Enriching the NArabizi Treebank: A Multifaceted Approach to Supporting an Under-Resourced Language

Arij Riabi **Menel Mahamdi** **Djamé Seddah**
Inria, Paris
{firstname,lastname}@inria.fr

Abstract

In this paper we address the scarcity of annotated data for NArabizi, a Romanized form of North African Arabic used mostly on social media, which poses challenges for Natural Language Processing (NLP). We introduce an enriched version of NArabizi Treebank (Seddah et al., 2020) with three main contributions: the addition of two novel annotation layers (named entity recognition and offensive language detection) and a re-annotation of the tokenization, morpho-syntactic and syntactic layers that ensure annotation consistency. Our experimental results, using different tokenization schemes, showcase the value of our contributions and highlight the impact of working with non-gold tokenization for NER and dependency parsing. To facilitate future research, we make these annotations publicly available. Our enhanced NArabizi Treebank paves the way for creating sophisticated language models and NLP tools for this under-represented language.

1 Introduction

Despite the abundance of rich and diverse dialects worldwide, each possessing distinctive features and characteristics, many of these dialects still lack the necessary resources and support to enable their speakers to access modern technologies in their own language (Joshi et al., 2020). Therefore, it is imperative to undertake endeavors aimed at creating annotated corpora, developing language models, and establishing dictionaries and grammars for low-resource dialects. These efforts are crucial for the preservation and advancement of these dynamic languages, which encapsulate unique cultures, histories, and experiences within their respective communities.

One notable example of such an effort is the Masakhane community, which is dedicated to enhancing natural language processing (NLP) research for African languages through significant initiatives such as MasakhaNER (Adelani et al., 2021). Similar efforts are ongoing for Indonesian languages (Cahyawijaya et al., 2022).

In addition, a long-standing and somewhat unrelated initiative known as the Universal Dependencies project (Nivre et al., 2020) originally aimed to provide a standardized set of syntactic guidelines for a limited number of languages turned out to become the recipient of numerous treebank initiatives for low-resource languages. These initiatives not only adopted the initial guidelines but also expanded upon them to accommodate the unique idiosyncrasies of each language.

In this work, we aim to enhance a pre-existing multi-view treebank devoted to a very low-resource language, namely the North-African Arabic dialect written in Latin script, collected from Algerian sources and denoted as the Narabizi treebank, the first available for this dialect, where Arabizi refers to both the practice of writing Arabic using the Latin alphabet and *N* for the North African dialect (Seddah et al., 2020). Made of noisy user-generated content that exhibits a high level of language variability, its annotations faced many challenges as described by the authors and contained remaining errors (Touileb and Barnes, 2021).

Our work builds on previous efforts to annotate and standardize treebank annotations for low-resource languages to enhance the quality and consistency of linguistic resources (Schluter and van Genabith, 2007; Sade et al., 2018; Türk et al., 2019; Zariquiey et al., 2022).

Following previous research, we consider the impact of refining annotation schemes on downstream tasks. Mille et al. (2012) examine how much a treebank's performance relies on its annotation scheme and whether employing a more linguistically rich scheme would decrease performance. Their findings indicate that using a fine-grained annotation for training a parser does not necessarily improve performance when parsing with a coarse-grained tagset. This observation is relevant to our study as

Proceedings of the 17th Linguistic Annotation Workshop (LAW-XVII), pages 266–278
July 13, 2023

we expect refining the treebank could enhance the parsing performance even though the inherent variability of this language, which, tied to its small size treebank, could bring a negative impact on such enhancements.

On the other hand, the experiments conducted by Schluter and van Genabith (2007) demonstrate that using a cleaner and more coherent treebank yields superior results compared to a treebank with a training set five times larger. This observation highlights the significance of high-quality dataset annotations, particularly for smaller datasets. This understanding primarily drives the goal of improving the NArabizi treebank's annotations.

In this context, we propose a heavily revised version of NArabizi treebank (Seddah et al., 2020) that includes two novel annotation layers for Named Entity Recognition (NER) and offensive language detection. One of the goals of this work is also to study the impact of non-gold tokenization on NER, a scenario almost never investigated by the community (Bareket and Tsarfaty, 2021). Our primary contributions are as follows:

- Using error mining tools, we release a new corrected version of the treebank, which leads to improved downstream task performance.
- We show that corrections made to a small size treebank of a highly variable language favorably impacts the performance of NLP models trained on it.
- We augment the treebank by adding NER annotations and offensive language detection, expanding its applicability in various NLP tasks.
- We homogenize tokenization across the dataset, analyze the impact of proper tokenization on UD tasks and NER and conduct a realistic evaluation on predicted tokenization, including NER evaluation.

The enhanced version of the Narabizi Treebank is freely available.[1]

2 Related work

NArabizi The Arabic language exhibits diglossia, where Modern Standard Arabic (MSA) is employed in formal contexts, while dialectal forms are used informally (Habash, 2010). Dialectal forms, which display significant variability across regions and predominantly exist in spoken form, lack standardized spelling when written. Many Arabic speakers employ the Latin script for transcribing their dialects online, using digits and symbols for phonemes not easily mapped to Latin letters (Seddah et al., 2020). This written form, known as Arabizi and its North African variant, NArabizi, often showcases code-switching with French and Amazigh (Amazouz et al., 2017). Textual resources for Arabizi primarily consist of noisy, user-generated content (Foster, 2010; Seddah et al., 2012; Eisenstein, 2013), complicating the creation of supervised models or collection of extensive pre-training datasets. The original NArabizi treebank (Seddah et al., 2020), contains about 1500 sentences. The sentences are randomly sampled from the romanized Algerian dialectal Arabic corpus of Cotterell et al. (2014) and from a small corpus of lyrics from Algerian dialectal Arabic songs popular among the younger generation. This treebank is manually annotated with morpho-syntactic information (parts-of-speech and morphological features), together with glosses and code-switching labels at the word level, as well as sentence-level translations to French. Moreover, this treebank also contains 36% of French tokens. Since its creation, this treebank spawned two derived versions that first added a transliteration to the Arabic script at the word level and sentiment and topic annotation at the sentence level (Touileb and Barnes, 2021). In parallel to our own corrections and annotation work[2], Touileb (2022) extended this work to include a named-entity annotation layer.

Treebanking for User-generated Content Treebanks and annotated corpora have greatly impacted NLP tools, applications, and research in general. Despite the challenges of constructing large and structurally consistent corpora, which requires considerable effort and time, many in the field considered this pursuit valuable and necessary (de Marneffe et al., 2021). However, constructing treebanks for user-generated content is more challenging due to the extensive variation in language usage and style, the prevalence of non-standard spellings and grammar, and the necessity for domain-specific annotations (Sanguinetti et al., 2022). Interest in treebanking user-generated content, such as social media posts and online forum discussions, has risen, and numerous efforts have been undertaken to create treebanks for user-generated content (Foster et al., 2011; Seddah et al., 2012; Sanguinetti

[1] https://gitlab.inria.fr/ariabi/release-narabizi-treebank

[2] Released on November 26th, 2022, the same day as the publication of (Touileb, 2022).

et al., 2018; Rehbein et al., 2019; Sanguinetti et al., 2020).

NER for Dialects and User-generated Content NER is an information extraction task that identifies and categorizes entities at the token level. It is an extensively investigated NLP task with numerous datasets and models for various languages. However, datasets for low-resource languages are rare, and NER datasets for social media platforms such as Twitter predominantly exist for English (Ritter et al., 2011; Derczynski et al., 2016, 2017).

A prominent NER dataset for *lower-than-English* resource languages is the CoNLL 2002 Shared Task dataset (Tjong Kim Sang, 2002), which provides NER annotations for four languages: Dutch, Spanish, Chinese, and Czech. Additionally, the WikiAnn dataset (Pan et al., 2017) includes NER annotations for several low-resource languages. Nevertheless, it is derived from Wikipedia content which is not well-suited for NER tasks involving user-generated content. As mentioned above, Touileb (2022) added a NER annotation for the first version of the NArabizi treebank. However, they did not address the tokenization issues inherent in the dataset and used a different annotation scheme. The following sections delve deeper into the tokenization challenges and the differences between the two datasets.

3 Extending a Low-resource Language treebank

In this section, we outline our methodology for expanding and enhancing the NArabizi treebank. We start by re-annotating tokenization, morpho-syntactic, and syntactic layers to ensure consistency, followed by detailing the annotation guidelines and procedures for NER and Offensive Language detection. We refer to the initial treebank introduced by Seddah et al. (2020) as NArabiziV1 and our extended version as NArabiziV2.

3.1 Maintaining Consistency in Treebank Annotations

We start with an extended clean-up of the NArabiziV1 formatting, which involves reinstating missing part-of-speech tags and rectifying Conllu formatting discrepancies. Then, we embark on general error mining in the lexical and syntactical annotation and correction phase. We implement this stage using semi-automated methods. We do not change the UD tagsets used in the original treebank.

Error Mining We use the UD validator Vr2.11 [3], a tool designed to assess the annotation of treebanks in UD and ensure compliance with the UD specifications. The validator is specifically employed to detect common errors, such as invalid dependency relations, incorrect part-of-speech tags, and inconsistent usage of features like tense and aspect. By leveraging the UD validator, we guarantee that our dataset is syntactically consistent and conforms to the standards established by the UD project. These changes encompass correcting cycle and projectivity issues and removing duplicates.

We also use Errator (Wisniewski, 2018), a data mining tool, to pinpoint inconsistencies in our dataset. It implements the annotation principle presented by Boyd et al. (2008), which suggests that if two identical word sequences have different annotations, one is likely erroneous.

We remove the duplicated sentences when the text field is an exact match and fix duplicated sentence identification for different sentences. We also fixed some problems with the original text, such as Arabic characters encoding and sentence boundaries.

Tokenization We address tokenization concerns to uphold consistency in the NArabizi Treebank annotations. Furthermore, we introduce targeted adjustments to resolve issues related to segmenting specific word classes, including conjunctions, interjections (e.g., “ya”), determiners, and prepositions, especially when adjacent to noun phrases. For example, we segment determiners at the initial vowel (“a” or “e”), as demonstrated in the examples “e ssalam” (“the peace”) and “e dounoub” (“the sins”). The lemma field for these terms is aligned with the French translation for the splitting (e.g., “e ssalam” ⇒ “la paix” (“the peace”)). For prepositions, we perform splitting at the first letter followed by “i” when possible, as seen in “brabi” ⇒ “b rabi” (“with my god”). We also establish rules for segmenting determiners and proper nouns. When possible, we separate prepositions at the initial letter and “i” and instituted guidelines for segmenting determiners and proper nouns. We implement these alterations for splitting using the Grew graph rewriting tool for NLP (Guillaume, 2021) to improve the consistency and quality of the treebank annotations. Additionally, we fix all the problems mentioned by Touileb and Barnes (2021) regarding the incoherence of the

[3] https://github.com/UniversalDependencies/tools/releases/tag/r2.11

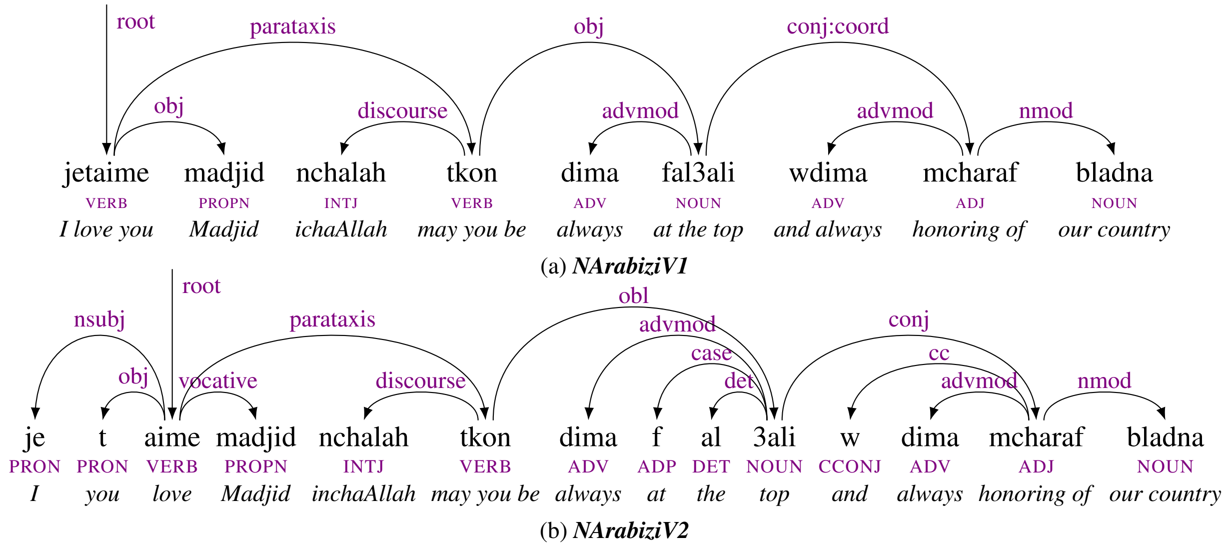

Figure 1: Illustration of an example from the NAarabizi treebank before and after the modifications.

tokenization, wrong translations, and incoherent annotations.

Translation The translation quality is also enhanced; previously, translations were not consistently carried out by Algerian speakers, resulting in local expressions and phrases being frequently misinterpreted, either in a literal manner or, at times, entirely inaccurately. This had implications for lexical and syntactical annotation. For instance, the term “skara” was initially annotated as “on purpose” but was later revised to “taunting”. Recognizing that “skara fi” represents a local expression facilitates annotation and promotes corpus harmonization.

Example In Figure 1, we illustrate a parse tree before and after applying several corrections. Tokenization errors in French were rectified (“jetaime” ⇒ “je t aime”), and Arabic prepositions, articles, and conjunctions were separated from the nouns or adverbs they were attached to (“fal3ali” ⇒ “f al 3ali”, “wdima” ⇒ “w dima”). We also correct some dependency relations: the previous “obj” relation between the verb “aimer” and the proper noun “madjid” was altered to “vocative”.

Interesting Properties The corpus displays several interesting linguistic features, including *parataxis*, *goeswith*, and dislocated structures, characteristic of oral productions and user-generated content. A deeper examination of the root/parataxis ratio and the average parataxis per tree in the corpus, which contains 2066 parataxis for 1287 sentences, shows that the corpus exhibits a high level of juxtaposed clauses resulting from the absence of punctuation. Given the initial data sources (web forums), it is likely that these end of sentences markers were initially present as carriage returns.

As pointed out by Seddah et al. (2020) the corpus also exhibits a high level of spelling variation, reflecting the speakers’ diversity in terms of geography and accents. Furthermore, analyzing the number of sentences without a verb and the average number of verbs per sentence shows that NArabizi speakers tend to favor nominalization, as seen in the abundance of ellipses (e.g., “rabbi m3ak” which translates in English to “God bless you”).

3.2 Annotation Methodology for NER and Offensive Language Detection

Named Entity Recognition Our NER annotation guidelines are based on the revised tokenization of the NArabizi treebank, which ensures consistency between token-level annotations, an essential aspect of multi-task learning. We use the Inception tool (Klie et al., 2018) for our manual annotation by two native speakers, adhering to the IOB2 Scheme (Tjong Kim Sang and Veenstra, 1999). Each word is labeled with a tag indicating whether it is at the beginning, inside, or outside of a named entity. In case of disagreement between annotators, the multiple annotations were subsequently discussed until agreement was reached, and one annotation was selected to be retained. We extend the FTB NER

(Sagot et al., 2012) French treebank annotations. Our annotation contains the following NE types: PER for real or fictional persons, ORG for organizations, LOC for locations, COMP for companies, and OTH for brands, events, and products.

In cases of ambiguity between products and companies, we adhere to the decision made in the FTB dataset. For person names, we exclude grammatical or contextual words from the mention. We annotate football teams as organizations, and we annotate mentions of ”Allah” or ”Rabi” as PERderivA. The PERderiv annotation is applied to groups of individuals who originate from or share the same location. Country names are consistently labeled as locations, irrespective of the context. TV channels and ambiguous brand names are annotated as companies, while religious groups are not designated entities. The names of football stadiums are classified under OTH, whereas journal names are identified as organizations.

Table 1 presents the distribution of entities, with a similar distribution observed across both the development and test splits. The most frequent entity type is PERderivA, while the least frequent is COMP.

Type	train	dev	test	Total
PER	371	61	47	479
LOC	358	58	50	466
ORG	200	23	28	251
COMP	6	5	3	14
OTH	44	6	7	57
PERderiv	96	14	13	123
PERderivA	386	57	66	509
Total	1461	224	214	1899

Table 1: Named entity type distribution across train, dev, and test splits.

Type	train	dev	test
nb sentences	1003	139	145
nb tokens	15522	2124	2118
nb unique tokens	6652	1284	1327

Table 2: Statistics of the deduplicated corpus across train, dev, and test splits. The train-dev intersection contains 549 tokens, the train-test intersection contains 551 tokens, and the dev-test intersection contains 266 tokens.

Table 2 displays the number of unique words which can provide information about the language used in the corpus. The fact that the count of unique tokens constitutes nearly half of the total tokens suggests that the language used in the corpus is complex and diverse, with a wide range of vocabulary and expressions. This can make it more challenging for NER algorithms to accurately identify and classify named entities in the corpus.

Touileb (2022) recently introduced NERDz, a version of the NArabizi treebank annotated for NER. As our dataset's annotation labels differ from theirs, we establish a mapping between the two annotation schemes to enable comparisons (cf. see Table 10 in the appendix A). Our schemes also differ in named entities' scope, as we split contracted forms, ours only cover the nominal phrase parts. Regarding nouns, such as “*bled*”, which means *country*, some are annotated as entity GPE in NERDz, which is not the case in our dataset. Also, the names of stadiums are annotated as LOC in NERDz while they are considered OTH in our dataset. Similarly, for “*equipe nationale*”, which means *national team* is annotated ORG in NERDz, while we do not consider it as an entity, following the FTB NER's guidelines. Added to annotator divergences, this may explain the differences in the count of the entities.

Offensive Language Classification The annotation process for offensive language classification was conducted manually by three annotators with diverse backgrounds. The annotators consisted of two females and one male, each bringing unique expertise to the task. One female annotator is a Ph.D. student in NLP, the other is a Ph.D. student in political sciences, and the male annotator is an engineer with in-depth knowledge of North African football, a prominent topic in the dataset.

The annotators were asked to annotate every sentence as offensive (OFF) or non-offensive (NOT-OFF). Offensive posts included any form of unacceptable language, targeted offense (veiled or direct), insults, threats, profane language, and swear words. To maintain objectivity and minimize potential bias, the annotators were not granted access to the other annotators' work and were not allowed to discuss their annotations with one another. This approach ensured the independence of their judgments, allowing for a more reliable evaluation of the offensive language classification process. For the offensive annotation, the two female annotators did not usually agree with the male annotator as they have different backgrounds and hence different opinions about football-related sentences. The final label is determined through a majority voting process. Additionally, we calculate the average pair-wise Cohen's κ (Cohen, 1960) to highlight how hard this task was. The average κ value is 0.54, indicating a moderate agreement between annotators, common in sentence level annotation for annotators with different backgrounds and topic familiarity (Bobicev and Sokolova, 2017). This disagreement likely stems from the interpretation of terms that can be considered offensive or non-offensive depending on either the dialect or context.

Table 3 presents the distribution of non-offensive and offensive language instances. The dataset features an imbalance between non-offensive and offensive classes, with non-offensive samples being considerably more frequent in each split.

Split	Non-Offensive	Offensive
Train	804	199
Dev	86	53
Test	118	27

Table 3: Offensive language detection distributions across train, dev, and test splits.

4 Dataset Evaluation

We evaluate the NarabiziV2 dataset on UD parsing tasks and NER using standard transfer learning architectures on which we vary the pre-trained language model and the tokenization scenario.

New NArabizi CharacterBert Model Following Riabi et al. (2021), we train a CharacterBERT (El Boukkouri et al., 2020) model, a character-based BERT variant, on a NArabizi new filtered corpus. The authors demonstrate that CharacterBERT achieves significant results when dealing with noisy data while being extremely data efficient.

We improve the initial pre-training dataset used by Riabi et al. (2021) by more stringently filtering non-NArabizi examples from the 99k instances provided by Seddah et al. (2020), as well as incorporating new samples from the CTAB corpus (Amara et al., 2021) and 12k comments extracted from various Facebook and forum posts, mostly in the Tunisian dialect taken from different datasets listed by Younes et al. (2020). This results in a 111k sentence corpus. To exclude non-NArabizi content, we first use a language detection tool (Nakatani, 2010) with a 0.9 confidence threshold to eliminate text in French, English, Hindi, Indonesian, and Russian, which are commonly found in mixed Arabizi data. Following the filtering process, a bootstrap sampling method is adopted to randomly select a subset of the remaining text for manual annotation. This annotated text is then used to train an SVM classifier for NArabizi detection. The final dataset, containing 91k annotated text instances after deduplication, focuses on North African Arabizi text. We make this corpus publicly available.

Sub-word Models We also evaluate the performance of subword-based language models, monolingual and multilingual. For the multilingual subword-based language model, we use mBERT, the multilingual version of BERT (Devlin et al., 2018). It is trained on data from Wikipedia in 104 different languages, including French and Arabic. Muller et al. (2020) demonstrated that such a model could be transferred to NArabizi to some degree. Finally, our monolingual model is DziriBERT (Abdaoui et al., 2021), a monolingual BERT model trained on 1.2M tweets from major and highly-populated Algerian cities scrapped using a set of popular keywords in the Algerian spoken dialect in both Arabic and Latin scripts.

5 Results

5.1 New Results for UD

For our updated version of the treebank, we present results for models trained and tested on NArabiziV2, as shown in Table 4 and highlighted by a red box. These results represent the new state-of-the-art performance for the treebank, and we report findings for three previously used models. The DziriBERT model exhibits the best performance; however, CharacterBERT delivers competitive results while being trained on a mere 7.5% of the data used for training DziriBERT. This observation is consistent with the conclusions drawn by Riabi et al. (2021).

In order to assess the influence of the implemented corrections, we use NArabiziV1 and eliminate duplicate sentences [4]. For this comparison, we focused on the DziriBERT model's performance when trained on either NArabiziV1 or NArabiziV2 and tested on NArabiziV2, as denoted by the blue highlights in Table 4. Training on NArabiziV2 enhances the average scores for UPOS, UAS, and LAS by 3.5 points, illustrating the favorable outcomes of the refinements introduced in the NArabiziV2 dataset. This observation is further substan-

[4] To use the prior version with an equivalent number of sentences, format errors must be rectified (earlier experiments with these sentences excluded them).

Model	Train \ Test	*NArabiziV1*			*NArabiziV2*		
		UPOS	UAS	LAS	UPOS	UAS	LAS
mBERT	*NArabiziV1*	77.42 ± 1.52	68.91 ± 0.65	56.19 ± 0.86	74.59 ± 1.42	66.01 ± 0.47	53.19 ± 0.87
DziriBERT		83.57 ± 0.92	73.97 ± 0.72	62.04 ± 0.54	80.19 ± 0.82	70.28 ± 0.83	58.63 ± 0.78
CharacterBERT		76.19 ± 2.48	68.78 ± 0.36	55.14 ± 0.38	73.01 ± 2.05	66.10 ± 0.48	52.41 ± 0.50
mBERT	*NArabiziV2*	74.48 ± 0.95	66.03 ± 0.35	52.82 ± 0.66	79.65 ± 0.90	70.56 ± 0.32	58.08 ± 0.76
DziriBERT		78.75 ± 1.29	70.51 ± 0.43	57.51 ± 0.67	83.10 ± 1.60	74.26 ± 0.27	62.66 ± 0.52
CharacterBERT		72.24 ± 2.62	65.74 ± 0.24	51.86 ± 0.51	76.34 ± 2.68	69.84 ± 0.27	56.27 ± 0.54

Table 4: Results for UD on test set, DEV set is used for validation (with gold tokenization) (We report average of F1 scores over 5 seeds with the standard deviation)

tiated by examining the performance of CharacterBERT and mBERT, reinforcing the validity of the noted improvements.

A comparative analysis of the results for models trained and tested on NArabiziV1, denoted by the blue box, and those for models trained and tested on NArabiziV2, denoted by the red box, reveals that NArabiziV2 generally yields superior evaluation scores. This observation underlines the impact of the treebank's consistency on the overall performance of the models. When we test on NArabiziV1, the model trained on NArabiziV1 gets better results than the model trained on NArabiziV2. The modifications in tokenization can explain this drop in performance.

5.2 Results for NER and Offensive Language Detection

NER Table 5 presents the results for NER[6]. The CharacterBERT model achieves the highest F1 scores for `LOC` and `OTH` categories, as well as the best performance for `PERderiv` and `PERderivA`. On the other hand, the DziriBERT model outperforms the other models in the `ORG` and `PER` categories. It is important to note that the performance varies significantly across the different categories, reflecting the diverse challenges posed by each entity type. For instance, some categories contain named entities with variations of the same word, such as "Allah"/"Alah"/"Elah", which translates into God for `PERderivA`. Since CharacterBERT uses character-level information, it is more robust to noise, which explains the high performances for those entities.

Offensive Language Detection The imbalance between non-offensive and offensive instances is challenging during the models' training and evaluation. For example, we fail to train mBERT as it only predicts non-offensive labels corresponding to the majority class. This can also be explained by how hard the distinction between offensive and non-offensive content is without context and external knowledge, as explained before. This also raises the question of how relevant is the backgrounds of the annotators for the offensive detection dataset (Basile et al., 2020; Uma et al., 2021; Almanea and Poesio, 2022).

6 Discussion

6.1 Impact of the Pre-training Corpus

In Appendix A, we present the results of all our experiments using the CharacterBERT model trained by Riabi et al. (2021). We observe a heterogeneous improvement in performance, with predominantly better outcomes for our CharacterBERT. We hypothesize that the impact of filtering the training data may not be overly beneficial, possibly due to some smoothing during the training process. Both models' final training data sizes are comparable: 99k for CharacterBERT (Riabi et al., 2021) and 91k for our CharacterBERT. Nevertheless, we believe this new corpus can be a valuable resource for this language.

6.2 Impact of Tokenization

In this section, we investigate the tokenization influence on the enhanced NArabizi Treebank, with a particular emphasis on the homogenization of the tokenization [7] and its subsequent impact on our tasks. We also evaluate the models in a realistic scenario where gold tokenization is unavailable. We use the UDPipe tokenizer (Straka et al., 2016) that employs a Gated Linear Units (GRUs) (Cho

[6]We use Seqeval (Nakayama, 2018) classification report.

[7]We follow the terminology of UD where a tokenizer performs token segmentation (i.e. source tokens).

Model	LOC	ORG	PER	OTH	PERderiv	PERderivA	macro avg
mBERT	82.93 $\pm$ 4.02	66.17 $\pm$ 6.61	61.84 $\pm$ 3.56	25.56 $\pm$ 14.64	57.98 $\pm$ 11.30	95.62 $\pm$ 1.24	65.02 $\pm$ 1.24
DziriBERT	85.84 $\pm$ 3.43	**73.67** $\pm$ 4.03	**73.42** $\pm$ 3.52	26.27 $\pm$ 4.23	57.47 $\pm$ 6.62	94.98 $\pm$ 1.39	68.61 $\pm$ 1.39
CharacterBERT	**87.98** $\pm$ 1.77	70.16 $\pm$ 3.63	69.35 $\pm$ 3.01	**31.27** $\pm$ 9.30	**64.19** $\pm$ 7.03	**96.13** $\pm$ 0.70	**69.85** $\pm$ 0.70

Table 5: NER average of F1 scores over 5 seeds with the standard deviation with gold tokenization[5].

Model	Off	Non-Off	macro avg
mBERT	0.00 $\pm$ 0.00	**89.73** $\pm$ 0.00	44.87 $\pm$ 0.00
DziriBERT	**36.77** $\pm$ 10.88	**84.78** $\pm$ 2.58	60.78 $\pm$ 6.21
CharacterBERT	24.58 $\pm$ 7.44	80.21 $\pm$ 3.66	52.39 $\pm$ 3.18

Table 6: Offensive language detection F1 scores, *off* for offensive and *Non-Off* for non offensive

et al., 2014) artificial neural network for the identification of token and sentence boundaries in plain text. It processes fixed-length segments of Unicode characters and assigns each character to one of three classes: token boundary follows, sentence boundary follows, or no boundary. The tokenizer is trained using the Adam stochastic optimization method, employing randomly shuffled input sentences to ensure effective tokenization across various NLP tasks.

Tokenizer	Prec	Recall	F1
Tokens	97.10 $\pm$ 0.35	95.49 $\pm$ 0.45	96.29 $\pm$ 0.39
Multiwords	79.74 $\pm$ 4.30	33.81 $\pm$ 2.87	47.35 $\pm$ 2.59
Words	92.92$\pm$ 0.65	88.06 $\pm$ 0.96	90.42$\pm$ 0.80

Table 7: Tokenization evaluation average scores over 5 folds

We conduct a 5-fold evaluation using the UDPipe tokenizer and assess its performance based on the token-level, multiword, and word-level scores. The results in Table 7 show high scores for the tokens and words F1 scores demonstrate the tokenizer's efficacy in handling various tokens and words, which shows that the tokenization for NArabizi is learnable. We also notice sub-optimal performance regarding multi-words, due to their random occurrence nature.[8].

For our following experiments, we train a tokenizer using the train and dev as held-out and tokenize the test set for evaluation. We do not predict the boundaries of the sentence.

[8]It is important to note that tokens refer to surface tokens (e.g., French "au" counts as one token), while words represent syntactic words ("au" is split into two words, "à" and "le").

Pos-tagging and Dependency Parsing Table 8 presents the results for models trained on the NArabiziV2 training set and tested on both the predicted tokenization and the previous version of tokenization with gold annotations from NArabiziV2. The outcomes for the predicted tokenization indicate that despite having a well-performing tokenizer, as demonstrated in Table 7, there is still a substantial loss in performance when compared to the gold tokenization results, highlighted by the red box in Table 4. Similarly, using the tokenization from NArabiziV1 and gold annotations from NArabiziV2 also exhibits a significant drop in performance. This observation first highlights the impact of the corrections brought to standardize the treebank tokenization and then, given the difference of performance between predicted and gold tokens, calls for the development of morphological-analysers, crucial for Arabic-based dialects, as UD tokenization is indeed a morpho-syntactic process.

Named Entity Recognition Evaluation on Non-Gold Tokenization The conventional evaluation methodology for NER typically assigns entities to distinct token positions. Nevertheless, this method proves inadequate when the token count for evaluation differs from the number of gold tokens, which is almost always the case when processing user-generated content.

As a result, we adopt the evaluation strategy devised by Bareket and Tsarfaty (2021), which associates entities with their forms instead of their indices. This approach yields F1 scores based on strict, exact matches of surface forms for entities, irrespective of the category distinctions, thereby offering a more accurate and reliable evaluation in scenarios with varying token counts. In other words, the gold and predicted NE spans must exhibit an exact match regarding their form, boundaries, and associated entity type.

Table 9 presents the NER scores, considering our three main NE categories: PER, LOC, and ORG. As expected, we observe a decline in performance when evaluating the models using predicted tokenization. The CharacterBERT model exhibits the

Model	*Predicted tokenization*			*NArabiziV1 tokenization*		
	UPOS	UAS	LAS	UPOS	UAS	LAS
mBERT	72.44 $^{\pm 0.87}$	61.40 $^{\pm 0.29}$	50.39 $^{\pm 0.64}$	75.84 $^{\pm 0.92}$	65.77 $^{\pm 0.40}$	54.15 $^{\pm 0.68}$
DziriBERT	76.27 $^{\pm 1.46}$	65.35 $^{\pm 0.39}$	55.04 $^{\pm 0.65}$	79.49 $^{\pm 1.63}$	70.04 $^{\pm 0.48}$	59.19 $^{\pm 0.70}$
CharacterBERT	70.03 $^{\pm 2.10}$	61.08 $^{\pm 0.18}$	49.13 $^{\pm 0.42}$	73.10 $^{\pm 2.33}$	65.37 $^{\pm 0.22}$	52.99 $^{\pm 0.50}$

Table 8: UD results for models trained on NArabiziV2 treebank and tested on test set with predicted tokenization and old tokenization from NArabiziV1

Model	*Gold*	*Predicted*
mBERT	71.79 $^{\pm 2.30}$	66.76 $^{\pm 1.52}$
DziriBERT	75.56 $^{\pm 2.13}$	68.89 $^{\pm 2.64}$
CharacterBERT	**76.30** $^{\pm 1.29}$	**70.54** $^{\pm 2.00}$

Table 9: Comparison of NER scores for `PER`/ `LOC`/ `ORG` entities F1 micro average on predicted tokenization and gold tokenization averaged across five seeds.

best performance on gold and predicted tokenization. Moreover, when evaluated using predicted tokenization, all models demonstrate a similar performance drop. This demonstrates that there is an important gap when evaluating using gold tokenization, which raises the question of how much the current evaluation of NER models reflects the actual model performance in a realistic setting for noisy UGC.

7 Conclusion

In this paper, we present a comprehensive study on the development and refinement of the NArabizi Treebank (Seddah et al., 2020) by improving its annotations, consistency, and tokenization, as well as providing new annotations for NER and offensive language. Our work contributes to the enhancement of the NArabizi Treebank, making it a valuable resource for research on low-resource languages and user-generated content with high variability. We explore the impact of tokenization on the refined NArabizi treebank, employing the UDPipe tokenizer for our evaluation. The results demonstrate the tokenizer's effectiveness in handling various tokens and multiword expressions. Our experiments show that training and testing on the NArabiziv2 improve the UD tasks performances. Furthermore, we show the impact of the tokenization for NER and UD tasks, and we report results using predicted tokenization for evaluation to estimate the models' performance on raw data.

Future research could emphasize expanding the NArabizi Treebank towards other dialects and examining the treebank's potential applications in various NLP tasks. Our dataset is made freely available as part of the new version of the Narabizi Treebank[9]. The next release will additionally contain a set of other sentence translations prepared by a Tunisian speaker. These translations will be interesting for cross-dialect studies, given that the Narabizi corpus is predominantly made of Algerian dialect.

Acknowledgements

We warmly thank the reviewers for their very valuable feedback. This work received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 101021607. We are grateful to Roman Castagné for his valuable feedback and proofreading and wish to gratefully acknowledge the OPAL infrastructure from Université Côte d'Azur for providing resources and support.

References

Amine Abdaoui, Mohamed Berrimi, Mourad Oussalah, and Abdelouahab Moussaoui. 2021. Dziribert: a pre-trained language model for the algerian dialect. *arXiv preprint arXiv:2109.12346*.

David Ifeoluwa Adelani, Jade Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, Stephen Mayhew, Israel Abebe Azime, Shamsuddeen H. Muhammad, Chris Chinenye Emezue, Joyce Nakatumba-Nabende, Perez Ogayo, Aremu Anuoluwapo, Catherine Gitau, Derguene Mbaye, Jesujoba Alabi, Seid Muhie Yimam, Tajuddeen Rabiu Gwadabe, Ignatius Ezeani, Rubungo Andre Niyongabo, Jonathan Mukiibi, Verrah Otiende, Iroro Orife, Davis David, Samba Ngom, Tosin Adewumi, Paul Rayson, Mofetoluwa Adeyemi, Gerald Muriuki, Emmanuel Anebi, Chiamaka Chukwuneke, Nkiruka Odu, Eric Peter Wairagala, Samuel Oyerinde, Clemencia Siro, Tobius Saul Bateesa,

[9]https://gitlab.inria.fr/ariabi/release-narabizi-treebank

Temilola Oloyede, Yvonne Wambui, Victor Akinode, Deborah Nabagereka, Maurice Katusiime, Ayodele Awokoya, Mouhamadane MBOUP, Dibora Gebreyohannes, Henok Tilaye, Kelechi Nwaike, Degaga Wolde, Abdoulaye Faye, Blessing Sibanda, Orevaoghene Ahia, Bonaventure F. P. Dossou, Kelechi Ogueji, Thierno Ibrahima DIOP, Abdoulaye Diallo, Adewale Akinfaderin, Tendai Marengereke, and Salomey Osei. 2021. MasakhaNER: Named entity recognition for African languages. *Transactions of the Association for Computational Linguistics*, 9:1116–1131.

Dina Almanea and Massimo Poesio. 2022. ArMIS - the Arabic misogyny and sexism corpus with annotator subjective disagreements. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2282–2291, Marseille, France. European Language Resources Association.

Amina Amara, Houcemeddine Turki, Mohamed Ali Hadj Taieb, Mohamed Ben Aouicha, and Kawthar Ellouze. 2021. Ctab: Corpus of tunisian arabizi. This corpus has been developed by the Data Engineering and Semantics Research Unit (DES- Unit), University of Sfax, Tunisia. It has been developed to increase the coverage of Latin Script in the NLP resources for Tunisian. It is included as a part of the Tunisian Arabic Corpus (http://www.tunisiya.org/).

Djegdjiga Amazouz, Martine Adda-Decker, and Lori Lamel. 2017. Addressing code-switching in french/algerian arabic speech. In *Interspeech 2017*, pages 62–66.

Dan Bareket and Reut Tsarfaty. 2021. Neural modeling for named entities and morphology (NEMO2). *Transactions of the Association for Computational Linguistics*, 9:909–928.

Valerio Basile et al. 2020. It's the end of the gold standard as we know it. on the impact of pre-aggregation on the evaluation of highly subjective tasks. In *CEUR WORKSHOP PROCEEDINGS*, volume 2776, pages 31–40. CEUR-WS.

Victoria Bobicev and Marina Sokolova. 2017. Inter-annotator agreement in sentiment analysis: Machine learning perspective. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 97–102, Varna, Bulgaria. INCOMA Ltd.

Adriane Boyd, Markus Dickinson, and W Detmar Meurers. 2008. On detecting errors in dependency treebanks. *Research on Language and Computation*, 6(2):113–137.

Samuel Cahyawijaya, Holy Lovenia, Alham Fikri Aji, Genta Indra Winata, Bryan Wilie, Rahmad Mahendra, Christian Wibisono, Ade Romadhony, Karissa Vincentio, Fajri Koto, Jennifer Santoso, David Moeljadi, Cahya Wirawan, Frederikus Hudi, Ivan Halim Parmonangan, Ika Alfina, Muhammad Satrio Wicaksono, Ilham Firdausi Putra, Samsul Rahmadani, Yulianti Oenang, Ali Akbar Septiandri, James Jaya, Kaustubh D. Dhole, Arie Ardiyanti Suryani, Rifki Afina Putri, Dan Su, Keith Stevens, Made Nindyatama Nityasya, Muhammad Farid Adilazuarda, Ryan Ignatius, Ryandito Diandaru, Tiezheng Yu, Vito Ghifari, Wenliang Dai, Yan Xu, Dyah Damapuspita, Cuk Tho, Ichwanul Muslim Karo Karo, Tirana Noor Fatyanosa, Ziwei Ji, Pascale Fung, Graham Neubig, Timothy Baldwin, Sebastian Ruder, Herry Sujaini, Sakriani Sakti, and Ayu Purwarianti. 2022. Nusacrowd: Open source initiative for indonesian nlp resources.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Ryan Cotterell, Adithya Renduchintala, Naomi Saphra, and Chris Callison-Burch. 2014. An Algerian Arabic-French code-switched corpus. In *Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools Workshop Programme*, page 34.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Leon Derczynski, Kalina Bontcheva, and Ian Roberts. 2016. Broad Twitter corpus: A diverse named entity recognition resource. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1169–1179, Osaka, Japan. The COLING 2016 Organizing Committee.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

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

Jacob Eisenstein. 2013. What to do about bad language on the internet. In *Proceedings of the 2013 conference of the North American Chapter of the association for computational linguistics: Human language technologies*, pages 359–369.

Hicham El Boukkouri, Olivier Ferret, Thomas Lavergne, Hiroshi Noji, Pierre Zweigenbaum, and Jun'ichi Tsujii. 2020. CharacterBERT: Reconciling ELMo and

BERT for word-level open-vocabulary representations from characters. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6903–6915, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jennifer Foster. 2010. "cba to check the spelling": Investigating parser performance on discussion forum posts. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 381–384, Los Angeles, California. Association for Computational Linguistics.

Jennifer Foster, Özlem Çetinoğlu, Joachim Wagner, Joseph Le Roux, Joakim Nivre, Deirdre Hogan, and Josef van Genabith. 2011. From news to comment: Resources and benchmarks for parsing the language of web 2.0. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 893–901, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Bruno Guillaume. 2021. Graph matching and graph rewriting: Grew tools for corpus exploration, maintenance and conversion. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 168–175.

Nizar Habash. 2010. *Introduction to Arabic Natural Language Processing*. Morgan and Claypool.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In *proceedings of the 27th international conference on computational linguistics: system demonstrations*, pages 5–9.

Simon Mille, Alicia Burga, Gabriela Ferraro, and Leo Wanner. 2012. How does the granularity of an annotation scheme influence dependency parsing performance? In *Proceedings of COLING 2012: Posters*, pages 839–852, Mumbai, India. The COLING 2012 Organizing Committee.

Benjamin Muller, Benoit Sagot, and Djamé Seddah. 2020. Can multilingual language models transfer to an unseen dialect? a case study on north african arabizi. *arXiv preprint arXiv:2005.00318*.

Shuyo Nakatani. 2010. Language detection library for java.

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Ines Rehbein, Josef Ruppenhofer, and Bich-Ngoc Do. 2019. tweeDe – a Universal Dependencies treebank for German tweets. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 100–108, Paris, France. Association for Computational Linguistics.

Arij Riabi, Benoît Sagot, and Djamé Seddah. 2021. Can character-based language models improve downstream task performances in low-resource and noisy language scenarios? In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 423–436, Online. Association for Computational Linguistics.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Shoval Sade, Amit Seker, and Reut Tsarfaty. 2018. The Hebrew Universal Dependency treebank: Past present and future. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 133–143, Brussels, Belgium. Association for Computational Linguistics.

Benoît Sagot, Marion Richard, and Rosa Stern. 2012. Annotation référentielle du corpus arboré de paris 7 en entités nommées. In *Traitement Automatique des Langues Naturelles (TALN)*, volume 2.

Manuela Sanguinetti, Cristina Bosco, Lauren Cassidy, Özlem Çetinoğlu, Alessandra Teresa Cignarella, Teresa Lynn, Ines Rehbein, Josef Ruppenhofer, Djamé Seddah, and Amir Zeldes. 2020. Treebanking user-generated content: A proposal for a unified representation in Universal Dependencies. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5240–5250, Marseille, France. European Language Resources Association.

Manuela Sanguinetti, Cristina Bosco, Lauren Cassidy, Özlem Çetinoğlu, Alessandra Teresa Cignarella, Teresa Lynn, Ines Rehbein, Josef Ruppenhofer, Djamé Seddah, and Amir Zeldes. 2022. Treebanking user-generated content: a ud based overview of guidelines, corpora and unified recommendations. *Language Resources and Evaluation*, pages 1–52.

Manuela Sanguinetti, Cristina Bosco, Alberto Lavelli, Alessandro Mazzei, Oronzo Antonelli, and Fabio Tamburini. 2018. PoSTWITA-UD: an Italian Twitter treebank in Universal Dependencies. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Natalie Schluter and Josef van Genabith. 2007. Preparing, restructuring, and augmenting a french treebank: Lexicalised parsers or coherent treebanks?

Djamé Seddah, Farah Essaidi, Amal Fethi, Matthieu Futeral, Benjamin Muller, Pedro Javier Ortiz Suárez, Benoît Sagot, and Abhishek Srivastava. 2020. Building a user-generated content North-African Arabizi treebank: Tackling hell. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1139–1150, Online. Association for Computational Linguistics.

Djamé Seddah, Benoit Sagot, Marie Candito, Virginie Mouilleron, and Vanessa Combet. 2012. The French Social Media Bank: a treebank of noisy user generated content. In *Proceedings of COLING 2012*, pages 2441–2458, Mumbai, India. The COLING 2012 Organizing Committee.

Milan Straka, Jan Hajič, and Jana Straková. 2016. UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4290–4297, Portorož, Slovenia. European Language Resources Association (ELRA).

Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Erik F. Tjong Kim Sang and Jorn Veenstra. 1999. Representing text chunks. In *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 173–179, Bergen, Norway. Association for Computational Linguistics.

Samia Touileb. 2022. Nerdz: A preliminary dataset of named entities for algerian. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, pages 95–101.

Samia Touileb and Jeremy Barnes. 2021. The interplay between language similarity and script on a novel multi-layer Algerian dialect corpus. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3700–3712, Online. Association for Computational Linguistics.

Utku Türk, Furkan Atmaca, Şaziye Betül Özateş, Abdullatif Köksal, Balkiz Ozturk Basaran, Tunga Gungor, and Arzucan Özgür. 2019. Turkish treebanking: Unifying and constructing efforts. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 166–177, Florence, Italy. Association for Computational Linguistics.

Alexandra N Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. 2021. Learning from disagreement: A survey. *Journal of Artificial Intelligence Research*, 72:1385–1470.

Guillaume Wisniewski. 2018. Errator: a tool to help detect annotation errors in the universal dependencies project. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Jihene Younes, Emna Souissi, Hadhemi Achour, and Ahmed Ferchichi. 2020. Language resources for maghrebi arabic dialects' nlp: a survey. *Language Resources and Evaluation*, 54(4):1079–1142.

Roberto Zariquiey, Claudia Alvarado, Ximena Echevarría, Luisa Gomez, Rosa Gonzales, Mariana Illescas, Sabina Oporto, Frederic Blum, Arturo Oncevay, and Javier Vera. 2022. Building an endangered language resource in the classroom: Universal Dependencies for kakataibo. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3840–3851, Marseille, France. European Language Resources Association.

A Appendix

A.1 Datasets

NERDz		Our dataset	
Entities	Count	Entities	Count
PER	467	PER	479
GPE/LOC	479	LOC	466
ORG	290	ORG/COMP	265

Table 10: Mapping of NER labels in our dataset to the Published NERDz dataset (Touileb, 2022).

A.2 Results with CharacterBERT from (Riabi et al., 2021)

Model	Train \ Test	*NArabiziV1* UPOS	*NArabiziV1* UAS	*NArabiziV1* LAS	*NArabiziV2* UPOS	*NArabiziV2* UAS	*NArabiziV2* LAS
CharacterBERT (Riabi et al., 2021)	*NArabiziV1*	75.33 ± 2.77	67.86 ± 0.95	54.40 ± 0.81	72.33 ± 2.60	65.17 ± 0.79	51.51 ± 1.05
CharacterBERT (Ours)	*NArabiziV1*	**76.19** ± 2.48	**68.78** ± 0.36	**55.14** ± 0.38	**73.01** ± 2.05	**66.10** ± 0.48	**52.41** ± 0.50
CharacterBERT (Riabi et al., 2021)	*NArabiziV2*	**72.46** ± 3.19	65.30 ± 0.50	51.84 ± 0.68	**79.65** ± 0.90	**70.56** ± 0.32	**58.08** ± 0.76
CharacterBERT (Ours)	*NArabiziV2*	72.24 ± 2.62	**65.74** ± 0.24	**51.86** ± 0.51	76.34 ± 2.68	69.84 ± 0.27	56.27 ± 0.54

Table 11: Results for UD on test set, DEV set is used for validation (with gold tokenization) (We report average of F1 scores over 5 seeds with the standard deviation)

Model	LOC	ORG	PER	OTH	PERderiv	PERderivA
CharacterBERT (Riabi et al., 2021)	86.80 ± 2.01	68.53 ± 6.09	65.36 ± 2.74	**45.16** ± 13.60	58.96 ± 10.42	95.00 ± 1.32
CharacterBERT (Ours)	**87.98** ± 1.77	**70.16** ± 3.63	**69.35** ± 3.01	31.27 ± 9.30	**64.19** ± 7.03	**96.13** ± 0.70

Table 12: NER average of F1 scores over 5 seeds with the standard deviation with gold tokenization[10].

Model	Off	Non-Off	macro avg
CharacterBERT (Riabi et al., 2021)	**36.29** ± 5.73	76.49 ± 3.81	**56.39** ± 2.95
CharacterBERT (Ours)	24.58 ± 7.44	**80.21** ± 3.66	52.39 ± 3.18

Table 13: Offensive language detection F1 scores, *off* for offensive and *Non-Off* for non offensive