information

MDPI

Article

A Comparative Study of Arabic Part of Speech Taggers Using Literary Text Samples from Saudi Novels

Reyadh Alluhaibi [1], Tareq Alfraidi [2,*], Mohammad A. R. Abdeen [3] and Ahmed Yatimi [4]

[1] Department of Computer Science, Taibah University, Madinah 41477, Saudi Arabia; rluhaibi@taibahu.edu.sa
[2] Department of Linguistics, Islamic University of Madinah, Madinah 42351, Saudi Arabia
[3] Department of Computer Science, Islamic University of Madinah, Madinah 42351, Saudi Arabia; mabdeen@iu.edu.sa
[4] Department of Literature, Islamic University of Madinah, Madinah 42351, Saudi Arabia; alyatimi@iu.edu.sa
* Correspondence: t.alfraidi@iu.edu.sa

Abstract: Part of Speech (POS) tagging is one of the most common techniques used in natural language processing (NLP) applications and corpus linguistics. Various POS tagging tools have been developed for Arabic. These taggers differ in several aspects, such as in their modeling techniques, tag sets and training and testing data. In this paper we conduct a comparative study of five Arabic POS taggers, namely: Stanford Arabic, CAMeL Tools, Farasa, MADAMIRA and Arabic Linguistic Pipeline (ALP) which examine their performance using text samples from Saudi novels. The testing data has been extracted from different novels that represent different types of narrations. The main result we have obtained indicates that the ALP tagger performs better than others in this particular case, and that Adjective is the most frequent mistagged POS type as compared to Noun and Verb.

Keywords: Arabic tagger; Part of Speech; Saudi novel; performance evaluation

check for updates

Citation: Alluhaibi, R.; Alfraidi, T.; Abdeen, M.A.R.; Yatimi, A. A Comparative Study of Arabic Part of Speech Taggers Using Literary Text Samples from Saudi Novels. *Information* **2021**, *12*, 523. https://doi.org/10.3390/info12120523

Academic Editor: Paulo Quaresma

Received: 26 September 2021
Accepted: 11 December 2021
Published: 15 December 2021

1. Introduction

Part of Speech (POS) tagging is the process of assigning each word in a text with the appropriate grammatical classification by using a set of tags [1–3]. This process is a critical step for many natural language processing (NLP) applications and corpus linguistics, and it is seen as one of the initial procedures that directly influence the performance of successive text processing steps [1,4]. Furthermore, this sort of tagging is valuable for corpus linguistics as it helps with issues of disambiguation related to word categories and allows for more focused search results [5]. Contrastingly, an untagged corpus provides limited search results.

The process of tagging words is implemented via software that is normally called 'tagger'. Each tagger has its own tag set which is an essential element for any POS tagger. This tag set contains all the possible tags that can be utilized to label the words morphologically [6]. There is, however, no standard tag set used in all existing Arabic taggers. This is because their developers followed different modeling methods and aimed at achieving particular objectives that fit with their own purposes [7].

The Arabic language possesses its own particularly rich and complex morphological system. This poses a big challenge to NLP researchers who develop Arabic POS taggers. For example, when a word has affixes, like *bi-qalami-hi* (with his pen) (We followed the transliteration system adopted by Library of Congress, available via: https://www.loc.gov/catdir/cpso/romanization/arabic.pdf, accessed on 15 October 2021), it should be stemmed by removing all the word's affixes before the tagging process is applied [8]. One of the main challenges that the POS tagging process faces is the ambiguity of a word; a phenomenon that exists in all languages. For example, in Arabic, the word علم (*'ilm*) can be a noun (i.e., *'ilm* as in *hādhā 'ilmun mufīdun* (this is beneficial knowledge)) or a verb (i.e., *'allama* as in *'allama al-ṭullāba* (he taught the students)). This happens due to the lack of short vowel representation (i.e., diacritics) in Arabic written texts [1,9]. Another challenge is the nature of the corpus that is to be tagged. This corpus may represent different genres and contexts from the training data of the tagger. Hence, unfamiliar words and context can minimize the performance of the tagger. For example, if a particular tagger is trained with data collected from newspaper articles, then it will likely struggle to tag words from a dataset of academic articles.

In this paper, we aim to evaluate the performance of five Arabic taggers, namely: the Stanford Arabic tagger (https://nlp.stanford.edu/software/tagger.shtml, accessed on 15 October 2021), CAMeL Tools (https://github.com/CAMeL-Lab/camel_tools, accessed on 15 October 2021), MADAMIRA (https://camel.abudhabi.nyu.edu/madamira/, accessed on 15 October 2021), Farasa (https://farasa.qcri.org/POS/, accessed on 15 October 2021) and Arabic Linguistic Pipeline (ALP) (http://arabicnlp.pro/, accessed on 15 October 2021) using data collected from Saudi novels. The outcomes emerging from their performance will then be compared. It will do so by exploring how these taggers perform with texts representative of language used in Saudi novels in terms of similarity and difference. In the relevant literature, there are a number of studies that have attempted to compare several Arabic taggers and evaluate their performance using different sets of data [9–12]. Therefore, none of these studies have used data from the genre of Arabic novels for such evaluation. On other hand, Arabic novels have been used as a set of data for different Arabic NLP purposes [13,14]. This paper will, therefore, fill this gap in the existing studies on Arabic POS taggers.

The motivation behind this study is twofold. It is first a step to a larger project from which we aim to build a tagged corpus for Saudi novels, which does not yet exist. Since the step of tagging the corpus has to be implemented carefully with a high level of accuracy, we believe that our decision has to be made on the basis of objective evidence; not on subjective preference of a certain (popular) tagger over another (less well-known one). Thus, the tagger that performs better than others will be chosen to tag our corpus. Second, it is claimed that most of the Arabic POS taggers (including the evaluated five taggers in this study) work more appropriately with Modern Standard Arabic (MSA) because they were trained mainly with data gathered from MSA sources [15,16]. The problem, however, is that MSA are used in different genres which differ lexically and stylistically. Our goal, therefore, is to address this generalized approach to MSA by observing how these taggers perform when the target texts are limited to a particular genre—in our case, the Saudi novel—especially as the vast majority of the data training in these taggers comes from the news genre [9,16].

The paper is structured in five sections. After outlining the research problem in Section 1, Section 2 reviews the works related to the area of Arabic POS tagging. Section 3 describes the methodology we applied to conduct the comparative analysis. Section 4 presents and discusses the experimental results. Finally, Section 5 draws the conclusion.

2. Related Works

There are several works in the existing literature regarding the development of tools for Arabic NLP. These tools include features such as tokenization, POS tagging, segmentation, and morphological analysis. In this section, we have particularly reviewed the five tools presented in this study with a focus on their POS tagging function. This review is an important step in enabling the readers to understand the similarities and differences among these tools in terms of their tag sets, modeling techniques, development, training data, and evaluation process.

In [17–19], the authors presented a maximum-entropy POS tagger which is one of the features within Standford CoreNLP. The tagger, unlike the others, is not specifically built for Arabic. It is, however, trained to tag words from some other languages. For Arabic, the developers used Penn Arabic Treebank (PATB) for training the tool. The performance of this tagger was then tested against data from the newswire. The accuracy reported was 96.5% [20]. However, the tagger was later evaluated on data collected from Arabic tweets.

The tagger poorly performed in this particular genre, achieving 49% accuracy. According to the authors, this poor performance was due to the challenges confronted by the tagger with tagging unknown words. In other words, it tagged new words, which were not part of the training data, incorrectly [11].

Pasha [21] presented MADAMIRA, which is a combined version of previously developed tools: MADA [22] and AMIRA [23]. MADA applies a set of models – Support Vector Machines (SVMs) and N-gram language models – and uses an underlying morphological analyzer to generate morphological features of each word (discretization, POS, lemma, and 13 inflectional and clitic features). AMIRA is an Arabic Language processing toolkit built on supervised machine learning with no overt knowledge of deep morphology. It provides various functions, such as a tokenization, POS tagging and a base phrase chunker (BPC). MADAMIRA was then developed to be speedy, extensible, simple to use and easy to maintain. The tool was trained onthe PATB corpus for MSA and the Egyptian Arabic Treebanks for the Egyptian dialect (EGY). The performance of MADAMIRA was evaluated through a blind test of a dataset. The results were compared with MADA. The results showed that MADAMIRA performed very slightly lower than MADA with respect to tagging MSA texts (95.9% as opposed to 96.1%), However, it presented half a percentage better performance than MADA for EGY (92.4% as opposed to 91.8%).

In [24–26], the authors presented an Arabic toolkit called Farasa. The tool produces various functions: segmentation, POS tagger, Arabic text discretization, and dependency parser for Arabic words and sentences. It is an SMV-based toolkit that utilizes various features for ranking. For training, a set of data extracted from PATB was used to train the tool. For testing, the tool designer created a data set from 70 WikiNews articles published between 2013 and 2014. The results obtained were compared with the performance of MADAMIRA. Both tools showed nearly identical tagging accuracy. However, Farasa presented faster performance and proved to be more accurate than MADAMIRA in terms of segmentation. Finally, the developers reported that confusion between nouns and adjectives is the most frequent error type that occurs during the evaluation process.

More recently, Obeid [27] exhibited an Arabic NLP toolset developed in Python: CAMeL. The toolset contained a number of features, such as pre-processing, morphological modeling (including POS tagging), dialect identification, Named Entity Recognition and Sentiment Analysis. The developers compared their toolset with others available in the literature such as MADAMIRA, Stanford CoreNLP and Farasa. They compared the performance of the POS tagging feature with MADAMIRA in two Arabic contexts: MSA and EGY. For MSA, they used the parts 1, 2, and 3 of PATB. For EGY they used the CALIMA ARZ database. The results of POS performance against MADAMIRA were essentially similar in MSA data (97.1% for MADAMIRA vs. 97.2% for CAMeL). For EGY, the results of the comparison were also almost identical. The tagging result for MADAMIRA was 91.7% while for CAMeL it was 91.8%. This implies that these two taggers perform very much alike in data similar to that which was applied in the testing process (i.e., PATB 1, 2 and 3).

Freihat [16] presented the Arabic Language Pipeline tool (ALP), that performs segmentation, POS tagging and Named Entity Recognition (NER) tasks in one single process (i.e., without implementing any preprocessing tasks). This, according to the developers, solves a major problematic case of ambiguity in Arabic. They also claimed that the tasks mentioned above are, by contrast, executed by other taggers through a number of consecutive processing steps. The training corpus of this tool consists of two million words collected from various MSA genres, such as newspapers, novels, medical consultancy web pages and social media. The results showed a significant difference from the other taggers, which were trained on a subset corpus of PATB. The developers evaluated the tool based on a corpus of data from the *Aljazeera* news website and the *Altibbi* medical consultancy portal. They used 9990 tokens and performed manual evaluation and validation of the results. The results turned out various types of errors including the coarse-grain and fine-grain POS tagging errors. The former referred to words that were segmented correctly but tagged incorrectly. The latter error meant that the segmentation and coarse-grain tagging was correct while the fine-grain information within the tag was incorrect. They also reported that the accuracy of the coarse-grained POS tagging was 98.7%, while the fine-grained accuracy was 97.9%.

One final point that should be mentioned here is that these tools do not provide unified tag sets with the exception of CAMeL and MADAMIRA. The tag sets differ in terms of numbers and labels. Table 1 shows a sample of these differences.

Table 1. A sample of the difference of the five Arabic POS taggers in terms of tag sets.

Item	Stanford	CAMeL	Farasa	MADAMIRA	ALP
Number of tags	32	35	16	35	58
Past tense verb tag	VB	verb	V	verb	PSTV
Preposition tag	IN	prep	PREP	prep	P

This section describes the methodology we followed to conduct the study. We split this section into sub-sections: data preparation and method. Samples were taken of the difference of the five Arabic POS taggers in terms of tag sets. In the above section, we reviewed the five taggers under consideration with focus on their POS tagging function. The next section presents the methodological steps we implemented to conduct the comparison.

3. Methodology

This section describes the methodology we followed to conduct the study. We split it into subsections: Data set and method.

3.1. Data Set

The data used in this study has been collected from a set of Saudi novels. We have selected ten texts from eight novels, each of about 200 words in size. The intention is to make these texts representative of various types of genres such as history, sociology, science fiction and others. As an example, we selected the following text [28]:

> نبدأ بالمنطقة الشرقية التي يبدو أن جنّها يتنقلون بسهولة متناهية بينها وبين بقية مناطق الخليج . هناك ، أولاً ، الجنية الأشهر في منطقة الإحساء ، أم السعف والليف . كان كل طفل إحسائي - أو حساوي وهي كلمة أخف على اللسان ! - يرتعد في فراشه كل ليلة وهو يسمع حفيف…

which represents a legendry narrative novel and it means the following in English:

> We start with the eastern region, where it seems that the Jinn move very easily between it and the rest of the Gulf regions. There is, first of all, the most famous fairy in the Al-Ahsa region, Umm the Leaves and the Leaf. Every child was an ehsa'i - or hasawy, a word lighter on the tongue! He shivered in bed every night when he heard a rustle…

Let us take another example of different genres as follows [29]:

> أدركت عموشة أن تاريخاً جديداً يختلف عن تاريخ أمّها وأبيها قد حلَّ ، عندما أقرَّ الملك فيصل قانون تحرير العبيد ، في ستينيات القرن العشرين . حينها ، ركض جوهر ومعه زوجته نوير ، بعد أن انطفأ نور عينيها وأكل وجهها بقايا الجدري ، نحو عمه عبد الرحمن ، وسأله : ما الذي يعنيه هذا القانون؟…

which represents a social narrative novel and it means the following in English:

> Amousha realized that a new history, different from the history of her mother and father, had occurred when King Faisal passed the law for the liberation of

> slaves in the sixties of the twentieth century. At that time, Jawhar and his wife Nuer, after the light in her eyes had gone out and her face had eaten the remains of smallpox, ran towards his uncle Abdul Rahman, and asked him: What does this law mean?...

Both examples were selected carefully and avoid repetition and bias with respect to their types.

This step was accomplished by an expert in the area of Saudi literature. The total number of words in these texts is 2059, with a mean average of 206 words per text. Table 2 provides a description of the data.

Table 2. A description of the collected Data.

No	Narration Type	Number of Words
1	Legendry narrative [28]	237
2	Dialogue [28]	211
3	Social narrative [29]	192
4	Place description [30]	181
5	Preamble to the Novel [31]	195
6	Descriptive narrative [32]	205
7	Science Fiction [33]	218
8	Using poetry in historical narrative [33]	212
9	Historical narrative [34]	205
10	Social narrative [35]	208
Total: 2059		

The texts, it should be noted, represent MSA. One of these texts, however, represents a mixture of MSA and Saudi dialect since Saudi novelists sometimes, although uncommonly, wish to make part of the novel closer to the spoken language used in the real world. The variability in the testing data is beneficial because it enables the evaluation of the taggers, which are mainly built to deal with MSA, on expressions of local dialect style. It is relevant to mention that there are some corpora of Arabic dialects including Saudi novels that represent some Saudi dialects [36,37].

For the next step, we manually tagged the words of these texts. This process was conducted by two experts who understand the meaning of the tag sets used and the meaning of Arabic words lexically and contextually. One expert holds a PhD in Arabic Linguistics and the other holds a Master's degree in the same field. Note that we manually tagged these texts using four POS categories: Noun, Verb, Adjective, Other (which represent all other POS types). We focused on these three because they represent the main content of POS types and the most frequent ones as reported in some Arabic corpora [6], and the ambiguity between them has been reported to be common [26,38]. Hence, we have 1236 joint words which cover 59.8% of the whole texts (2059). Moreover, other POS categories were included but were not the main focus in this paper.

3.2. Method

The method to evaluate the performance of the five taggers is shown in Figure 1. These steps are described as follows:

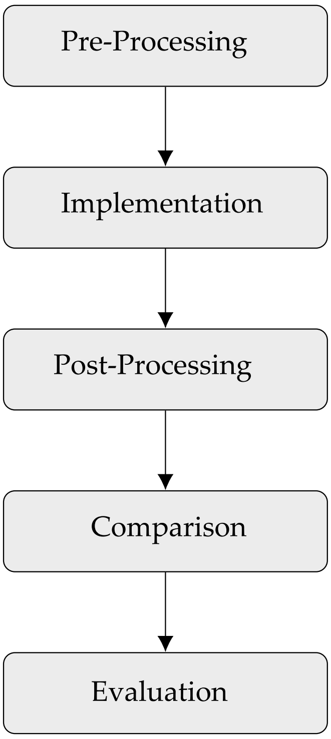

Figure 1. The method flowchart for evaluating the taggers' performance.

1. Pre-processing: We first collected the tag sets of the five taggers and then grouped the tags that are related to nouns, verbs, adjectives and remaining POS categories into four main categories: Noun, Verb, Adjective, Other. Mapping different tag sets of the five taggers into only four simplified and unified tags allow for efficient comparison. Table 3 presents a sample of tag sets from the selected taggers and their corresponding simplified tags. This mapping was done for simplification purposes, particularly for validating and evaluating the outcomes.
2. Implementation: We tagged the original ten Arabic texts via the five taggers included in this study (Stanford, CAMeL, Farasa, MADAMIRA and ALP). It should be mentioned here that the selected texts were tagged without segmentation. This is due to existing reports that segmentation increases ambiguity of the words tagged [9,39].
3. Post-processing of results: We used the tag sets mapping to transform all the taggers' results into the four POS categories: Noun, Verb, Adjective and Other, as explained earlier. This simplification is necessary for comparison purposes.
4. Results comparison: we compared the outputs of the tagging results with those performed using classification report function from Pandas library in the Python Language.
5. Evaluation: after applying the classification report function, we evaluated the generated precision, recall and F1 scores for each text and for all taggers as explained in the in the following section.

Table 3. A sample of tag sets and their mapping into simplified tags.

Sample of Tag Sets Mapping					
Stanford	**Farasa**	**MADAMIRA**	**CAMeL**	**ALP**	**Simplified Tags**
DTNN, DTNNP, NN, NNS.	NOUN, NSUFF, FOREIGN.	noun, noun_prop, noun_quant.	noun, noun_prop, noun_quant.	SMN, SFN, DMN, DFN, PMN, PFN.	Noun
VB, VBD, VBN, VBP.	V, VSUFF.	verb, verb_pseudo.	verb, verb_pseudo.	PRSV, PSTV, PPRSV, PPSTV, IMPV.	Verb
JJ, JJR, DTJJ.	ADJ.	adj, adj_comp.	adj, adj_comp.	SMAJ, SFAJ, DMAJ, DFAJ, PMAJ.	Adjective
PUNC, WP, RB, WP.	CONJ, RP, WP, CD.	pron_rel, conj_sub, adv, pron.	prep, part_voc, part_neg, pron_dem.	PX, REL, C, C+LC.	Other

4. Results and Discussion

Tables 4–6 present the results of various metrics (precision, recall, and F_1 scores) obtained by evaluating the five taggers under consideration.

Each one of the three tables shows the results of a specific metric resulting from evaluating the five taggers while considering the ten text samples extracted from the Saudi novels. Table 4, as an example, shows the recall of various taggers against the ten text samples. The table shows that the ALP tagger performed best among all considered taggers. With only one exception, the ALP tagger scored an average value of 92%. This one exception was for the MADAMIRA tagger, where that tagger was only 1% higher than the ALP tagger. It is also to be noted that Farasa and CAMeL taggers showed significantly lower recall values (86% and 83%, respectively). On the other hand, the Stanford tagger performed poorly with the lowest level of recall among the five taggers (70%), showing a significant performance gap against the other taggers.

Table 5 shows the precision of various taggers against the ten text samples. The table show that the ALP tagger performed best among all considered tagger except two instances with CAMeL and Farasa which are better. Nevertheless, the ALP tagger scored an average value of 93%. It is also to be noted that Farasa and CAMeL taggers showed significantly lower precision values (87% and 85%), respectively. On the other hand, Stanford tagger performed poorly with the lowest precision value among the five taggers (71%), showing a significant performance gap against other taggers.

Table 6 shows the F_1 of various taggers against the ten text samples. The table shows that the ALP tagger performed best among all considered tagger except one instance with MADAMIRA which are both equal. The ALP tagger scored an average value of 92%. It is also to be noted that the Farasa and CAMeL taggers showed significantly lower F_1 values (86% and 84%),respectively. On the other hand, the Stanford tagger performed poorly, with the lowest level of F_1 score among the five taggers (67%), showing a significant performance gap against other taggers. The weak performance of the Stanford tagger can be related to the "unknown words" (i.e., words that were possibly not part of the training corpus). This interpretation might be supported by what is shown in Section 2 [18], which showed that this tool delivered lower performance with tagging unknown words, achieving 84.5%

accuracy as opposed to 96.7% for known words. Nevertheless, the other taggers could have possibly encountered this problem in the implementation process, but they appeared to have tackled it in a better way than Stanford.

Table 4. The recall performance of the five taggers (The number indicates percentages).

Recall Comparison											
Taggers/No.Text	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Average**
Stanford	65	76	73	70	71	57	73	77	68	74	70
CAMeL	77	78	79	88	90	82	90	83	86	80	83
Farasa	81	90	83	92	92	83	90	82	**88**	82	86
MADAMIRA	85	94	82	89	93	84	89	**88**	87	87	88
ALP	**91**	**97**	**87**	**98**	**94**	**95**	**94**	87	**88**	**93**	**92**

Table 5. The precision performance of the five taggers (The number indicates percentages).

Precision Comparison											
Taggers/No.Text	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Average**
Stanford	69	70	67	77	70	67	73	68	80	70	71
CAMeL	80	78	79	91	88	87	**93**	83	89	84	85
Farasa	86	90	**86**	94	89	86	90	83	88	82	87
MADAMIRA	86	93	81	92	91	88	92	88	93	85	89
ALP	**89**	**99**	85	**99**	**94**	**94**	91	**90**	**95**	**95**	**93**

Table 6. The F_1 performance of the five taggers (The numbers indicates percentages).

F_1 Comparison											
Taggers/No.Text	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Average**
Stanford	61	69	66	68	67	59	71	69	72	68	67
CAMeL	76	78	79	89	89	84	91	82	87	81	84
Farasa	83	90	84	93	90	84	89	82	88	81	86
MADAMIRA	85	94	81	90	92	85	90	**88**	89	86	88
ALP	**90**	**98**	**86**	**99**	**94**	**95**	**92**	**88**	**91**	**94**	**92**

The ALP's high metric values (precision, recall, F1 score) can be explained with relation to the nature of its training corpus. It is stated explicitly by the developers that training data is more diverse and part of this data is derived from Arabic novel genre [16], which represents 15% of the overall corpus. This type of data, by contrast, is absent from the training corpus of other taggers, as they were trained with data that comes from PABT which only focused on Arabic news texts [9,40]. In addition to this, the size of the training data of the ALP corpus is over two million words, while the size of the PATB (which the other four tagger are based upon) is about 500,000 words [40]. Not only does ALP score the highest average numbers (precision, recall, and F1 score), but it also almost always outperforms other taggers in the eight sample texts. Another reflection emerging from Table 6 is that CAMeL and MADAMIRA scored different performance results (F1 scores of 84% and 88%), although it is reported in section 2 that, according to [27], both tools performed remarkably similar (97.2% vs. 97.1%) in a dataset that is collected from PATB, where CAMeL was evaluated using the Multitask Learning disambiguator. However, in this study, the two taggers showed different performance results. It is our opinion that this variation is mainly due to the fact that we used CAMeL with the Maximum Likelihood Estimation (MLE) disambiguator.

One interesting observation that can also be spotted in Table 6 is that the taggers have shown a relatively low performance in Text 8. One possible reason behind this low performance is that this text contains six verses from a classical Arabic poem. This appears to have negatively affected the taggers' performance. This interpretation may be justified by the fact mentioned above (Section 1), which is that the five taggers have been trained with MSA corpus. Hence, these taggers generally could not recognize such classical words or their contexts. This justification is supported by the findings obtained by Alosaimy [9] and Alrabiah [12] who experimentally found that when the taggers (e.g., MADAMIRA, Farasa, etc.), which were designed for MSA were used to tag Classical Arabic texts, their accuracy decreased.

Tables 7–9 present the performance of the taggers from a different angle. They show the results of various metrics (precision, recall, and F_1 scores) obtained by evaluating the five taggers with respect to tagging the four POS categories (Noun, Verb, Adjective and Other).

Table 7. The recall performance of the five taggers among the POS classes (the numbers indicate percentages).

Recall Comparison				
POS Taggers	**Noun**	**Verb**	**Adjective**	**Other**
Stanford	91	73	68	50
CAMeL	93	92	60	89
Farasa	94	87	73	89
MADAMIRA	**95**	95	70	91
ALP	**95**	**96**	**84**	**95**

Table 8. The precision performance of the five taggers among the POS classes (the numbers indicate percentages).

Precision Comparison				
POS Taggers	**Noun**	**Verb**	**Adjective**	**Other**
Stanford	57	67	65	93
CAMeL	80	91	68	97
Farasa	81	**96**	74	97
MADAMIRA	84	**96**	79	98
ALP	**91**	**96**	**85**	**98**

Table 9. The F_1 performance of the five taggers among the POS classes (the numbers indicate percentages).

F_1 Comparison				
POS Taggers	**Noun**	**Verb**	**Adjective**	**Other**
Stanford	69	69	63	66
CAMeL	86	92	64	70
Farasa	87	92	74	80
MADAMIRA	89	69	63	66
ALP	**93**	**97**	**84**	**88**

Table 7 shows the recall of various taggers with respect to shown categories. As can be seen, ALP scored higher than the other taggers in the four POS categories. Another noteworthy remark is that nouns have been tagged correctly by all the taggers more than verbs and adjectives. Such a result indicates that recognizing nouns in general is easier than recognizing verbs and adjectives. However, the taggers differ in their performance and show varying degrees in tagging the three POS.

Table 8 shows the precision of various taggers with respect to shown categories. ALP scored higher than the other taggers in Noun and Adjective categories while in Verb and Other categories ALP, MADAMIRA and Farasa are almost the same.

Table 9 shows the F_1 of various taggers with respect to each category. As can be seen, ALP scored higher than the other taggers in the four POS categories. It is also noted from the three tables that the Adjective category appears to be the most commonly mistagged POS class. Tables 7–9 show that none of the taggers scored above 85% in the three metrics. This low performance may have emerged by the confusion between adjectives and nouns in Arabic since most of the incorrect cases showed that the taggers have defined adjectives as nouns. For example, CAMeL tagged adjectives with a recall value of 60%. Table 10 shows a sample of the adjectives that were incorrectly tagged as Noun by CAMeL in Text 1 and Text 6:

Table 10. A sample of the mistagged adjectives by CAMeL.

Text 1	Text 6
متناهية	حجري
الأشهر	منحوت
إحسائي	الغابرين
حساوي	كاغد
أخف	ثمين
الثقيل	المتداينون
الباطن	المتشاكلون
مباشر	المتباغضون
شابة	المأكولة
حسناء	المتنازعون
عجوز	جلدي
غاية	سميك
الحسناء	
الشوهاء	
المؤقت	
الحمارية	

This shows a great level of uncertainty that encounters the taggers when tagging adjectives. This confusion between these two POS classes has also been observed by [39], who claimed that in Arabic, the distinction between nouns and adjectives is hazy, since adjectives can be used as nouns (e.g., the words *gharīb* ‘stange’, *ghāyah* ‘desire’, *‘ulwī* ‘high’ from the tested data) and behave morphologically similarly to nouns. In other words, adjectives obtain the plural and feminine markers in the same way that nouns do (e.g., from the data we examined the words *alsamīkah* ‘thick’, *shābbah* ‘young person’, *almutabāghiḍīn* ‘people who hate each other’, *alghābirīn* ‘those who lived in the past’). A similar view has been adopted by [38]. It appears that what makes the ambiguity between the two classes more complex is the absence of diacritics from the Arabic writing system. An example from the tested data is the word المجاورة: it is an adjective when the *waw* is followed by the short vowel -i (transliterated. *almujāwirah*), which means ‘neighboring’, while it is a noun when the waw is followed by the short vowel -a (transliterated. *almujāwarah*), which means ‘neighborhood’.

5. Conclusions

In this paper, we presented a comparative study of the performance of five Arabic POS taggers (Stanford, CAMeL, Farasa, MADAMIRA and ALP) in Saudi novels. The data set used in this study consisted of ten samples from eight different Saudi novels. Those samples were carefully extracted from the novel text by an expert such that it spans various types of Arabic narratives. The performance metrics used were the precision, the recall, and the F1 score. The results showed that the ALP tagger performed, in almost all cases, better than the the remaining four taggers. Furthermore, the results showed that the Noun phrase was the most correctly tagged one. The Verb and the adjective phrases followed. We suggest that the overall performance of a given tagger is governed by the ability of the tagger to correctly tag the adjective and the verb phrases. Therefore, to improve the performance of those taggers, we propose to include data sets with more verbs and adjectives. Despite the relatively limited size of the data set, we think it served the purpose of this preliminary study. Larger datasets extracted from more novels will be considered in future works.

Author Contributions: Conceptualization, software, methodology, formal analysis R.A.; project administration, investigation, validation, funding acquisition, writing—original draft preparation T.A.; conceptualization, resources, writing—review & editing, M.A.R.A.; investigation, resources, A.Y. All authors have read and agreed to the published version of the manuscript.

Funding: The research was funded by the Deanship of Scientific Research at Islamic university of Madinah. Grant number (581).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors wish to thank Abed Alhakim Freihat (University of Trent, Italy) and Abdulmohsen Al-Thubaity (King Abdulaziz City for Science and Technology) for their scientific support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Albared, M.; Omar, N.; Ab Aziz, M.J. Developing a competitive HMM arabic POS tagger using small training corpora. In Proceedings of the Asian Conference on Intelligent Information and Database Systems, Daegu, Korea, 20–22 April 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 288–296.
2. El Hadj, Y.; Al-Sughayeir, I.; Al-Ansari, A. Arabic part-of-speech tagging using the sentence structure. In Proceedings of the Second International Conference on Arabic Language Resources and Tools, Cairo, Egypt, 22–23 April 2009; pp. 241–245.
3. Habash, N.Y. Introduction to Arabic natural language processing. *Synth. Lect. Hum. Lang. Technol.* **2010**, *3*, 1–187. [CrossRef]
4. Marquez, L.; Padro, L.; Rodriguez, H. A machine learning approach to POS tagging. *Mach. Learn.* **2000**, *39*, 59–91. [CrossRef]
5. Randi, R. Building a corpus: What are the key considerations? In *The Routledge Handbook of Corpus Linguistics*; Routledge: London, UK, 2010; pp. 31–37.
6. Alkhazi, I.S.B. *Compression-Based Parts-of-Speech Tagger for the Arabic Language*; Bangor University: Bangor, UK, 2019.
7. Zeroual, I.; Lakhouaja, A.; Belahbib, R. Towards a standard Part of Speech tagset for the Arabic language. *J. King Saud Univ.-Comput. Inf. Sci.* **2017**, *29*, 171–178. [CrossRef]
8. Khoja, S. APT: Arabic part-of-speech tagger. In Proceedings of the Student Workshop at NAACL, Pittsburgh, PA, USA, 2–7 June 2001; Citeseer: State College, PA, USA, 2001; pp. 20–25.

9. Alosaimy, A.; Atwell, E. Tagging classical Arabic text using available morphological analysers and part of speech taggers. *J. Lang. Technol. Comput. Linguist.* **2017**, *32*, 1–26.
10. Alashqar, A.M. A comparative study on Arabic POS tagging using Quran corpus. In Proceedings of the 2012 8th International Conference on Informatics and Systems (INFOS), Giza, Egypt, 14–16 May 2012; p. NLP-29.
11. Albogamy, F.; Ramsay, A. Fast and robust POS tagger for Arabic tweets using agreement-based bootstrapping. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia, 23–28 May 2016; pp. 1500–1506.
12. Alrabiah, M.; Al-Salman, A.; Atwell, E.; Alhelewh, N. KSUCCA: A key to exploring Arabic historical linguistics. *Int. J. Comput. Linguist. (IJCL)* **2014**, *5*, 27–36.
13. Al Khalil, M.; Habash, N.; Jiang, Z. A large-scale leveled readability lexicon for Standard Arabic. In Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, 11–16 May 2020; pp. 3053–3062.
14. Inoue, G.; Alhafni, B.; Baimukan, N.; Bouamor, H.; Habash, N. The interplay of variant, size, and task type in Arabic pre-trained language models. *arXiv* **2021**, arXiv:2103.06678.
15. Alkhazi, I.S.; Teahan, W.J. BAAC: Bangor Arabic Annotated Corpus. *Mach. Transl.* **2018**, *22*, 23. [CrossRef]
16. Freihat, A.A.; Bella, G.; Mubarak, H.; Giunchiglia, F. A single-model approach for Arabic segmentation, POS tagging, and named entity recognition. In Proceedings of the 2018 2nd International Conference on Natural Language and Speech Processing (ICNLSP), Algiers, Algeria, 25–26 April 2018; pp. 1–8.
17. Green, S.; Manning, C.D. Better Arabic parsing: Baselines, evaluations, and analysis. In Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), Beijing, China, 23–27 August 2010; pp. 394–402.
18. Toutanova, K.; Klein, D.; Manning, C.D.; Singer, Y. Feature-rich part-of-speech tagging with a cyclic dependency network. In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, Edmonton, AB, Canada, 27 May–1 June 2003; pp. 252–259.
19. Toutanova, K.; Manning, C. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In Proceedings of the 2000 Joint SIGDAT Conference EMNLP/VLC, Hongkong, China, 7–8 October 2000; pp. 63–71.
20. El-Haj, M.; Koulali, R. KALIMAT a multipurpose Arabic Corpus. In Proceedings of the Second Workshop on Arabic Corpus Linguistics (WACL-2), Lancaster, UK,22 July 2013; pp. 22–25.
21. Pasha, A.; Al-Badrashiny, M.; Diab, M.T.; El Kholy, A.; Eskander, R.; Habash, N.; Pooleery, M.; Rambow, O.; Roth, R. Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), Reykjavik, Iceland, 26–31 May 2014; Citeseer: State College, PA, USA, 2014; Volume 14, pp. 1094–1101.
22. Habash, N.; Rambow, O.; Roth, R. MADA+ TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. In Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR), Cairo, Egypt, 21–23 April 2009; Volume 41, p. 62.
23. Diab, M. Second generation AMIRA tools for Arabic processing: Fast and robust tokenization, POS tagging, and base phrase chunking. In Proceedings of the 2nd International Conference on Arabic Language Resources and Tools, Cairo, Egypt, 22–23 April 2009; Volume 110, p. 198.
24. Abdelali, A.; Darwish, K.; Durrani, N.; Mubarak, H. Farasa: A fast and furious segmenter for arabic. In Proceedings of the 2016 conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, San Diego, CA, USA, 12–17 June 2016; pp. 11–16.
25. Darwish, K.; Mubarak, H. Farasa: A new fast and accurate Arabic word segmenter. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portorož, Slovenia, 23–28 May 2016; pp. 1070–1074.
26. Darwish, K.; Mubarak, H.; Abdelali, A.; Eldesouki, M. Arabic pos tagging: Don't abandon feature engineering just yet. In Proceedings of the Third Arabic Natural Language Processing Workshop, Valencia, Spain, 3–4 April 2017; pp. 130–137.
27. Obeid, O.; Zalmout, N.; Khalifa, S.; Taji, D.; Oudah, M.; Alhafni, B.; Inoue, G.; Eryani, F.; Erdmann, A.; Habash, N. CAMeL tools: An open source python toolkit for Arabic natural language processing. In Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, 13–15 May 2020; pp. 7022–7032.
28. Ghazi, A. Aljinniyah. In *Almu assaah Al arabiyyah li-Dirāsāt wa Alnashr*; Beirut, Lebanon, 2016. Available online: http://airpbooks.com/?url=ar/BookDetails?BookID:2397 (accessed on 5 October 2021).
29. Badria, A. Hind wa Al skar. In *Dār Alsāqī*; Beirut, Lebanon, 2013. Available online: https://www.daralsaqi.com/book/%D9%87%D9%86%D8%AF-%D9%88%D8%A7%D9%84%D8%B9%D8%B3%D9%83%D8%B1 (accessed on 5 October 2021).
30. Abdulrahman, M. Umm Alnudhūr. In *Almu assaah Al arabiyyah li-Dirāsāt wa Alnashr*; Beirut, Lebanon, 2005. Available online: https://www.neelwafurat.com/itempage.aspx?id=lbb136677-96909&search=books (accessed on 5 October 2021).
31. Abdulrahman, M. Urwat Alzmān Albāhī. In *Almu assaah Al arabiyyah li-Dirāsāt wa Alnashr*; Beirut, Lebanon, 2007. Available online: https://www.neelwafurat.com/itempage.aspx?id=lbb29569-27908&search=books (accessed on 5 October 2021).
32. Omaima, A. Msrā Alghrānīq. In *Dār Alsāqī*; Beirut, Lebanon, 2017. Available online: https://www.daralsaqi.com/book/%D9%85%D8%B3%D8%B1%D9%89-%D8%A7%D9%84%D8%BA%D8%B1%D8%A7%D9%86%D9%8A%D9%82-%D9%81%D9%8A-%D9%85%D8%AF%D9%86-%D8%A7%D9%84%D8%B9%D9%82%D9%8A%D9%82 (accessed on 5 October 2021).
33. Monther, Q. Qarīn. In *Aldār Al arabiyyah li-l ulūm*; Beirut, Lebanon, 2016. Available online: http://www.aspbooks.com/books/bookpage.aspx?id=254828-237549 (accessed on 5 October 2021).

34. Magbol, A. Ziryāb. In *Dār Alsāqī*; Beirut, Lebanon, 2014. Available online: https://www.daralsaqi.com/book/%D8%B2%D8%B1%D9%8A%D8%A7%D8%A8 (accessed on 5 October 2021).
35. Qmasha, A. Unthā Al ankabūt. In *Dār Alkifāḥ li-Nashr wa Altawzī*; Dammam, Saudi Arabia, 2000. Available online: https://www.neelwafurat.com/itempage.aspx?id=lbb87629-0&search=books (accessed on 5 October 2021).
36. Khalifa, S.; Habash, N.; Abdulrahim, D.; Hassan, S. A large scale corpus of Gulf Arabic. *arXiv* **2016**, arXiv:1609.02960.
37. Khalifa, S.; Habash, N.; Eryani, F.; Obeid, O.; Abdulrahim, D.; Al Kaabi, M. A morphologically annotated corpus of Emirati Arabic. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 May 2018.
38. Diab, M.; Hacioglu, K.; Jurafsky, D. Automatic tagging of Arabic text: From raw text to base phrase chunks. In Proceedings of HLT-NAACL 2004, Boston, MA, USA, 2–7 May 2004; pp. 149–152. Short Papers.
39. Mohamed, E.; Kübler, S. Arabic Part of Speech Tagging. In Proceedings of the Seventh International Conference on Language Resources and Evaluation, Valletta, Malta, 17–23 May 2010.
40. Maamouri, M.; Bies, A.; Buckwalter, T.; Mekki, W. The penn arabic treebank: Building a large-scale annotated arabic corpus. In Proceedings of the NEMLAR Conference on Arabic Language Resources and Tools, Cairo, Egypt, 22–23 September 2004; Volume 27, pp. 466–467.