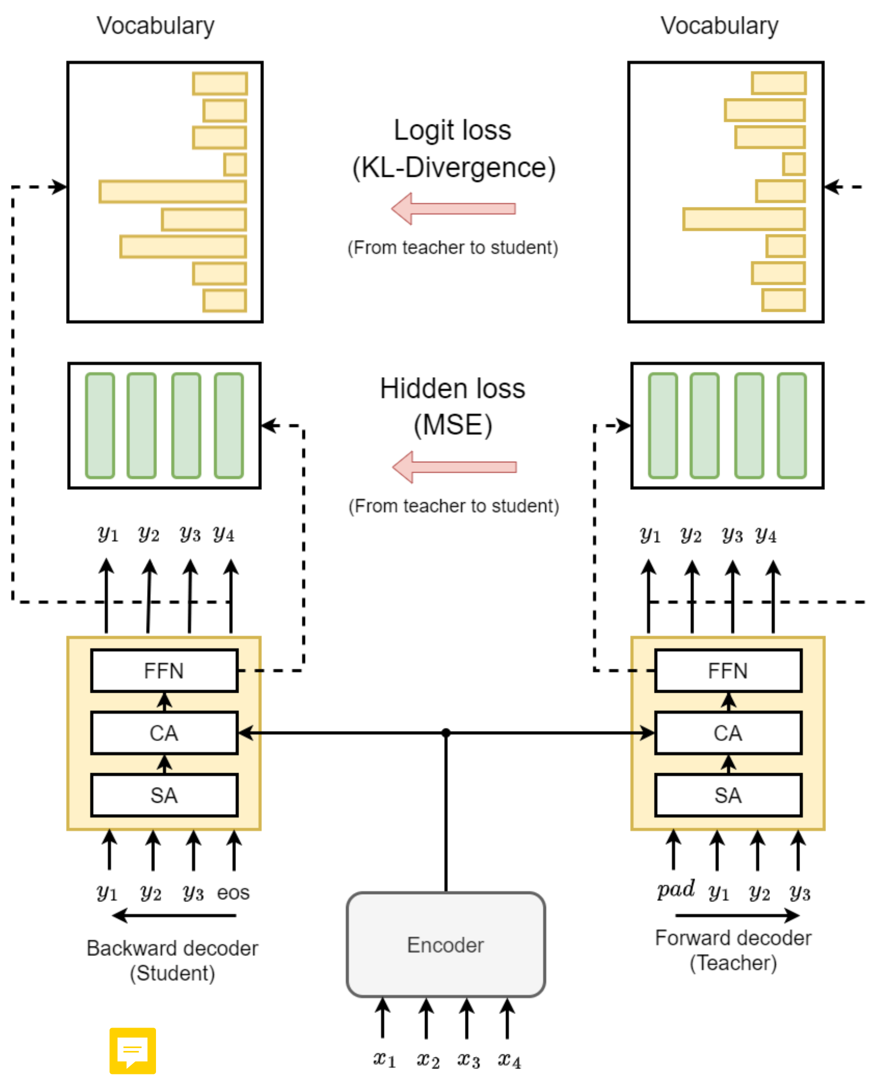


Figure 2. The design of our BKDGEC model incorporates two decoders, labeled as Backward and Forward, represented by yellow boxes. These decoders consist of Self-Attention (SA), Cross-Attention (CA), and a Feed-Forward Neural Network (FFN).

In the realm of neural-based techniques, Logit alludes to the predictive vector that can be produced using the last layer of the decoder. This layer has the same dimension as the vocabulary size and is employed to determine the token that should be predicted in the present time step. In this work, Kullback-Leibler (KL) Joyce (2011) was utilized to quantify the divergence between the logit probability distributions of the backward and forward decoders at the same position.

Equations (4) and (5) demonstrate the implementation of this method:

$$L_{logit} = \sum_{n=1}^{T} KL(P(y_t|y_{1:t-1}, \mathbf{X}\overrightarrow{\theta})||P(y_t|y_{t+1:T}, \mathbf{X}; \overleftarrow{\theta})), \tag{4}$$

$$KL(P(y_t|y_{1:t-1}, \mathbf{X}\overrightarrow{\theta})||P(y_t|y_{t+1:T}, \mathbf{X}; \overleftarrow{\theta})) = \sum_{w \in V} P(y_t = w|y_{1:t-1}, \mathbf{X}; \overrightarrow{\theta})) \times \mathbf{log} \frac{\mathbf{P}(\mathbf{y_t} = \mathbf{w}|\mathbf{y_{1:t-1}}, \mathbf{X}; \overrightarrow{\theta})}{\mathbf{P}(\mathbf{y_t} = \mathbf{w}|\mathbf{y_{t+1:T}}, \mathbf{X}; \overleftarrow{\theta})}, \tag{5}$$

Here, V represents the output vocabulary, and T denotes the target length. Consequently, this led to the distillation of hidden states, which can be depicted through the following equation.

$$L_{hd} = MSE(\overleftarrow{HW_h}, \overrightarrow{H}) \tag{6}$$

where MSE is a loss function stands to mean squared error, $\overleftarrow{H} \in R^{l \times \acute{d}}$ and $\overrightarrow{H} \in R^{l \times d}$ refers to the hidden states of the both decoders R2L and L2R, respectively. Furthermore, $W_h \in R^{\acute{d} \times d}$ is a linear function that adjusts the L2R hidden states to have the same dimension as the R2L hidden states, and $\acute{d}, d$ are the hidden dimension of both the decoders and have the same value. In this work, two knowledge distillation functions were utilized to encourage the backward decoder to grasp future representations. In addition, a joint training framework was constructed to optimize both the decoders iteratively, as shown in Equation (7).

$$L(\theta) = \sum -logP(\overrightarrow{y}|X, \overrightarrow{\theta}) - logP(\overleftarrow{y}|X, \overleftarrow{\theta}) + L_{kd}(\overrightarrow{y}, \overleftarrow{y}), \tag{7}$$

$$L_{kd} = L_{logit} + L_{hd}, \tag{8}$$

As explained, the knowledge distillation learning process in this work is based on a student model imitating the teacher model. This might raise concerns as the student's potential might be constrained by the teacher's performance, resulting in limited ability to surpass the teacher Clark et al. (2019). Consequently, the student model could rely heavily or excessively on the teacher model. To tackle this challenge in our BKDGEC framework, we applied two distillation methods. These methods help the R2L decoder gain a better understanding of future knowledge and drive the model to place more emphasis on the L2R decoder as training progresses. To this end, an annealing mechanism was proposed that is fitting for BKDGEC. It adjusts the training objective to consider the agreement between both decoders as in Equation (9).

$$L(\theta) = \sum_{i=1}^{n} \left[-(1-\lambda) \cdot \left(\log P_{\overleftarrow{\theta}}(y_i|X_i) \right)^2 - \lambda \cdot \log P_{\overrightarrow{\theta}}(y_i|X_i) + (1-\lambda)\lambda \cdot L_{kd}(y_i, \hat{y}_i) \right], \tag{9}$$

where $\lambda \in [0,1]$ is a hyperparameter that controls the balance between the forward decoder $P_{\overrightarrow{\theta}}$ and the backward decoder $P_{\overleftarrow{\theta}}$. Here, y_i is the ground truth label for the i-th input sample X_i, and $\hat{y}_i$ is the output label from the forward decoder. The value of λ is determined based on the current training step c_{step} and the warm start step w_{step}. Specifically, if $c_{step} < w_{step}$, then $\lambda = 1$, and the training objective function only considers the output of the forward decoder $P_{\overrightarrow{\theta}}$ to help the backward decoder $P_{\overleftarrow{\theta}}$ acquire sufficient knowledge. Otherwise, $\lambda = \frac{w_{step}}{c_{step}}$, indicating that the number of training steps is greater than w_{step}. In this case, the effect of the backward decoder $P_{\overleftarrow{\theta}}$ (also known as the teacher) increases, and the initial value of the divergence in agreement $L_{kd}(y_i, \hat{y}_i)$ also increases during training, while the output of the forward decoder $P_{\overrightarrow{\theta}}$ (also known as the student) decreases over time.

EXPERIMENTS

Data

The seed of the synthetic parallel training data was CC100-Arabic created by Conneau et al. (2020) from Facebook AI. After data preprocessing, we applied our confusion method EDSE presented in Section to generate parallel training data subdivided into train and development sets. The authentic data QALB-2014 was utilized for fine-tuning, which is the only AraGEC that contains 20430 examples. The data was collected from English articles translated into Arabic, and Arabic Learners Written Corpus (CERCLL) Alfaifi et al. (2014). Furthermore, the users comments on the Aljazeera news platform, which contains most of the possible grammatical errors because the writers were from different perspectives and different countries (different Arabic dialects). The data was corrected and double-checked twice by a team of ten native speakers and linguistic experts.

Model setting

The baseline was Transformer-based, which has been modified during experiments according to the primary results Vaswani et al. (2017). The model size and batch size were reduced from 512 to 256, and 2048 to 128, respectively. The original values achieved poor results because our proposed model used chunks of 2-to-4 characters instead of words. The number of layers was reduced during experiments from six to four, whereas the number of heads attention was kept to eight as the original. In the same context, the first layers in the encoder and decoder were used for positional encoding instead of the static encoding as in BERT Devlin et al. (2019); however, the label smoothing was not applied. Instead of warm-up and cool-down steps learning rate, we applied Adam optimizer Kingma and Ba (2015) to address over-fitting with a value of 0.003 during training and 0.001 in the fine-tuning. To avoid exceeding the gradient, a gradient clipping was applied with a value of 1.0, and dropout was applied with probabilities of 0.15 and 0.10 for training and fine-tuning, respectively. The algorithm of Byte Pair Encoding (BPE) was utilized to split unknown tokens into sub-tokens that addressed the challenge of the rare words Sennrich et al. (2016).

Early stop was applied during training, which led to 27 epochs using the monolingual parallel synthetic data and three epochs for fine-tuning using a monolingual parallel authentic of QALB-2024. A checkpoint of the best model was created after each epoch. Due to the small chunks of input sequences, the maximum length of input sequences was set to 400 tokens in training and testing. The tokenizer was the BPE algorithm with 1000 vocabulary size. Beam search was applied during inference with a five-beam size. The outputs of the test set have been tuned after inference using a simple data preprocessing method to remove the repetitions of words, characters, and some punctuation errors that the model failed to correct well.

Evaluation

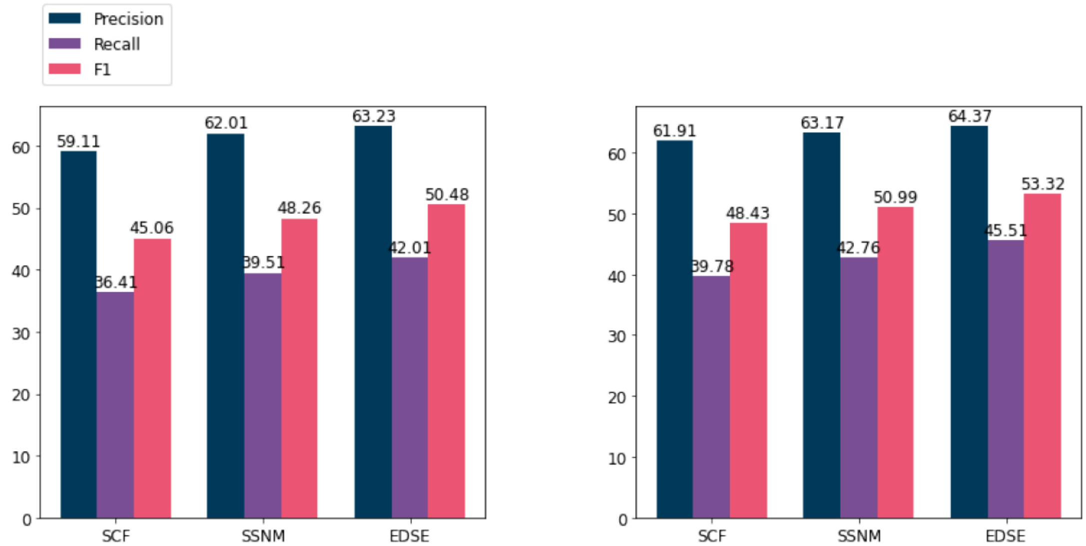
The proposed framework was evaluated on two benchmarks as same as in the second Arabic automatic grammar correction shared task Rozovskaya et al. (2015). MaxMatch Dahlmeier and Ng (2012) was applied to evaluate the performance using the same tool in the same shared task Rozovskaya et al. (2015) to measure the word-level edits in the output compared to the golden target sentences, and reported precision, recall, and F_1 score using different scenarios during training. In addition, BLEU-4 score was applied to evaluate the quality of the machine correction compared to high-quality human-corrected sentences.

RESULTS

This section investigates the performance of the proposed framework, including the impact of the synthetic data, the bidirectional knowledge distillation method, as well as fine-tuning and re-ranking L2R as an improvement. We also investigated the performance against the most powerful bidirectional methods in NMT: asynchronous and synchronous decoding.

Impact of synthetic data

The constructed synthetic data have more diverse training examples that have been used to train different versions of our BKDGEC models. Table 1 shows the effectiveness of our EDSE method



(a) QALB-2014 (b) QALB-2015

Figure 3. Illustration performance of EDSE compared to the classical SSMN and SCF approaches using precision, recall, F_1 using (a) QALB-2014 and (b) QALB-2015.

to construct more reliable data compared to previous approaches such as a semi-supervised confusion function (SCF) Solyman et al. (2021) and a simple spelling noise method (SSNM) Solyman et al. (2022) using the same data size consisting of 250 k examples. The three synthetic sets have been used to train the baseline Transformer- base without fine-tuning, BPE was applied with a vocabulary of 30k to reduce the confusion caused by unknown words during training. EDSE performed better than SCF and SSNM in the benchmark QALB-2014 as illustrated in Figure 3. This highlights the importance of multi-training patterns in the training data, in which SCF contains only spelling errors, while SSNM has more training patterns but is still limited compared to our synthetic data.

	QALB-2014			QALB-2015		
Training data	Prec.	Recall	F_1	Prec.	Recall	F_1
SCF	59.11	36.41	45.06	61.91	39.78	48.43
SSNM	62.01	39.51	48.26	63.17	42.76	50.99
EDSE	**63.23**	**42.01**	**50.48**	**64.37**	**45.51**	**53.32**

Table 1. Performance of asynchronous and synchronous decoding in AraGEC using the same baseline (Transformer) compared to BKDGEC.

Eventually, the performance was investigated using the full systematic data for training the model. Table 2 shows that F1 score increased + 18.71 and +3.06 for QALB-2014 and QALB-2015, respectively. This emphasizes the importance and ability of producing synthetic data to raise the level and effectiveness of the GEC systems during training, and also the impact of 10k vocabulary of the BPE algorithm.

Impact of Bid-knowledge distillation

The performance of different versions of the proposed GEC framework, utilizing two benchmarks, is presented in Table 2. The baseline model was a Transformer-based approach trained on the QALB-2014 authentic corpus, with slight modifications. The results demonstrate that the proposed BKDGEC regularization technique can significantly enhance the framework's performance, as indicated by the F_1 scores of 0.62 and 0.85 for QALB-2014 and QALB-2015, respectively. Notably, the bid-knowledge distillation approach proved to be particularly effective in improving the framework's performance, highlighting the backward decoder's ability to predict the forward decoder's concurrent states accurately. These findings have significant implications for the development of more effective GEC frameworks.

Impact of fine-tuning

BKDGEC has been carefully fine-tuned to improve its accuracy and performance. This fine-tuning process involved using the original parallel corpus of QALB-2014 and a monolingual dataset called CC-100[4], consisting of 1k clean sentences. The results of this process were presented in Table 2, which achieved the best results among all models with an F1 score of 70.29% for QALB-2014 and 73.13% for QALB-2015. The impact of fine-tuning on both datasets was remarkable, as demonstrated by the significant improvement in the model's accuracy. Notably, the parallel corpus yielded better results, likely due to the inclusion of additional authentic examples. These findings highlight the importance of using high-quality datasets for fine-tuning language models, as it can have a significant impact on GEC performance.

Re-ranking n-best list

We applied re-ranking from NMT to enhance the performance after inference, which achieved significant improvement Liu et al. (2016). Initially, three different models were trained on both sides (R2L and L2R) using BKDGEC method from scratch which utilized the synthetic data for training and which was tuned using QALB-2014. This enriches the hypothesis list which contains three different n-best lists with the corresponding scores of the R2L and L2R models. Each n-best list of the L2R models is passed to each R2L model to integrate both lists into a union relation resulting from the summation of the scores and reordered to obtain the k-best list, which is the final output. This notably improves the precision and F1 score, as shown in Table 2, which increases the F1 by 1.22 and 0.90 in QALB-2014 and QALB-2015, respectively. The impact of joint search in the n-best lists R2L and L2R led the system to improve the accuracy of prefixes and suffixes in the output.

	QALB-2014			QALB-2015		
Model	Prec.	Recall	F_1	Prec.	Recall	F_1
Transformer (Baseline)	75.61	55.82	64.22	74.78	60.86	67.10
Transformer + EDSE data	77.14	62.73	69.19	75.36	67.53	71.23
Transformer + EDSE data + BKDGEC	77.91	63.11	69.73	76.17	68.42	72.08
Transformer + EDSE data + BKDGEC + Fine-tuning	78.12	63.90	70.29	76.89	69.73	73.13
Transformer + EDSE data + BKDGEC + Fine-tuning + L2R re-ranking	**78.61**	**65.59**	**71.51**	**78.21**	**70.28**	**74.03**

Table 2. Comparisons of precision, recall, and F_1 of the baseline, with EDSE data, bidirectional knowledge distillation method (BKDGEC), fine-tuning, as well as L2R re-ranking.

Bidirectional decoding optimization

This subsection investigates the impact of the most common NMT bidirectional decoding techniques in GEC compared to bidirectional knowledge distillation.

[4]https://data.statmt.org/cc-100/

Asynchronous bidirectional decoding

Zhang et al. (2018) proposed an asynchronous bidirectional decoding method that employs a standard encoder-decoder along with a backward decoder. In this work, the existing L2R decoder was used as a backward decoder and the R2L decoder as a forward decoder. R2L decoder generates the correction from right to left, considering the bidirectional source and reversed hidden states of the backward decoder to improve the correction accuracy. Asynchronous bidirectional decoding achieved F_1 scores of 68.83 and 71.14 for QALB-2014 and QALB-2015, respectively, as shown in Table 3.

Synchronous bidirectional decoding

To circumvent the limitation of bidirectional decoding, Zhou et al. (2019) proposed to integrate the R2L and L2R decoders into a synchronous and bidirectional framework instead of performing independent bidirectional decoding. The same technique has been applied, which used a single decoder to generate the output correction R2L and L2R in an interactive and simultaneous decoding process. The simultaneous decoding achieved 69.22 and 71.56 F1 scores in the QALB-2014 and QALB-2015, respectively, as shown in Table 3. This technique allows the GEC framework to take advantage of the history (backward decoding) and future (backward decoding) information into an interactive decoding process that uses R2L and L2R at the same time.

	QALB-2014			QALB-2015		
Model	Prec.	Recall	F_1	Prec.	Recall	F_1
Asynchronous bidirectional decoding	77.34	62.02	68.83	75.61	67.18	71.14
Synchronous bidirectional decoding	77.59	62.48	69.22	75.67	67.89	71.56
BKDGEC	**77.91**	**63.11**	**69.73**	**76.17**	**68.42**	**72.08**

Table 3. Performance of asynchronous and synchronous decoding in AraGEC using the same baseline (Transformer) compared to BKDGEC.

Bidirectional knowledge distillation differs from the above methods, allowing the system to utilize richer target-side contexts for corrections. This occurs when L2R target-side context and R2L corrections are integrated into an end-to-end joint framework and take the agreement between decoders as a regulation term. Hence, it will much alleviate the error propagation of the reverse target-side context. In summary, Table 3 shows that our bid-knowledge distillation without fine-tuning and re-ranking achieved the best improvement over both methods.

BLEU score

In this subsection, we assess the performance of the proposed framework using the BLEU score to compare the quality of its outputs to the reference or golden sentences, as well as to the baseline performance (Transformer-based), which is an extra human evaluation. Initially, source sentences of the benchmark QALB-2015 were grouped into eight different lengths, also different settings have been used including n-grans with n from 1 to 4.

Table 4 shows that the proposed model achieved the highest scores in different lengths compared to the baseline trained using the same dataset and hyperparameters. The performance of both models gradually increased with the sentence length, and BLEU score settings changed, with our model being superior as shown in Figure 4. Once again, this demonstrates the efficiency of the BKDGEC for low-resource GEC systems, which leads to overcome the challenge of exposure bias problem and improved performance without the need for extra resources or training additional models.

The proposed AraGEC framework has been compared with the existing approaches including MT-based, NMT-based, and hybrid systems as shown in Table 5. CLMB-1 is the best system in the first Arabic shared-task Mohit et al. (2014) which is a hybrid system of machine-learning techniques and linguistic knowledge. SCUT is a neural-based model that employed

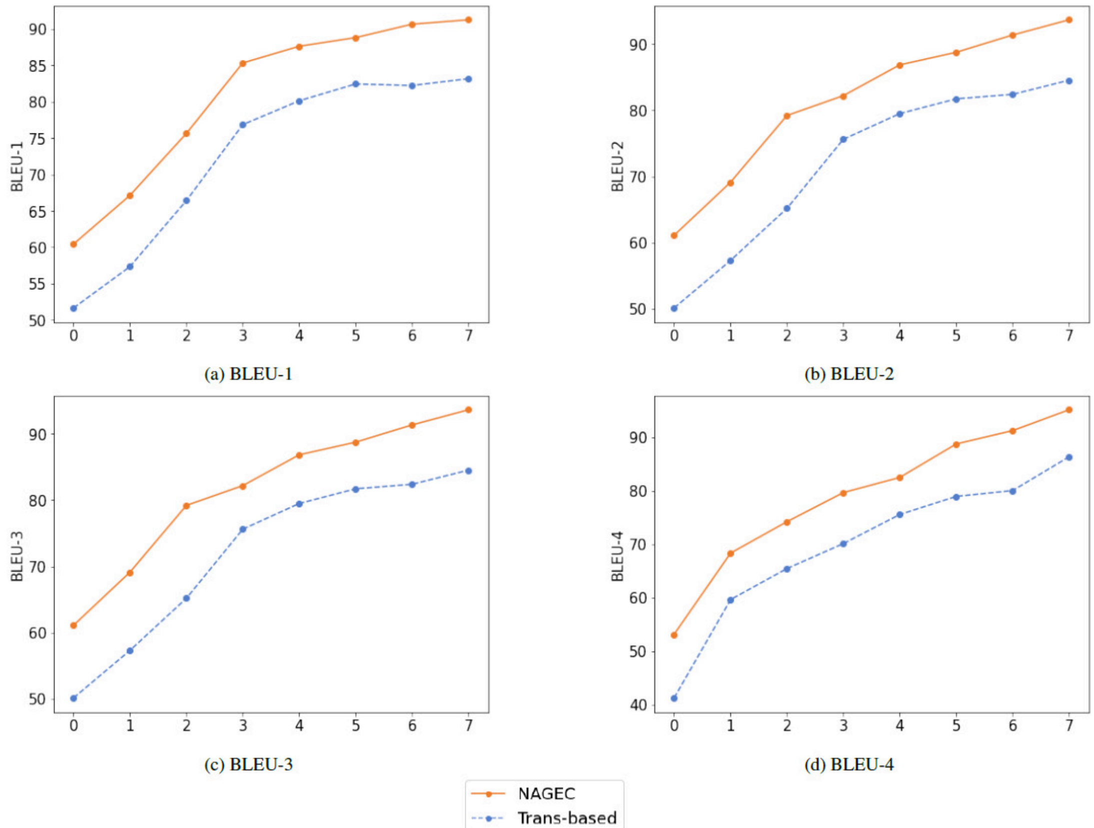


Figure 4. Performance achieved using different settings of BLEU score.

Sentence lengths in words	Unigram		Bigram		Trigram		Fourgram	
	Transf.	BKDGEC	Transf.	BKDGEC	Transf.	BKDGEC	Transf.	BKDGEC
1 to 29	51.65	**60.43**	49.80	**56.14**	50.14	**61.11**	41.20	**53.11**
30 - 35	57.34	**67.13**	56.73	**62.63**	57.32	**69.13**	59.6	**68.31**
36 - 45	66.41	**75.63**	65.18	**71.13**	65.18	**79.20**	65.42	**74.18**
46 - 55	76.84	**85.31**	75.42	**80.03**	75.61	**82.19**	70.09	**79.64**
56 - 65	80.11	**87.60**	78.92	**86.21**	79.49	**86.84**	75.53	**82.47**
66 - 75	82.42	**88.79**	81.96	**88.39**	81.71	**88.74**	78.92	**88.73**
76 - 85	82.21	**90.63**	82.13	**90.72**	82.39	**91.35**	81.02	**91.23**
> 85	83.14	**91.23**	83.22	**91.72**	84.52	**93.64**	86.32	**95.14**

Table 4. Performance of our AraGEC framework using different settings of BLEU score and different lengths.

System	2014	2015
CLMB-1 Rozovskaya et al. (2014)	67.91	N/A
SCUT Solyman et al. (2021)	N/A	70.91
CUFE Nawar (2015)	N/A	72.87
AHMADI Sina (2017)	50.34	N/A
WATSON Watson et al. (2018)	70.39	73.19
PAJAK Pajak and Pajak (2022)	N/A	69.81
BKDGEC (Our model)	**71.51**	**74.03**

Table 5. Comparisons of F_1 of our AraGEC framework and existing approaches using two benchmarks.

CNN and attention mechanism. CUFE is a systematic rule-based system for Arabic text correction that achieved the best score in the second shared-task Rozovskaya et al. (2015) of Arabic GEC. AHMADI and WATSON are neural-based models that exploit bidirectional RNN in different settings such as *Fasttext* pre-trained embeddings. PAJAK is a multi-lingual neural-based model tuned for GEC. In closing, BKDGEC achieves significant improvements over all AraGEC baselines in two benchmarks as shown in Figure 5.

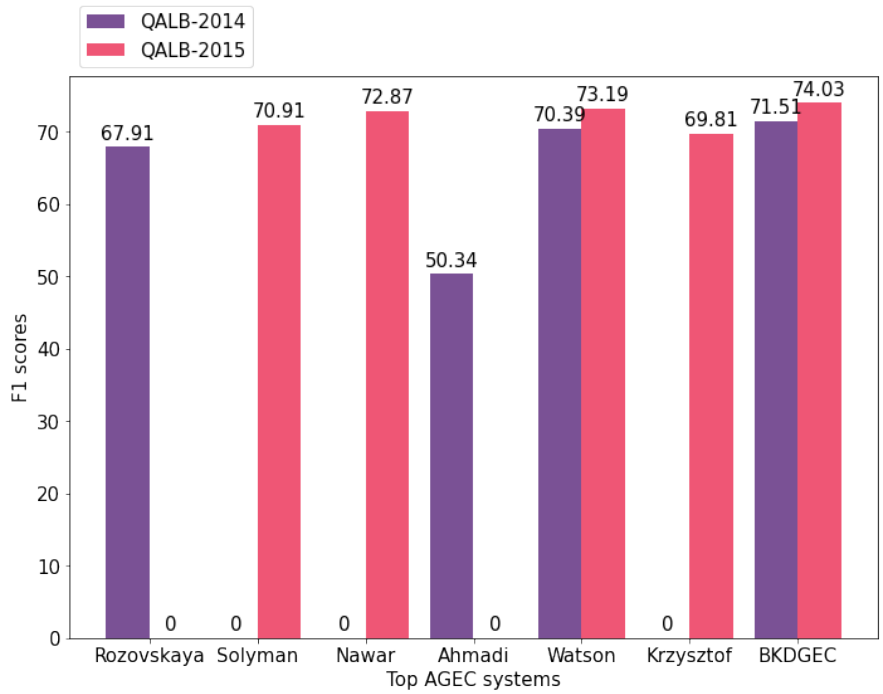


Figure 5. An illustrated F_1 score of the top systems in ArAraGEC using QALB-2014 and QALB-2015 benchmarks.

Case study

In this subsection, we investigate the performance of different versions of the GEC framework using a real-world example. The given example is from the QALB-2015 test set has 24 different errors, 18 spelling errors labeled as (sp), errors number 5(sy), and 13(sy) as syntax errors, while punctuation errors are in 4(pt), 6(pt), 16(pt), and 4(pt). Table 6 shows the output of the baseline, the baseline trained using EDSE data, the proposed model BKDGEC, and BKDGEC with fine-tuning. Furthermore, we provide the source, target, and English translation. Initially, the baseline that was trained using small training data of QALB-2014, corrected fifteen errors and failed to correct seven spelling and two punctuation errors.

Whereas a version of the baseline trained using our data EDSE has successfully corrected most of the reported errors except for three spelling and punctuation errors in 5(pt) and 6(pt) and caused a new punctuation error labeled "new". BKDGEC model corrected all the errors except two spelling errors in 17(sp), 18(sp), and the new punctuation error. BKDGEC with fine-tuning has been made significant improvements and corrected all the reported errors except the punctuation in "new"

This indicates that BKDGEC has been somewhat successful in challenging the scarcity of training data and also address the exposure bias problem. However, it is still far from being perfect as it fails to correct some punctuation, dialectal words, and challenging grammatical errors when the output of the test set has been checked sentence by sentence. Therefore, extra effort is needed to correct the dialectal words, punctuation, and the most complex grammatical errors.

Type	Example
Source	[1(sp)]الصحافه عندنا فى السودان وحتى [2(sp)]الان تفتقد [3(sp)]للمصداقيه وتعتمد فى نجاحها واستمراريتها على التطبيل [*4(pt)] [5(sy)]و النفاق [*6(pt)] والانحياز [7(sp)] الى [8(sp)] جماعه [9(sp)] او [10(sp)] فئه [11(sp)] معينه من [12(sp)]اجل تلميعها فقط [13(sp)] و ليس من [14(sp)] اجل [15(sp)] الصحافه [16(pt)] * [17(sp)] فبلدى [18(sp)] تحتاج [19(sp)] الى [20(sp)] اقلام [21(sp)] حره [22(sp)] شريفه من [23(sp)] اجل ذلك [*24(pt)]
Target	الصحافة عندنا فى السودان وحتى الآن تفتقد للمصداقية وتعتمد فى نجاحها واستمراريتها على التطبيل ، والنفاق ، والانحياز إلى جماعة أو فئة معينة من أجل تلميعها فقط وليس من أجل الصحافة ، فبلدي يحتاج إلى أقلام حرة شريفة من أجل ذلك .
English	The press we have in Sudan up to now lacks credibility and it depends on hypocrisy and polishing up a particular group for its success and continued existence, not for the sake of the press. For that, my country needs free and honest writers.
Baseline (Transformer)	الصحافة عندنا فى السودان وحتى [2(sp)]الان تفتقد [3(sp)]للمصداقيه وتعتمد فى نجاحها واستمراريتها على التطبيل [*4(pt)] والنفاق [*6(pt)] والانحياز [7(sp)] الى [8(sp)] جماعه أو فئة معينة من أجل تلميعها فقط وليس من أجل الصحافة ، [17(sp)] فبلدى [18(sp)] تحتاج إلى أقلام حرة [22(sp)] شريفه من أجل ذلك .
Baseline + EDSE data	الصحافة عندنا فى السودان وحتى [2(sp)]الان تفتقد للمصداقية وتعتمد فى نجاحها واستمراريتها على التطبيل [*4(pt)] والنفاق [*6(pt)] والانحياز إلى جماعة أو فئة معينة من أجل تلميعها فقط وليس من أجل الصحافة [new]. [17(sp)] فبلدى [18(sp)] تحتاج إلى أقلام حرة شريفة من أجل ذلك .
BKDGEC + EDSE data	الصحافة عندنا فى السودان وحتى الآن تفتقد للمصداقية وتعتمد فى نجاحها واستمراريتها على التطبيل ، والنفاق ، والانحياز إلى جماعة أو فئة معينة من أجل تلميعها فقط وليس من أجل الصحافة [new]. [17(sp)] فبلدى [18(sp)] تحتاج إلى أقلام حرة شريفة من أجل ذلك .
BKDGEC + EDSE data + Fine-tuning	الصحافة عندنا فى السودان وحتى الآن تفتقد للمصداقية وتعتمد فى نجاحها واستمراريتها على التطبيل ، والنفاق ، والانحياز إلى جماعة أو فئة معينة من أجل تلميعها فقط وليس من أجل الصحافة [new]. فبلدي يحتاج إلى أقلام حرة شريفة من أجل ذلك .

Table 6. Examples of output from different versions of BKDGEC framework, incorrect words are colored in red.

CONCLUSION AND FUTURE WORK

This paper introduced an AraGEC framework based on the Transformer-based equipped with Bidirectional Knowledge Distillation to overcome the exposure bias problem. Furthermore, the proposed model applied a process of knowledge distillation using a Kullback-Leibler divergence method as a regularization term to incorporate the learning information from the backward decoder to the forward decoder. To address the challenge of sparse data in GEC, a novel approach was proposed that utilized a supervised confusion function called the equal distribution technique for syntactic errors, which is used to construct massive synthetic data. The generated data has more diverse training patterns and consists of 25.162 million examples as the largest AraGEC training data. Experimental results on two benchmarks demonstrated that the synthetic data makes a significant improvement, which reported the highest F_1 score over the previous AraGEC systems.

In the future, we aim to investigate the influence of the confusion method in producing trustworthy syntactic training data for low-resource languages like Italian, Russian, and Indonesian. In the same context, we are also interested in investigating the impact of bidirectional knowledge distillation on other sequence-to-sequence tasks, such as text classification, image captioning, and conversational models.

REFERENCES

Acheampong, K. N. and Tian, W. (2021). Toward perfect neural cascading architecture for grammatical error correction. *Applied Intelligence*, 51:3775–3788.

Alfaifi, A., Atwell, E., and Hedaya, I. (2014). Arabic learner corpus (alc) v2: a new written and spoken corpus of arabic learners. In *Proceedings of Learner Corpus Studies in Asia and the World 2014*, volume 2, pages 77–89. Kobe International Communication Center.

Baghdadi, N. A., Malki, A., Balaha, H. M., AbdulAzeem, Y., Badawy, M., and Elhosseini, M. (2022). An optimized deep learning approach for suicide detection through arabic tweets. *PeerJ Computer Science*, 8:e1070.

Brown, P. F., Della Pietra, V. J., Desouza, P. V., Lai, J. C., and Mercer, R. L. (1992). Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–480.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., and Child, R. (2020). Language models are few-shot learners.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. (2022). Palm: Scaling language modeling with pathways.

Clark, K., Luong, M.-T., Khandelwal, U., Manning, C. D., and Le, Q. (2019). Bam! born-again multi-task networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5931–5937.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2020). Unsupervised cross-lingual representation learning at scale. In *ACL*.

Dahlmeier, D. and Ng, H. T. (2012). Better evaluation for grammatical error correction. In *Proceedings of the 2012 Conference of the North American: Human Language Technologies*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Ge, T., Wei, F., and Zhou, M. (2018). Fluency boost learning and inference for neural grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

Grundkiewicz, R., Junczys-Dowmunt, M., and Heafield, K. (2019). Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*.

Hagiwara, M. and Mita, M. (2020). Github typo corpus: A large-scale multilingual dataset of misspellings and grammatical errors. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6761–6768.

He, D., Xia, Y., Qin, T., Wang, L., Yu, N., Liu, T.-Y., and Ma, W.-Y. (2016). Dual learning for machine translation. *Advances in neural information processing systems*, 29.

Joyce, J. M. (2011). Kullback-leibler divergence. In *International encyclopedia of statistical science*, pages 720–722. Springer.

Junczys-Dowmunt, M. and Grundkiewicz, R. (2016). Phrase-based machine translation is state-of-the-art for automatic grammatical error correction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1546–1556.

Kingma, D. P. and Ba, J. L. (2015). Adam: A method for stochastic optimization. In *ICLR 2015 : International Conference on Learning Representations 2015*.

Kiyono, S., Suzuki, J., Mizumoto, T., and Inui, K. (2020). Massive exploration of pseudo data for grammatical error correction. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2134–2145.

Liu, L., Finch, A., Utiyama, M., and Sumita, E. (2016). Agreement on target-bidirectional lstms for sequence-to-sequence learning. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Madi, N. and Al-Khalifa, H. (2020). Error detection for arabic text using neural sequence labeling. *Applied Sciences*, 10(15):5279.

Masri, A. and Al-Jabi, M. (2023). A novel approach for arabic business email classification based on deep learning machines. *PeerJ Computer Science*, 9:e1221.

Mohit, B., Rozovskaya, A., Habash, N., Zaghouani, W., and Obeid, O. (2014). The first qalb shared task on automatic text correction for arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 39–47.

Náplava, J. and Straka, M. (2019). Grammatical error correction in low-resource scenarios. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 346–356.

Nawar, M. (2015). CUFE@QALB-2015 shared task: Arabic error correction system. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*.

Pajak, K. and Pajak, D. (2022). Multilingual fine-tuning for grammatical error correction. *Expert*

Systems with Applications, page 116948.

Rozovskaya, A., Bouamor, H., Habash, N., Zaghouani, W., Obeid, O., and Mohit, B. (2015). The second qalb shared task on automatic text correction for arabic. In *Proceedings of the Second workshop on Arabic natural language processing*, pages 26–35.

Rozovskaya, A., Habash, N., Eskander, R., Farra, N., and Salloum, W. (2014). The Columbia system in the QALB-2014 shared task on Arabic error correction. In *Proceedings of the EMNLP 2014 Workshop on Arabic*.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Simmons, R. F. (1978). Rule-based computations on english. In *Pattern-Directed Inference Systems*, pages 455–468. Elsevier.

Sina, A. (2017). Attention-based encoder-decoder networks for spelling and grammatical error correction.

Solyman, A., Wang, Z., and Tao, Q. (2019). Proposed model for arabic grammar error correction based on convolutional neural network. In *2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*.

Solyman, A., Zhenyu, W., Qian, T., Elhag, A. A. M., Rui, Z., and Mahmoud, Z. (2022). Automatic arabic grammatical error correction based on expectation maximization routing and target-bidirectional agreement. *Knowledge-Based Systems*, page 108180.

Solyman, A., Zhenyu, W., Qian, T., Elhag, A. A. M., Toseef, M., and Aleibeid, Z. (2021). Synthetic data with neural machine translation for automatic correction in arabic grammar. *Egyptian Informatics Journal*.

Sun, X., Ge, T., Ma, S., Li, J., Wei, F., and Wang, H. (2022). A unified strategy for multilingual grammatical error correction with pre-trained cross-lingual language model. *arXiv preprint arXiv:2201.10707*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wan, Z., Wan, X., and Wang, W. (2020). Improving grammatical error correction with data augmentation by editing latent representation. In *Proceedings of the 28th International Conference on Computational Linguistics*.

Watson, D., Zalmout, N., and Habash, N. (2018). Utilizing character and word embeddings for text normalization with sequence-to-sequence models. In *Proceedings of the 2018 Conference on Empirical Methods*.

Yuan, Z., Stahlberg, F., Rei, M., Byrne, B., and Yannakoudakis, H. (2019). Neural and FST-based approaches to grammatical error correction. pages 228–239, Florence, Italy. Association for Computational Linguistics.

Zhang, X., Shen, L., Pan, D., Wang, L., and Miao, Y. (2022). Look backward and forward: Self-knowledge distillation with bidirectional decoder for neural machine translation. *arXiv preprint arXiv:2203.05248*.

Zhang, X., Su, J., Qin, Y., Liu, Y., Ji, R., and Wang, H. (2018). Asynchronous bidirectional decoding for neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Zhang, Z., Wu, S., Liu, S., Li, M., Zhou, M., and Xu, T. (2019). Regularizing neural machine translation by target-bidirectional agreement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 443–450.

Zhao, W., Wang, L., Shen, K., Jia, R., and Liu, J. (2019). Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 156–165.

Zhou, L., Zhang, J., and Zong, C. (2019). Synchronous bidirectional neural machine translation. *Transactions of the Association for Computational Linguistics*, 7:91–105.