languages

MDPI

Article

A Bi-Gram Approach for an Exhaustive Arabic Triliteral Roots Lexicon

Ebtihal Mustafa [1,*] and Karim Bouzoubaa [2]

[1] Collage of Computer Science and Information Technology, Sudan University of Science and Technology, Khartoum HGX7+M5F, Sudan
[2] Mohammadia School of Engineers, Mohammed V University in Rabat, Rabat 10090, Morocco; karim.bouzoubaa@emi.ac.ma
* Correspondence: ebtihal99@hotmail.com

Abstract: With the rapid development of science and technology, many new concepts and terms appear, especially in English. Other languages try to express these concepts with words from their vocabulary. In Arabic, there are many ways to find a counterpart for a particularly new concept, such as using an existing word to denote the new concept, derivation, and blending. When these methods fail, the new concepts are phonetically transliterated. Unfortunately, most of the transliterated terms do not conform to the rules of the Arabic language, and many languages, including Arabic, avoid the use of such terms. Some modern linguists call for using the generation strategy to translate new terms into Arabic based on the idea of the meanings of the Arabic letters. Therefore, it is necessary to provide a resource that contains all Arabic roots with a categorization of what is used, what is available for use, and what is rejected according to the phonetic system. This work provides a comprehensive lexicon that contains all possible triliteral roots and determines the status of each root in terms of usage and acceptability. Additionally, it provides a mechanism for giving preference to roots when there is more than one root that indicates the desired meaning.

Keywords: Arabic language; Arabic roots; lexicons; phonetic system; bigram frequencies; roots weight; Artificial Intelligence; NLP; Arabic NLP

Citation: Mustafa, Ebtihal, and Karim Bouzoubaa. 2023. A Bi-Gram Approach for an Exhaustive Arabic Triliteral Roots Lexicon. *Languages* 8: 83. https://doi.org/10.3390/languages8010083

Academic Editor: Jeanine Treffers-Daller

Received: 4 April 2022
Revised: 4 March 2023
Accepted: 6 March 2023
Published: 13 March 2023

1. Introduction

Arabic is one of the oldest languages that originated in the Arabian Peninsula during pre-Islamic times. It belongs to the Semitic family along with Amharic, Aramaic, and Hebrew. It is the most widely spoken and studied language in this family (Al-Huri 2015) and also the religious language of all Muslims.

The Arabic alphabet contains 28 letters, and Arabic is a highly derivational language. The vocabulary of Arabic words is essentially derived from roots. These roots may consist of three, four, or five letters, such as ك ت ب (ktb)[1], د ح ر ج (dHrj), and س ف ر ج ل (sfrjl) (Al-kabeerm et al. 1981). Unlike English and other languages, words are not derived by adding suffixes and prefixes. Instead, words are derived according to specific patterns. Therefore, the letters of the root can be interrupted by affixes of the pattern. For example, applying the pattern فاعِل (fAEil) on the triliteral root ك ت ب (ktb) results in the lexical form كاتِب (kAtib/writer).

All letters are consonants, each of which can be extended using short vowels known as diacritics. For example, the letter (SEEN, س) can have the sound "sa" (written in Arabic as سَ), "su" (written as سُ), and "si" (written as سِ). Some patterns add additional letters to the root like the previous example while other patterns can add just diacritics. For instance, the pattern (فُعِل/foEil) is used to drive the passive voice form of the root, such as كُتِب (kutib/written) from the root ك ت ب (ktb).

Unlike English and other languages, words are not derived by adding suffixes and prefixes. Instead, words are derived according to specific patterns. Therefore, the letters of the root can be interrupted by affixes of the pattern. For example, applying the pattern فاعِل (fAEil) on the triliteral root ك ت ب (ktb) results in the lexical form كاتِب (kAtib/writer).

However, not all combinations of the 28 letters are used as roots. The unused combinations may or may not be subject to the phonetic rules of the language, and many linguists discuss the reason behind this phenomenon in different languages (Kishli 1996; Hindawi 1993; Balabaki 1987; Chomsky and Halle 1968; Crystal 2011), as we will see in Section 2. Arabic phoneticians have focused primarily on triliteral roots since quadriliteral and quinqueliteral largely share the properties of triliteral roots.

Unfortunately, the current situation regarding Arabic roots and corresponding words needs to keep pace with continuous scientific development. In fact, many new terms are emerging with the development of science and technology and all fields of life. Some studies have estimated that more than 50% of the vocabulary of developed countries is scientific terms (Dwaidri 2010). Consequently, many countries are trying to follow scientific trends and are making efforts to expand their languages to accompany this development.

Concerning Arabic, several strategies can be used to handle new terms. These strategies are: (1) modifying the original concept of an existing word to incorporate the new concept, such as سيارة(syArp; car) since the Arabic word had in ancient times the meaning of a group of walking people or convoy (Al-kabeerm et al. 1981), while today it is more known as a car; (2) Arabizing foreign words according to the Arabic forms (al-taʿrib; Arabization), such as تلفاز(tilfaz; television); (3) merging two words into one (al-naht; blending), such as برمائي(brmaai; amphibious); and, finally, (4) deriving new expressions from original Arabic roots (al-iŠtiqāq; derivation), such as حاسوب(HAswub; computer), which is a new word derived from the root (ح س ب/Hsb), which means compute (Brakhw and Milad 2019).

Modifying the word's original meaning to fit the new concept is one of the most effective methods of creating new terms. The resulting term is easy to understand, but sometimes it is impossible to have an old Arabic word suitable for the new intended meaning, so the new term must be created using one of the other methods.

Arabizing may produce words that do not conform to the phonetic system of Arabic. For example, the term "hydroxy" is translated as "هيدروكسي" whose pattern "فيعلولي/faialolly" is not Arabic. Moreover, in Arabization, it is not possible to maintain the relationship between the Arabic root and the Arabized term (Al-Shbiel 2017); for example, the use of the Arabic term محرك(muHarik) as an equivalent for the English term motor, associated with the Arabic root ح ر ك(Hrk/move). The Arabicized term موتور(mwutwur/motor), on the other hand, is not associated with any Arabic root.

Blending plays an influential role in handling affixations and abbreviations of long Arabic terms such as (لافقاري; invertebrate) and (كهروميغناطيسي; electromagnetic). However, there are restrictions on blending, and it may only be used for scientific necessity (Elmgrab 2011). These restrictions are due to the fact that in blending, there are no rules that must be followed during the process, while Arabic has specific rules and patterns that cannot be eliminated (Ali Al-foadi 2018).

Derivation is the best choice as suggested by many authors/works (Ali Al-foadi 2018). Indeed, as mentioned above, modifying the old word's meaning to fit the new one does not always work, and the terms created by Arabization and blending may be incompatible with Arabic (Al-Salih 1968). Therefore, some linguists suggest using new roots for new terms by deriving the corresponding Arabic words from these new roots (Abbas 1998).

It is worth noting that the methods for generating terms were proposed by linguists and not handled by natural language processing (NLP) researchers (Elmgrab 2011). As described in Section 2, most of the research in the NLP field concerned either collecting statistics on the used roots or studying the phonetic system of the Arabic language. How-

ever, the results of these efforts were not exploited to generate new terms (Musa 1978; Alm and Al-Faham 1983).

In order to help linguists propose new Arabic scientific terms using the generation strategy, this study aims to develop an algorithm that generates all possible triliteral roots, determines whether they are used or not, are phonetically accepted or not, and to what extent they are compatible with the phonetic system of the Arabic language. These roots can then be combined with patterns to generate new lexical forms that can be evaluated by lexicographers.

The rest of the paper is as follows. Section 2 reviews previous works. Section 3 describes the methodology, and Section 4 shows the results. The paper concludes in Section 5.

2. Related Work

Arab scholars have been interested in lexicography since ancient times and excelled in this field in variety and perfection. They have used various methods to collect and arrange vocabulary in lexicons (Omer 1995). Among the most famous Arabic lexicons are Al-Sahah (Attar 1987), Lisan Al-Arab (Al-kabeerm et al. 1981), Taj Al-Arous (Shiri 1994), Al-Wassit (Anees et al. 2004), and Al-Moassir (Omer 2008). In most of these lexicons, the vocabulary is divided into groups; each group belongs to the root from which it is derived. For example, the words مدرسة (mdrsp/school), دراسة (drAsp/study), and دارس (dArs/student) belong to the root د ر س (drs). The vocabulary size varies from one lexicon to another due to the differences in time and collection method; in each period, some terms appear and become popular, while the use of others decreases or disappears. Table 1 shows statistics about the roots of the mentioned Arabic lexicons.

Table 1. Lexicon statistics.

	Triliteral	Quadriliteral	Quinqueliteral	Total
Al-Sahah	4814—86% of total	766	38	5618
Lisan Al-Erab	6538—71%	2548	187	9273
Taj Al-Arous	7597—63%	4081	300	11,978
Al-Wassit	5155—78%	1332	153	6640
Al-Moassir	3292—67%	1092	535	4919

Table 1 shows that the most comprehensive lexicon is Taj Al-Arous, and the triliteral roots are the most used in the language. However, if we compare the used roots in these lexicons with the total number of roots that can be formulated from the twenty-eight letters of the Arabic language, it becomes clear that there is a large gap between them, as shown in Table 2.

Table 2. Letters Combinations Statistics.

	Possible Roots	Used Roots in Taj Al-Arous	Percentage
Triliteral	21,952	7597	34.6%
Quadriliteral	548,800	4081	0.74%
Quinqueliteral	9,765,625	300	0.003%

The total combinations of quadriliteral are 614,656, and quinqueliteral are 17,210,368, which are mathematically calculated by ($28 \times 28 \times 28 \times 28$) and ($28 \times 28 \times 28 \times 28 \times 28$), respectively. However, quadriliteral cannot start with vowel letters (أ/>, و/w and ي/y). Therefore, they are excluded from the first position, and the combinations are calculated

by (25 × 28 × 28 × 28), which equals 548,800. Quinqueliteral cannot contain any vowels; therefore, they are excluded from the letters set and combinations are calculated by (25 × 25 × 25 × 25 × 5), which equals 9,765,625 (Nowas et al. 2009). Regarding triliteral roots, there is no exclusion of any letter from any position; therefore, the number of possible combinations equals the number of possible roots.

The gap between the possible and used roots or words is a known linguistics phenomenon. The first Arabic scholar to notice the gap between the possible roots and the used roots was Al-Khalil, who called this phenomenon "Al Muhmal" (i.e., the unused) and explained that it is caused by difficulties in pronunciation (Kishli 1996). A pronunciation difficulty is an insufficient justification for unused combinations because most of them have no pronouncing difficulty. Therefore, many linguists after Al-Khalil studied this phenomenon to discover the reasons for the unused combinations. Ibn Duraid (Balabaki 1987) added the "disharmony of letters" as another reason. Ibn Jinni (Hindawi 1993) justified the unused combinations by arguing that there is no need for such terms or a lack of unison between the sounds that make up the root and the intended meaning.

The linguists of generative phonology study the unused combinations in the language. They categorized them into two groups: the first group contains words that do not exist because they do not obey the phonetic rules of the language (systematic gap). The second one contains the words that do not exist despite that which is permissible by the phonetic rules of that language (accidental gap). The most common examples of these groups are bnick and blick words, respectively (Chomsky and Halle 1968; Crystal 2011).

On the other hand, NLP researchers also studied and analyzed lexicons to know how Arabic words are formulated in order to use them in developing and expanding the language or to ensure the accuracy of the results obtained by ancient scholars. The first use of computers in Arabic linguistics was in the 1970s when (Musa 1978) conducted a statistical study on the roots of the Al-Sahah lexicon to investigate some linguistic phenomena. Al-fozan (Alfozan 1989) also devoted research to study, enumerate, and summarize the impossible combinations from ancient books. He collected more than 80 phonetic rules and corrected some rules addressed by Al-Khalil ibn Ahmed, such as the combination of the letters " أ/>" and "هـ/h", which are combined in the root "أهل/> hl".

Alm and Al-Faham (1983) studied the combination of letters using the bigram frequencies of Arabic roots. They wanted to verify the accuracy and completeness of the results obtained by Al-Khalil concerning letters that could not be combined in any Arabic root and resulted in many combinations not being used. By performing their experiment on five lexicons, they proved the strength of Al-Khalil's results.

Hegazi (2016) also shed light on the gap between the possible roots and the roots by creating a lexicon that includes all possible triliteral roots. In this lexicon, Arabic roots are generated by applying permutations to the Arabic letters. Then, he applied the Arabic patterns to the roots to obtain the words or vocabulary. The drawback to Hegazi's study is that he did not consider the combinations that were not used. He excluded only the 28 roots consisting of three redundant letters, then applied Arabic patterns to the 21,924 (21,952—28) remaining roots.

As we will show, NLP studies were conducted not only in theoretical terms but also in practice. For example, Abdoalrasool (2010) exploited the unused combinations in the context of optical character recognition to improve the output quality depending on the Arabic language features without using spell-checking or morphological analysis. Additionally, Abusair (2012) and Al-Radaideh and Masri (2011) combine Arabic bigrams with prediction methodologies to improve the writing of Arabic SMS messages on 12-key cell phones.

As we have already mentioned, languages are constantly evolving and expanding with the development of life. As one of the most widespread languages, the Arabic language must keep pace with this development and evolve in its own way without blurring its characteristics. In this regard, many researchers are exploring the characteristics, challenges,

and state-of-the-art of Arabic NLP (Habash 2010; Darwish et al. 2021; Imane et al. 2021) to explore different opportunities in using human language technology.

Indeed, language expansion is a matter of most languages. Many studies in the English language handled new terms and word formation, such as Aqchaboyevna (2020) and Yenikeyeva and Klymenko (2021). Additionally, there are several studies that have discussed the inclusion or creation of new terms in Arabic, such as the studies that were conducted by Kossmann (2013), Elmgrab (2016), and Hassan (2017). Kossmann talked about borrowing in Arabic as a means of expressing new concepts. He explained that Arabic has borrowed terminology since ancient times from several languages, such as Persian, Iranian, Greek, French, and English. He also explained that the rate of using borrowed words is somewhat rare outside the technical field because, in many languages, there is an explicit wish to keep the language free from foreign influences (Kossmann 2013). Elmgrab (2016) tried to find a suitable technique for creating new terms in Arabic, and Hassan (2017) proposed an approach to translate the new terms into Arabic automatically. Both studies found that the best strategy for introducing new terms in Arabic is derivation. As we have shown above, there is much research conducted by linguists and NLP researchers on Arabic lexicons. Most of them served the purpose of studying linguistic phenomena that characterize the Arabic language, while others used these phenomena for practical applications. However, the obtained results were not used to extend the language, especially through NLP tools. The work in this area was limited to linguists, although using NLP tools will significantly help.

In this context, Dwaidri (2010) points out in her book the necessity to create a bank for all Arabic roots to unify the Arabic lexicon in order to be used in expanding the language. This bank must include existing Arabic roots from well-known lexicons such as the Al-Sahah, Lisan Al-Arab, and Taj Al-Arous, as well as the unused and phonetically rejected roots.

In this work, an attempt is made to use these phenomena to create a lexicon that will help the linguists who use the generation strategy to propose new Arabic terms to develop and expand the language.

3. Methodology

This study presents an approach to generating all Arabic triliteral roots. For each generated root, we determine whether it is used in Arabic or not. For the unused roots, we explain whether they are accepted or not according to the Arabic phonetic system. Then, we assign a weight to each root indicating the compatibility of the root is compatible with the Arabic phonetic system. To do this, we proceed in several steps, as shown in Figure 1.

The first three modules are independent and can be run in parallel. Module 1 generates all combinations consisting of three of the 28 Arabic letters. Module 2 collects existing roots from the lexicons. The output of Module 1 and Module 2 are passed to Module 4 in order to mark each generated root from Module 1 as used or unused according to the output of Module 2.

Module 3 and Module 5 collect the letters that cannot be combined in a root. Most of these impossible letter combinations are addressed in ancient Arabic books, and the unaddressed ones are extracted from the existing roots.

According to the output of Module 5, Module 6 marks each generated root as accepted or rejected. Module 7 assigns a weight to each root to indicate the compatibility of the root with the Arabic phonetic system. Finally, we obtain a lexicon that contains all mathematically possible triliteral roots, which are assigned specific labels such as the acceptance and usage of the root in the language.

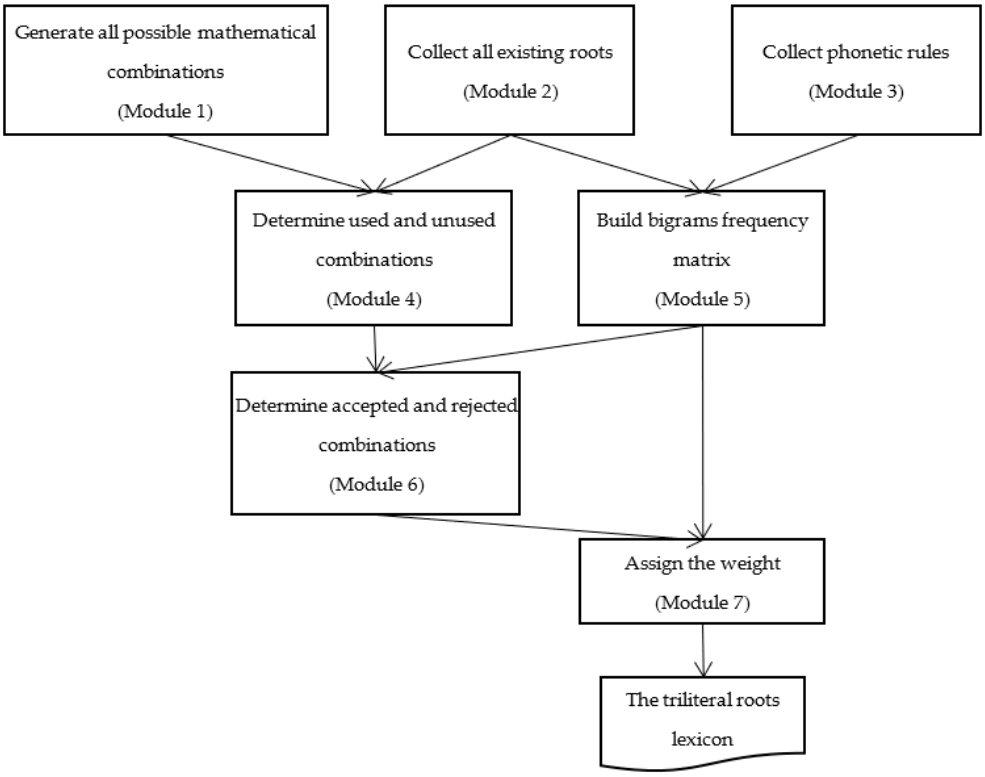

Figure 1. Proposed Approach.

Figure 2 shows a simple example of the output of each module from Figure 1. All modules are explained in detail in the following subsections.

3.1. Generating All Roots

The proposed generation algorithm is based on mathematical combinations where all possible triliteral combinations of the twenty-eight Arabic letters were generated, reaching a total number of 21,952 combinations ($28 \times 28 \times 28$).

The first generated root in Module 1 is "ءءء/>>>", followed by "ءءب/>> b" until it ends with the root "يي/yyy". Some generated roots are already used in Arabic, while others are not. To determine whether a root is used or not, we consider the five mentioned lexicons as explained below.

3.2. Collecting the Existing Roots

The triliteral roots are collected from five selected lexicons, as shown in Table 1, assuming they ensure completeness. After merging their roots and removing redundancy, we obtain 8426 distinct ones.

The existing roots are collected (in Module 2) to distinguish between used and unused ones (from Module 1) and obtain information about the phonetic system from the practiced language and how letters are combined to formulate the roots.

Figure 2 shows that the first existing triliteral root following the alphabetical order is "ب ا أ/<Ab" while the last one is "يي/yyy".

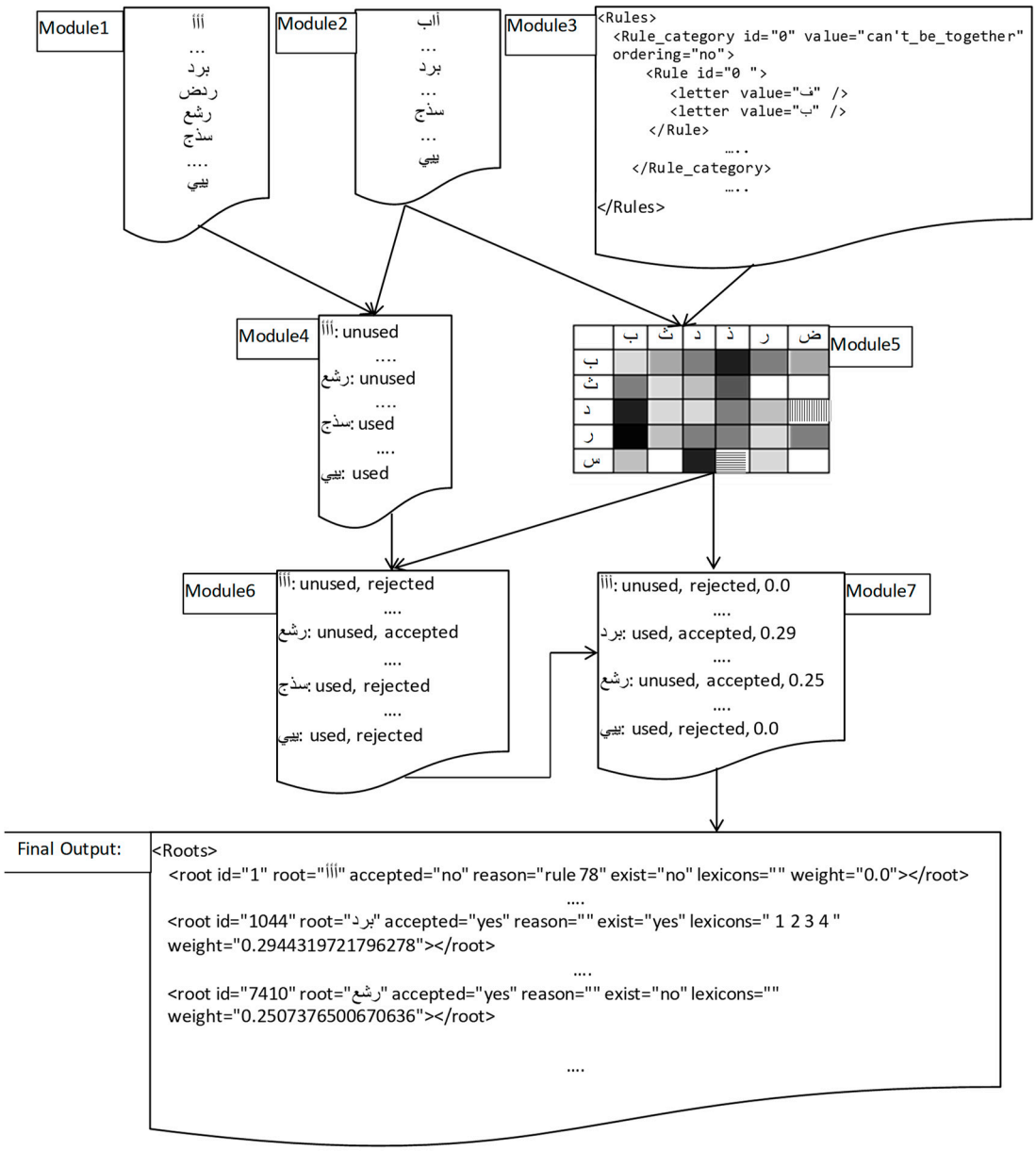

Figure 2. Modules Output Samples.

3.3. Collecting Phonetic Rules

As mentioned earlier, some letters cannot be combined in a root because of the difficulty of their pronunciation or their incompatibility with each other, such as the letters "س/s" and "ث/v". The roots that contain such an impossible combination are phonetically not accepted and must be excluded; this means there are phonetic rules that control the acceptance of the root in the language. These unused combinations were used as phonetic rules to recognize Arabicized roots.

As explained in the previous section, some modern linguists are interested in collecting phonetic rules (Alfozan 1989; Alm and Al-Faham 1983). Our effort in this regard is to organize the phonetic rules and put them into a standardized digital format that is accessible to everyone and easy to use. We have put all the addressed phonetic rules in an XML file. Each rule has an ID, a category, and letters that cannot be combined according to the rule. Figure 3 shows an example of the phonetic rules file.

```
<Rules>
    <Rule_category id="1" value="can't_be_together" ordering="no">
        <Rule id="1 ">
          <letter value="ف" />
          <letter value="ب" />
      </Rule>
            …..
    </Rule_category>
    <Rule_category id="2" value="can't_be_followed_by" ordering="yes">
        <Rule id="32 ">
          <letter value="س" order="1"/>
          <letter value="ش" order="2"/>
      </Rule>
            …..
    </Rule_category>
    <Rules_category id="3" value="composed_of_identical_letters">
      <Rule id="50" lett1="ا" lett2="ا" lett3="ا"></Rule>
            …..
    </Rules_category>
    <Rules_category id="4" value="start_with_identical_letters">
      <Rule id="78" lett1="ا" lett2="ا" ></Rule>
            …..
    </Rules_category>
</Rules>
```

Figure 3. Phonetic rules XML file example.

As can be seen in Figure 3, there are four categories of rules; the last two categories contain phonetic rules that apply to all letters, namely that the root must not consist of three identical letters "composed_of_identical_letters" and must not start with two repeating letters "start_with_identical_letters", such as "ففف\fff" and "ففر/ffr", respectively.

The first two categories, "can't_be_together" and "can't_be_followed_by," on the other hand, contain rules that prevent the co-occurrence of some letters in a root. For example, the letters "ف/f" and "ب/b" cannot be combined in a root, regardless of their order. So, this rule belongs to the "can't_be_together" category, where it does not matter which of the two letters precedes the other.

The letter "د/d" cannot be followed by the letter "ت/t" in any root, whereas the letter "ت/t" can be followed by the letter "د/d", as in "وتد/wtd". So, this rule belongs to the "can't_be_followed_by" category, where the letters can be combined in a root only in a specific order.

Nevertheless, not all phonetic rules are addressed in the ancient books due to the lack of capabilities at that time. Therefore, the unaddressed rules are extracted by analyzing the combinations in existing roots using a bigram frequency matrix. One can think that building a trigram matrix might also be of interest. However, this is not applicable since Arabic phonological rules concern the homogeneity of two letters only. This is explained in more detail in the next section.

3.4. Building Bigrams Frequency Matrix

In the context of natural language processing, a bigram is a sequence of two adjacent elements from a string of tokens, usually letters, syllables, or words. The frequency distribution of each bigram in a string is used in many applications, such as computational linguistics and speech recognition for statistical text analysis.

In order to obtain the bigram frequencies from Arabic lexicons, a 28 × 28 matrix is created. Each row and column represents an Arabic letter. The cell where the rows and columns intersect indicates how often these two letters occur in all lexicon entries. In order to fill the matrix, each root is split into three bigrams. For example, the root كتب is split into تب, كت and كب, and the cell corresponding to each bigram is incremented by one. The corresponding cell for the bigram تب is the cell located at the intersection of the row representing the letter ت and the column representing the letter ب.

For a more detailed representation, we obtain three matrices. The first matrix represents the first bigram (the bigram representing the letters in the first and second positions, كت in the previous example), and the second matrix represents the second bigram (تب in the previous example). In contrast, the third matrix represents the first and third bigram (كب in the previous example). Moreover, we can combine the three matrices into one matrix to get a global view of the frequency of bigrams. The bigram frequency matrix is statistically known as the correlation matrix.

The bigram frequency matrix can be represented in the form of a heatmap, which is a graphical representation of data where values are represented by colors and/or textures. The heatmap makes it easier to visualize and understand the data at a glance. Figure 4 shows the correlation matrix between Arabic bigrams extracted from five lexicons and visualized using the heatmap. The letters in the axes of the heatmap are represented using the International Phonetic Alphabet (IPA).

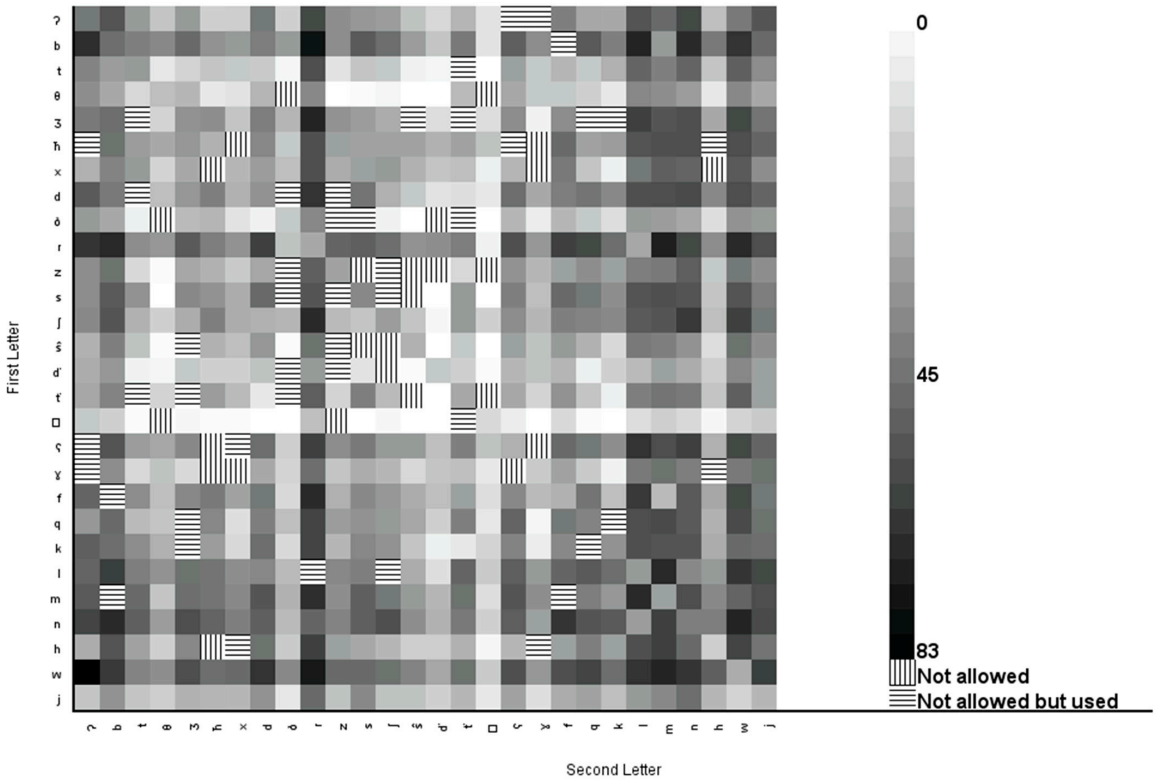

Figure 4. Arabic Roots Bigram Frequencies.

The darkest cell means this bigram is more frequent, and the frequency decreases when the cell is lighter. A white cell means the corresponding bigram does not occur in any root. For example, the frequency of the bigram طب is less than the frequency of the ر ب, the frequency of the bigram ذهـ is less than the two bigrams mentioned before, while the frequency of the bigram ظش is zero (shown in white), which means that there is no existing root containing ش and ظ.

As explained earlier, to obtain all phonetic rules, we cannot rely only on the addressed rules, since they are not complete. We also cannot rely only on the root analysis result since some Arabicized roots contain impossible letter combinations, such as "سذج/s*j", and such roots may affect the analysis result. Therefore, the root analysis process must include information about the addressed phonetic rules to avoid the effects of such exceptions. Therefore, the addressed rules appear in the bigrams frequency matrix as vertical and horizontal stripes.

The vertical stripes indicate that the corresponding bigram is not allowed by the addressed phonetic rules of Arabic, such as ثذ. However, some Arabicized roots may contain non-allowed bigrams, and horizontal stripes indicate such cases. For example, although there is a rule prohibiting the combination of س and ذ in a root, we found the bigram سذ extracted from the Arabized root سذج in four of the five selected lexicons.

As mentioned above, many bigrams in Arabic cannot occur together in one root. Many of them are addressed in Arabic books and are denoted in vertical stripes in the heatmap. However, the heatmap helps identify unaddressed bigrams because they are shown in white color. There are about 84 addressed phonetic rules, while there are 107 rules extracted from the matrix; this means there are more than 20 rules that are not addressed. We create an XML file with the Arabic phonetic rules, whether they are addressed or obtained by analyzing the used roots[2].

To determine whether the generated root of Module 4 is phonologically acceptable, we divide the root into bigrams and then compare these bigrams with the bigram frequency matrix. If one of the root bigrams corresponds to the white or striped cell, then that root is phonetically unacceptable. Otherwise, it is phonetically acceptable. For example, in Figure 2, the bigrams "ثض", "عح", and "مف" are phonetically unacceptable because they correspond to white, vertically striped, and horizontally striped cells, respectively.

Arabic phoneticians divide the degree of acceptability of root sounds into three types: suitable, less suitable, and unsuitable. This means not all phonetically acceptable roots have the same degree, but some are preferred over others depending on the letters that compose them.

With the suitable type, pronunciation is not difficult because the sounds are articulated far apart. An example of this type is the root م ل أ (alm). The less suitable type contains two identical letters, such as ك ك م (makk) and ب ب س (sabb) (Frisch et al. 2004). The unsuitable type is the one that contains sounds that are difficult to combine because they are articulated very closely, especially those that are articulated in the throat, such as ع ح ه (hHE) (Hindawi 1993). The next step is to assign a weight to each root expressing this degree in numbers.

3.5. Assigning the Weight

As previously explained, ease of pronunciation has been expressed by linguists in rules representing the possibility of the coexistence (or non-coexistence) of two letters in a root (Kishli 1996). Therefore, the idea is to calculate the weight of a root by calculating the weight of each of its three bi-grams. To do so, we first assign a weight to each bigram individually and then combine these weights to calculate the global weight of the root.

We use probability theory to assign a weight to a bigram (Sherlock and Ormell 1970). The weight of the bigram is calculated as follows:

$$w_{(xy)} = \frac{freq_{(xy)}}{freq_{(bigrams)}} \quad (1)$$

where

$w_{(xy)}$: weight of the bigram (xy)
$freq_{(xy)}$: frequency of the bigram (xy)
$freq_{(bigrams)}$: frequencies of all bigrams.

The frequency of the bigram is obtained from the corresponding cell in the bigram frequency matrix, while the frequency of all bigrams is obtained by summing up the frequencies from the corresponding matrix.

After assigning a weight to each bigram, the next step is to aggregate these weights into a value that is assigned to the root. The aggregation formula consists of multiplying the weights of the bigrams, as in the following equation.

$$w_{(root)} = (w_{12} * w_{23} * w_{13}) \quad (2)$$

where

$w_{(root)}$: weight of the root
w_{12} : weight of the first and second letters bigram
w_{23} : weight of the second and third letters of bigram
w_{13} : weight of the first and third letters bigram.

Multiplication ensures that the value of the total weight of the root is high only if the values of all bigrams are high, and if one bigram is un-accepted, the value of the root weight is zero.

According to the proposed weighting scheme, the unused roots "حتس" and "رشع" have a weighting value of 0.01 and 0.07, respectively, while the used roots "رأي" and "شرب" have a weighting value of 0.54 and 0.37, respectively. The roots that violate any of the phonetic rules have a weighting value of zero, regardless of whether they are used in the Arabic language or not.

In order to accomplish these tasks, we used SAFAR framework (Bouzoubaa et al. 2021), which is a monolingual NLP framework dedicated to the Arabic language. SAFAR possesses more than 50 tools and resources that can be exploited either using its API or web interface. Among the components that were actually used in the current work context, we can mention normalization, lemmatization, stopwords removal, and pattern detection, as well as resources such as a machine-readable version of Al-wassit and Al-moassir lexicons.

4. Results

Through this work, we want to build a lexicon containing all triliteral combinations, determining which ones are phonetically rejected, which ones are used, and which ones are available to be used by linguists to extend the language. In order to achieve our primary goal, we have gone through several stages; some of them had intermediate results, such as the Arabic phonetic rules file. These results are available to researchers in the field, as explained earlier. The main result is a lexicon of triliteral roots, as shown in Figure 5, where each root has several attributes.

```
<Roots>
    <root id="1" root="ااا" accepted="no" reason="rule 1" exist="no" lexicons="" weight="0.0"></root>
                            ……
    <root id="29" root="ابا" accepted="yes" reason="" exist="yes" lexicons=" 1   2   4 " weight="0.68"></root>
    <root id="30" root="أبب" accepted="yes" reason="" exist="yes" lexicons="  1    2    3    4    5  "
weight="0.25"></root>
                            …..
    <root id="7410" root="رشع" accepted="yes" reason="" exist="no" lexicons="" weight="0.07"></root>


                            …..
    <root id="8853" root="سذج" accepted="no" reason="rule 84" exist="yes" lexicons="  1    2    4    5  "
weight="0.0"></root>
                            …..
    <root id="21952" root="ييي" accepted="no" reason="rule 28" exist="yes" lexicons=" 1 " weight="0.0"></root>
</Roots>


<Rules>
     <Rules_Category cat_id="1" value="composed of identical letters">
          <Rule id="0" letter1="ا" letter2="ا" letter3="ا"></Rule>
        ….
     </Rules_Category>
        ….
</Rules>


<Lexicons>
```

Figure 5. Structure and content sample of the proposed lexicon.

The first attribute, "id", is the root's identification number, which has a value between 1 and 21,952, based on the root's alphabetical order. The "root" attribute is a three-letter combination of Arabic letters. The "accepted" attribute determines whether the root is acceptable or not according to the phonetic system of the Arabic language. If the root is phonetically rejected, the reason for rejection is explained in the "reason" attribute by specifying the ID of the phonetic rule that the root violated.

The "exists" attribute determines whether the root is used in the language and is present in the Arabic lexicons or not. If it is used, the lexicon attribute contains the IDs of the lexicons that contain the root. If the root is not used, the value of the lexicon attribute is empty. The last attribute is the "weight", whose value determines the root's compatibility with the Arabic language's phonetic system. If the root violates one or more phonetic rules, the value of the weight attribute is zero, even if the root is used.

For example, the first root "ااا" has one id and is not accepted according to rule 1, which states that the root must not consist of three repeating letters. This root is not used in Arabic, therefore, the value of its "exists" attribute is equal to zero, and the value of its "lexicons" attribute is empty. Since the root violates a phonetic rule, the value of its "weight" attribute is zero.

As previously mentioned, not all triliteral combinations are used. Some are not subject to the Arabic phonetic system, while others are phonetically accepted. Some linguists have advocated using these roots to expand the language rather than borrowing many terms that could blur the language (Abbas 1998).

Applying permutation to the twenty-eight letters of the Arabic alphabet yields 21,952 three-letter combinations that can be divided into two main categories: phonetically accepted and phonetically rejected. Each of these categories is, in turn, divided into used and unused. This results in four categories: phonetically accepted used category, phonetically accepted unused category, phonetically rejected used category, and phonetically rejected unused category. Table 3 provides statistics for each of these categories.

Table 3. Three-Letter Combinations Statistics.

All possible combinations 21,952			
Phonetically accepted combinations 13,410		Phonetically rejected combinations 8542	
Unused 5383	Used 8027	Used 399	Unused 8143
8426			

The phonetically accepted used category (8027) forms the vast majority of the current language; the phonetically rejected used category includes the exceptions in the current language (399), such as Arabicized roots. Thus, the current Arabic language uses 8426 (8027+399) forms. In turn, the phonetically rejected unused roots category (8143) contains the roots that do not follow the Arabic phonetic system and are not used, and the phonetically accepted unused category (5383) can be used to expand the language.

This last number (more than 5300) shows that a wide range of roots is accepted and not used and can be used to extend the language. If we compare the number of words in the lexicon with the number of roots, there are, on average, 13 words derived from a root. This means that unused roots can produce as many as 70,000 new words. Words are generated from the accepted-unused-roots by applying Arabic patterns. For example, some of the words that can be derived from the root "رشع/r$E" are "راشع/rA$iE", "مرشوع/mr$uwE", "مرشعة/mir$Ep", "مرشاع/mir$AE", and "رشعة/r$Ep".

5. Conclusions

In this paper, we have attempted to provide researchers with a comprehensive triliteral root lexicon containing information on what is used, what may not be used, and what can be used to extend the language. We relied on a mathematical combination and permutation theory to generate the roots to ensure that all roots are processed. Then, we merged five Arabic lexicons to know which roots are actually used. To determine the acceptability of each root, we used a bigram frequency approach based on the merged lexicon to create a corresponding heatmap matrix. In addition to the linguistically addressed phonetic rules, this matrix is used to (i) extract other phonetic rules on the one hand and (ii) calculate the weight of the roots on the other hand, which indicates the compatibility of the root with the Arabic phonetic system. The results show that there is a large space of available combinations that can be used by linguists to extend the language. Future research is needed to determine how researchers can use this space to extend the language and how to assign meaning to each root.

Author Contributions: Conceptualization, E.M. and K.B.; methodology, E.M. and K.B.; software, K.B.; validation, E.M. and K.B.; formal analysis, E.M. and K.B.; investigation, E.M. and K.B.; resources, E.M. and K.B.; data curation, E.M.; writing—original draft preparation, E.M.; writing—review and editing, E.M. and K.B.; visualization, E.M.; supervision, K.B.; project administration, K.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All research files are available at http://arabic.emi.ac.ma/alelm/#Resources (accessed on 4 March 2023).

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

Notes

[1] All words transliterated according to Buckwalter transliteration (Buckwalter 2002).

[2] The file is available at http://arabic.emi.ac.ma/alelm/#Resources/ (accessed on 4 March 2023).

References

Abbas, Hassan. 1998. *Khasais Al-Horoof Al-Arbea Wa Maaneeha*, 1st ed. Damascus: Etihad Al-Kottab Al-Arb.

Abdoalrasool, Amro Jumaa. 2010. Tatweer Alta'rof Alali Ala Alhoroof Alarabiea Min Khilal Aliea Loghawiea. In *International Computing Conference in Arabic*. Edited by Yasmine Hammamet, Moncef Charfi and Hani Ammar. Tunisia: Phillips Publishing. Available online: http://www.phillips-publishing.com/ (accessed on 7 October 2020).

Abusair, Mai I. 2012. Improving Arabic Text Entry Methods Using Word Bigrams Prediction And Keys Reassignment. Paper presented at International Conference on Intelligent Computational Systems, Dubai, United Arab Emirates, January 7–8.

Alfozan, Abdulrahman Ibrahim. 1989. *Assimilation in Classical Arabic: A Phonological Study*. Scotland: University of Glasgow.

Al-Huri, Ibrahim. 2015. Arabic Language: Historic and Sociolinguistic Characteristics. *English Literature and Language Review* 1: 28–36.

Ali Al-foadi, Raheem. 2018. Derivation as the Main Way of Adapting New Terms to Arabic. *Modern Journal of Language Teaching Methods (MJLTM)* 8: 194–99.

Al-kabeerm, Abdollah, Mohammed Ahmed Hasboallah, and Hashim Al-shazli. 1981. *Lisan Al-Arab Li Ibn Manzour*. Cairo: Dar Al-Maarif. Available online: www.lesanarab.com (accessed on 4 March 2023).

Alm, Yahya Meer, and Shakir Mohammed Al-Faham. 1983. *Derasa Ehsaiea Lidwaran Alhoroof Fi Aljozoor Al-Arabiea*. Damascus: Damascus University.

Al-Radaideh, Qasem A., and Kamal H. Masri. 2011. Improving Mobile Multi-Tap Text Entry for Arabic Language. *Computer Standards & Interfaces* 33: 108–13.

Al-Salih, Subhi. 1968. *Dirasast Fi Fiqh Al-Lugha*, 3rd ed. Lebanon: Dar al-ilm.

Al-Shbiel, Abeer Obeid. 2017. Arabization and Its Effect on the Arabic Language. *Journal of Language Teaching and Research* 8: 469. [CrossRef]

Anees, Ibrahim, Abdoalhaleem Muntasir, Ateea Al-Swalhi, and Mohammed Khalf-Allah Ahmed. 2004. *Al-Mujam Al-Waseet*, 4th ed. Cairo: MujamE Allogha Alarbeia-maktabat Alshorooq Aldowaliea.

Aqchaboyevna, Xasanova Mahfuza. 2020. Word-formation in modern english. *Science and Education* 1: 174–76.

Attar, Ahmed AbdoAlghafoor. 1987. *Kitab Al-Sahah Li Aljawhary*, 4th ed. Bayrut: Dar al-ilm.

Balabaki, Ramzi Monir. 1987. *Jamhrat Al-Logha Li Ibn Duraid*, 1st ed. Bayrut: Dar al-ilm.

Bouzoubaa, Karim, Younes Jaafar, Driss Namly, Ridouane Tachicart, Rachida Tajmout, Hakima Khamar, Hamid Jaafar, Si Lhoussain Aouragh, and Abdellah Yousfi. 2021. A Description and Demonstration of SAFAR Framework. Paper presented at the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Kiev, Ukraine, April 19–23; pp. 127–34.

Brakhw, Abobaker Ali, and Rabea Mansur Milad. 2019. Appropriate strategies to transfer neologisms from english into arabic. *International Journal of Research in Humanities, Arts and Literature* 7: 351–60.

Buckwalter, Tim. 2002. *Buckwalter Arabic Morphological Analyzer Version 1.0. Linguistic Data Consortium*. Philadelphia: University of Pennsylvania.

Chomsky, Noam, and Morris Halle. 1968. *The Sound Pattern of English*. New York: Harper & Row.

Crystal, David. 2011. *A Dictionary of Linguistics and Phonetics*. New York: John Wiley & Sons.

Darwish, Kareem, Nizar Habash, Mourad Abbas, Hend Al-Khalifa, Huseein T. Al-Natsheh, Samhaa R. El-Beltagy, Houda Bouamor, Karim Bouzoubaa, Violetta Cavalli-Sforza, Wassim El-Hajj, and et al. 2021. A Panoramic Survey of Natural Language Processing in the Arab World. *Communications of the ACM* 64. [CrossRef]

Dwaidri, Rajaa Waheed. 2010. *Al-Mostalah Al-Elmi Fi Al-Logha Al-Arabiea, Omqaho Al-Turathi Wa Boadho Al-Moassir*, 1st ed. Damascus: Dar al-fikr.

Elmgrab, Ramadan Ahmed. 2011. Methods of Creating and Introducing New Terms in Arabic. *IPEDR-International Proceedings of Economics Development and Research* 26: 491–500.

Elmgrab, Ramadan Ahmed. 2016. The Creation of Terminology in Arabic. *American International Journal of Contemporary Research* 6: 75–85.
Frisch, Stefan A., Janet B. Pierrehumbert, and Michael B. Broe. 2004. Similarity Avoidance and the OCP. *Natural Language & Linguistic Theory* 22: 179–228.
Habash, Nizar. 2010. *Introduction to Arabic Natural Language Processing*. New York: Columbia University.
Hassan, Sameh Saad. 2017. Translating Technical Terms into Arabic: Microsoft Terminology Collection (English-Arabic) as an Example. *Translation & Interpreting* 9: 67–86.
Hegazi, Mohamed Osman. 2016. An Approach for Arabic Root Generating and Lexicon Development. *International Journal of Computer Science and Network (IJCSNS)* 16: 9.
Hindawi, Hassan. 1993. *Sir Sinaat Al-Erab Li Ibn Jinni*. Damascus: Dar Al-Qalam.
Imane, Guellil, Houda Saâdane, Faical Azouaou, Billel Gueni, and Nouvel Damien. 2021. Arabic Natural Language Processing: An Overview. *Journal of King Saud University-Computer and Information Sciences* 33: 497–507.
Kishli, Ḥikmat. 1996. *Kitab Alain Lil-Khalil Ibn Ahmed Al-Farahidi*. Bayrut: Dar Al-Kutub Al-Ilmiyah.
Kossmann, Maarten. 2013. Borrowing. In *The Oxford Handbook of Arabic Linguistics*. Edited by Jonathan Owens. Oxford: Oxford University Press, pp. 349–68.
Musa, Ali Hilmi. 1978. *Dirasa Ihsaeia Lijzoor Muajm Al-Sahah Bistikhdam Al-Computer*, 1st ed. Ciro: Al-haiaa Al-masriea Al-ama lilkitab.
Nowas, Kefah Ibrahim Mahmoud, Yahia Jabr, and Mohammed Alnory. 2009. *Zahirat Al-Osool Almuhmala Fi Alarabiea Abadoha Wa Elaloha*. Nanlus: Alnajah Alwataneia.
Omer, Ahmed Mukhtar. 1995. *Muhadrat Fi Ilm Alloghah Al-Hadeeth*. Ciro: Ealm Alkutub Lilnashr wa altawzeee wa altebaea.
Omer, Ahmed Mukhtar. 2008. *Mujam Al-Logha Al-Arabea Al-Moassira*, 1st ed. Cairo: Aalm Al-Kutub.
Sherlock, Alan, and C. P. Ormell. 1970. *An Introduction to Probability and Statistics. The Mathematical Gazette*. Cambridge: Cambridge University Press, vol. 54. [CrossRef]
Shiri, Ali. 1994. *Taj Al-Arous Min Jawahir Al-Qamoos Li Al-Zobaidy*. Bayrut: Dar Al-Fikr.
Yenikeyeva, Saniya, and Olga Klymenko. 2021. Synergy of Modern English Word-Formation System. *Linguistics and Culture Review* 5. [CrossRef]