EasyChair Preprint

№ 9573

Evaluation of Arabic Named Entity Recognition Models on Sahih Al-Bukhari Text

Ibtisam Khalaf Alshammari, Eric Atwell and
Mohammad Ammar Alsalka

January 16, 2023

Evaluation of Arabic Named Entity Recognition Models on Sahih Al-Bukhari Text

Ibtisam Khalaf Alshammari[1,2,a], Eric Atwell[1,b] and Mohammad Ammar Alsalka[1,c]

[1]University of Leeds, Leeds, United Kingdom

[2]University of Hafr Al-Batin, Hafr Al-Batin 39524, Kingdom of Saudi Arabia

[a]ML18IKFA, [b]E.S.ATWELL, [c]M.A.ALSALKA@leeds.ac.uk

Abstract

In this paper, the following four Arabic named entity recognition (ANER) models were applied to the *Sahih Al-Bukhari* (صحيح البخاري) dataset: CAMeLBERT-CA Hatmimoha, Marefa-NER, and Stanza. This study's main aim is to identify the best-performing model for use with other Hadith datasets. The Stanza and Marefa-NER models are best because they obtained F1-scores of 0.826191 and 0.807396, respectively. Then, a new test dataset of approximately 5,000 words was created based on the CANERCorpus annotation. The four models were evaluated using the latest test dataset and had disappointing F1-scores, although Hatmimoha had the best results. This problem likely arose as a result of the small dataset. However, we observed that since the model has many named entity classes and matches the CANERCorpus labels, it could obtain a high performance, as the Hatmimoha and Marefa-NER models did.

Keywords: Arabic NER Models, CANERCorpus Annotation, Models Evaluation, Sahih Al-Bukhari.

1. Introduction

A popular method of information extraction in natural language processing (NLP) is entity recognition, according to a term coined in 1996 by Grishman and Sundheim. Named entity recognition (NER) is a two-step technique for detecting vital information (entities) that includes (a) identifying and (b) classifying a named entity into predefined categories from unstructured text. Entities include the names of people, organisations, locations, dates, and more. The NER technique can contribute to solving different complex NLP applications, such as information retrieval, knowledge bases, question-answering systems, and semantic searches.

There are various ways to perform the NER process. Much NER literature is English-language focused; however, a few NER studies have focused on the low-resourced classical Arabic language. Several studies have presented Arabic named entity approaches, such as the rule-based, machine learning, and hybrid approaches. The rule-based NER is based on a collection of manually crafted rules that linguists have extracted. The machine learning method depends on feature engineering and statistical models. The hybrid approach combines the rule-based and machine learning techniques. Deep learning approaches have recently achieved notably high performance in many NLP tasks, including NER. Therefore, deep learning is a subfield of machine learning because it uses multi-layer neural networks.

This study aims to use the following four state-of-the-art ANER tools to annotate the classical Arabic text *Sahih Al-Bukhari* (صحيح البخاري): Computational Approaches to Modeling Language Bidirectional Encoder Representations from Transformers-Classical Arabic (CAMelBERT-CA), Hatmimoha, Marefa-NER, and Stanza. Then, the best performance of ANER tools was identified through comparisons.

This paper is organised as follows: Section 2 provides an overview of previous studies conducted on Arabic text. Section 3 explains the datasets used in this study. Section 4 illustrates the phases of the proposed method in detail. Section 5 presents and discusses the results. The final section concludes with possible future research directions.

2. Related Work

Arabic NER is an important task in NLP applications. Numerous studies have used a collection of classification techniques for extracting Arabic NER. Kim and Kan (2010) developed a NER method using N-gram phrase extraction and a straightforward rules model; the outcomes showed excellent precision. Harrag et al. (2011) extracted named entities from prophetic narration texts using a finite state transducer allocated a conceptual class among a set of classes. Those classes include Num-Kitab, Title-Kitab, Num-Bab, Title-Bab, Num-Hadith, Saned, Matn, Taalik, and Atraf. The model got a 52% F1-score on a collection of prophetic narration texts from the *Sahih Al-Bukhari* dataset. Researchers also suggested finite state machines and graph transformation methods in order to extract Arabic personal names (narratives) and relations from Hadith, and their techniques achieved an accuracy of 93 percent (Zaraket & Makhlouta, 2012).

Sajadi and Minaei (2017) used machine learning to distinguish named entities into person, location, and organisation classes. They also improved model performance by using part-of-speech (POS) and keywords as features. Their model had 96% and 67% F-measure values on NoorCorp and ANERcorp, respectively. Aldali's (2018) NER system combined machine learning classifiers, including conditional random fields (CRF), maximum entropy (ME), and support vector machine (SVM). More precise NER improved the system's performance.

Jaber & Saad (2016) utilized CRF and SVM to identify optimal sets of attributes and compared the two methods' outcomes. Both studies (Ekbal & Bandyopadhyay, 2010; Saad, 2014) found that SVM and CRF worked equally well, except that SVM outperformed CRF on datasets with randomised backgrounds. Fairouz et al. (2020) extracted prophetic ontological narratives using Hadith association rules.

Alkhatib and Shaalan's (2020) early work – in which they developed annotated corpora termed ANERcorp, as a manually labeled corpus intended to be utilized in Arabic NER functions for training and testing – is one example of the beneficial resources for NER tasks. Aldumaykhi et al. (2022) presented a paper that evaluated the performance of the CAMeL, Hatmi, and Stanza models on 30 articles written in Modern Standard Arabic (MSA); they annotated all the entities manually. They found similarities between Stanza's and Hatmi's performances. Then, the authors combined the model results using the merge and vote methods. They concluded that the merge approach had the highest recall and F1-score performance.

3. Datasets

This section provides an overview of the datasets used in this study. First, the *Sahih Al-Bukhari* corpus was used and downloaded from the Leeds and King Saud University (LK) Hadith corpus. The LK Hadith is a bilingual corpus of English-Arabic Hadith constructed by Altammami et al. (2019). The *Sahih Al-Bukhari* dataset contains 97 files, covering over 7,000 Hadiths. It also consists of 39,038 annotated Hadiths, and the annotations include Chapter_Number, Chapter_English, Chapter_Arabic, Section_Number, Section_English, Section_Arabic, Hadith_number, English_Hadith, English_Isnad, English_Matn, Arabic_Hadith, Arabic_Isnad, Arabic_Matn, Arabic_Comment, English_Grade, and Arabic_Gradw. In this experiment, we focused on Arabic_Matn as illustrated in Fig.1 in bold text.

Arabic:
حَدَّثَنَا مُحَمَّدُ بْنُ الْمُثَنَّى، قَالَ حَدَّثَنَا يَحْيَى، قَالَ حَدَّثَنَا إِسْمَاعِيلُ، قَالَ حَدَّثَنَا قَيْسٌ، عَنْ جَرِيرِ بْنِ عَبْدِ اللَّهِ، قَالَ بَايَعْتُ رَسُولَ اللَّهِ صلى الله عليه وسلم عَلَى إِقَامِ الصَّلاَةِ، وَإِيتَاءِ الزَّكَاةِ، وَالنُّصْحِ لِكُلِّ مُسْلِمٍ.

English Translation:
Narrated Jarir bin `Abdullah**: I gave the pledge of allegiance to Allah's Messenger (ﷺ) for to offer prayers perfectly, to pay Zakat regularly,and to give good advice to every Muslim.**

Figure 1: Hadith Matn.

Second, CANERCorpus was used in our study to revise and correct the test dataset. CANERCorpus is a classical Arabic NER corpus manually annotated by human experts. This corpus consists of over 7,000 Hadiths from *Sahih Al-Bukhari*. It has a total of 258,241 words, of which over 72,000 are classified as named entities, with other words making up the remaining approximately 186,133 words. The named entities are organised into 21 classes, which are person (Pers), location (Loc), organisation (Org), measurement (Meas), money (Mon), book (Book), date (Date), time (Time), clan (Clan), natural object (NatOb), crime (Crime), day (Day), number (Num), God (Allah), prophet (Prophet), religion (Rlig), sect (Sect), paradise (Para), hell (Hell), month (Month), and other (O) (Salah and Binti Zakaria, 2018).

4. Proposed Methodology

In this section, we present in detail the methodological steps that followed in this study, as illustrated in Fig.2. Several experiments were conducted using the suggested four models CAMeLBERT-CA, Hatmimoha, Marefa-NER, and Stanza Stanford NLP. Then, fine-tuned the test dataset to identify the proper NER tags.

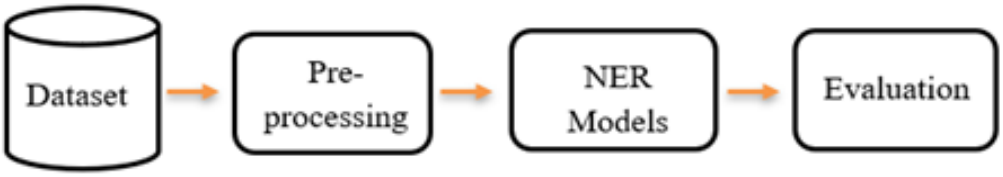

Figure 2: NER Models Evaluation Process.

4.1 Data Pre-processing

Preparing data is crucial in many NLP applications to get more significant data from raw data. In this work, we eliminate unnecessary data such as diacritics, punctuations, and digits using Araby, NumPy, and Panda libraries. Then, camel_tools is used in the normalization step to reduce various forms of words to one form, for instance, (Alef) (أ,إ,ء,) to the normal way (ا). Finally, the text is split into tokens to identify each sentence's boundaries.

4.2 NER Tools

4.2.1 CAMeLBERT-CA

Computational Approaches to Modeling Language (CAMeL) is a set of open-source tools for Arabic natural language processing in Python. It can be used for performing various NLP tasks, for instance, pre-processing, morphological modeling, dialect identification, sentiment analysis, and NER (Obeid et al., 2020). Researchers developed the CAMeLBERT tool from the CAMeLlab at New York University, Abu Dhabi (Inoue et al., 2021). They also released various pretrained language models for MSA, dialectal Arabic (DA), and classical Arabic (CA). This study used CAMeLBERT-CA for the CA in the NER task. The CAMeLBERT-CA NER model used HugginFace Transformers (Wolf et al., 2020) to fine-tune AraBERT. The authors stated that CAMeLBERT-CA was pretrained on the CA dataset and evaluated on ANERcorp, and that the overall F1 result was 67% (Benajiba et al., 2007). This model includes four classes: location (LOC), miscellaneous (MISC), organisation (ORG), and person (PERS) with an IOB (inside, outside, and beginning) tagging format.

4.2.2 Hatmimoha

Mohamed Hatmi developed an Arabic NER tool called Hatmimoha, a pretrained BERT-based Arabic NER model. Although there is no peer-reviewed paper describing and evaluating the model, the author stated that when testing this model on a valid corpus containing 30,000 tokens, the F1-score result was 87%. This model was pretrained using a BERT-based ArabicBERT base, is available via HugginFace, and has nine named entity types: competition, date, disease, event, location, person, organisation, prize, and product; it also has an IOB tagging format (Hatmi, 2020).

4.2.3 Marefa-NER

Marefa-NER is also called the TEBYAN model. According to the authors, this is a large Arabic NER model that aims to extract up to nine different entity tags from a new dataset. This tool has no peer-reviewed paper, but the authors indicated that they evaluated their model against a test set of 1,959 sentences. The F1 score for the weighted average was 0.859008. The named entity classes are similar to those in the Hatmimoha model: artwork, event, job, location, nationality, organisation, person, product, and time, as well as an IOB tagging format. The Marefa-NER model can be accessed via HugginFace (Marefa-NLP, 2021).

4.2.4 Stanza

The Stanford NLP Group developed Stanza as a natural language processing toolkit to enhance NLP tasks such as POS, tokenization, and NER (Qi et al., 2020). Stanza is a collection of pretrained neural models that support the linguistic analysis of various human languages, including Arabic, English, and French. It consists of four named entity classes: LOC, MISC, ORG, and PERS, as well as involves the IOB tagging format. For the Arabic model, the developers evaluated it using the American and Qatari Modeling of Arabic AQMAR (Mohit et al., 2012) corpus. The Arabic NER model was downloaded for this work since it is accessible as a Python package.

5. Results and Discussion

All experiments were conducted on the Google Collab platform. The training dataset was used to train the models, and the test dataset was used with the hyper-parameters shown in Table 1.

Table 1: Models Hyperparameters.

Parameter	Value
Train Batch Size	24
Test Batch Size	24
Learning Rate	2e-5
Epochs	8

Three standard metrics were used to compare the previous models' performance for evaluation purposes. Namely, Precision, Recall, and F1-score as defined in Eq. 1, Eq. 2, and Eq. 3, respectively. The F1-score can be defined as the harmonic of precision and recall (Li et al., 2020). F1-score values range from [0,1], where 0 represents the worst performance, and one represents the best. The metrics formulas are defined as follows:

- Precision represents the percentage of correct identification of named entities that were right.

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \quad (1)$$

- Recall indicates the percentage of true positives detected successfully.

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \quad (2)$$

- F1-score represents the weighted average of precision and recall.

$$F1 - score = 2\frac{Precision*Recal}{Precision+Recall} \quad (3)$$

Over 300,000 words were annotated as a result of this study. Table 2 shows the evaluation results of the four models that have been trained on the LK *Sahih Al-Bukhari* dataset. As can be seen, Stanza and Marefa-NER methods obtained the best results for each metric compared to others. Highlighted in bold are the highest performance of the used models for the F1-score metric. In addition, recall results for Stanza, Marefa-NER, and CAMeLBERT-CA were similar, but the CAMeLBERT-CA model had the worst value, **0.822746**. Similarly, precision values for the Stanza and Marefa-NER models were higher, with a slightly different between them. The possible reason being Marefa-NER model had the highest performance compared to CAMeLBERT-CA and Hatmimoha models is that it has various named entity types and matches the CANERCorpus tags. For example, the word (المدينه) was annotated in Marefa-NER as a location that matches the CANERCorpus tag.

Table 2: Models Performance on *Sahih Al-Bukhari* Corpus.

Model Name	Precision	Recall	F1-Score
CAMeLBERT-CA	0.733668	0.822746	0.775658
Hatmimoha	0.730825	0.798629	0.763224
Marefa-NER	0.780237	0.836513	**0.807396**
Stanza	0.786966	0.869532	**0.826191**

Subsequently, a new test set was built based on CANERCorpus annotation, around only 5 thousand words, to measure the performance of all the mentioned methods. Over 50 Hadith were randomly selected, reviewed, and re-labeled depending on CANERCorpus. Then, testing the previous models using this new test set. The results were disappointing, as shown in Table 3. The worst values were probably obtained since the test dataset was small compared to the training dataset. However, the observation is that the probability of achieving better performance in the model with large named entity types might be high because it matches as many CANERCorpus tags as possible. For example, the Hatmimoha model's performance was the best since it achieved **0.238705**, which is approximately close to the Marefa-NER model performance.

Table 3: Models Performance after Correcting Test Data Tags Based on CANERCorpus.

Model Name	Precision	Recall	F1-Score
CAMeLBERT-CA	0.149599	0.188130	0.166667
Hatmimoha	0.306449	0.197260	**0.238705**
Marefa-NER	0.275554	0.193562	**0.227331**
Stanza	0.209820	0.173099	0.190854

6. Future Directions and Conclusion

This paper compares and evaluates the performance of the four Arabic NER tools based on the classical text *Sahih Al-Bukhari*. These models were tested in order to find the best performance. Stanza and Marfa-NER models were the best. However, different results were founded when tested all the previous models on the new test set. This is because of the small number of annotated words which is considered a limitation for this work. Hatmimoha achieved the best score compared to Marefa-NER and Stanza. We think that if the model contains many named entity classes and matches the CANERCorpus tags, it will give a high score.

Islamic text has unique words that differ from the Modern Standard text because it contains the God and prophet names. Therefore, for future work, the plan is to contribute to improving a new model for the classical Arabic text, especially for the Islamic text. We also intend to annotate all the other LK- Hadith datasets.

7. Acknowledgments

The first author would like to express her deepest gratitude to the Ministry of Education in Saudi Arabia, the University of Hafr Al-Batin, University of Leeds. Ibtisam is also thankful for the suggestions of the anonymous reviewers.

References

Aldali, N. M. (2018). A Combination Method Of Linguistic Features And Machine Learning Techniques For Identifying Arabic Named Entities.

Aldumaykhi, A., Otai, S., & Alsudais, A. (2022). Comparing Open Arabic Named Entity Recognition Tools. arXiv preprint arXiv:2205.05857.

Alkhatib, M., & Shaalan, K. (2020, May). Boosting arabic named entity recognition transliteration with deep learning. In The thirty-third international flairs conference.

Altammami, S., Atwell, E., & Alsalka, A. (2019). The Arabic–English Parallel Corpus of Authentic Hadith. Paper presented at the International Journal on Islamic Applications in Computer Science And Technology-IJASAT.

Benajiba, Y., Rosso, P., & Benedíruiz, J. M. (2007, February). Anersys: An arabic named entity recognition system based on maximum entropy. In International Conference on Intelligent Text Processing and Computational Linguistics, pp. 143–153. Springer, Berlin, Heidelberg.

Ekbal, A., & Bandyopadhyay, S. (2010). Named entity recognition using support vector machine: A language independent approach. International Journal of Electrical and Computer Engineering, 4(3), 589–604.

Fairouz, B., Taleb, N., & Arari, A. N. (2020). An Ontological Model of Hadith Texts. International Journal of Advanced Computer Science and Applications, 11(4).

Grishman, R. and Sundheim, M. B. (1996). Message understanding conference-6: A brief history.

Harrag, F., El-Qawasmeh, E. and Salman Al-Salman, A.M. (2011). Extracting named entities from prophetic narration texts (hadith), Software Engineering and Computer Systems, pp. 289–297. Available at: https://doi.org/10.1007/978-3-642-22191-0_26.

Hatmi, M. (2020). Arabic Named Entity Recognition Model.

Inoue, G., Alhafni, B., Baimukan, N., Bouamor, H., & Habash, N. (2021). The interplay of variant, size, and task type in Arabic pre-trained language models. arXiv preprint arXiv:2103.06678.

Jaber, M. J., & Saad, S. (2016). NER in English translation of hadith documents using classifiers combination. Journal of Theoretical and Applied Information Technology, 84(3), 348.

Kim, S. N., Baldwin, T., & Kan, M. Y. (2010). Evaluating n-gram based evaluation metrics for automatic keyphrase extraction. In Proceedings of the 23rd International Conference on Computational Linguistics, pp. 572–580.

Li, J., Sun, A., Han, J., & Li, C. (2020). A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering, 34(1), 50–70.

Marefa-NLP (marefa NLP) (2021) marefa-nlp (Marefa NLP). Available at: https://huggingface.co/marefa-nlp (Accessed: November 03, 2022).

Mohit, B., Schneider, N., Bhowmick, R., Oflazer, K., & Smith, N. A. (2012, April). 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, pp. 162–173.

Obeid, O., Zalmout, N., Khalifa, S., Taji, D., Oudah, M., Alhafni, B., ... & Habash, N. (2020, May). CAMeL tools: An open source python toolkit for Arabic natural language processing. In Proceedings of the 12th language resources and evaluation conference (pp. 7022–7032).

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python natural language processing toolkit for many human languages. arXiv preprint arXiv:2003.07082.

Saad, S. (2014). Ontology learning and population techniques for English extended quranic translation text (Doctoral dissertation, Universiti Teknologi Malaysia).

Sajadi, M. B. and Minaei, B. (2017). Arabic named entity recognition using boosting method, 2017 Artificial Intelligence and Signal Processing Conference (AISP) [Preprint]. Available at: https://doi.org/10.1109/aisp.2017.8324098.

Salah, R. E., & Zakaria, L. Q. B. (2018, March). Building the classical Arabic named entity recognition corpus (CANERCorpus). In 2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP), pp. 1–8. IEEE.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., & Rush, A. M. (2020, October). Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations, pp. 38–45.

Zaraket, F., & Makhlouta, J. (2012, May). Arabic cross-document NLP for the hadith and biography literature. In Twenty-Fifth International FLAIRS Conference.