Lemmatizing Dialectal Arabic with Sequence-to-Sequence Models

Mostafa Saeed and **Nizar Habash**
Computational Approaches to Modeling Language (CAMeL) Lab
New York University Abu Dhabi
{mostafa.saeed,nizar.habash}@nyu.edu

Abstract

Lemmatization for dialectal Arabic poses many challenges due to the lack of orthographic standards and limited morphological analyzers. This work explores the effectiveness of Seq2Seq models for lemmatizing dialectal Arabic, both without analyzers and with their integration. We assess how well these models generalize across dialects and benefit from related varieties. Focusing on Egyptian, Gulf, and Levantine dialects with varying resource levels, our analysis highlights both the potential and limitations of data-driven approaches. The proposed method achieves significant gains over baselines, performing well in both low-resource and dialect-rich scenarios.

1 Introduction

Arabic lemmatization is particularly challenging due to Arabic's complex root and pattern morphology, and orthographic ambiguity caused by optional diacritics. These challenges are further amplified by the wide variation across dialects, which lack standardized spelling and differ significantly from Modern Standard Arabic (MSA) in vocabulary, syntax, and morphology, limiting the effectiveness of conventional NLP methods.

Lemmatization is a task of reducing a word to its base form, that abstracts away from its inflectional variants, which is a fundamental step in many NLP pipelines. Accurate lemmatization is crucial for downstream tasks such as Arabic diacritization (Habash and Rambow, 2007), summarization (El-Shishtawy and El-Ghannam, 2014), machine translation (Yeong et al., 2016) and and readability prediction (Liberato et al., 2024).

While lemmatization for MSA has been widely explored through systems such as (Abdelali et al., 2016; Obeid et al., 2020; Jarrar et al., 2024; Saeed and Habash, 2025), dialectal lemmatization remains significantly underexplored. Prior work has primarily focused on the Egyptian dialect, including efforts such as (Pasha et al., 2014; Zalmout and Habash, 2020a,b). More recently, CAMeL Tools (Obeid et al., 2020) has developed dialect-specific disambiguators for Egyptian (EGY), Gulf (GLF), and Levantine (LEV) Arabic, which we adopt as our primary baselines in this study.

Dataset	Lemmatizer		
	MSA_{CT}	DIA_{CT}	Our Sys
MSA	**98.0**	–	–
EGY	69.2	90.4	**90.9**
GLF	64.0	79.1	**93.7**
LEV	64.4	58.7	**79.5**

Table 1: Lemma accuracy (L) on MSA and dialectal test sets using CAMeL Tools (CT)'s MSA and Dialectal (EGY, GLF, and LEV) disambiguators, and our system.

As shown in Table 1, applying an MSA-trained disambiguator to MSA data performs well, but its effectiveness drops sharply on dialectal data, highlighting the limitations of cross-dialect generalization without dialect-specific resources. When such disambiguators are available, performance improves significantly, with an average gain of 10.2% over the MSA disambiguator. Our proposed system further boosts accuracy by 12% over the dialect-specific setups. Overall, the improvement from MSA disambiguation on dialects to our system reaches 22.2%, demonstrating its effectiveness in both low-resource and dialect aware scenarios. We explore these gains in detail as we examine Seq2Seq performance without analyzers and how it improves when integrated with them. All code and models are released to support continued research in Arabic lemmatization.[1]

The paper is structured as follows: §2 reviews background, related work, and datasets, §3 outlines our methodology, §4 presents the evaluation results, and §5 provides an in-depth error analysis.

[1] https://github.com/CAMeL-Lab/seq2seq-arabic-dialect-lemmatization

Proceedings of The Third Arabic Natural Language Processing Conference, pages 117–129
November 8-9, 2025

Diacritization		Lemma		POS	English
وِحْدَة	*wiH.daħ*	وِحْدَة	*wiH.daħ*	noun	unit
وِحْدَة	*wiH.daħ*	وِحْدَة	*wiH.daħ*	noun	loneliness
وَحِدَة	*waHidaħ*	حِدَة	*Hidaħ*	noun	separately
وَحِدَّة	*waHid∼aħ*	حِدَّة	*Hid∼aħ*	noun	intensity
وَحدَة	*waH.daħ*	وَحدَة	*waH.daħ*	noun_num	one
وَحْده	*waH.dh*	وَحْد	*waH.d*	noun	alone
وَحَدُّه	*waHad∼uh*	حَدّ	*Had∼*	verb	delimit
وَحِّدُه	*waH∼iduh*	وَحِّد	*waH∼id*	verb	unite

Table 2: Example surface forms and corresponding lemmatization variations.

2 Background and Related Work

2.1 Arabic Lemmas

Arabic is a morphologically rich and orthographically ambiguous language, characterized by complex root-and-pattern derivation and frequent omission of diacritics. This leads to significant surface ambiguity, where a single word form may correspond to multiple lemmas, parts of speech (POS), morphological features, such as gender, number, person, aspect, and a long list of attachable clitics and senses.

Table 2 illustrates this ambiguity using variants of the form وحده *wHdh*.[2] While some surface forms have distinct diacritics, others are not, and can differ in part-of-speech (POS), e.g. noun vs. verb, as well as meaning, e.g., 'unit', 'intensity', 'alone', 'to unite'. These distinctions are non-trivial, especially in dialects that lack standardized orthography and diacritic usage.

2.2 Lemmatization Resources

Several morphological databases and lexicons exist to support Arabic dialects lemmatization; however, these resources remain limited in coverage, with certain dialects lacking dedicated resources entirely, thereby significantly increasing the complexity of the task. Tharwa Lexicon (Diab et al., 2014) is a comprehensive three-way electronic lexicon linking Dialectal Arabic (initially Egyptian), Modern Standard Arabic, and English, with over 73K entries compiled from diverse sources. Maknuune Lexicon (Dibas et al., 2022) is a large open-resource lexicon for Palestinian Arabic, containing over 36K entries from around 17K lemmas, including diacritized orthography, phonological transcriptions, and English glosses. Qabas Lexicon (Jarrar and Hammouda, 2024) is an extensive open-source Arabic lexicon with around 58K lemmas, compiled from 110 lexicons and linked to 12 annotated corpora (2M tokens). It covers Classical Arabic, MSA, dialects, and transliterated foreign words.

In this research, we utilize the morphological taggers developed by CAMeL Tools (Obeid et al., 2020; Inoue et al., 2022) for Egyptian, Gulf, and Levantine. The quality of these analyzers connected to the taggers varies considerably. The Egyptian analyzer was manually annotated using expert linguistic annotations, resulting in high-quality morphological outputs (Habash et al., 2012b). In contrast, the Gulf and Levantine analyzers were automatically generated using paradigm completion techniques (Eskander et al., 2013; Khalifa et al., 2020), which may introduce inconsistencies and limit their accuracy due to the absence of manual validation.

Several Arabic dialect lemmatization benchmark datasets have been created as part of larger annotation efforts, including ARZATB for Egyptian Arabic (Maamouri et al., 2012, 2014), Curras for Palestinian (Levantine) Arabic (Jarrar et al., 2016), Gumar Annotated Corpus for Gulf Arabic (Khalifa et al., 2018), a six-dialect corpus covering Saudi, Moroccan, Iraqi, Syrian, Yemeni, and Jordanian Arabic (Alshargi et al., 2019), Baladi for Lebanese Arabic (Al-Haff et al., 2022), Nabra for Syrian Arabic (Nayouf et al., 2023), and Lîsan dataset covering Iraqi, Yemeni, Sudanese, and Libyan dialects (Jarrar et al., 2023). In this research, we focus on lemmatization for three Arabic dialects: Egyptian, Gulf, and Levantine. We examine the structure, coverage, and consistency of these corresponding datasets and report lemmatization results using both baseline and proposed approaches.

2.3 Lemmatization Approaches

Arabic lemmatization has been a central task in morphological analysis, and it has been extensively explored through a variety of computational approaches over the years. These include rule-based finite state machines (MINNEN et al., 2001), which utilize manually crafted morphological rules and transition systems to derive lemmas from surface forms. Lexicon-based selection methods depend on comprehensive dictionaries or morphological databases to select the correct lemma based on the observed word and its context (Roth et al., 2008; Ingason et al., 2008; Jongejan and Dalianis,

[2]Arabic in HSB Romanization (Habash et al., 2007).

2009; Mubarak, 2018; Ingólfsdóttir et al., 2019; Zalmout and Habash, 2020a; Jarrar et al., 2024). Tagging-based frameworks approach lemmatization as a classification task by predicting a set of morphological tags (e.g., POS, gender, number), which are then used to infer the lemma (Gesmundo and Samardzic, 2012; Müller et al., 2015). More recently, Seq2Seq neural models have been adopted, treating lemmatization as a generation task that maps inflected word forms to lemmas using deep neural architectures trained on large corpora, often leveraging contextual embeddings for improved generalization (Sennrich and Haddow, 2016; Bergmanis and Goldwater, 2018; Kondratyuk et al., 2018; Zalmout and Habash, 2020b; Sahala, 2024).

Despite the richness and variety of approaches for Modern Standard Arabic (MSA), research on dialectal Arabic lemmatization remains significantly underdeveloped. Most existing work has focused almost exclusively on Egyptian Arabic, which benefits from relatively better linguistic resources. In contrast, other dialects have received little to no attention in lemmatization studies, despite their widespread use and linguistic diversity. This highlights a major gap in the field and underscores the need for broader efforts to develop lemmatization tools that can effectively handle the morphological complexity and variability of Arabic dialects.

Zalmout and Habash (2020a) proposed a unified model for joint morphological tagging and lemmatization. A Bi-LSTM tagger predicts non-lexicalized features using full sentence context and character embeddings, while lexicalized features are generated by character-level decoders conditioned on tags and encoder states. Gradient flow from decoder to tagger is blocked, and CODA normalization is applied to address dialectal variation in MSA and Egyptian Arabic.

Zalmout and Habash (2020b) proposed a lemmatization method for MSA that integrates heuristic and unsupervised subword features, including stems, patterns, roots, and segments from morphological analysis. These are fed into a character-level Seq2Seq model with context, and the architecture supports multitask learning by jointly training lemmatization and subword prediction.

Our work is inspired by Saeed and Habash (2025), who demonstrated that Seq2Seq models can be trained for lemmatization without relying on external resources, and that integrating morphological analyzers can enhance performance. Building on this, we show that cross-dialectal approaches leveraging shared datasets and analyzers not only support generalization but also improve lemmatization accuracy within individual dialects.

Dataset	Train	Dev	Test
EGY (Maamouri et al., 2012)	133,746	21,146	20,462
GLF (Khalifa et al., 2018)	161,815	20,181	20,089
LEV (Jarrar et al., 2016)	45,018	5,823	5,854

Table 3: Number of words in the train, dev, and test splits for the dialectal dataset we study.

2.4 Datasets

We conduct our experiments on three publicly available datasets: ARZATB for **EGY** (Maamouri et al., 2012, 2014), Gumar Annotated Corpus (henceforth Gumar) for **GLF** (Khalifa et al., 2018), and the Curras corpus for **LEV** (Jarrar et al., 2016).

All of these sets provide reliable lemmatization annotations suitable for robust evaluation. Other available dialectal datasets were excluded due to major inconsistencies in lemma diacritization, such as irregular treatment of initial vowels or selective retention of final vowels and tanween. To be usable, these datasets would require normalization based on standardized conventions like the Conventional Orthography for Dialectal Arabic (CODA) (Habash et al., 2012a), which would help align them with consistent diacritization rules and make them valuable for expanding cross-dialectal lemmatization research.

To provide an overview of the scale and distribution of our data, Table 3 reports the number of words in the train, dev, and test splits for each of the three dialectal datasets used in our experiments. Understanding the size of each split is essential, as it highlights the relative richness of the training resources and the robustness of the evaluation sets. These statistics offer insight into the potential learning capacity and generalization behavior of the lemmatization models trained on each dialect.

In addition to the above, we use multiple MSA data sets: ATB (Maamouri et al., 2004), NEMLAR (Yaseen et al., 2006), Quranic Corpus (Dukes and Habash, 2010), WikiNews (Mubarak, 2018), ZAEBUC (Habash and Palfreyman, 2022), and the BAREC dataset lemmas annotated version (Elmadani et al., 2025; Saeed and Habash, 2025). We specifically use these datasets in experiments with **ATB** alone and with all MSA sets combined (**MSA**) (see Table 4).*

3 Approach

We explore and evaluate a range of approaches for lemmatizing Arabic dialects, aiming to address the linguistic complexity and morphological richness inherent in these varieties. Our primary focus is on the effectiveness of Seq2Seq models in generating accurate diacritized lemmas across different dialects. We investigate how these models perform when used independently, as standalone lemmatizers, as well as how they can be integrated into larger morphological analysis pipelines to refine outputs. We discuss the different lemmatization strategies considered in this study next.

Disambiguator (Tagger) This approach uses a dialect-specific POS taggers trained on annotated data, primarily focusing on the Egyptian, Gulf, and Levantine models by Inoue et al. (2022). Each word is assigned a ranked list of morphological analyses, and each analysis includes over 37 features, including pos, gender, number, clitics, along with the lemma and **pos-lex** (POS-Lemma) log-probability. The top 1 scoring analysis is selected, with the **pos-log** probability used to break ties. This setup serves as our main baseline.

Standalone Seq2Seq Model Our first proposed approach treats lemmatization as a standalone Seq2Seq task, where the model takes a target word along with a two-word context window on each side and is trained to generate the diacritized lemma for this target word. We experiment with six training configurations to systematically assess the impact of different supervision settings:

1. **Dialect Specific (DS) S2S** trains a separate model for each dialect using only its own data; each dialectal model is also evaluated on the other dialects to assess cross-dialect generalization.
2. **ATB S2S** trains a model solely on the Penn Arabic Treebank (ATB) data.
3. **Dialect+ATB (DS+ATB) S2S** augments each dialect's data with ATB.
4. **All Dialects (AD) S2S** trains a unified model on a combined dataset that includes EGY, GLF and LEV.
5. **MSA-only (MSA*) S2S** uses only the MSA datasets (See Section 2.4)
6. **All Dialects+MSA (AD+MSA*) S2S** augments each dialect's data with all available MSA resources.

These variations enable us to explore the effects of dialect-specific training, MSA-based supervision, and cross-dialectal learning, allowing for a fine-grained comparison of their contributions to lemmatization performance.

Seq2Seq-Guided Single Tagger The second proposed approach integrates the Seq2Seq model as a filtering stage applied to the output of a dialect-specific morphological tagger. The analyzer not only narrows down the candidate space significantly but also provides the pos tag, addressing a limitation of the standalone Seq2Seq model. We use the lemma predicted by the Seq2Seq model to filter the tagger set of lex-pos candidates, retaining only the candidates whose lemma matches the Seq2Seq output, and if no match exists, we fall back to the top-ranked candidate from the tagger. All the training configurations used in the standalone Seq2Seq approach whether dialect only, ATB augmented, MSA enriched, or cross-dialectal are reused in this setup to examine how different levels of supervision influence the filtering stage. Additionally, we explore two variants of this strategy: (i) one that filters over all tagger generated candidates (**All**), and (ii) another that filters only within the top scoring subset (**Top**). This enables us to evaluate the trade off between broad exploration and high confidence disambiguation.

Seq2Seq-Guided Multi Tagger Building on the two previous approaches, this strategy also combines Seq2Seq outputs with morphological taggers, but differs in the number of taggers used, integrating outputs from all three dialect specific taggers: Egyptian, Gulf, and Levantine. The goal is to enhance the performance of GLF and LEV analyzers, which are automatically generated and less reliable, by leveraging the higher quality Egyptian tagger that benefits from expert manual annotation. This cross dialect tagger setup enables weaker resourced dialects to benefit from morphological signals present in more robust analyzers. These approaches allow us to examine how integrating generative models with multiple taggers affects lemmatization quality and whether cross dialect Seq2Seq models can outperform single dialect models. They also help assess the extent to which support from high quality resources like the Egyptian tagger can improve performance in lower resource dialects.

Dialect	DS	ATB	DS+ATB	AD	MSA*	AD+MSA*
EGY	133,746	503,015	636,761	340,579	1,141,165	1,481,744
GLF	161,815	503,015	664,830	340,579	1,141,165	1,481,744
LEV	45,018	503,015	548,033	340,579	1,141,165	1,481,744

Table 4: Number of words used for training across setups. **DS** (Dialect Specific) refers to the dialect in the corresponding row; **AD** (All Dialects) refers to the union of all dialectal data

Dialect	DS		ATB		DS+ATB		AD		MSA*		AD+MSA*	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
EGY	5.5	6.5	32.2	29.7	4.1	4.5	4.8	5.6	28.1	25.4	3.5	3.6
GLF	2.0	2.1	46.4	45.8	1.5	1.6	1.5	1.5	42.8	41.5	1.3	1.3
LEV	13.3	13.5	35.0	35.7	8.8	8.5	8.4	8.5	32.4	31.4	6.7	6.4

Table 5: OOV lex (%) in Dev and Test sets. **DS** (Dialect Specific) refers to the dialect in the corresponding row; **AD** (All Dialects) refers to the union of all dialectal data.

Dialect	DS		ATB		DS+ATB		AD		MSA*		AD+MSA*	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
EGY	16.4	18.1	29.3	30.9	11.9	13.4	14.1	15.5	26.2	26.9	10.0	11.4
GLF	8.1	8.2	35.3	34.7	6.3	6.5	6.3	6.5	30.4	29.6	5.3	5.6
LEV	28.8	30.3	32.5	33.0	19.9	19.7	18.2	19.7	28.5	28.7	14.5	15.0

Table 6: OOV word forms (%) in Dev and Test sets. **DS** (Dialect Specific) refers to the dialect in the corresponding row; **AD** (All Dialects) refers to the union of all dialectal data.

4 Evaluation

4.1 Experiments Setup

Seq2Seq Models Hyperparameters We followed the Seq2Seq architecture introduced by Saeed and Habash (2025). Models are trained for 100 epochs with a learning rate of 5e-5, using batch sizes of 64 (train) and 32 (eval), and gradient checkpointing. The best model was selected based on validation accuracy at the end of each epoch. Training was conducted on three A100 GPUs, taking between 2–5 hours for dialect specific models and up to 24 hours for the all-dialects model, depending on the size of the training data.

Seq2Seq Models Data All development and evaluation in this work focused on the three dialectal datasets mentioned earlier, EGY, GLF, and LEV, which have been previously used in the morphosyntactic tagging paper by Inoue et al. (2022), making them a consistent and validated choice for lemmatization. Model variations were tuned using the last 10% of the training set as the tuning set, with evaluation performed on the corresponding dev set. Tuning was carried out separately for each dialectal training set; however, for models involving AD, the tuning set was constructed by taking 10% from each of the dialectal training sets (EGY, GLF, and LEV), while for the ATB and MSA setup, the tuning data was drawn only from the MSA portion.

To further analyze the training data, we report three complementary statistics. Table 4 presents the total number of words used for training across the different setups. Table 5 reports the percentage of unseen lex entries (OOV lex) that appear in the dev and test sets but were not present in training, and Table 6 provides the percentage of unseen surface word forms (OOV words) that occur in the dev and test sets. For both OOV lex and OOV word analyses, we first extracted the unique words from each training set, ensuring that repeated tokens were excluded from these calculations.

Seq2Seq Models Tokenizer We used the AraT5v2-base-1024 tokenizer, which is the latest release of AraT5. This version provides improved handling of Arabic text and is capable of processing diacritics, allowing us to preserve important linguistic information during tokenization.

Metrics We report results using two evaluation metrics, with **lemma accuracy (L)** serving as our primary evaluation metric. Lemma accuracy (L) is computed by comparing the predicted lemma to the gold lemma after removing from both any sukuns and any diacritics preceding ي,و,ا *A,w,y* (used to indicate long vowels). We also report **normalized**

Analyzer	Tagger set	S2S	Metric	DS	GLF	LEV	ATB	DS+ATB	AD	MSA*	AD+MSA*
Single	Top	-	L	90.0	–	–	–	–	–	–	–
–	–	S2S	L	79.2	42.9	40.4	59.8	66.8	83.0	56.4	70.6
Single	Top	S2S	L	**90.4**	87.8	88.7	83.0	83.9	**90.4**	83.3	82.8
Single	All	S2S	L	89.5	85.3	87.3	76.8	80.3	89.3	78.1	79.2
Multiple	Top	S2S	L	90.0	79.0	85.1	82.1	83.2	88.8	82.3	81.8
Multiple	All	S2S	L	89.2	76.8	84.2	76.3	80.0	88.0	76.9	78.7
Single	Top	-	L′	96.3	–	–	–	–	–	–	–
–	–	S2S	L′	85.1	57.8	49.9	73.8	78.9	90.9	68.8	83.7
Single	Top	S2S	L′	96.4	95.9	96.3	94.2	94.7	**96.5**	94.4	94.4
Single	All	S2S	L′	95.8	94.6	95.3	90.8	92.6	96.1	91.1	91.9
Multiple	Top	S2S	L′	96.1	94.3	95.3	93.6	94.2	96.0	93.6	93.8
Multiple	All	S2S	L′	95.6	93.1	94.6	90.4	92.4	95.7	89.9	91.5

Table 7: Comparison of lemmatization techniques on the **EGY** dev set across different training setups. The table summarizes system components for each configuration, including tagger type (single or multi), tag set (top or all), use of a Seq2Seq model, and granularity level (L vs. L′). Columns represent the various training setups introduced earlier.

lemma accuracy (L′), which offers a more lenient evaluation by further removing all diacritics and normalizing all forms of Alef to a standard form. This allows us to assess the robustness of the model to surface level variations while maintaining **L** as the central measure of lemma correctness.

For initial evaluation, we applied the CAMeL Tools tagger, relying on its top one ranked analysis for each word as our baseline. We then advanced to the proposed approach, which begins with training a Seq2Seq model under the various training configurations described earlier. In this setup, the Seq2Seq models can be applied on their own, where they serve not only as an additional point of comparison but also as a simple yet robust baseline or being empowered by integrating them with the outputs of the dialectal taggers, allowing us to better exploit cross-dialectal information and enhance the overall predictive performance.

4.2 Results

Development Phase We begin by presenting the results of the proposed approaches on the dev sets of EGY, GLF, and LEV datasets. These initial evaluations allow us to analyze performance during model development. We then report results on the corresponding test sets of these three datasets. In the following tables, we experiment with eight different models: **DS** (each trained on one dialect and additionally evaluated on the other two dialects), **ATB**, **DS+ATB**, **AD**, **MSA***, and **AD+MSA***.

For the **EGY** dev set, as shown in Table 7, only the top tagger set with a single analyzer improves lemma accuracy (L) over the baseline, whether using the DS model or the AD model, achieving the highest score of 90.4%. Notably, multiple taggers did not enhance L, indicating that the Egyptian analyzer alone delivers high quality outputs without requiring additional taggers. In addition to that, the Seq2Seq model on its own, without the analyzer did not surpass the baseline. As for L′, most configurations with the DS and AD models outperformed the baseline, with the AD setup achieving the highest score of 96.5%, again excluding the standalone Seq2Seq model, which underperformed in the absence of analyzer support.

For the **GLF** dev set, as shown in Table 8, in the Seq2Seq-only setup the model outperforms the baseline on both DS and AD, achieving 92.2% and 92.9% in lemma accuracy (L), and 93.7% and 95.5% in normalized lemma accuracy (L′), respectively. When the tagger is integrated with the Seq2Seq model, L improves over the baseline across all single-tagger setups, regardless of whether the top or all tagsets are used. Performance further increases with multiple taggers, particularly in the DS and AD setups, with the DS model yielding the highest results 93.9% for L and 96.9% for L′. Overall, tagger integration generally enhances performance for L′, with only a few configurations failing to surpass the baseline, which highlights the benefit of using multiple analyzers when the dialect specific analyzer is not that good.

For the **LEV** dev set, as shown in Table 9 the Seq2Seq models on their own outperform the baseline for L in the DS, AD, and AD+MSA* setups. For L′, only the AD and AD+MSA* configurations show improvement over the baseline. When

Analyzer	Tagger set	S2S	Metric	DS	EGY	LEV	ATB	DS+ATB	AD	MSA*	AD+MSA*
Single	Top	-	L	78.7	–	–	–	–	–	–	–
–	–	S2S	L	92.2	51.5	47.3	56.2	69.6	92.9	56.2	70.6
Single	Top	S2S	L	88.3	81.6	81.5	82.9	85.8	88.2	82.9	85.3
Single	All	S2S	L	89.9	81.5	81.8	82.4	86.1	89.7	82.0	85.6
Multiple	Top	S2S	L	**93.9**	73.9	78.0	75.0	83.6	92.9	76.6	78.5
Multiple	All	S2S	L	93.4	71.9	76.5	71.8	82.1	92.5	73.1	76.5
Single	Top	-	L′	88.8	–	–	–	–	–	–	–
–	–	S2S	L′	93.7	66.3	56.1	71.2	79.6	95.5	69.3	85.4
Single	Top	S2S	L′	95.3	91.4	90.6	91.2	93.5	95.3	91.2	93.3
Single	All	S2S	L′	95.5	90.5	90.1	89.6	93.1	95.5	89.6	92.6
Multiple	Top	S2S	L′	**96.9**	91.7	91.1	89.6	94.0	**96.9**	90.3	92.7
Multiple	All	S2S	L′	96.3	90.0	89.9	86.9	93.0	96.4	87.1	91.4

Table 8: Comparison of lemmatization techniques on the **GLF** dev set across different training setups. The table summarizes system components for each configuration, including tagger type (single or multi), tag set (top or all), use of a Seq2Seq model, and granularity level (L vs. L′). Columns represent the various training setups introduced earlier.

Analyzer	Tagger set	S2S	Metric	DS	EGY	GLF	ATB	DS+ATB	AD	MSA*	AD+MSA*
Single	Top	-	L	60.2	–	–	–	–	–	–	–
–	–	S2S	L	62.1	58.6	49.6	56.6	56.1	74.2	55.5	62.5
Single	Top	S2S	L	66.5	64.1	64.0	63.8	63.7	67.3	63.9	65.0
Single	All	S2S	L	69.2	66.0	66.0	64.8	64.8	69.6	64.9	66.6
Multiple	Top	S2S	L	**78.8**	72.6	67.9	66.5	66.3	76.7	68.4	68.3
Multiple	All	S2S	L	74.0	68.3	63.1	62.9	62.5	74.2	64.3	65.3
Single	Top	-	L′	77.5	–	–	–	–	–	–	–
–	–	S2S	L′	64.5	68.9	63.1	73.1	72.5	85.4	68.4	78.6
Single	Top	S2S	L′	81.9	80.5	80.6	80.1	80.3	82.9	80.3	81.2
Single	All	S2S	L′	81.9	80.2	80.3	79.7	79.9	82.7	79.8	80.7
Multiple	Top	S2S	L′	88.2	86.8	86.5	85.9	85.9	**90.0**	85.8	86.7
Multiple	All	S2S	L′	83.4	82.8	81.9	82.5	82.1	87.3	82.0	83.6

Table 9: Comparison of lemmatization techniques on the **LEV** dev set across different training setups. The table summarizes system components for each configuration, including analyzer type (single or multiple), tagger set (top or all), use of a Seq2Seq model, and granularity level (L vs. L′). Columns represent the various training setups introduced earlier.

integrating taggers, all single tagger setups using both the top and all tagsets surpass the baseline in L. Multi-tagger configurations also consistently outperform the baseline and single tagger experiments for each setup, with the best result (78.8%) achieved using the DS model with the top tagger set. For L′, both single and multi-tagger setups outperform the baseline across the board, with the highest result obtained using the AD model, with the multi-tagger top set setup achieving 90.0%.

In the development phase, the Seq2Seq models alone outperformed the baseline for GLF and LEV in terms of lemma accuracy (L) using DS or AD setups, but not for EGY. When combined with taggers, multi-tagger setups produced substantially better results for Gulf and Levantine compared to single-tagger setups, whereas the single tagger configuration worked best for EGY, likely due to the already high quality of the EGY analyzer. These findings highlight the effectiveness of cross-dialectal integration, whether through training data as in the DS or AD setup or through tagger combinations, in improving lemma prediction for lower-resource dialects. The highest L scores were achieved using the DS model with multi-taggers for GLF 93.9% and LEV 78.8%, while the single tagger for EGY with 90.4% accuracy.

Testing Phase Based on the findings from the development phase, we evaluate the best performing models on the test sets of EGY, GLF, and LEV. Specifically, we test the baseline of each dataset using the single analyzer Top tagger configuration

Dataset	Analyzer	Tagger set	S2S	Metric	DS
EGY	Single	Top	-	L	90.4
EGY	Single	Top	S2S	L	**90.9**
EGY	Single	Top	-	L′	96.1
EGY	Single	Top	S2S	L′	**96.3**
GLF	Single	Top	-	L	79.1
GLF	Multiple	Top	S2S	L	**93.7**
GLF	Single	Top	-	L′	89.1
GLF	Multiple	Top	S2S	L′	**97.2**
LEV	Single	Top	-	L	58.7
LEV	Multiple	Top	S2S	L	**79.5**
LEV	Single	Top	-	L′	76.4
LEV	Multiple	Top	S2S	L′	**88.3**

Table 10: Top tagger results on EGY, GLF, and LEV **test** sets.

for EGY data, while applying the multiple analyzer Top tagger setup for GLF and LEV. For all three datasets, we use the DS Seq2Seq model as it consistently showed the strongest performance during development.

As shown in Table 10, the key insights from the development phase generalize well to the test phase. In all three dialect datasets, the DS Seq2Seq model consistently outperforms the baseline. For EGY, the performance gains are marginal, reflecting the already high quality of its tagger, improving from 90.4% to 90.9%. In contrast, GLF and LEV show more substantial improvements rising from 79.1% to 93.7% and from 58.7% to 79.5%, respectively, when leveraging multi- analyzer outputs, highlighting the value of cross-dialectal support. These results reinforce the effectiveness of our selected configurations for robust lemmatization across diverse dialects.

5 Error Analysis

To better understand the limitations of our lemmatization system, we conduct a manual error analysis on a sample of 300 words: 100 each from the development sets of Egyptian, Gulf, and Levantine Arabic. For each instance, we annotate three aspects: (1) whether the gold lemma is a valid lemmatization (i.e., free of annotation errors), (2) whether the model prediction is fully correct, plausibly acceptable, or clearly incorrect, and (3) the specific error type in case of errors.

Table 11 summarizes the distribution of the first two judgments (Gold validity and Prediction correctness) across the full sample and each of the three dialects. We find that around 23% of the total errors are due to problems with the gold reference itself, such as annotation inconsistencies or outright mistakes. This highlights the difficulty of ensuring high-quality gold annotations for dialectal Arabic, especially given orthographic variation and limited guidelines.

Gold	Prediction	All	EGY	GLF	LEV
Valid	**Wrong**	56%	37%	75%	57%
Valid	**Plausible**	11%	20%	4%	9%
Valid	**Correct**	10%	19%	4%	7%
Error	**Wrong**	8%	6%	9%	10%
Error	**Correct**	14%	18%	8%	17%
Valid	–	77%	76%	83%	73%
Error	–	23%	24%	17%	27%
–	**Wrong**	65%	43%	84%	67%
–	**Plausible**	11%	20%	4%	9%
–	**Correct**	24%	37%	12%	24%

Table 11: Manual analysis of 300 lemmatization errors sampled from dev sets (100 per dialect). Judgments reflect gold lemma validity and prediction correctness.

When the gold lemma is valid, our system's errors are actually correct 10% of the time, and plausibly acceptable in an additional 11%, suggesting that some "errors" may be more a matter of interpretation. Only 56.3% of the predictions are clearly incorrect relative to the gold.

Dialect-specific trends are also noteworthy: Gulf Arabic has the highest share of correct gold references but also the highest proportion of clearly wrong predictions, indicating robustness issues in generalization. Egyptian, conversely, has the highest proportion of plausibly correct outputs and the lowest share of outright wrong predictions.

Our manual analysis of error types reveals several key challenges in dialectal Arabic lemmatization. The most frequent error category is **Hallucination** (14.0%), where the model generates a lemma unrelated to the input word's meaning, often due to overgeneralization or ambiguity in surface forms. **Verb pattern confusion**, especially within the Form I vs. Form II paradigms (e.g., وَقَّف *waq~af* vs. وَقَف *waqaf*), is another significant source of error (10.7%), highlighting the difficulty of capturing subtle morphological distinctions without context or diacritics.

Nominal derivation confusions (e.g., **Nominal Patterns** and **Nominal-Verbal** errors, 14.7% combined) further indicate that the model struggles to distinguish between semantically related noun and

Error Type	%	Word		Gold Lemma			Predicted Lemma		
Hallucination	14.0	تسرقه	*tsrqh*	سَرَق	*saraq*	to steal	سَمَّى	*sam∼aý*	to name
Verbal Patterns	10.7	وقف	*wqf*	وَقَّف	*waq∼af*	halt	وَقَف	*waqaf*	stand up
Nominal Patterns	7.7	مجتمعة	*mjtmςħ*	مُجتَمِع	*muj.tamiς*	gathering	مُجتَمَع	*muj.tamaς*	community
Nominal-Verbal	7.0	جنية	*jnyħ*	جِنِّيّ	*jin∼iy∼*	genie	جَنَى	*janaý*	to reap
Clitic Confusion	7.3	لهدرجة	*lhdrjħ*	دَرَجَة	*darajaħ*	degree	هَدرَجَة	*hadrajaħ*	hydrogenation
Diacritization	5.7	بتعبر	*btςbr*	عَبَّر	*ςab∼ar*	to express	عبّر	*ςb∼r*	to express (sp)
Input Typo	4.7	نفول	*nfwl*	قال	*qAl*	to say	نَفَل	*nafal*	to loot
Lemma Choice	3.7	القتلة	*Alqtlħ*	قاتِل	*qAtil*	killer	قَتلَة	*qatlaħ*	killers
Spelling	4.0	ونضارات	*wnDArAt*	نَظّارَة	*naĎ∼Araħ*	glasses	نَضّارَة	*naD∼Araħ*	glasses (sp)

Table 12: Representative lemmatization errors by category. Each row includes the original dialectal word, the gold lemma and gloss, and the predicted lemma and gloss.

verb forms. **Clitic segmentation errors** (7.3%) suggest issues with boundary detection in fused forms, a known challenge in dialects lacking standard orthography.

Errors due to **input noise (typos)** or **spelling variation** (8.7%) show the importance of robust preprocessing and orthographic normalization. Finally, some **diacritic-related mismatches** (5.7%) reflect annotation inconsistencies or cases where both gold and prediction are plausible, indicating the limits of purely form-based evaluation.

These findings suggest that integrating contextual modeling, improved orthographic handling, and richer morphological priors could further enhance lemmatization performance in dialectal settings.

6 Conclusion and Future Work

This work introduced Arabic dialect lemmatization as a Seq2Seq task, evaluating both standalone models and configs that integrate taggers. Results show that some standalone Seq2Seq setups for LEV and GLF outperform the baseline, while this is not the case for EGY. With taggers, LEV and GLF surpass the baseline with single tagger setup, and the best results come from multi tagger DS and AD configs. For EGY, the top performance is with single tagger setups under DS and AD. Notably, while combining taggers or applying cross-dialectal approaches does not always benefit dialects with high-quality resources, such strategies greatly improve performance for under-resourced dialects.

Future work includes addressing occasional hallucinations from Seq2Seq models, possibly through constrained decoding, and exploring the integration of additional morphological features(e.g., POS tags, affix patterns) to enrich input representations and better guide training; and applying CODA normalization (Habash et al., 2012a) to remaining dialectal datasets to standardize lemma annotations particularly since no prior work has systematically reported on these datasets for lemmatization task.

Acknowledgments

We acknowledge the support of the High Performance Computing Center at New York University Abu Dhabi. We thank Salam Khalifa, Go Inoue, Bashar Alhafni, Ossama Obeid and Kurt Micallef for helpful discussions.

Limitations

While our Seq2Seq lemmatization approach shows strong performance across dialects, several limitations remain. First, the system relies heavily on supervised data, which is limited in both quantity and quality for dialectal Arabic. In particular, we found that a notable portion of evaluation errors stem from inconsistencies or inaccuracies in gold annotations. Second, the model operates purely at the surface level without explicit morphological structure or linguistic constraints, which may hinder generalization to rare or unseen forms. Although integration with existing analyzers improves results, such tools are only available for a few dialects and vary in coverage. Future work could explore unsupervised or semi-supervised techniques, richer features, and broader dialect coverage to enhance robustness and reduce dependence on annotated resources.

References

Ahmed Abdelali, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. Farasa: A fast and furious segmenter for Arabic. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 11–16, San Diego, California. Association for Computational Linguistics.

Karim Al-Haff, Mustafa Jarrar, Tymaa Hammouda, and Fadi Zaraket. 2022. Curras + baladi: Towards a Levantine corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 769–778, Marseille, France. European Language Resources Association.

Faisal Alshargi, Shahd Dibas, Sakhar Alkhereyf, Reem Faraj, Basmah Abdulkareem, Sane Yagi, Ouafaa Kacha, Nizar Habash, and Owen Rambow. 2019. Morphologically annotated corpora for seven Arabic dialects: Taizi, Sanaani, Najdi, Jordanian, Syrian, Iraqi and Moroccan. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 137–147, Florence, Italy. Association for Computational Linguistics.

Toms Bergmanis and Sharon Goldwater. 2018. Context sensitive neural lemmatization with lematus. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1391–1400.

Mona T Diab, Mohamed Al-Badrashiny, Maryam Aminian, Mohammed Attia, Heba Elfardy, Nizar Habash, Abdelati Hawwari, Wael Salloum, Pradeep Dasigi, and Ramy Eskander. 2014. Tharwa: A Large Scale Dialectal Arabic-Standard Arabic-English Lexicon. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, pages 3782–3789, Reykjavik, Iceland.

Shahd Salah Uddin Dibas, Christian Khairallah, Nizar Habash, Omar Fayez Sadi, Tariq Sairafy, Karmel Sarabta, and Abrar Ardah. 2022. Maknuune: A large open Palestinian Arabic lexicon. In *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 131–141, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Kais Dukes and Nizar Habash. 2010. Morphological Annotation of Quranic Arabic. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, Valetta, Malta.

Tarek El-Shishtawy and Fatma El-Ghannam. 2014. A lemma based evaluator for semitic language text summarization systems. *arXiv preprint arXiv:1403.5596*.

Khalid N. Elmadani, Nizar Habash, and Hanada Taha-Thomure. 2025. A large and balanced corpus for fine-grained Arabic readability assessment. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 16376–16400, Vienna, Austria. Association for Computational Linguistics.

Ramy Eskander, Nizar Habash, and Owen Rambow. 2013. Automatic extraction of morphological lexicons from morphologically annotated corpora. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1032–1043, Seattle, Washington, USA. Association for Computational Linguistics.

Andrea Gesmundo and Tanja Samardzic. 2012. Lemmatisation as a tagging task. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 368–372.

Nizar Habash, Mona Diab, and Owen Rambow. 2012a. Conventional orthography for dialectal Arabic. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 711–718, Istanbul, Turkey. European Language Resources Association (ELRA).

Nizar Habash, Ramy Eskander, and Abdelati Hawwari. 2012b. A morphological analyzer for Egyptian Arabic. In *Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology*, pages 1–9, Montréal, Canada. Association for Computational Linguistics.

Nizar Habash and David Palfreyman. 2022. ZAEBUC: An annotated Arabic-English bilingual writer corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 79–88, Marseille, France. European Language Resources Association.

Nizar Habash and Owen Rambow. 2007. Arabic diacritization through full morphological tagging. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 53–56, Rochester, New York. Association for Computational Linguistics.

Nizar Habash, Abdelhadi Soudi, and Tim Buckwalter. 2007. On Arabic Transliteration. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*, pages 15–22. Springer, Netherlands.

Anton Karl Ingason, Sigrún Helgadóttir, Hrafn Loftsson, and Eiríkur Rögnvaldsson. 2008. A mixed method lemmatization algorithm using a hierarchy of linguistic identities (holi). In *Advances in natural language processing: 6th international conference, GoTAL 2008 Gothenburg, Sweden, August 25-27, 2008 Proceedings*, pages 205–216. Springer.

Svanhvít Lilja Ingólfsdóttir, Hrafn Loftsson, Jón Friðrik Daðason, and Kristín Bjarnadóttir. 2019. Nefnir: A high accuracy lemmatizer for Icelandic. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 310–315, Turku, Finland. Linköping University Electronic Press.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 92–104, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Go Inoue, Salam Khalifa, and Nizar Habash. 2022. Morphosyntactic tagging with pre-trained language mod-

els for Arabic and its dialects. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1708–1719, Dublin, Ireland. Association for Computational Linguistics.

Mustafa Jarrar, Diyam Akra, and Tymaa Hammouda. 2024. Alma: Fast lemmatizer and pos tagger for arabic. *Procedia Computer Science*, 244:378–387.

Mustafa Jarrar, Nizar Habash, Faeq Alrimawi, Diyam Akra, and Nasser Zalmout. 2016. Curras: an annotated corpus for the Palestinian Arabic dialect. *Language Resources and Evaluation*, pages 1–31.

Mustafa Jarrar and Tymaa Hasanain Hammouda. 2024. Qabas: An open-source Arabic lexicographic database. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13363–13370, Torino, Italia. ELRA and ICCL.

Mustafa Jarrar, Fadi A Zaraket, Tymaa Hammouda, Daanish Masood Alavi, and Martin Wählisch. 2023. Lisan: Yemeni, iraqi, libyan, and sudanese arabic dialect corpora with morphological annotations. In *2023 20th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA)*, pages 1–7. IEEE.

Bart Jongejan and Hercules Dalianis. 2009. Automatic training of lemmatization rules that handle morphological changes in pre-, in-and suffixes alike. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 145–153.

Salam Khalifa, Nizar Habash, Fadhl Eryani, Ossama Obeid, Dana Abdulrahim, and Meera Al Kaabi. 2018. A morphologically annotated corpus of Emirati Arabic. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Salam Khalifa, Nasser Zalmout, and Nizar Habash. 2020. Morphological analysis and disambiguation for Gulf Arabic: The interplay between resources and methods. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3895–3904, Marseille, France. European Language Resources Association.

Daniel Kondratyuk, Tomáš Gavenčiak, Milan Straka, and Jan Hajič. 2018. LemmaTag: Jointly tagging and lemmatizing for morphologically rich languages with BRNNs. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4921–4928, Brussels, Belgium. Association for Computational Linguistics.

Juan Liberato, Bashar Alhafni, Muhamed Khalil, and Nizar Habash. 2024. Strategies for Arabic readability modeling. In *Proceedings of the Second Arabic Natural Language Processing Conference*, pages 55–66, Bangkok, Thailand. Association for Computational Linguistics.

Mohamed Maamouri, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *Proceedings of the International Conference on Arabic Language Resources and Tools*, pages 102–109, Cairo, Egypt.

Mohamed Maamouri, Ann Bies, Seth Kulick, Michael Ciul, Nizar Habash, and Ramy Eskander. 2014. Developing an Egyptian Arabic Treebank: Impact of dialectal morphology on annotation and tool development. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Mohamed Maamouri, Ann Bies, Seth Kulick, Dalila Tabessi, and Sondos Krouna. 2012. Egyptian Arabic Treebank DF Parts 1-8 V2.0 - LDC catalog numbers LDC2012E93, LDC2012E98, LDC2012E89, LDC2012E99, LDC2012E107, LDC2012E125, LDC2013E12, LDC2013E21.

GUIDO MINNEN, JOHN CARROLL, and DARREN PEARCE. 2001. Applied morphological processing of english. *Natural Language Engineering*, 7(3):207–223.

Hamdy Mubarak. 2018. Build fast and accurate lemmatization for Arabic. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Thomas Müller, Ryan Cotterell, Alexander Fraser, and Hinrich Schütze. 2015. Joint lemmatization and morphological tagging with lemming. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2268–2274, Lisbon, Portugal. Association for Computational Linguistics.

Amal Nayouf, Tymaa Hammouda, Mustafa Jarrar, Fadi Zaraket, and Mohamad-Bassam Kurdy. 2023. Nâbra: Syrian Arabic dialects with morphological annotations. In *Proceedings of ArabicNLP 2023*, pages 12–23, Singapore (Hybrid). Association for Computational Linguistics.

Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash. 2020. CAMeL tools: An open source python toolkit for Arabic natural language processing. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 7022–7032, Marseille, France. European Language Resources Association.

Arfath Pasha, Mohamed Al-Badrashiny, Ahmed El Kholy, Ramy Eskander, Mona Diab, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In *In Proceedings of LREC*.

Ryan Roth, Owen Rambow, Nizar Habash, Mona Diab, and Cynthia Rudin. 2008. Arabic morphological tagging, diacritization, and lemmatization using lexeme models and feature ranking. In *Proceedings of ACL-08: HLT, Short Papers*, pages 117–120, Columbus, Ohio. Association for Computational Linguistics.

Mostafa Saeed and Nizar Habash. 2025. Lemmatization as a classification task: Results from Arabic across

multiple genres. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Suzhou, China.

Aleksi Sahala. 2024. Neural lemmatization and postagging models for coptic, demotic and earlier egyptian. In *Proceedings of the 1st Workshop on Machine Learning for Ancient Languages (ML4AL 2024)*, pages 87–97.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, pages 83–91, Berlin, Germany. Association for Computational Linguistics.

Mustafa Yaseen, Mohammed Attia, Bente Maegaard, Khalid Choukri, Niklas Paulsson, Salah Haamid, Steven Krauwer, Chomicha Bendahman, Hanne Fersøe, Mohsen A Rashwan, et al. 2006. Building annotated written and spoken arabic lrs in nemlar project. In *LREC*, pages 533–538. Citeseer.

Yin-Lai Yeong, Tien-Ping Tan, and Siti Khaotijah Mohammad. 2016. Using dictionary and lemmatizer to improve low resource english-malay statistical machine translation system. *Procedia Computer Science*, 81:243–249.

Nasser Zalmout and Nizar Habash. 2020a. Joint diacritization, lemmatization, normalization, and fine-grained morphological tagging. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8297–8307, Online. Association for Computational Linguistics.

Nasser Zalmout and Nizar Habash. 2020b. Utilizing subword entities in character-level sequence-to-sequence lemmatization models. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4676–4682, Barcelona, Spain (Online). International Committee on Computational Linguistics.

A License

We list below the licenses of the data and tools used in this work, all of which are employed in accordance with their intended use.

- Arabic Treebank Parts 1-3 (LDC2010T13, LDC2011T09, LDC2010T08) (Maamouri et al., 2004): LDC User Agreement for Non-Members.
- Egyptian Arabic Treebank Parts 1-8 (LDC2012E93, LDC2012E98, LDC2012E89, LDC2012E99, LDC2012E107, LDC2012E125, LDC2013E12, LDC2013E21) (Maamouri et al., 2012, 2014): LDC User Agreement for Non-Members.
- BAREC Corpus (Elmadani et al., 2025): Creative Commons Attribution-NonCommercial-ShareAlike 4.0
- Curras Corpus (Jarrar et al., 2016): Creative Commons Attribution 4.0 International license.
- Gumar Annotated Corpus (Khalifa et al., 2018): NYU Abu Dhabi Non-commercial, research-only license.
- NEMLAR Corpus (Yaseen et al., 2006): Non Commercial Use - ELRA END USER
- Quran Corpus (Dukes and Habash, 2010): GNU General Public License
- WikiNews Corpus (Mubarak, 2018): Creative Commons Attribution 4.0 License
- ZAEBUC Corpus (Habash and Palfreyman, 2022): Creative Commons Attribution-NonCommercial-ShareAlike 4.0
- CAMeL Tools (Obeid et al., 2020) and CAMeLBERT (Inoue et al., 2021): MIT License.