TArC: Tunisian Arabish Corpus First complete release

Elisa Gugliotta, Marco Dinarelli

Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 1125–1136
Marseille, 20-25 June 2022

TArC: Tunisian Arabish Corpus
First complete release

Elisa Gugliotta, Marco Dinarelli
Laboratoire d'Informatique de Grenoble (LIG), LIDILEM, "Sapienza" University of Rome
Campus Universitaire de Saint-Martin-D'Hères, 38000 Grenoble,
Circonvallazione Tiburtina 4, 00185, Rome
{elisa.gugliotta,marco.dinarelli}@univ-grenoble-alpes.fr

Abstract

In this paper we present the final result of a project on Tunisian Arabic encoded in Arabizi, the Latin-based writing system for digital conversations. The project led to the creation of two integrated and independent resources: a corpus and a NLP tool created to annotate the former with various levels of linguistic information: word classification, transliteration, tokenization, POS-tagging, lemmatization. We discuss our choices in terms of computational and linguistic methodology and the strategies adopted to improve our results. We report on the experiments performed in order to outline our research path. Finally, we explain why we believe in the potential of these resources for both computational and linguistic researches.
Keywords: Tunisian Arabizi, Annotated Corpus, Neural Network Architecture

1. Introduction

In this paper we describe the methodology we adopted for building a dialectal Arabic Corpus from scratch. Our motivations for building a corpus from scratch are related to the utility we envisioned with its release, both from a linguistic and a NLP points of view. Along with the path identified to achieve our goal, we give the linguistic motivations and describe the computational experiments that guided the choice of our approach, leading us to the final corpus structure. The corpus is the result of a semi-automatic annotation procedure carried out using an NLP tool based on neural models, that we developed specifically to create the corpus. Our architecture produces in cascade the different levels of annotation that we decided to have in the corpus. For many reasons, mentioned below, our approach is *hybrid*. First of all, the project lies at the intersection of different research fields: Arabic dialectology, corpus linguistics and deep learning. Secondly, the texts collected in our corpus are in an Arabic dialect. Arabic dialects are notoriously under-resourced linguistic systems. In particular, the texts collected in our corpus are encoded in a script that, on the one hand represents some phonetic phenomena of Tunisian Arabic (e.g., the article assimilation, unlike the encoding in Arabic script), on the other hand is a writing system. Such script arose in a diamesically influenced context, namely digital environments. The encoding we refer to uses the Latin alphabet, as well as some numbers, for Arabic phonemes, without correspondence in the Latin script. This encoding is known as Franco-Arabic, Arabizi, Arabish *et alias*, depending on the Arabic country. We focused on the system in use for writing Tunisian Arabic.

The structure of the paper is as follows: we describe the state-of-the-art in section 2. In section 3 we explain the reasoning behind the planning of our work. We will describe our corpus building steps in section (4). In section 5 we will present the neural architecture created to annotate our corpus.[1] Finally, we will discuss the linguistic-computational methodology to improve our results in section 6. In section 6.1 we describe the procedure to add the lemma annotation level in our corpus. We conclude the paper in section 7.

2. Related Work

Being a Semitic language, Arabic has a complex inflectional and derivational morphology which lead to complex NLP challenges. Recently, there has been a significant increase in NLP research for morphological processing of both Modern Standard Arabic (MSA) and Dialectal Arabic (DA) (Gugliotta et al., 2020). This growth follows two significant independent trends: **1.** the extension of the application domains of deep learning and **2.** the rise of social media, leading to the widening of available research data (Darwish et al., 2021).[2] Among many, a relevant work is the RNN-based model for Arabic morphological disambiguation proposed by Zalmout and Habash (2017). The authors exploited the Penn Arabic Treebank (PATB) (Maamouri et al., 2004a) and employed LSTM achieving good results. The increasing availability of easily accessible DA data led to a growing interest in applying Arabic NLP (ANLP) to DA processing (Al-Sabbagh and Girju, 2012; Bouamor et al., 2014; Bouamor et al., 2018; Diab et al., 2010; El-Haj, 2020; Gadalla et al.,

[1] Available at https://gricad-gitlab.univ-grenoble-alpes.fr/dinarelm/tarc-multi-task-system/

[2] The success of word embeddings (Mikolov et al., 2013; Pennington et al., 2014) trained on unlabelled data and the resulting enhanced performance on NLP tasks has also helped the growth of interest in Arabic NLP (Al Sallab et al., 2015; Farha and Magdy, 2019; Soliman et al., 2017).

1997; Sadat et al., 2014; Salama et al., 2014; Harrat et al., 2014).

The number of DA corpora increased in the last few years, the majority of them covers the Arabic varieties of Egyptian, Gulf Arabic and the two geographical macro-areas of Levant and Maghreb. Regarding the last area, one of the most studied variety is Algerian, which, as far as we know, is only represented by three corpora: one collected from newspapers and focusing on French code-switching (Cotterell et al., 2014); one dealing with Youtube comments (Abidi et al., 2017); and one built on an English lexicon automatically translated into Algerian and designed for Sentiment Analysis (Guellil et al., 2018). There is also a treebank for Algerian, following the Universal Dependencies formalism, containing romanized user-generated contents (Seddah et al., 2020). There is a good amount of corpora addressing Tunisian Arabic (TA), however often they are not publicly available.

Among the TA corpora publicly available there are the Tunisian Dialect Corpus Interlocutor (TuDiCoI) (Graja et al., 2010; Graja et al., 2013) and the Spoken TA Corpus (STAC) (Zribi et al., 2015). The latter is morpho-syntactically annotated with a tag set based on the Tunisian-*Al-Khalil* conventions (Zribi et al., 2013b). STAC is composed of 42,388 words resulting from transcriptions of audio files from TV channels and radio station. The employed transcription convention is OTTA (Zribi et al., 2013a).[3] Another domain-specific corpus is the TA Railway Interaction Corpus (TARIC) (Masmoudi et al., 2014) built from oral conversations between staff and clients of the Tunisian railway stations.[4] There are also two parallel corpora, one of which is the Parallel Arabic DIalect Corpus (PADIC) (Meftouh et al., 2015; Meftouh et al., 2018), consisting of 6,400 sentences from six Arabic dialects, with TA among them, aligned at sentence level. A similar corpus collection was employed by Bouamor et al. (2014) for the collection of 2k Egyptian words manually translated into various dialects. The source texts are part of the Egyptian-English corpus Zbib et al. (2012). Also the MADAR parallel corpus (Bouamor et al., 2018) has been gathered by translating sentences from English and French into Arabic dialects. The source texts were collected from the Basic Traveling Expression Corpus (BTEC) (Takezawa et al., 2007). A version of MADAR in *CODA* orthography has recently been published.[5] The CODA MADAR Corpus includes 10,000 sentences, together with their original raw version. The sentences are from MADAR CORPUS-6 which covers the dialects of Beirut, Cairo, Doha, Rabat and TA (Eryani et al., 2020).

Concerning Arabizi processing, Guellil et al. (2018) show that a transliteration task is a required preprocessing stage to decrease the ambiguity of Arabizi, resulting from its lack of spelling conventions.[6] In fact, most efforts in Arabizi processing focused on automatic transliteration from Arabizi to Arabic script, such as Chalabi and Gerges (2012); Darwish (2014); Al-Badrashiny et al. (2014); Masmoudi et al. (2015); Younes et al. (2016); Younes et al. (2018); Younes et al. (2020).

Studies on other Arabizi features include Eskander et al. (2014), which focused on foreign words and automatic processing of Arabic social media texts written in Roman script. Guellil and Azouaou (2016) presented an approach for social media dialectal Arabic identification based on supervised methods, using a pre-built bilingual lexicon of 25,086 words proposed by Guellil and Faical (2017) and Azouaou and Guellil (2017). A different method is presented in Younes et al. (2018), where the authors present a sequence-to-sequence model for Tunisian Arabizi-Arabic characters transliteration (Sutskever et al., 2014).

As in the case of (Younes et al., 2020), most research on automatic processing of Arabizi involves a preliminary phase of, i) corpus collection, ii) model training and testing. However, creating corpora from scratch is a time and energy demanding practice, and the only available corpora for Tunisian Arabizi are the Electronic Tunisian Dialect (LETD), which gathers 43,222 messages in Latin script collected from the web (Younes and Souissi, 2014); the TLD and TAD which respectively include 420,897 and 160,418 words in both Latin and Arabic script (Younes et al., 2015).[7] The Tunisian Sentiment Analysis Corpus (TSAC) collects 17,060 Tunisian Facebook comments in Arabic and Latin script, and it is manually annotated with polarity (Mdhaffar et al., 2017).

3. Corpus Usefulness

3.1. From a Computational Point of View

As we can conclude from section 2, the available resources for TA and in particular for its Arabizi encoding, are not enough to adequately focus research efforts on the development of automatic TA-Arabizi processing systems. There is indeed a need for data. As for the collection of DA corpora, the problem of non-standardized encoding affects all Arabic dialects with different degrees. Whatever type of analysis or computational use of the corpus one wishes to perform, the corpus needs to follow a normalized encoding, as also demonstrated by (Guellil et al., 2018). This problem was addressed by Habash et al. (2012) by presenting the Conventional Orthography for DA (CODA). This set of guidelines was initially dedicated only to Egyptian dialect, but was later extended to other DA, such as

[3] OTTA is a TA dedicated convention for orthographic transcription, oriented to TA phonetics.

[4] TARIC is a task-oriented resource, built for Automatic Speech Recognition (ASR).

[5] Please see the section 3.1 for a definition of CODA.

[6] In this research, the primary task was the sentiment classification of an Algerian Arabizi corpus.

[7] The latter two corpora are automatically constructed from the web.

Algerian (Saadane and Habash, 2015), Tunisian (Zribi et al., 2014), Maghrebi (Turki et al., 2016) or Gulf Arabic (Khalifa et al., 2016). Finally, Habash et al. (2018) proposed *CODA Star*, a common set of orthographic rules focused on features of individual dialects so that to help in creating dialect specific conventions. CODA Star is the convention we have chosen to make our corpus compatible with other DA corpora. We have provided our corpus with a number of annotation levels that constitute useful information for both computational and linguistic purposes (section 3.2). Our work can serve as a starting point for numerous and various ANLP research projects thanks to the variety of linguistic annotations it contains and the used methodology. In fact, our corpus was created with a semi-automatic procedure, including a manually check phase (section 4.2). Our code and our data are freely available, the experiments described in section 6 are thus reproducible.[8]

The neural architecture was designed for our main goal: building the multi-level annotated Tunisian Arabish Corpus (TArC), and it could be used for extending our project in the future. Moreover, it could be possible to adapt the same tool to other DA texts, in whatever encoding they are written in, creating other annotated twin corpora covering other Arabic dialects.

3.2. From a Linguistic Point of View

As the TA lacks resources for automatic processing, it also lacks resources for its linguistic study. There are very few resources available for this purpose. Regarding TA self-learning there are a number of scientific studies that describe its feature (e.g., Gibson (2011); Baccouche (2011); Ritt-Benmimoun (2014); Mion (2017)), some grammar manuals (e.g., Abdelkader (1977); Stumme (1896); Singer (1984)), digital corpora questionable via interface, such as the *Tunisian Arabic Corpus* (McNeil, 2018) and the corpus released in the context of the *TUNICO* project (Moerth et al., 2014; Moerth et al., 2017). Some dictionaries are also available through web interfaces (e.g., that of the *TUNICO* project), or downloadable (e.g., *Le Karmous* (Abdellatif, 2010), as well as some ontologies (Karmani and Alimi, 2015; Karmani et al., 2014; Moussa et al., 2015)). However, as much as these tools are unquestionably useful in supporting research, TA still remains rather uncovered either in terms of global materials, or in terms of support tools for linguistic analyses. This is the reason why our corpus has been annotated with various linguistic information and metadata. The amount of data is certainly one of the most important elements in NLP research, but the accuracy of the data and the levels of linguistic information are no less important and are certainly necessary for linguistic research. For this reason, we chose to concentrate on a reduced amount of data in order to focus on a methodology that could meet linguistic needs.[9] In order to enable different types of research through our work, we decided to provide as many levels of information as possible on the selected texts. For instance, we chose to collect texts from different Digital Networked Writing (DNW) environments, such as forums, blogs and social networks. In this way, TArC allows analyses that correlate language and spelling adopted by users according to the media platform employed. In order to be able to carry out comparative analyses among the different 'textual genres', we decided to limit our intervention on the original data as much as possible. This means that we have not removed any element from the texts, i. e. punctuation, symbols, typos, nor all those para-textual elements typical of the DNW.[10] The subdivision of paragraphs into sentences respected textual semantics as much as possible, relying on both syntax and end-of-clause punctuation (i.e. '.', '!', '?'). All these measures motivated by linguistic reasons involved compromises with the typical NLP *modus operandi*. For example, we chose to treat the different 'textual genres' separately, in different blocks so as not to compromise the natural order of communicative exchanges or sentences in paragraphs. In doing so, however, we sacrificed the homogeneous distribution of 'textual genres' in the annotation blocks. This, on the other hand, allowed us to observe different model learning behaviors at different stages of text annotation. In this regard, in section 6, we discuss how we trace these different behaviors to the different nature of the texts processed at each annotation phase. Finally, these observations on the architecture behavior supported our linguistic analyses by confirming some structural differences between 'text genres'. Along with analyses of the different 'textual genres', our data also allow to observe some typical traits of Tunisian Arabic and of its spontaneous writing system (Gugliotta et al., in prep).[11] The data extracted from the lyrics of rap songs encoded in Arabizi allow comparison with their encoding in Arabic characters, as we have chosen songs that we know are widespread in both writing systems. Thanks to the fact that, in the data collection phase (section 4.1), we collected both the texts and their metadata, it is possible, for example, to carry out diachronic, diatopic, diastratic and generally sociolinguistic analyses on TArC. Thanks to the annotation levels provided within our corpus, all these linguistic analyses can concern the orthographic, morphological and syntactic levels.

[8] The code is available at: https://gricad-gitlab.univ-grenoble-alpes.fr/dinarelm/tarc-multi-task-system, the Corpus is available at: https://github.com/eligugliotta/tarc.

[9] However, we plan to increase the size of the corpus by exploiting TArC as a gold standard.

[10] E.g., emoticons or smileys, which in our word classification system converge in the *emotag* class (section 4.2.2).

[11] We have already carried out preliminary analyses of this type (Gugliotta et al., in prep).

4. The Corpus

4.1. Data collection

Considering the nature of the Arabizi encoding, we primarily identified three sources of digital conversations in Tunisian Arabic to collect the data we wanted to include in our corpus. These sources of written texts are conversations on social networks, forums, and texts extracted from blogs. We were also interested in extracting some musical texts for the purpose of comparison between the two writing systems, considering that for the most popular songs it is possible to find both encodings. So we also extracted some lyrics of rap music (the most popular genre among young Tunisians). Our goal was to collect texts of varying lengths, from medium to long, so that they contain as much context as possible and are rich in linguistic information. Thus, concerning social network texts, we ruled out the possibility of using Twitter and with it the possibility of automatic filtering texts by location. In order to quickly identify Tunisian texts, we had to organize our crawling based on a keyword search so as to guide the automatic online texts searching and URLs list creating. Thus, we adopted a methodological approach similar to that one outlined in Saadane et al. (2018). With the aim of building a corpus as representative as possible of the linguistic system, we considered useful to identify wide thematic categories that could represent the most common topics of online daily conversations. In this regard, two instruments with a similar thematic organisation have been employed, i.e., 'A Frequency Dictionary of Arabic' and in particular its 'Thematic Vocabulary List' (TVL) (Buckwalter and Parkinson, 2014) and the 'Loanword Typology Meaning List', which is a list of 1460 meanings (LTML) (Haspelmath and Tadmor, 2009). By aiming to prevent introducing relevant query biases, it was decided to avoid the use of category names in the query, but to generate a range of keywords (Schäfer and Bildhauer, 2013). Therefore, each category was associated with a set of keywords in Arabizi belonging to the basic Tunisian vocabulary. Three meanings for each semantic category was found to be enough to obtain a sufficient number of keywords and URLs for each category. E.g., for the category 'family', the meanings: 'child', 'marriage', 'divorce' were associated with all their TA variants, resulting in an average of 10 keywords for each macro-category.[12] After manually checking the URLs list, in order to ensure the compatibility of the identified pages with the project aims, the following stage was the automatic scraping of the selected pages. In this way, we built a corpus that covers the basic lexicon terms in a balanced way. Moreover, this methodology avoids any bias typically introduced by manual queries based on thematic keywords (Rinke et al., 2022).

Some quantative information about the extracted data, are reported in table 1.

	Sentences	*Words*	*Avg sentence len.*
Total	4,797	43,327	9.0
forum	755	11,909	15.8
social	3,162	16,056	5.1
blog	366	6,671	18.2
rap	514	8,691	16.9

Table 1: Statistics of our corpus

4.2. Semi-Automatic Annotation

In order to make the collection of TArC as fast and as easy as possible for human annotators, deep learning techniques have been employed to implement a semi-automatic annotation procedure (Gugliotta and Dinarelli, 2020). In particular the multi-task sequence-to-sequence neural architecture described in Gugliotta et al. (2020) has been used. Such a system takes one or more input as sequences, and generates one or more outputs as sequences. The number of outputs is dynamically and automatically detected by the system based on the data format. The same system has been thus used for different phases of the annotation procedure, where a different number of annotation levels was available (see below). A high level schema of the multi-task system is given in the figure 1, where the system is instantiated to generate all the annotation levels of the corpus, taking Arabizi text as input.

The iterative semi-automatic procedure adopted to collect the TArC corpus consists in splitting the data to be annotated in blocks of roughly the same size and then:

1. Annotating automatically a block of data with a model;
2. Correcting manually the automatic annotation;
3. Adding the new annotated block of data to the training data of the model;
4. Training a new model;
5. Restarting from step 1 with a new block of data.

The TArC data have been split into seven blocks of roughly 6,000 tokens, requiring thus seven steps of the procedure to annotate the whole corpus. Concerning the annotator, he is a non-native speaker with a background in Tunisian Arabic.[13] He could rely on the tools mentioned in section 3.2 and the constant feedback from native speakers.

4.2.1. First Annotation Phase

In order to facilitate correspondence with existing tools and studies concerning dialectal Arabic processing, we considered essential to provide our corpus with a normalized encoding level. Arabizi is indeed a spontaneous writing-system and does allow different encodings for a given Tunisian lexeme. In order to achieve this normalization, we decided to transliterate the texts in a conventional orthography created *ad hoc* for Arabic dialects processing, the CODA Star orthography (Habash et al., 2018). Since at first no annotated data existed, in order to start the semi-automatic procedure some data must be manually annotated to train

[12] This phase is more deeply described in Gugliotta and Dinarelli (2020).

[13] Having first specialized in Standard Arabic and having been living in Tunisia for a few years. In fact, he attended the *IBLV* and the *WALI* in Tunis, focusing on Tunisian Arabic.

Step	Train. tokens	Tasks (Accuracy) Cl	Ar	Tk	POS
Corpus: MADAR					
Step0	12,391	99.83	-	88.83	72.71
$\text{Step0}^{(*)}_{\text{complete}}$	12,391	99.58	76.77	74.83	67.59
Corpus: MADAR+TArC					
Step1	17,261 (4,870)	92.69	-	77.66	59.56
Step2	22,173 (9,780)	97.21	-	87.53	74.30
Step3	27,270 (14,870)	96.69	-	91.47	76.38
Corpus: TArC					
Step4	22,150	96.83	75.30	73.38	69.76
Step5	27,435	97.17	75.08	73.07	66.24
$\text{Step4}_{\text{smart-init}}$	22,150	95.91	76.55	74.96	72.57
$\text{Step5}_{\text{smart-init}}$	27,435	97.08	77.83	75.69	69.76
Corpus: $\text{MADAR}_{\text{Arabizi}}$+TArC					
$\text{Step4}_{\text{concat}}{}^{(*)}$	34,541 (22,150)	96.59	78.94	77.38	74.54
$\text{Step4}_{\text{reloaded}}{}^{(*)}$	34,541 (22,150)	96.38	79.72	77.88	73.69
$\text{Step6}_{\text{concat}}{}^{(*)}$	46,197 (33,806)	96.45	79.97	77.81	70.33
$\text{Step6}_{\text{concat}}{}^{(*)}$ **fix**	46,197 (33,806)	97.63	83.29	81.94	81.02
$\text{Final-Step}_{\text{concat}}{}^{(*)}$ lstm	42,895 (30,504)	98.56	82.98	81.84	82.84
$\text{Final-Step}_{\text{concat}}{}^{(*)}$ transformer	42,895 (30,504)	95.99	75.37	74.34	71.30
$\text{Final-Step}_{\text{concat}}{}^{(*)}$ input:Ar lstm	42,895 (30,504)	98.67	-	96.78	86.31
$\text{Final-Step}_{\text{concat}}{}^{(*)}$ input:Ar transformer	42,895 (30,504)	99.95	-	95.93	82.49

Table 2: Summary of results, in terms of accuracy, obtained on the TArC data at the different steps of the iterative procedure for semi-automatic annotation of the corpus. The tasks are indicated with **Cl** for classification, **Ar** for Arabic script encoding, **Tk** for tokenization, and **POS** for POS tagging. $^{(*)}$ indicates results obtained with the MADAR data translated into Arabizi.

the model from scratch. A block of the TArC corpus has been manually annotated with the Tunisian Arabic transliteration of Arabizi tokens. We must point out that Arabizi encoding, being a Latin-based script, allows a consistent employ of code-switching and transfers. In case of Tunisian Arabic, these elements are mostly coming from French language as can be seen in table 4.2.2. Considering the linguistic aims of our work, transliterating French tokens into CODA would have generated confusion in the data, namely hiding word etymology, and that would have made it difficult to perform linguistic analyses on our corpus.[14] Furthermore, the correspondence between the phonological and orthographic levels of Tunisian and European languages is necessarily asymmetric. These asymmetries would have definitely resulted in noise for an automatic transliteration model. Aware of needing a better solution (addressed in the second phase 4.2.2), we decided to work only with non-code-switched data, manually tagging the other tokens as *foreign*. In this way, our first block of data was reduced from roughly 6,000 tokens to 5,000. The same has been done on two more blocks of data. The automatic annotation accuracy at the end of the first phase was roughly 65%. For more information regarding this phase we refer to Gugliotta and Dinarelli (2020). Since the Arabizi transliteration into CODA, shown in table 4.2.2, is the most difficult and ambiguous phase due to the spontaneous nature of Arabizi (Gugliotta et al., in prep), before going on with the annotation of the other data blocks, it has been decided to implement a more comprehensive strategy. The latter consists of the second phase, and can be resumed in the following points:

1. Adding an Arabizi token classification level before the transliteration level;
2. Improving performance in the transliteration task exploiting the information included in the other annotation layers we wanted to perform.

Hence, the intuition to continue the corpus annotation, instead of through separate modules, using a multi-task architecture that could allow the different modules to benefit from shared information. This is what we refer to as the second phase, outlined in the next section.

4.2.2. Second Annotation Phase

The levels of annotation provided in our corpus are classification of tokens in three categories (*arabizi*, *foreign* and *emotag*), transliteration into Arabic conventional script (CODA), word reduction in its morphemes (tokenization), Part-of-Speech tagging and lemmatization.[15]

Regarding the procedure employed to classify our data we refer to Gugliotta et al. (2020). Concerning the other levels, these have been performed in the following way. First of all, in order to extend the amount of Tunisian Arabic data, it has been decided to exploit a morpho-syntactically well-formed corpus. Thus, 2,000 Tunisian Arabic sentences (roughly 12,400 to-

[14] For example, the hypothetical transliteration of the French insertion 'ma grossesse' in the 4.2.2 table could have been ما فروساس, where the transcription of the possessive adjective overlaps with that of the Tunisian particle (ما) as well as with the noun used to denote 'water'.

[15] The POS-tagging formalism employed includes 184 tags and it is an adaptation of the Buckwalter tag set for MSA. We followed the guidelines of the PATB (Maamouri et al., 2004b).

CODA	Tokeniz.	POS	Lemma
انا	انا	PRON_1S	هو
بعد	بعد	ADV	بعد
ma	foreign	foreign	foreign
grossesse	foreign	foreign	foreign
حوايجي	حوايجيي	NOUN+ POSS_PRON_1S	حوايج
ال	ال	DET	ال
قدم	قدم	ADJ	قديم
ال	ال	DET	ال
كلّهم	كلّهم	NOUN_QUANT+ PRON_3P	كلّ
ولّاوا	ولّاوا	PV-PVSUFF_ SUBJ:3P	ولّى
motivation	foreign	foreign	foreign

Table 3: An excerpt of TArC annotation levels of the Arabizi sentence *ena ba3d ma grossesse houayji el kdom el kollehom waleou motivation* ('after my pregnancy all my previous clothes became my motivation').

kens) from the MADAR parallel corpus (Bouamor et al., 2018) have been annotated with the annotation levels planned for our corpus, namely classification, tokenization and POS tags, applying the semi-automatic procedure described in 4.2.[16] After annotating the MADAR corpus with the mentioned levels, the semi-automatic procedure for annotating the whole TArC corpus started.[17] Results in terms of accuracy at each step are reported in table 2, and are described in details in section 6.

5. The Architecture

As previously mentioned, considering the correlation between the different annotation levels designed for the TArC corpus, we had the intuition to produce these levels with a neural Multi-Task Architecture (MTA) rather than generating them one by one with mono-task models. Indeed, a neural architecture learns to factorize information across tasks, even when employing different modules for different tasks, which are learned jointly and interdependently. Thanks to these properties, the predictions at different levels should improve their individual performance. Based on our intuition about the need to filter Arabizi data through a level of classification, we put this level as first in the MTA, followed respectively by the modules dedicated to lemmatization, transliteration in Arabic script (CODA), tokenization and POS-tagging. Each of these modules corresponds to a *Decoder* of our neural architecture in figure 1. Thus, we have the $Decoder_{cl}$ for classification task, $Decoder_{lm}$ for lemmatization, $Decoder_{ar}$ for transliteration into Arabic script, $Decoder_{tk}$ for tokenization and $Decoder_{pos}$ for PoS tagging. The input (x), consisting of the Arabizi text embedding, is converted into context-aware hidden representations by the $Encoder$. Each decoder is provided with a number of attention mechanisms equivalent to the number of previous modules (including the encoder), it receives thus as input the encoder's hidden state (h_E) together with the hidden state of each previous decoder. Each decoder returns its predicted output ($\hat{o}_i$ for $i = 1 \dots 5$), which is used to learn the corresponding task by computing a loss comparing the predicted to the expected output (i.e. $\mathcal{L}_i(o_i, \hat{o}_i)$). The whole architecture, and thus all the tasks, is learned end-to-end by computing a global loss $\mathcal{L}$ via the sum of each individual loss ($\mathcal{L} = \sum_{i=1}^{5} \mathcal{L}_i(o_i, \hat{o}_i)$), represented with the circled $+$ symbol at the upper part of the figure 1.

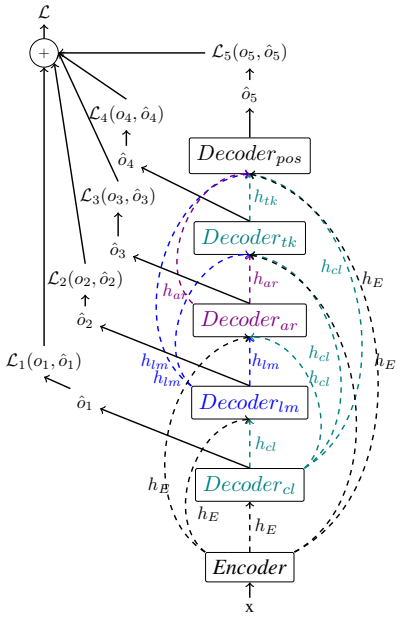

Figure 1: *A high-level schema of our architecture*

6. Experiments

In table 2 we report results for all annotation levels except lemmatization, which has been added afterwards and will be described in section 6.1. We note that results up to the block marked with *Corpus: TArC* included, have been already described in Gugliotta et al. (2020), and in Gugliotta et al. (in prep). We report them here for completeness, but we give only a short description.

In the table 2, **Train. tokens** indicates the number of tokens used for training the model at each step, in parenthesis we show the number of tokens from the TArC corpus (the others are from the MADAR corpus). Accuracy for the four annotation levels are reported on the rightmost part of the table. The line corresponding to $\text{Step0}^{(*)}_{\text{complete}}$ describes an experiment performed in a later time, with respect to the other steps reported in the first lines of the table 2. Such an experiment will be explained later on in this section.

[16] Due to time constraints, we started working on lemmatization level in recent times. The inherent procedure is described in section 6.1.

[17] We decided to use the MADAR corpus instead of the Tunisian corpora mentioned in section 2, as we considered it more compatible for our corpus-building goals and future projects, namely to extend towards other Maghrebi varieties.

Step	Train. tokens	Tasks (Accuracy)				
		Cl	Ar	Tk	POS	Lm
Corpus: MADAR$_{Arabizi}$+TArC						
Step1	17,509 (5,118)	98.61	73.61	73.15	73.92	72.22
Step2	22,272 (9,881)	97.33	79.10	77.53	78.82	75.14
Step3	27,399 (15,008)	98.31	80.69	79.81	80.38	79.00
Step4	33,069 (20,678)	99.13	81.77	80.94	82.30	80.36
Step5	38,681 (26,290)	98.72	85.79	84.89	85.44	83.69
Step6	44,792 (32,401)	97.13	85.96	84.81	83.11	84.38
Final Step global-split	42,559 (30,168)	97.14	82.34	81.45	80.95	80.48
Final Step genre-split	42,559 (30,168)	98.47	82.93	81.77	80.33	81.40
Step	**Train. tokens**	**Cl**	**Lm**	**Ar**	**Tk**	**POS**
Final Step 2xlstm	42,559 (30,168)	98.42	81.81	82.65	81.58	81.60
Final Step 3xtransformer	42,559 (30,168)	96.48	68.89	69.72	68.18	68.37
Final Step 2xlstm input:Ar	42,559 (30,168)	98.77	92.40	-	96.74	85.90
Final Step 3xtransformer input:Ar	42,559 (30,168)	96.91	83.10	-	93.43	74.09

Table 4: Summary of results, in terms of accuracy, for the semi-automatic procedure for TArC lemmatization. **Lm** stays for Lemma, the other annotation levels are like in table 2.

In order to facilitate the understanding of the different annotation steps of the procedure, we note that at step i, i blocks of TArC data are used for training the model. At step 0 thus, only MADAR is used as training data to annotate the first block of TArC data. This is the bootstrap step to start the procedure. As reported in table 2, the incremental procedure has been applied up to step 3 for annotating data with Classification, Tokenization, and POS tags using Tunisian Arabic written in CODA as input. We note that for these three annotation steps, accuracies are relatively high (at best 91.47% and 76.38% at step 3 for Tokenization and POS tagging, respectively).

In step 4 and 5 of the incremental annotation procedure, Arabizi has been used as input to the system, since only the first three data blocks were annotated with tokens transliterated in CODA. For these two steps thus, not only the system has less training data, but also an additional decoder is instantiated to predict the CODA annotation level. This means also that the model has more parameters to train. All of that translates overall into a drop in performance when using Arabizi as input to the system (accuracies are 77.83%, 75.69%, 69.76% for transliteration into CODA, Tokenization and POS tagging, respectively, at step 5).

In order to further reduce data sparsity and ambiguity, the latter especially related to predicting Arabic script from Arabizi, MADAR data have been also annotated with Arabizi encoding level. This has been performed once again with a semi-automatic procedure. The step 0 has been performed again, this time with Arabizi as input to predict the other 4 levels, allowing to compare results with the very first step 0 of the annotation procedure. Such an experiment is indicated with Step$0^{(*)}_{\text{complete}}$ in the table 2 (meaning that MADAR data have now all the annotation levels). We note that, once again predicting all the information levels from Arabizi leads to an overall drop in performances. Since MADAR is composed of morpho-syntactically well-formed text, this confirms that predicting CODA level from Arabizi is a difficult task, and the most difficult among those performed in this multi-task setting. As we explain in a forthcoming work (to appear), while concatenating the MADAR data to the TArC data provides similar performances with respect to initializing the model with one pre-trained on MADAR data only (Step4_{concat} vs. Step4_{reloaded} in the table), the former is slightly more accurate on POS tagging and doesn't require to pre-train a model, allowing to save resources. The concatenation strategy has thus been chosen for the remainder of the experiments. We refer the reader to (Gugliotta and Dinarelli, 2020; Gugliotta et al., 2020) for more details on this part of the experiments and on experimental settings.

The step 6 is the last step for annotating the data block 7 of the TArC corpus. Results on this step are similar to those of previous steps, with a small drop on POS tagging. We attribute this to the fact that the block 7 is of a different genre with respect to previous blocks.

At this point of the annotation procedure we decided to update our multi-task system adding the Transformer model (Vaswani et al., 2017). This allowed to find a weight initialization problem in the system.[18] Solving this problem not only allowed to have a working system with both LSTM and Transformer (Hochreiter and Schmidhuber, 1997; Vaswani et al., 2017), but also improved drastically the performances of the system when using LSTM. This can be seen in table 2 comparing the lines Step6_{concat} and Step6_{concat} **fix**. Despite the huge change in performances, we decided to not perform again the previous experiments. This first of all because the experiments were needed to pre-annotate data, while pre-annotation and manual correction up to this step had already been performed. Second in order to save resources, and third because even with the performance change the main message stays the same: the most difficult and ambiguous task is still the transliteration of Arabizi tokens into CODA script. This is confirmed by the fact that performance drops (rela-

[18] Basically parameters were initialized with the default of Pytorch (https://pytorch.org/docs/stable/index.html), which we replaced with the more effective *Xavier initialization* (Bengio, 2012).

tively) very little from Ar to POS prediction (83.29% to 81.02%). The remainder of the experiments has been performed with the corrected system.

Once the whole corpus has been annotated with all the annotation levels, we split the data into training, development and test splits after a random shuffle at sentence level. This has been performed by splitting separately each text genre with 70/15/15 ratios for the three splits, respectively, and then concatenating the corresponding splits of all genres. Experiments with this split are reported in table 2 with Final-Step$_{concat}$, and have been performed with both LSTM and Transformer for comparison. In addition, we performed also experiments using Tunisian in CODA script as input (input:Ar) to predict the other levels (except for Arabizi). Similar experiments have been performed for annotation steps from 1 to 3, and they could be useful for automatically annotating more data to be used in a similar strategy as *back translation* (Sennrich et al., 2016).

6.1. Lemmatization

Though the lemmatization annotation level was planned since the begin of our project, it was the last level we produced for practical reasons. Despite its unarguable usefulness, transliteration in CODA, tokenization and POS tagging were the most crucial annotation layers, in particular for linguistic analyses. However, even lemmatization represents a tool of fundamental importance both for linguistic analysis and automatic data processing (Zalmout and Habash, 2019). The lemmatization level was produced using the same semi-automatic annotation procedure used for the other levels. Again, we exploited data from the MADAR corpus, which were semi-automatically lemmatized, using a first block of manually lemmatized TArC data for bootstrapping the annotation procedure. The TArC lemmas are encoded in Arabic CODA Star orthography, also used for the transliteration level.

The results for the lemmatization procedure are reported in table 4, with the same format as previous results. We note that results obtained with LSTMs up to step 6 are substantially better than those in table 2. This is due to the use of the corrected system for all steps (see previous section), but also to the presence of the lemmas. Indeed, by comparing the step 6 in table 2 and in table 4, all performed with the corrected system, we can see that results are better when lemmas are used, confirming our intuition on the usefulness of this annotation level.

We note also that, overall, Transformers are substantially less effective than LSTMs on these data. We attribute this to the fact that while we try to keep layers of the same size with the two models, Transformers lead to larger models, roughly 32M vs. 24M parameters, which can't probably be trained effectively on our small amount of data. Additionally, the character-level data format used as input to our models (please refer to Gugliotta et al. (2020)) creates structured information on which LSTMs are notoriously more effective due to their computational power (Weiss et al., 2018; Hahn, 2020).

Finally, we compare different final steps experiments in table 4. The first two, right below *Step6*, are performed keeping the lemma as the last information level like in the annotation phase. In the first (*global-split*), randomization is performed at sentence level at whole corpus level. In the second (genre-split), randomization is performed like in the previous section. The latter leads to improvements on all levels except for POS, we keep thus this randomization strategy for the following experiments. In the bottom block of table 4 we report results from models using the lemmas as the second level of information (note the new header for the bottom block). As we already mentioned in previous sections, the lemma should allow disambiguating the Arabic script and POS prediction, thus it should be put before them in the decoder's cascade. As we can see, results are further improved on POS tagging and lemmatization (respectively 81.60 and 81.81 vs. 80.95 and 81.40). Overall results are slightly worse than the corresponding ones in table 2. While we find this surprising, we think this can be due to the fact that, while adding lemmas should improve overall results, it requires an additional decoder and thus additional parameters to train, in addition to the negative effect mistakes on the lemma level can have on transcription in CODA and tokenization levels.

7. Conclusion

In this article we presented the outcomes of a three-year project that resulted in the creation of two tools to support research on Tunisian Arabic: a corpus of Tunisian Arabic encoded in Arabizi, namely a writing system for informal digital texts, and a neural network architecture created to annotate the corpus at various levels of linguistic information. We discussed the choices we made in terms of computational and linguistic methodology and the strategies we adopted to improve our results. We also reported some of the experiments that helped us to decide our path, by optimizing the available resources. Finally, we explained the reasons why we believe in the potential of these tools for both linguistic and computational research.

8. Acknowledgements

This work was supported by the CREMA project (*Coreference REsolution into MAchine translation*) from the French National Research Agency (ANR), contract number ANR-21-CE23-0021-01.

9. Bibliographical References

Abdelkader, B. (1977). *Peace Corps English-Tunisian Arabic Dictionary*. ERIC Cleringhouse, Washington D.C.

Abdellatif, K. (2010). Dictionnaire "le Karmous" du tunisien. *Online unter: https://www. fichier-pdf. fr/2010/08/31/m14401m/, Zugriff*, 25:2017.

Abidi, K., Menacer, M. A., and Smaili, K. (2017). Calyou: A comparable spoken algerian corpus harvested from youtube. In *18th Annual Conference of the International Communication Association (Interspeech)*.

Al-Badrashiny, M., Eskander, R., Habash, N., and Rambow, O. (2014). Automatic transliteration of romanized dialectal Arabic. In *Proceedings of the eighteenth conference on computational natural language learning*, pages 30–38.

Al-Sabbagh, R. and Girju, R. (2012). Yadac: Yet another dialectal Arabic corpus. In *Proceedings of the 8th Language Resources and Evaluation Conference (LREC)*, pages 2882–2889.

Al Sallab, A., Hajj, H., Badaro, G., Baly, R., El-Hajj, W., and Shaban, K. (2015). Deep learning models for sentiment analysis in Arabic. In *Proceedings of the second workshop on Arabic natural language processing*, pages 9–17.

Azouaou, F. and Guellil, I. (2017). Alg/fr: A step by step construction of a lexicon between algerian dialect and french. In *The 31st Pacific Asia Conference on Language, Information and Computation PACLIC*, volume 31.

Baccouche, T. (2011). Tunisia. *Encyclopedia of Arabic language and linguistics*, 4:571–577.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. *CoRR*, abs/1206.5533.

Bouamor, H., Habash, N., and Oflazer, K. (2014). A Multidialectal Parallel Corpus of Arabic. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC)*, pages 1240–1245.

Bouamor, H., Habash, N., Salameh, M., Zaghouani, W., Rambow, O., Abdulrahim, D., Obeid, O., Khalifa, S., Eryani, F., Erdmann, A., et al. (2018). The madar arabic dialect corpus and lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Buckwalter, T. and Parkinson, D. (2014). *A frequency dictionary of Arabic: Core vocabulary for learners*. Routledge.

Chalabi, A. and Gerges, H. (2012). Romanized arabic transliteration. In *Proceedings of the Second Workshop on Advances in Text Input Methods*, pages 89–96.

Cotterell, R., Renduchintala, A., Saphra, N., and Callison-Burch, C. (2014). An algerian arabic-french code-switched corpus. In *Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools Workshop Programme*, page 34.

Darwish, K., Habash, N., Abbas, M., Al-Khalifa, H., Al-Natsheh, H. T., Bouamor, H., Bouzoubaa, K., Cavalli-Sforza, V., El-Beltagy, S. R., El-Hajj, W., et al. (2021). A panoramic survey of natural language processing in the arab world. *Communications of the ACM*, 64(4):72–81.

Darwish, K. (2014). Arabizi detection and conversion to Arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 217–224.

Diab, M., Habash, N., Rambow, O., Altantawy, M., and Benajiba, Y. (2010). Colaba: Arabic dialect annotation and processing. In *Lrec workshop on semitic language processing*, pages 66–74. Citeseer.

El-Haj, M. (2020). Habibi - a multi dialect multi national Arabic song lyrics corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1318–1326, Marseille, France, May. European Language Resources Association.

Eryani, F., Habash, N., Bouamor, H., and Khalifa, S. (2020). A spelling correction corpus for multiple arabic dialects. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4130–4138.

Eskander, R., Al-Badrashiny, M., Habash, N., and Rambow, O. (2014). Foreign words and the automatic processing of Arabic social media text written in roman script. In *Proceedings of The First Workshop on Computational Approaches to Code Switching*, pages 1–12.

Farha, I. A. and Magdy, W. (2019). Mazajak: An online Arabic sentiment analyser. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 192–198.

Gadalla, H., Kilany, H., Arram, H., Yacoub, A., El-Habashi, A., Shalaby, A., and McLemore, C. (1997). Callhome egyptian Arabic transcripts ldc97t19. *Web Download. Philadelphia: Linguistic Data Consortium*.

Gibson, M. (2011). Tunis arabic. *Encyclopedia of Arabic language and linguistics*, 4:563–571.

Graja, M., Jaoua, M., and Hadrich Belguith, L. (2010). Lexical study of a spoken dialogue corpus in tunisian dialect. In *Proceedings of The International Arab Conference on Information Technology, benghazi–libya*. Citeseer.

Graja, M., Jaoua, M., and Belguith, L. H. (2013). Discriminative framework for spoken tunisian dialect understanding. In *International Conference on Statistical Language and Speech Processing*, pages 102–110. Springer.

Guellil, I. and Azouaou, F. (2016). Arabic dialect identification with an unsupervised learning (based on a lexicon). application case: Algerian dialect. In *2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES)*, pages 724–731. IEEE.

Guellil, I. and Faical, A. (2017). Bilingual lexicon for algerian Arabic dialect treatment in social media. *WiNLP: Women & Underrepresented Minorities in Natural Language Processing (Co-located with ACL 2017)*.

Guellil, I., Adeel, A., Azouaou, F., and Hussain, A. (2018). Sentialg: Automated corpus annotation for algerian sentiment analysis. In *International Conference on Brain Inspired Cognitive Systems*, pages 557–567. Springer.

Gugliotta, E. and Dinarelli, M. (2020). Tarc: Incrementally and semi-automatically collecting a tunisian arabish corpus. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6279–6286.

Gugliotta, E., Dinarelli, M., and Kraif, O. (2020). Multi-task sequence prediction for Tunisian Arabizi multi-level annotation. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 178–191, Barcelona, Spain (Online), December. Association for Computational Linguistics.

Gugliotta, E., Massaro, A., Mion, G., and Dinarelli, M. (in prep.). Definiteness in tunisian arabizi: Some data from statistical approaches.

Habash, N., Diab, M., and Rambow, O. (2012). Conventional orthography for dialectal Arabic. In *Proceedings of the 8th Language Resources and Evaluation Conference (Proceedings of the 8th Language Resources and Evaluation Conference (LREC))*, pages 711–718.

Habash, N., Eryani, F., Khalifa, S., Rambow, O., Abdulrahim, D., Erdmann, A., Faraj, R., Zaghouani, W., Bouamor, H., Zalmout, N., et al. (2018). Unified guidelines and resources for arabic dialect orthography. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Hahn, M. (2020). Theoretical limitations of self-attention in neural sequence models. *Transactions of the Association for Computational Linguistics*, 8:156–171.

Harrat, S., Meftouh, K., Abbas, M., and Smaili, K. (2014). Building resources for algerian arabic dialects. In *15th Annual Conference of the International Communication Association Interspeech*.

Haspelmath, M. and Tadmor, U. (2009). The loanword typology meaning list: Electronic databases of 29 languages. *A collaborative project coordinated by the Max Planck Institute for Evolutionary Anthropology, Department of Linguistics*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November.

Karmani, B. M. N. and Alimi, A. M. (2015). Construction d'un wordnet standard pour l'arabe tunisien. In *Proceedings of Colloque pour les Étudiants Chercheurs en Traitement Automatique du Langage naturel et ses applications, Sousse, Tunisia*.

Karmani, N. B., Soussou, H., and Alimi, A. M. (2014). Building a standardized wordnet in the iso lmf for aeb language. In *Proceedings of the Seventh Global Wordnet Conference*, pages 71–77.

Khalifa, S., Habash, N., Abdulrahim, D., and Hassan, S. (2016). A large scale corpus of Gulf Arabic. *Proceedings of the 10th Language Resources and Evaluation Conference (LREC)*.

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004a). The penn arabic treebank: Building a large-scale annotated arabic corpus. In *NEMLAR conference on Arabic language resources and tools*, volume 27, pages 466–467. Cairo.

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004b). The Penn Arabic Treebank: Building a large-scale annotated Arabic corpus. In *NEMLAR conference on Arabic language resources and tools*, volume 27, pages 466–467. Cairo.

Masmoudi, A., Khmekhem, M. E., Esteve, Y., Belguith, L. H., and Habash, N. (2014). A Corpus and Phonetic Dictionary for Tunisian Arabic Speech Recognition. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC)*, pages 306–310.

Masmoudi, A., Habash, N., Ellouze, M., Estève, Y., and Belguith, L. H. (2015). Arabic transliteration of romanized tunisian dialect text: A preliminary investigation. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 608–619. Springer.

McNeil, K. (2018). Tunisian arabic corpus: Creating a written corpus of an 'unwritten'language. *Arabic Corpus Linguistics*, 30.

Mdhaffar, S., Bougares, F., Esteve, Y., and Hadrich-Belguith, L. (2017). Sentiment analysis of tunisian dialects: Linguistic ressources and experiments. In *Third Arabic Natural Language Processing Workshop (WANLP)*, pages 55–61.

Meftouh, K., Harrat, S., Jamoussi, S., Abbas, M., and Smaili, K. (2015). Machine translation experiments on padic: A parallel Arabic dialect corpus. In *The 29th Pacific Asia conference on language, information and computation*.

Meftouh, K., Harrat, S., and Smaïli, K. (2018). Padic: extension and new experiments. In *7th International Conference on Advanced Technologies ICAT*.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mion, G. (2017). À propos du futur à tunis. *Tunisian and Lybian Arabic Dialects, Common Trends, Recent Developments, Diachronic Aspects*, pages 205–217.

Moerth, K., Procházka, S., and Dallaji, I. (2014). Laying the foundations for a diachronic dictionary of tunis arabic. a first glance at an evolving new language resource. In *Proceedings of the XVI EURALEX International Congress: The User in Focus*, pages 377–387.

Moerth, K., Schopper, D., and Siam, O. (2017). Linking instead of lemmatising: Enriching the tunico corpus with the dictionary of tunis arabic. *Tunisian*

and Lybian Arabic Dialects, Common Trends, Recent Developments, Diachronic Aspects, pages 218–238.

Moussa, N. K. B., Soussou, H., and Alimi, A. M. (2015). Tunisian Arabic aeb wordnet: Current state and future extensions. In *2015 First International Conference on Arabic Computational Linguistics (ACLing)*, pages 3–8. IEEE.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Rinke, E. M., Dobbrick, T., Löb, C., Zirn, C., and Wessler, H. (2022). Expert-informed topic models for document set discovery. *Communication Methods and Measures*, 16(1):39–58.

Ritt-Benmimoun, V. (2014). The Tunsian hilāl and sulaym dialects. a preliminary comparative study. In *Alf lahğa wa lahğa, Proceedings of the 9th Aida Conference*, pages 351–360. Lit Verlag, Berlin.

Saadane, H. and Habash, N. (2015). A conventional orthography for algerian Arabic. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 69–79.

Saadane, H., Seffih, H., Fluhr, C., Choukri, K., and Semmar, N. (2018). Automatic identification of maghreb dialects using a dictionary-based approach. In *Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Sadat, F., Kazemi, F., and Farzindar, A. (2014). Automatic identification of arabic dialects in social media. In *Proceedings of the Second Workshop on Natural Language Processing for Social Media*, pages 22–27. Dublin, Ireland.

Salama, A., Bouamor, H., Mohit, B., and Oflazer, K. (2014). YouDACC: the Youtube dialectal Arabic comment corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1246–1251, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

Schäfer, R. and Bildhauer, F. (2013). Web corpus construction. *Synthesis Lectures on Human Language Technologies*, 6(4):1–145.

Seddah, D., Essaidi, F., Fethi, A., Futeral, M., Muller, B., Suárez, P. J. O., Sagot, B., and Srivastava, A. (2020). Building a user-generated content north-african arabizi treebank: Tackling hell. In *ACL 2020-58th Annual Meeting of the Association for Computational Linguistics*.

Sennrich, R., Haddow, B., and Birch, A. (2016). Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany, August. Association for Computational Linguistics.

Singer, H. R. (1984). *Grammatik der arabischen Mundart der Medina von Tunis*. Walter de Gruyter.

Soliman, A. B., Eissa, K., and El-Beltagy, S. R. (2017). Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*, 117:256–265.

Stumme, H. (1896). *Grammatik des tunisischen arabisch: nebst Glossar*. JC Hinrichs.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. *arXiv preprint arXiv:1409.3215*.

Takezawa, T., Kikui, G., Mizushima, M., and Sumita, E. (2007). Multilingual spoken language corpus development for communication research. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 12, Number 3, September 2007: Special Issue on Invited Papers from ISCSLP 2006*, pages 303–324.

Turki, H., Adel, E., Daouda, T., and Regragui, N. (2016). A conventional orthography for maghrebi arabic. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC), Portoroz, Slovenia*.

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

Weiss, G., Goldberg, Y., and Yahav, E. (2018). On the practical computational power of finite precision RNNs for language recognition. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 740–745, Melbourne, Australia, July. Association for Computational Linguistics.

Younes, J. and Souissi, E. (2014). A quantitative view of Tunsian dialect electronic writing. In *5th International Conference on Arabic Language Processing*, pages 63–72.

Younes, J., Achour, H., and Souissi, E. (2015). Constructing linguistic resources for the tunisian dialect using textual user-generated contents on the social web. In *International Conference on Web Engineering*, pages 3–14. Springer.

Younes, J., Souissi, E., and Achour, H. (2016). A hidden markov model for the automatic transliteration of romanized Tunsian dialect. In *Proceedings of the 2nd international conference on arabic computational linguistics*.

Younes, J., Souissi, E., Achour, H., and Ferchichi, A. (2018). A sequence-to-sequence based approach for the double transliteration of Tunsian dialect. *Procedia computer science*, 142:238–245.

Younes, J., Achour, H., Souissi, E., and Ferchichi, A. (2020). Romanized tunisian dialect transliteration using sequence labelling techniques. *Journal of King Saud University-Computer and Information Sciences*.

Zalmout, N. and Habash, N. (2017). Don't throw those

morphological analyzers away just yet: Neural morphological disambiguation for Arabic. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 704–713.

Zalmout, N. and Habash, N. (2019). Joint diacritization, lemmatization, normalization, and fine-grained morphological tagging. *arXiv preprint arXiv:1910.02267*.

Zbib, R., Malchiodi, E., Devlin, J., Stallard, D., Matsoukas, S., Schwartz, R., Makhoul, J., Zaidan, O., and Callison-Burch, C. (2012). Machine translation of Arabic dialects. In *Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 49–59.

Zribi, I., Graja, M., Khmekhem, M. E., Jaoua, M., and Belguith, L. H. (2013a). Orthographic transcription for spoken Tunisian Arabic. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 153–163. Springer.

Zribi, I., Khemekhem, M. E., and Belguith, L. H. (2013b). Morphological analysis of tunisian dialect. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 992–996.

Zribi, I., Boujelbane, R., Masmoudi, A., Ellouze, M., Belguith, L. H., and Habash, N. (2014). A Conventional Orthography for Tunisian Arabic. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC)*, pages 2355–2361.

Zribi, I., Ellouze, M., Belguith, L. H., and Blache, P. (2015). Spoken Tunsian Arabic corpus "STAC": transcription and annotation. *Research in computing science*, 90:123–135.