Tafsir Dataset: A Novel Multi-Task Benchmark for Named Entity Recognition and Topic Modeling in Classical Arabic Literature

Sajawel Ahmed[1,2,3], Rob van der Goot[3], Misbahur Rehman[2], Carl Kruse[2], Ömer Özsoy[2], Alexander Mehler[1], Gemma Roig[1]
[1]Faculty for Computer Science and Mathematics, Goethe University Frankfurt
[2]Faculty for Linguistics, Cultures, and Arts, Goethe University Frankfurt
[3]Department of Computer Science, IT University of Copenhagen
{sahmed}@em.uni-frankfurt.de

Abstract

Various historical languages, which used to be lingua franca of science and arts, deserve the attention of current NLP research. In this work, we take the first data-driven steps towards this research line for Classical Arabic (CA) by addressing *named entity recognition* (NER) and *topic modeling* (TM) on the example of CA literature. We manually annotate the encyclopedic work of *Tafsir Al-Tabari* with span-based *NEs*, sentence-based *topics*, and span-based *subtopics*, thus creating the *Tafsir Dataset* with over 51,000 sentences, the first large-scale multi-task benchmark for CA. Next, we analyze our newly generated dataset, which we make open-source available, with current language models (lightweight BiLSTM, transformer-based MaChAmP) along a novel *script compression method*, thereby achieving state-of-the-art performance for our target task *CA-NER*. We also show that *CA-TM* from the perspective of historical topic models, which are central to Arabic studies, is very challenging. With this interdisciplinary work, we lay the foundations for future research on automatic analysis of CA literature.

1 Introduction

All languages deserve equal technologies. Named entity recognition (NER) and topic modeling (TM) are a crucial part of various downstream tasks in natural language processing (NLP), such as Entity Linking, Relation Extraction, and ultimately Question Answering. For such tasks, many research institutes and individual scholars put their emphasis on popular, high-resource languages like English, where there is already a large amount of previous work and resources available (Rajpurkar et al., 2018; Dzendzik et al., 2021; Cambazoglu et al., 2021). This definitely accelerates the progress of the ongoing data-driven NLP. However, many historical languages, such as Ancient Egyptian, Ancient Greek, and especially Classical Arabic (CA), which used to be the lingua franca of science and arts, have been mostly neglected by the NLP community. These languages possess large volumes of historical literature (CA: e.g. *Liber Algebrae et Almucabola*, *Canon Medicinae*, *Tafsir Al-Tabari*), which were and still are to this date relevant for many communities and societies, lay their foundations and even shape their further development. In order to perform historical analysis which are relevant for our modern age, we need to let these *forgotten* low-resource languages benefit from the wave of machine learning (ML) progress, thus making historical texts accessible to modern studies and approaching ethically an egalitarian state of NLP research.

To this end, within the project *Linked Open Tafsir* (Ahmed et al., 2022), firstly, we create the *Tafsir Dataset* by annotating the CA encyclopedic books of *Tafsir Al-Tabari* on exegetical studies of law, ethics and philosophy. This is done with respect to span-based *NEs*, sentence-based *topics* and span-based *subtopics*, thereby producing over 51,000 sentences and presenting the first multi-task benchmark for CA with three independent tasks.

Rasm + I'jam + Tashkil (Vocalized Arabic)
قَالَ اَحْمَدُ لِسَارِيَةَ فِيْ مَكَّةَ: كُلُوْا وَاشْرَبُوْا هَنِيْئًا
Rasm + I'jam (Standard Arabic)
قال احمد لسارية فى مكة: كلوا واشربوا هنيئا
Rasm (Skeleton Arabic)
قال احمد لساريه فى مكه : كلوا واسربوا هسا
NER & TM Output (Skeleton Arabic)
#topic=kalam قال [احمد]PER ل[ساريه]PER فى [مكه]LOC: كلوا واسربوا هسا

Figure 1: Example for Arabic script-dependent preprocessing layers for the sentence *"Ahmed said to Saria in Mecca: eat and drink with happiness"* along NER & TM output.

Proceedings of the 29th International Conference on Computational Linguistics, pages 3753–3768
October 12–17, 2022.

Secondly, we develop a novel *script compression method* for Arabic text in order to examine its influence on the performance of neural models (see Figure 1). For this, we take the modern vocalized Arabic script and gradually transform it to its antique form of skeleton script *Rasm* from the 7th century by removing first, the vocalization marks *Tashkil* (consisting of dashes and circles), and second, the diacritic marks *I'jam* (consisting of dots), thus lowering the vocabulary size drastically by reducing the number of distinct letters from 280 (vocalized) over 28 (standard) to 16 (skeleton). From a historical critical perspective, the usage of this skeleton script is quite interesting as this was the first one to be used for documenting the text of the Quran. Thus, on a side note, by analyzing this ancient script, we shed light on the historical critical question of its readability.

Thirdly, we analyze our newly generated dataset, apply the leightweight BiLSTM (Lample et al., 2016; Ahmed and Mehler, 2018) and contrast its usage with *MaChAmp* (van der Goot et al., 2021), a toolkit for multi-task learning in NLP. This toolkit ideally fits to our multi-task benchmark, allowing us to conduct over 119 many-fold experimental setups with various Arabic pre-trained language models (LM), such as AraBERT, AraElectra, RemBERT. With these optimization steps, we produce the first major results for *CA-TM* and on top establish a state-of-the-art performance for *CA-NER* by achieving a value of up to *95.58% F1-score*.

Our work facilitates an automatic extraction of theological information so far buried in the bulk of paper manuscripts and volumes. By creating the necessary training data for tackling the task of NER and TM with various ML algorithms, we provide an open-source gold standard for the NLP community and hereby lay the foundations for future work on digitization of historical Arabic juridical and theological studies.

The remainder of the paper is organized as follows: Section 2 reviews related work, Section 3 presents the dataset, its historical source and provides details on the annotation tasks and their guidelines, Section 4 presents a sketch of the underlying methods, Section 5 reports and discusses our results, and, finally, Section 6 draw the conclusion.

2 Related Work

Not much work has been done in the field of NLP for CA as this language suffers from *resource poverty* in the ML community. For Modern Standard Arabic (MSA), there are only a handful of studies and resources open-source available. Noteworthy work specifically for MSA-NER has been done so far mainly by Benajiba et al. (2007) on *ANERCorp dataset* and by Mohit et al. (2012) on *AQMAR dataset*; both datasets along their NER models will be used as baselines here (see Table 1). Although these datasets are relatively small compared to those which are used for other languages in the community, to this date we do not have any other alternatives. For MSA-TM, again only few resources are freely available (El Kah and Zeroual, 2021), however, these are all built on modern web texts mainly from the genre of newspapers and social media. For the case of CA-TM, no prior work is known to the authors. Hence with our work, we lay the foundations for future research in this interdisciplinary field of historical NLP.

3 Tafsir Dataset: Annotation of Classical Arabic Literature

In this section, we describe the data source, the textual conversions performed to prepare the annotation task, the annotation guidelines and the annotation process itself.

3.1 Data Source: Raw Text to TEI Format

Al-Tabari *Al-Tabari*, in full *Abu Ja'far Muhammad ibn Jarir al-Tabari*, (born c. 839, Amol, Tabiristan, Iran—died 923, Baghdad, Iraq), is a religious scholar, author of enormous compendiums of early Islamic history and Quranic exegesis, who made a distinct contribution to the consolidation of Sunni thought during the 9th century. He condensed the vast wealth of exegetical and historical erudition of the preceding generations of Muslim scholars and laid the foundations for both Quranic and historical sciences. His major works were the *Exegesis of Al-Tabari* (Tafsir Al-Tabari) and the *History of Prophets and Kings*. In this study, we are focusing on his former work.

Edition of the book and TEI format Tafsir Al-Tabari has been published in various editions, the *Turki Edition* from 2001 is the most extensive and complete one, hence, this was chosen for our study. It is published in 26 volumes consisting of a total of 18,594 pages. The original text of this edition, which is vocalized, is freely available from different online sources such as the *King Saud University*, the *Shamela Software*, and from the well-known

Corpus	*Sent.*	*PER*	*LOC*	*ORG*	*TME*	*OTH*
ANERCorp-2007	5,887	3,598	*4,429*	2,231	n/a	1,115
AQMAR-2012	2,646	1,468	1,443	450	n/a	*2,474*
Tafsir-2022	**51,704**	**176,105**	**5,583**	**22,026**	**4,160**	**12,453**

Table 1: Major open-source NER Datasets for Arabic along our NER annotations in the Tafsir Dataset.

resource platform *Gawami' al-Kalim*[1], whose text is the most refined and accurate one according to a review of the linguists in our annotation team.

The raw text was transformed to the TEI format (with an adapted TEI model), which was selected due to its extensive usage in Digital Humanities (Maraoui et al., 2017). Furthermore, this format can be useful for additional data analytical inquiries (e.g. with XQuery).

Sentence splitting heuristic Sentence splitting has been addressed by various approaches (Schweter and Ahmed, 2019). However, if there is no punctuation available, it becomes challenging for many algorithms to find a stable solution. In the case of CA literature, we rarely find regular punctuation. In fact in this ancient literature, there was no concept of sentences in the modern sense. Therefore, we apply a heuristic, which first uses all possible punctuation (which are introduced by modern editing authors), then looks for some specific sense splitting words, e.g. and (*wa*), so (*fa*), then (*thumma*). With this, we achieve an average sentence length of 30 words, which proves to be useful according to our initial downstream task evaluations.

3.2 Annotation Tasks

We developed annotation guidelines for generating the Tafsir Dataset. For NER, we extended the standard task to the domain of theology. Our guidelines built on those developed for the NER dataset on German historical literature (Ahmed et al., 2019). We took the original German guideline text and adjusted it by incorporating domain-specific needs for CA. For TM, we categorized the number of topics according to the classical understanding of *tafsir studies* and its 15 fields (Al-Suyuti, 1505), and refined them further during our discussion sessions with the annotation team. The appendix shows the material which was provided to the annotation team, including the introductory example of annotations. Overall, the raw text was annotated chapter-wise by considering each verse as a single annotation task. By this scheme, we ensured that annotators had the contextual information they needed to make their interpretations.

3.2.1 Named Entities

NEs are entities that are referred to in natural language texts by proper nouns (PN) as unique individuals (e.g. Mecca, Asia, Tabari, Shia). PN are contrasted by *common names* (CN) which refer to classes of entities (e.g. city, continent, person, organization).

In our task of CA-NER, we focus on PN. However, it is not easy to differentiate between PN and CN. In the following, we provide details for each class of NE which we used to annotate our raw text (for annotation results see Table 1, for further examples of NEs see Appendix A).

Person (PER) Naming can be a complex process in classical Arab society (comparable to ancient Hebraic naming) (Almuhanna and Prunet, 2019). Full names are made of chains of single names, which can include the name of the city where the person was living. Once the full name is mentioned, short forms are usually used throughout the remainder of a text (e.g. Al-Tabari). In CA-NER, we consider all naming conventions found in the raw texts.

Location (LOC) Location names are mostly straightforward (either classical Arabic names, or names going back to ancient age of Babylonia). Sometimes, there is a ambiguity in their semantics, e.g. the word *Medina* (city) is not a PN per se, however, when it is used a short form for *Medina Al Munawwarah* ("The Enlightened City"), then it becomes a PN. Obviously, the word's meaning is highly context dependent.

Organization (ORG) We extended the modern definition of this class to the classical context of religious organizations (Jews, Christians, Muslims), their subgroups (Sunni, Shia, Ismailities), theological school of thoughts (Hanafi, Maliki, Shafi'i,

[1] https://gk.islamweb.net

Topic/Subtopic	Sent.	Span
adyan (non-Islamic relig.)	13,564	1,063
asbab (occas. of revelation)	3,086	997
fiqh (jurisprudence)	9,782	7,707
israiliyat (Judeo-Christian)	3,260	0
kalam (Islamic theology)	17,208	3,066
lugha (linguistics)	14,444	9,543
mushkilat (problem)	61	0
mutashabih (allegorical)	153	0
naskh (abrogation)	544	223
qiraat (recitation style)	1,525	2,519
sirah (prophetic biography)	1,193	215
sufism (mysticism)	7,749	881
takhsis (specification)	146	0
tikrar (repetition)	174	0
ulum (science)	2,520	823
total annotations	*75,409*	*27,037*

Table 2: Statistics for sentence-based topic and span-based subtopic annotation data.

Hanbali), tribes and clans (Hashim, Quraysh), and ethnic groups (Arabs, Greeks, Persians).

Time (TME) In the early 7th century, the moon calendar was still in its primary form, hence there was not a proper usage of numerical format like in our modern days. Therefore, dates were mostly written out in words, either only by day name, or sometimes including the month name, and rarely, the year. In CA-NER, we consider all possible variants and annotate them accordingly. Also well-known temporal entities, such as the *Day of Judgment* (Yawm Al-Din), are annotated with the tag *TME*.

Other (OTH) All NEs which did not fit into the former class were annotated with the tag *OTH*, such as name of languages (Arabic, Greek, Latin), angels (Gabriel, Michael, Raphael), and (polytheistic) deities (Al Uzza, Al Lat, Manat, Baal).

3.2.2 Topic Modeling

TM is the task of mapping (segments of) texts to a fixed set of *topics* according to a multiclass setting (Blei et al., 2003). This task is important for higher-level NLP tasks such as Semantic Search, Text Summarization and Question Answering. There is no standard number of topics, as this depends on the application domain, the desired thematic resolution and the specifics of the underlying texts. In our case of historical-exegetical tafsir studies, we determined a set of 15 sentence-based topics and span-based subtopics. Table 2 shows them along their amount of annotation data. The totals include multiple counts due to multiple annotations of the same topic. If there are lines with 0 spans and several sentences (e.g. for israiliyat), that means that only sentences have been annotated according to the 15 topics. However, no specific spans (inside the sentences) could be identified by the annotators and marked accordingly. Hence, both tasks, namely sentence-based TM and span-based TM, are displayed in Table 2, indicating that they are independent from each other.

3.3 Annotation Process

Annotation Team The annotation team consisted of 4 domain experts, who were historical linguists and orientalists by background. For NER, we let the annotators train on a smaller subset of the text (i.e., chapter 50, verse 1-22) until they reached a high inter annotators agreement (IAA) value of 97% (Cohen's kappa; (Cohen, 1960)). Thus we let them continue their annotations for the remaining volumes of text individually. For TM, we did not compute any IAA value initially, as there were only 2 domain experts available for our topics. However, we ensured a high quality of topic annotation by cross-validating and correcting them directly by the other annotator.

Tool selection & issues Selecting the right tool for our annotation task was challenging. First, CA caused many problems: It is not only a low resource-language per se; even its *right-to-left* script is low-resourced to some extent, as there are not many tools that can handle it. Second, our intention was to use the TEI standard as the target data format due to its extensive usage in Digital Humanities. Third, we required a user friendly environment as our annotators did not have any technical background. Reflecting these points, we preferred the annotation tool *Oxygen XML Editor*[2] over other candidates (such as WebAnno or BRAT). Figure 2 gives a glimpse into the annotation environment.

Data format For our final training data, we use the CoNLL format (with the BIO/IOB2 tagging scheme) and extend it for the annotation of topics and subtopics. In this adjusted 3-column format, each sentence is written vertically along its *Arabic*

[2] https://oxygenxml.com/

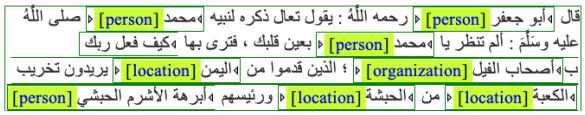

Figure 2: Screenshot of annotation working environment in *Oxygen XML Editor*.

token, *NE-tag* and *subtopic-tag*. Besides, for topics, a binary matrix structure is used at the beginning of each sentence to model all the occurrences of each *15 topics* (e.g. `# kalam: 1`, see sample excerpt in Appendix C).

After randomizing the order of the sentences, we divided the Tafsir Dataset into *train, dev, test* files according to the conventional ratio of 80:10:10 percentages. These resulting data files are used for our empirical evaluations, whose setups are described in the next section.

4 Methods

4.1 Script Compression

Arabic is a language with rich morphological variety of words. Besides, it has a distinct type of writing system (*Abjad*), which contains many layers of information developed in the course of the first centuries after the advent of Classical Arabic written tradition in the 7th century CE. The Arabic writing system is made of a basic skeleton script (*Rasm*), which 1-2 centuries later was extended to the standard Arabic script with the diacritic points (*I'jam*) to reduce the ambiguity of over 25 letters. Further 1-2 centuries later, the vocalization marks consisting of dashes and circles (*Tashkil*) were added which allowed a proper vocalized reading of theological literature.

Thus to deal with these variants, we propose the analytical setup shown in Figure 1. We use three textual variants, namely *skeleton*, *standard*, and *vocalized*, which denote the above mentioned stages the Arabic script went through during its historical development. We utilize the Python libraries *camel tools v1.3.1* (Obeid et al., 2020) and *rasmipy v0.2*[3], both applying rule-based preprocessing methods for generating our respective layers.

We hypothesize that $F_1(vocalized) < F_1(standard) \leq F_1(skeleton)$: The vocalization introduces noise, thus creating many different word embeddings of one word, which in turn lowers the overall *vocabulary coverage* of the LM for the training data. Hence, the standard/skeleton text will suite best to transformer-based neural models. Besides, current contextualized word embeddings are able to deal better with incoming textual data which has been the least preprocessed and overloaded with details (i.e. low feature engineering), which is the case for the standard/skeleton scripts. Moreover, for historical experts of the skeleton script, the ambiguity of each word decreases once longer contexts are provided, as they narrow down the possibilities of proper reading. Thus, we postulate that depending on the context, the model will be able to disambiguate the word itself and deliver an actual proper reading of the Arabic script. In Section 5, we will see that indeed our assumption has been right, and we find results which support this postulation.

4.2 Word Embeddings

We train word embeddings from scratch on large text corpora. For MSA, we take the *LeipzigArabic-2020* corpus (Goldhahn et al., 2012) with 13.55 Mio. sentences, which is already preprocessed such that it contains per line a sentence. For CA, we crawl the platform of *OpenITI* (Miller et al., 2018), containing the largest collection of online-available historical books for CA. Next, we apply our sentence splitting heuristic and tokenization from camel tools to produce a final text data file which again contains per line a sentence. With this, we get 134.17 Mio. sentences (with 17 GB of raw text data), the largest amount yet to be used for CA.

We calculate our optimized word embeddings with the extended version of the *Word2vec* algorithm (Mikolov et al., 2013), namely *Wang2vec* (Ling et al., 2015), with dimension 100, windows size 8, and min. word count 4. Although since 2019/2020 static word embeddings (which are context-independent after their training) are being replaced by their transformer-based generalization of pre-trained LMs, such as *BERT*, *XLNet*, *GPT-3* (which consider the context after their training), we still inspect the former method due to it allowing us to calculate a LM according to our chosen layer from Figure 1, and thus consider a *full analytical setup*. Furthermore, this allows us to examine how improvements can be achieved while using lightweight neural models, compared to data and computation intensive transformer-based LMs, which are on top expensive to train from scratch, and have a fixed vocabulary of subword units.

[3] https://pypi.org/project/rasmipy/

Data	Embeddings	skeleton	cov	standard	cov	vocalized	cov
ANERCorp	n-gram	n/a	n/a	55.23	Benajiba (2007)	n/a	n/a
AQMAR	SVM	n/a	n/a	69.33	Mohit (2012)	n/a	n/a
ANER	LeipzigAr	79.13	0.97	**79.14**	0.96	68.91	0.16
AQMAR	LeipzigAr	68.34	0.97	**70.93**	0.94	59.51	0.27
Tafsir	*OpenITI*	*87.13*	*0.99*	***87.41***	*0.99*	*82.97*	*0.52*

Table 3: BiLSTM results for NER on Tafsir Dataset for each layer (full setup). Coverage denotes the percentage of words from the training data that occur in the pre-trained embeddings.

4.3 Neural Models

This section provides details on the neural models which were used to examine the Tafsir Dataset along the script compression method.

4.3.1 BiLSTM

We use the neural model of BiLSTM-CRF (Lample et al., 2016; Ahmed and Mehler, 2018) with default hyperparameters for the task of CA-NER. In short, this model consists of stacked LSTM layers which receive the embedded tokens of an incoming sentence and compute a hidden representation, which in turn is used by the last CRF layer to predict the output NE-tags (i.e. PERson, LOCation, ORGanization, OTHers, O). For further details, we refer to the original papers.

4.3.2 MaChAmp

For our experiments with transformer-based LMs we use MaChAmp (van der Goot et al., 2021), a toolkit focused on multitask learning for NLP. We used v0.3 beta with default hyperparameters and compare all Arabic LMs we could find on the *Hugging Face* (Wolf et al., 2020) hub. In MaChAmp, each task has its own decoder, while the encoder (i.e. LM) is shared. We empirically saw that adding a CRF layer was beneficial (see Appendix E, Table 10), so we enabled it for NER as well as the subtopic task layer. Because the sentences can be annotated with multiple topics, we model each topic as a separate binary task. For the multi-task setups, we use an equal loss weight for all tasks, and process all tasks simultaneously.

5 Results

In this section, we present the results which are obtained while utilizing the methods and their setups described in the previous section. The evaluation of the NER predictions are performed by running the official evaluations script from the CoNLL 2003 shared task (Tjong Kim Sang and De Meulder, 2003) on the test set of the Tafsir Dataset.

5.1 BiLSTM Evaluation for CA-NER

In the single training setup, the results for our Tafsir Dataset is given which is preprocessed according to the layers outlined in Section 4.1. Most importantly, in contrast to transformer-based networks, this lightweight model allows us to not only process the training data according to our script compression method, but also the underlying LM of Word2vec (i.e. full setup). Table 3 shows the results for this setup.

First, we can see that the vocalized layer gives the lowest performance which confirms our original assumption. This performance is clearly linked to the low vocabulary coverage of this layer in respect to the pre-trained word embeddings on our selected corpora. Next, we see that the performance for standard and skeleton is relatively high. We can see that the skeleton layer continuously approaches the performance of the standard one. This behavior is stable across all three datasets and two languages (namely CA and MSA). This shows, that the skeleton layer is actually robust and almost as good as the standard one.

These results already demonstrate that our approach of script compression is noteworthy. Reducing the size of specific "redundant" letters does not lead to any significant reduction of the downstream performance. On the LM level, however, we save a relatively large amount of memory, e.g. for the Word2vec model calculated on the OpenITI corpus, we go down from 1.5 GB (standard) to 1.2 GB (skeleton) model size. Thus our first results on script compression appear to reveal a promising research direction.

MLM (standard)	skeleton	cov	standard	cov	vocalized	cov
aubmindlab/bert-base-arabertv02	85.37	0.87	**95.58**	1.00	80.26	0.85
aubmindlab/bert-large-arabertv2	85.13	0.86	**95.24**	1.00	80.14	0.84
CAMeL-Lab/bert-base-arabic-camelbert-ca	89.12	0.91	**95.43**	1.00	80.31	0.85
aubmindlab/araelectra-base-generator	84.94	0.87	**94.89**	1.00	80.06	0.85
bert-base-multilingual-cased^{+}	88.85	0.90	**95.15**	1.00	94.36	1.00
xlm-roberta-large^{+}	95.00	1.00	**95.29**	1.00	94.88	1.00
google/rembert^{+}	95.26	1.00	**95.32**	1.00	94.73	1.00

Table 4: MaChAmp results for NER on Tafsir Dataset with selected MLMs (all pre-trained on the standard layer), where for each layer (skeleton, standard, vocalized) its respective coverage (cov) is given.

5.2 MaCHAmp Evaluation

CA-NER In this section, we examine the Tafsir Dataset with various pre-trained Masked Language Models (MLM) from Hugging Face in over 119 multi-learning setups in MaChAmP. We start by utilizing all available Arabic MLMs (only pre-trained on the standard layer) and examining them along adding an optional CRF layer (see Appendix E, Table 10). Next, we cross test the Tafsir Dataset on the final selected MLMs, giving our major results in Table 4.

Although in respect to the script-dependent analysis, this is not the justified full setup, we can still get an idea what the impact of each script layer can be while fine-tuning the model. We see that the standard layer performs the best, confirming one part of our hypothesis that $F_1(vocalized) < F_1(standard)$ holds. Moreover, it is clearly demonstrated how the different layers influence the vocabulary coverage, which in turn influences the downstream performance. We can observe that in cases where $cov(vocalized) < cov(skeleton)$ holds, $F_1(vocalized) < F_1(skeleton)$ holds as well. In the opposite case, $vocalized$ is outperforming $skeleton$. Besides, we have noteworthy cases of MLMs marked with $^{+}$: For all these large multi-lingual models, their *word piece algorithm* is able to handle the vocalization by splitting it from each character, thus automatically producing the standard layer for the vocalized input. Last but not least, we can see that transformer-based models with an additional CRF layer outperform the lightweight BiLSTM thoroughly, even on the mismatched layers of $vocalized$ and $skeleton$. With this, we establish a state-of-the-art performance for CA-NER with 95.58% F-score. Thus, this comprehensive analysis allows researchers to use our dataset with the described model configurations to train a NER tagger that can confidently annotate related CA literature.

CA-TM & Multi-Task Learning In this setup, we fine-tuned the MLMs on the full Tafsir Dataset, first for each task separately, then joined within the setup of multi-task learning. Although the performance for CA-NER has been high, our results show that it is not beneficial for the task of CA-TM (see Appendix E, Table 11). However, multi-task learning is not always beneficial, as the cost of parameter sharing can become higher than the benefits of knowledge sharing. Besides, we hypothesize that TM is a very hard task on our unbalanced data which has many topics with small amount of training samples (see Table 2). A second reason that makes CA-TM a very challenging task is the fact that the topics were chosen mainly on the basis of normative considerations of a historical author: They should accompany the interpretation of religious texts in a normative way, so to speak, and are therefore of importance for the historical research of CA. TM has here the special task to reflect that the topics have been normatively pre-selected in a historical context that may not be directly available to contemporary annotators (for the purpose of generating appropriate training data). Nevertheless, these historical topic labels cannot simply be ignored, since they de facto shape research on CA.

5.2.1 Learning Curve over CA-NER Annotation Data

In order to evaluate the importance of our large-scale annotation work, we analyze the influence of the annotation data size on the final performance by plotting a learning curve over the annotation data. For each step of the size 5k sentences, we calculate the F1-score for CA-NER (on the test set) with the best observed model *bert-base-arabertv02*.

Figure 3 shows the learning curve displaying the downstream performance according to the progress of our annotation work.

Interestingly, we can see that the annotators' work has been worth it. The curve is quite steep, i.e. with every additional generation of annotation data we increased the performance steadily for our target task of CA-NER until 30k sentences. After that, the gradient starts to decrease at which the curve begins to slowly approach the max performance value of 95.58% F-score. Thus, we conclude that large amount of gold data is indeed beneficial for CA-NER, which contrasts previous findings for other low-resource languages such as Danish (Plank et al., 2020).

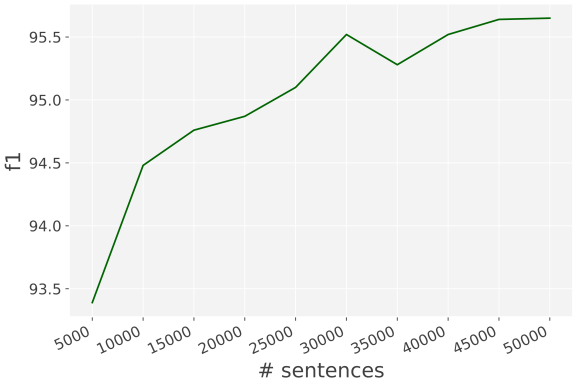

Figure 3: Learning curve over annotation data for NER (standard layer) in steps of 5k sentences.

5.3 Error Analysis for CA-NER

Our manual error analysis on the test set has revealed that the following errors exist: A majority of (1) prediction errors, where the model does not tag those NEs which are annotated by the annotators, and a minority of (2) annotation errors, where the model tags those NEs which are falsely not annotated by the annotators. However, most of the annotation errors were found in the false positives.

The Arabic language contains various words with polysemy (i.e. one word has many meanings). Especially if a word is not vocalized, and the sentence context is small, it can become difficult for the common reader to understand the underlying meaning. Then, only a domain expert can provide the precise meaning. For prediction errors, our manual error analysis has shown that the model is mistaken exactly in such cases, where there is a NE in very short sentences (e.g. 2-word *nominal sentences*). We hypothesize this is because the model has only access to one sentence, whereas the domain expert annotators have more advantages by knowing the full context via their chapter-wise view.

6 Conclusion

In this work, we presented the Tafsir Dataset, the first large-scale multi-task benchmark on NER and TM in Classical Arabic literature. We demonstrated how useful resources can be for languages which have been historically important but now forgotten by the ongoing NLP research. Besides, we also performed a first evaluation of this newly generated dataset. While doing so, we empirically saw that adding a CRF layer was beneficial to the transformer-based models, with which we ultimately established a state-of-the-art performance for CA-NER. Although TM was not the primary focus of this paper, we generated first results for CA-TM, thereby leaving room for future improvements. This refers to a scenario of TM in which topic labels were originally determined in a historical, normative, exegetical setting, whereas they need to be learned using modern NLP tools, based on their relevance to CA research. Such scenarios are likely to be increasingly encountered as more historical languages come into NLP focus. We therefore believe that our benchmark induces a new challenge for the NLP community that can lead to progress for our target low-resource language.

The Tafsir Dataset and its accompanying material are made open-source available for the research community. Furthermore, a website[4] is published which offers a comprehensive research tool in English and Arabic for accessing our dataset in a more user-friendly environment and performing various search queries on it. The web-based tool is freely available and provides over 400 filter options along the categories of our dataset. Additionally, it provides the option of graphical visualization (bubble or pie chart) of the dataset and of the query results performed on it. This digital tool makes it possible for scholars from historical and theological fields to access the dataset without any prior technical skill sets, thus allowing them to find systematically the answers to their long-lasting research questions.

On a side note, by analyzing the historical skeleton script, we shed light on a centuries-old historical critical question regarding the readability of the Rasm text: Whether the first Quranic manuscripts

[4] https://linkedopentafsir.de/

(i.e. *Uthmanic codex*) can provide a precise reading of the canonized oral text, or whether there is a large amount of ambiguity in it. Our script-dependent analysis shows that from an information retrieval perspective, the usage of the skeleton script is robust enough to deliver a similar performance compared to the usage of the standard script. We can thus conclude that if the ML model is able to deal with the skeleton script, then humans will also not face major difficulties after gaining sufficient training on the same ancient script.

Future work Our work gives indications that script compression seems to be a promising direction to reduce the amount of data and tackle the question of which resource-size actually matters (Ahmed and Mehler, 2018). In this work, for the case of Arabic we came down from 28 to 16 letters while keeping the performance stable. This shows that we do not need (1) vowels, and (2) different letters for each phoneme. In fact, just some minimum amount of *consonantal distinction* is needed. What is this amount, can we determine it exactly for each target language? Phonetic algorithms such as *Metaphone* (Philips, 1990) pose to be a first language-independent approach, be that as it may, only future work can give us the answers.

Acknowledgments

This interdisciplinary research work has been conducted as part of the project *Linked Open Tafsir*, which is funded by the *AIWG*[5], in turn supported by the *BMBF* and *Stiftung Mercator*. We acknowledge the IT University of Copenhagen HPC for the resources made available for conducting the research reported in this paper. Special thanks go to U. Brucker and the International Office (Goethe University Frankfurt) for enabling the collaboration with the IT University of Copenhagen, Prof. B. Plank and further colleagues from the IT University of Copenhagen for their critical feedback on the experimental evaluations, Prof. U. Meyer and GRADE (Goethe University Frankfurt) for funding the final publication process, and Prof. G. Hirst (University of Toronto) along Dr. J. Kindermann (Fraunhofer Institute IAIS) and Prof. R. V. Zicari (Goethe University Frankfurt) for their closing remarks on the manuscript.

[5] https://aiwg.de/

References

Sajawel Ahmed and Alexander Mehler. 2018. Resource-Size matters: Improving Neural Named Entity Recognition with Optimized Large Corpora. In *Proceedings of the 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Florida, Orlando, USA.

Sajawel Ahmed, Misbahur Rehman, Joshua Tischlik, Carl Kruse, Edin Mahmutovic, and Ömer Özsoy. 2022. Linked Open Tafsir—Rekonstruktion der Entstehungsdynamik (en) des Korans mithilfe der Netzwerkmodellierung früher islamischer Überlieferungen. In *8. Jahrestagung des Verbandes Digital Humanities im deutschsprachigen Raum (DHd)*.

Sajawel Ahmed, Manuel Stoeckel, Christine Driller, Adrian Pachzelt, and Alexander Mehler. 2019. BIOfid dataset: Publishing a German gold standard for named entity recognition in historical biodiversity literature. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 871–880, Hong Kong, China. Association for Computational Linguistics.

Jalal Al-Din Al-Suyuti. 1505. *Al-itqan Fi 'ulum Al-Qur'an (The Perfect Guide to the Sciences of the Qu'ran)*. Garnet Publishing; Bilingual edition (May 1, 2012).

Amin Almuhanna and Jean-Francois Prunet. 2019. From Classical to Modern Arab Names and Back. *Anthropological Linguistics*, 61(4):405–458. Copyright - Copyright University of Nebraska Press Winter 2019; Last updated - 2021-10-04; SubjectsTermNotLitGenreText - Arabian Peninsula; Kuwait.

Yassine Benajiba, Paolo Rosso, and José Miguel Benedíruiz. 2007. Anersys: An arabic named entity recognition system based on maximum entropy. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 143–153. Springer.

Emily M Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Armin Burkhardt. 2004. 2004. Nomen est omen? : zur Semantik der Eigennamen. In *Landesheimatbund Sachsen-Anhalt e. V. (Hrsg.): "Magdeburger Namenlandschaft" : Orts- und Personennamen der Stadt und Region Magdeburg*.

B Barla Cambazoglu, Mark Sanderson, Falk Scholer, and Bruce Croft. 2021. A review of public datasets in question answering research. In *ACM SIGIR Forum*, volume 54, pages 1–23. ACM New York, NY, USA.

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

Daria Dzendzik, Jennifer Foster, and Carl Vogel. 2021. English machine reading comprehension datasets: A survey. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8784–8804, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Anoual El Kah and Imad Zeroual. 2021. Arabic topic identification: A decade scoping review. In *E3S Web of Conferences*, volume 297, page 01058. EDP Sciences.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *LREC*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270. Association for Computational Linguistics.

Wang Ling, Chris Dyer, Alan Black, and Isabel Trancoso. 2015. Two/Too Simple Adaptations of word2vec for Syntax Problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Hajer Maraoui, Kais Haddar, and Laurent Romary. 2017. Encoding prototype of al-hadith al-shareef in tei. In *International Conference on Arabic Language Processing*, pages 217–229. Springer.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Matthew Thomas Miller, Maxim G. Romanov, and Sarah Bowen Savant. 2018. Digitizing the textual heritage of the premodern islamicate world: Principles and plans. *International Journal of Middle East Studies*, 50(1):103–109.

Behrang Mohit, Nathan Schneider, Rishav Bhowmick, Kemal Oflazer, and Noah A Smith. 2012. Recall-oriented learning of named entities in arabic wikipedia. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 162–173.

Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash. 2020. CAMeL tools: An open source python toolkit for Arabic natural language processing. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 7022–7032, Marseille, France. European Language Resources Association.

Lawrence Philips. 1990. Hanging on the metaphone. In *Computer Language*, volume 7, pages 39–43.

Barbara Plank, Kristian Nørgaard Jensen, and Rob van der Goot. 2020. DaN+: Danish nested named entities and lexical normalization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6649–6662, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Stefan Schweter and Sajawel Ahmed. 2019. Deep-EOS: General-Purpose Neural Networks for Sentence Boundary Detection. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS)*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Rob van der Goot, Ahmet Üstün, Alan Ramponi, Ibrahim Sharaf, and Barbara Plank. 2021. Massive choice, ample tasks (MaChAmp): A toolkit for multi-task learning in NLP. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 176–197, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

A Examples for annotating named entities in Tafsir Al-Tabari books

1) ذُكر أن [المشركين]ORG سألوا رسول الله صلى الله عليه وسلم عن نسب ربّ العزّة, فأنـزل الله هذه السورة جوابا لهم. (112:1)
2) حدثنا [أبو كُرَيب]PER , قال: ثنا [ابن إدريس]PER , عن [عبد الملك]PER , عن [طلحة]PER , عن [مجاهد]PER , مثله. (112:4)
3) يقول تعالى ذكره لنبيه [محمد]PER صلى الله عليه وسلم : إذا جاءك نصر الله يا [محمد]PER على قومك من [قريش]ORG , والفتح: فتح [مكة]LOC (110:2)
4) قال: ثنا [ابن ثور]PER ، عن [معمر]PER ، عن [ابن طاوس]PER ، عن أبيه، قال: ما من شيء أقرب إلى الشرك من رَقّية المجانين . (113:4)
5) حدثني [يعقوب]PER ، قال: ثنا [هشيم]PER ، قال: أخبرنا [العوّام بن عبد الجبار الجولاني]PER ، قال: قَدم رجل من [أصحاب رسول الله]ORG صلى الله عليه وسلم [الشأم]LOC ، قال: فنظر إلى دور [أهل الذمة]ORG ، وما هم فيه من العيش والنضارة، وما وُسّع عليهم في دنياهم، قال: فقال: لا أبا لك أليس من ورائهم الفلق؟ قال: قيل: وما الفلق؟ قال: بيت في [جهنم]LOC إذ فُتح هَرَ [أهْلُ النار]ORG . (113:1)
6) يقول تعالى ذكره لنبيه [محمد]PER صلى الله عليه وسلم: ألم تنظر يا [محمد]PER بعين قلبك, فترى بها كَيْفَ فَعَلَ رَبُّكَ بِ [أَصْحَابِ الْفِيلِ]ORG الذين قَدِموا من [اليمن]LOC يريدون تخريب [الكعبة]LOC من [الحبشة]LOC ورئيسهم [أبرهة الحبشيّ الأشرم]PER (105:1)
7) حدثنا [ابن المثنى]PER ، قال: ثني [عبد الأعلى]PER ، قال: ثنا [داود]PER ، عن [عكرِمة]PER ، عن [ابن عباس]PER ، قال: نـزل [القرآن]OTH كله مرة واحدة في [ليلة القدر]TME في [رمضان]TME إلى السماء الدنيا، فكان الله إذا أراد أن يحدث في [الأرض]LOC شيئًا أنـزله منه حتى جمعه. (97:1)

Figure 4: Examples for annotating named entities (i.e. PER, LOC, ORG, TME, OTH) in 7 verses from the raw text of Tafsir Al-Tabari books.

B Annotation Guidelines (German version)

Guidelines für die Named Entity Recognition. Sie bauen auf den arabisierten Guidelines von Ahmed et al. (2019) auf.

B.1 Einführung: Named Entity Recognition

Unter der Named Entity Recognition (NER) versteht man die Aufgabe, Eigennamen (named entities) in Texten zu erkennen. Technisch gesehen sind hierzu zwei Schritte notwendig. Zuerst müssen in einem laufenden Text die Token gefunden werden, die zu einem Eigennamen gehören (Named Entity Detection: NED), danach können diese Eigennamen semantischen Kategorien zugeordnet werden (Named Entity Classification). Prototypisch ist dabei der Unterschied zwischen Eigennamen und Appellativa der, dass letztere eine Gattung oder eine Klasse beschreiben, während erstere einzelne Individuen oder Sammlungen von Individuen unabhängig von gemeinsamen Eigenschaften bezeichnen (Burkhardt, 2004). Die vorliegenden Guidelines sollen es Annotatoren ermöglichen, Eigennamen in Texte aus Standard und Nichtstandard-Varietäten konsistent zu annotieren. In diesen Guidelines werden die beiden Aufgaben der NED und NEC nicht unterschieden, da die Konzentration auf Beispiele in diesem Dokument, die Trennung künstlich erzeugen müsste und nicht zu erwarten ist, dass die Resultate sich dadurch verbessern würden. In Anlehnung an die oben genannten Guidelines für Zeitungssprache werden in NoSta-D-Tafsir fünf semantische Hauptklassen für klassische arabische Texte unterschieden (Personen, Organisationen, Orte, Zeiten und Andere).

B.2 Wie finde ich eine NE?

Schritt 1: Nur volle Nominalphrasen können NEs sein. Pronomen und alle anderen Phrasen können ignoriert werden.
Schritt 2: Namen sind im Prinzip Bezeichnungen für einzigartige Einheiten, die nicht über gemeinsame Eigenschaften beschrieben werden.
Beispiel:
[Der Struppi] folgt [seinem Herrchen].
Hier gibt es zwei Nominalphrasen als Kandidaten für einen Eigennamen (NE). “Der Struppi” bezeichnet eine einzige Einheit. Es kann auch mehrere Struppis geben, aber diese haben an sich keine gemeinsamen Eigenschaften, bis auf den gemeinsamen Namen, daher handelt es sich um einen Eigennamen. *"seinem Herrchen"* bezeichnet zwar (typischerweise) auch nur eine einzige Person allerdings können wir diese nur über die Eigenschaft identifizieren, dass sie ein Herrchen ist und dass dies für Struppi zutrifft. Struppi könnte auch mehrere Herrchen haben, die alle die Eigenschaften teilen, die ein Struppi-Herrchen beinhaltet (z.B. darf Struppi streicheln, muss ihn ausführen und füttern etc.)
Schritt 3: Determinierer sind keine Teile des Namens.
Beispiel: *Der [Struppi]NE folgt seinem Herrchen.*
Schritt 4: Eigennamen können mehr als ein Token beinhalten. Beispiel:
Viele Personennamen (PER für person):
[Abu Jafar Muhammad Ibn Jarir Al Tabari]PER
Buchtitle (OTH für other):
[Jami Al Bayan Al Tawil Ay Al Quran]OTH
Schritt 5: Eigennamen können auch in einander verschachtelt sein. Beispiel:

Personennamen in Buchtiteln:
[Sunan [Abi Dawud]PER]OTH
Orte (LOC für location) in Vereinsnamen (ORG für organisation):
[Hebarium Senckenbergianum [Frankfurt]LOC]ORG
Schritt 6: Titel, Anreden und Besitzer gehören NICHT zu einem komplexen Eigennamen. Besitzer können natürlich selber Eigennamen sein. Beispiel:
Referenz auf Musiktitel:
[Vivaldis]PER [Vier Jahreszeiten]OTH
Referenz auf Personen:
Landesvorsitzende Frau Vorstandsvorsitzende Dr. [Ute Wedemeier]PER
Schritt 7: Wenn das Gesamttoken einen Eigennamen darstellt, dann wird dieser annotiert. Beispiel:
Stiftungen: *[[Böll]PER-Stiftung]ORG*
Schritt 8: Kann in einem Kontext nicht entschieden werden, ob eine NP sich als Eigennamen oder Appellativ verhält, wird es nicht als NE markiert. Beispiel:
Ortsnamen vs. -beschreibungen:
...und zogen mit ihren grossen Transparenten gestern vom [Steintor] über den [Ostertorsteinweg]LOC zum [Marktplatz].
Schritt 9: Wenn ein Name als Bezeichnung für bestimmte Gegenstände in die Sprache übergegangen ist und in seiner Nutzung nicht als NE fungiert, so wird dieser nicht annotiert. Beispiel:
[Teddybär] (NICHT PER)
[Colt] (NICHT PER)
Schritt 10: Bei Aufzählungen mit Hilfe von Bindestrichen oder Vertragen eines Teils der NE auf spätere Wörter, wird die NE so annotiert, als sei sie voll ausgeschrieben.
Beispiel:
[Frühe]OTH und [Späte Bronzezeit]OTH
[Süd-]LOC und [Nordafrika]LOC

B.3 Zu welcher semantischen Klasse gehört ein Eigenname?

Wenn der Eigenname in eine der Klassen in der Liste Faustregel zur Unterscheidung einer Klassenbezeichnung und eines Namens gehört, dann annotiere die zugehörige Klasse. Sollte die gefundene NE Rechtschreibfehler enthalten, wird sie dennoch annotiert. In Zweifelsfällen hilft auch die Tabelle NoSta-D-Tafsir-TagSet und alle Untertabellen, insbesondere die Beispiele mit dem weiter.
Jahreszahlen in ORGanisationen werden markiert.
Beispiel:
[COLING]ORG [2022]TIME
[Fussball-WM]ORG [2014]TIME
Wenn der Eigennamen in KEINE der vorhandenen Klassen passt, markiere diesen mit ***UNCLEAR***, notiere dir bitte das Beispiel und schicke uns eine E-Mail an: `X.Y@email.com`. So können wir die Guidelines sukzessiv verbessern.

B.4 Faustregel zur Unterscheidung einer Klassenbezeichnung und eines Namens:

- Elemente der fraglichen Einheit verbinden die gleichen Eigenschaften → Klasse → keine NE
- Die Elemente der fraglichen Einheit verbindet nur der Name oder Element ist Einheit bezeichnet ein spezifisches Individuum → Name → NE
- *"Paleocene"* bezeichnet spezifische Epoche → NE (TME)

NoSta-D-Tafsir-Tagset

Table 5: Kategorie 'PER-Person'

Subkategorie	Beispiele
Person	*Ibn Ahmed, Saria, Al Tabari*
Künstlernamen	*Abu Nuwas*
Charaktere	*Ali Baba*
Superhelden	*Aladin, Sindbad*

Table 6: Kategorie 'LOC-Ort'

Subkategorie	**Beispiele**
Bezirke	*Makkah Aziziyah, Schöneberg*
Sehenswürdigkeiten, Moscheen	*Mada'in Saleh, Al Masjid Al Haram*
Planeten	*Erde, Mars*
Landschafts-bezeichnungen	*Al Nefud, Königsheide*
Straßen, Plätze	*Al Tariq Al Maliki Al Farsi*
Einkaufszentren	*Suq Ukadh, Nordwestzentrum*
Berge, Seen, Flüsse	*Jabal Arafat, Al Bahr Al Ahmar, Wadi Hanifa*
Kontinente	*Asien, Europa*
Länder, Staaten	*Saudi-Arabien, Hessen, Iran*
Städte	*Mekka, Babylon*
Regionen	*Al Hijaz*
Qiraat-Orte	*Al Amsar*

Table 7: Kategorie 'ORG-Organisation'

Subkategorie	**Beispiele**
Organisationen	*Ahl Al Hadith, Sunni, Shia, Ismailiten, GEFIS, EU, Landgericht Frankfurt*
Religionsgruppen	*Juden, Christen, Muslime*
Unternehmen	*Karimis, Microsoft*
Sammelbezeichung	*Umran*
Madhahib	*Kufiyun*
Qabilah	*Quraish*
Volksgruppen	*Araber, Perser, Römer*
Universitäten	*Al-Azhar University*
Bibliotheken	*Bayt Al Hikmah*

Table 8: Kategorie 'TIME'

Subkategorie	**Beispiele**
Tag	*Freitag*
Monat	*Rabi' Al Awwal*
Jahr	*570*
dd.mm.yyyy	*12.03.0570*
Jahrhundert	*5. Jahrhundert*
Epochen	*Jahiliyyah, Paleocene*

Table 9: Kategorie 'OTH-Andere'

Subkategorie	**Beispiele**
Buch-, Filmtitel etc.	*Sahih Al Bukhari, Faust*
Währungen	*Dinar, Dirham, Euro*
Sprachen	*Arabisch, Deutsch, Latein*
Buchtitel mittels Autor	*Helbig et al., ([[Helbig]PER et al.]OTH)*
Gottheiten	*Al Uzza, Al Lat, Manat, Ba'al, Nasr, Suwa', Wadd, Yaghuth*
Engel	*Jibril, Mikail, Israfil*
Dschinn	*Iblis*
Mythol. Tiere	*Hudhud*

C Sample Excerpt from Tafsir Dataset

```
# adyan: 0
# asbab: 0
# fiqh: 0
# kalam: 1
# lugha: 1
# mushkilat: 0
# mutashabih: 0
# naskh: 0
# qiraat: 0
# science: 0
# sirah: 0
# sufism: 0
# takhsis: 0
# tikrar: 0
وَقَوْلُهُ    O     O
:     O     O
وُجُوهٌ    O     O
يَوْمَئِذٍ    O     O
نَاضِرَةٌ    O     O
.     O     O
يَقُولُ    O     O
تَعَالَى    O     O
ذكره    O     O
:     O     O
وُجُوهٌ    O     B-lugha
يَوْمَئِذٍ    O     O
.     O     O
يَعْنِي    O     O
:     O     O
يَوْمَ    B-TME O
الْقِيَامَةِ    I-TME O
،     O     O
نَاضِرَةٌ    O     B-lugha
.     O     O
```

Figure 5: Tafsir Dataset in CoNLL format, showing the binary topic matrix before the sentence start, afterwards the Arabic tokens along their NER tag (1st column) and subtopic tag (2st column).

D Data Statement

In accordance with (Bender and Friedman, 2018), the following outlines the data statement for the Tafsir Dataset:

A. CURATION RATIONALE Manual annotation of literature in Classical Arabic, which is to date a low-resource language, for identification of named entities in different historical text domains, complemented with topic modeling annotation. The generation of such training data enables machine learning applications for the research fields of historical NLP and digital humanities.

B. LANGUAGE VARIETY The canonical text data of *Tafsir Al-Tabari* was collected from the online resource platform *Gawami' al-Kalim* (https://gk.islamweb.net).

C. SPEAKER DEMOGRAPHIC For various text samples in the historical collections of narrations, it is Classical Arabic speakers. Gender, age, race-ethnicity, socioeconomic status can be inferred from the extensive classical literature of biographical evaluation (*'Ilm Al-Rijal*) on narrators and their biographies (books such as *Al-Tarikh Al-Kabir* ("The Great History") by Imam Bukhari, *Kitab Al-Tabaqat Al-Kabir* ("The Book of the Major Classes") by Ibn Sa'd, or *Ikhtiyar Ma Rifat Al-Rijal* ("The Selection of the Knowledge of the Men") by Shaykh Tusi).

D. ANNOTATOR DEMOGRAPHIC Four scientific staff members and two students (age range: 25-60), gender: male and female. European with Middle Eastern background. Native language: German, Modern Standard Arabic, Classical Arabic. Socioeconomic status: university faculty and higher-education student in Classical Arabic studies.

E. SPEECH SITUATION Sopken Classical Arabic, which was later edited by the collector (here: Al-Tabari). Time frame of data between 7th century and 923 CE. Place: Middle East.

F. TEXT CHARACTERISTICS Exegetical literature: Sentences made of chain of narrators (*Isnad*) and the actual content of narrations (*Matn*) along exegetical prose elaborations for each verse of the Quran.

PROVENANCE APPENDIX N/A

E Extended Results

MLM	SEQ	CRF	Coverage
aubmindlab/bert-base-arabert	79.34	79.91	0.74
aubmindlab/bert-base-arabertv01	79.49	80.07	0.65
aubmindlab/bert-base-arabertv02	79.81	80.26	0.85
aubmindlab/bert-base-arabertv2	79.43	80.14	0.84
aubmindlab/bert-large-arabertv2	79.18	80.29	0.84
asafaya/bert-base-arabic	94.99	95.31	1.00
asafaya/bert-mini-arabic	94.02	94.50	1.00
asafaya/bert-large-arabic	94.90	94.92	1.00
asafaya/bert-medium-arabic	94.93	94.87	1.00
CAMeL-Lab/bert-base-arabic-camelbert-ca	79.56	80.31	0.85
CAMeL-Lab/bert-base-arabic-camelbert-mix	79.61	80.19	0.85
CAMeL-Lab/bert-base-arabic-camelbert-msa	79.40	80.23	0.85
UBC-NLP/ARBERT	95.04	95.29	0.88
UBC-NLP/MARBERT	94.83	94.92	0.88
aubmindlab/araelectra-base-generator	79.37	80.06	0.85
bert-base-multilingual-cased	93.89	94.36	1.00
xlm-roberta-base	94.13	94.49	1.00
xlm-roberta-large	94.36	94.88	1.00
google/rembert	94.43	94.73	1.00

Table 10: Results for CA-NER w/ and w/o CRF

MLM	NER		Topic		Subtopic	
	st	mt	st	mt	st	mt
aubmindlab/bert-base-arabertv02	95.99	95.87	26.11	13.73	21.18	20.47
aubmindlab/bert-large-arabertv2	95.53	95.26	18.94	14.43	18.28	19.44
asafaya/bert-base-arabic	95.61	94.94	20.63	11.84	19.23	18.36
asafaya/bert-large-arabic	95.65	95.80	22.15	20.46	21.68	20.58
asafaya/bert-medium-arabic	95.13	95.17	20.15	9.46	18.67	17.45
CAMeL-Lab/bert-base-arabic-camelbert-ca	96.06	95.99	24.75	15.81	19.68	17.42
UBC-NLP/ARBERT	95.46	95.45	22.16	20.37	22.05	20.56
aubmindlab/araelectra-base-generator	95.08	94.95	18.92	6.86	14.85	14.65
bert-base-multilingual-cased	95.04	94.79	23.11	11.58	18.54	16.72
xlm-roberta-large	95.54	95.22	16.97	13.80	21.18	20.46

Table 11: Multi-task learning results for each task. st=single task, mt=multitask

Topic	Macro-F1
adyan (non-Islamic religion)	27.93
asbab (occasions of revelation)	22.74
fiqh (jurisprudence)	16.66
israliyat (Judeo-Christian)	23.17
kalam (Islamic theology)	26.61
lugha (linguistics)	30.06
mushkilat (problem)	19.97
mutashabih (allegorical)	20.00
naskh (abrogation)	19.76
qiraat (recitation style)	*41.45*
sirah (biography)	21.96
sufism (mysticism)	14.87
takhsis (specification)	19.99
tikrar (repetition)	19.98
ulum (science)	18.41

Table 12: Fine-grained TM results obtained with the measure of Macro-F1 from MaChAmp on Tafsir Dataset (arabertv02).

NE category	Precision	Recall	F1
PER	*97.12*	*97.60*	*97.36*
LOC	72.53	66.93	69.62
ORG	82.00	89.31	85.50
TME	78.00	79.90	78.94
OTH	79.59	76.38	77.95

Table 13: Fine-grained NER results obtained by running the official CoNLL-2003 script on Tafsir Dataset (arabertv02).