CLASSICAL ARABIC POETRY CATEGORIZATION USING N-GRAM FREQUENCY STATISTICS

Iqbal AbdulBaki Mohammad
College of Technical Medical & Health, Foundation of Technical Education, Baghdad - Iraq

Abstract

Most of the Arabic language vocabulary is built from the roots derivation. These roots are words composed of three to five consonants letters. Any performance in Arabic language for the purpose of information retrieval needs to deal with the language morphological and structural changes first (which is called the stemming process) then a statistical method for extracting information is implemented. This approach presents a method for categorizing the Classical Arabic Poetry (CAP) into its categorizations: Ghazal, Medeh, Wasef, Hijaa',..etc. by combining the algorithm of a light stemmer (which identify sets of prefixes and suffixes in an Arabic word in order to reach to the word root after removing the suffixes and prefixes) with "N-gram" statistical method (which retrieves the information independently of the language complexity). Two measures will be implemented: the "Manhattan distance" dissimilarity coefficient and the "Dice's measure" similarity coefficient for the purpose of categorization.

تصنيف الشعر العربي الكلاسيكي بأستخدام ترددات N-Gram الأحصائية

اقبال عبد الباقي محمد
كلية التقنيات الطبية والصحية، هيئة التعليم التقني ، بغداد – العراق.

الخلاصة

معظم مفردات اللغة العربية مبنية من أشتقاقات جذور الكلمات. هذه الجذور هي كلمات مؤلفة من ثلاثة الى خمسة أحرف ساكنة. أي عملية تجرى على اللغة العربية لأغراض أسترجاع المعلومات تتطلب التعامل مع صرفيات اللغة وتغييرات بنائها أولاً (هذه العملية تسمى التجذير) ثم نستخدم طريقة أحصائية لأسترجاع المعلومات. هذا البحث يقدم طريقة لتصنيف الشعر العربي الكلاسيكي الى أصنافه المعروفة وهي: الغزل، المدح، الوصف، الهجاء،... الخ وذلك باستخدام كل من خوارزمية التجذير الخفيف (التي تميز مجاميع من الأضافات الأمامية والنهائية في الكلمة العربية ومن ثم حذفها لغرض الوصول الى جذر الكلمة) مع طريقة N-gram الأحصائية (والتي تسترجع المعلومات دون الخوض في تعقيدات اللغة). نوعين من القياسات سوف يتم أستخدامهما وهما مسافة Manhattan المعامل الغير مماثل، وقياس Dice وهو المعامل المماثل لأغراض التصنيف.

Introduction

Arabic language is a real complex and rich in nature and for this any development in text categorization systems become a challenging task. There are many problems with Arabic language like various spelling of certain words, irregular and inflected derived forms, short diacritics and long vowels, and most of its words contain affixes. It consists of 28 letters and written from right to left. It has a very complex morphology that made its major words have a tri-letter root and the rest have quad or penta or hexa -letter root. The Classical Arabic Poetry CAP is written in a certain way it has got a verse which is known as bayt or abyat, and is divided into two halves also known as shater or shatrayn.

Text categorization is the process of structuring a set of poems according to a group structure that is known in advance [1]. The aim of this paper is to propose a preliminary categorization of CAP using a method of two steps the first is called "Stemmer" which requires specific knowledge about the language [2]. The second is called "N-gram" which is a statistical approach for categorization.

Stemming is used to reduce variant word forms to common roots and thereby improve the ability of the system to match query and vocabulary [3]. It makes the text compact and easy to process. The N-gram frequency profiles provide a simple and reliable way to categorize documents in a wide range of categorization tasks. It can be found if two words are semantically similar or dissimilar from the structures of characters of these words [4].

Many research works used the N-gram method in categorizing Arabic text like [5] who developed the first automatic classification technique based on the character structure of words. Dice's similarity coefficient is computed from the number of matching bi-grams (2-gram) in pairs of character strings, and used to cluster sets of character strings. [6] used tri-grams for indexing Arabic documents without any prior stemming. [7] used N-grams with and without stemming for text searching. Their results indicated that the use of tri-grams combined with stemming improved the performance of search retrieval. [8] assessed the performance of two N-gram matching techniques for Arabic root-driven string searching. [9] used N-grams for searching Arabic text documents. They investigated di-grams and tri-grams without using stemming. They concluded that the N-gram technique is not an efficient approach to corpus-based Arabic word conflation. [1] used N-gram frequency statistics technique for classifying Arabic text documents. For each document to be classified, the N-gram frequency profile was generated and compared against the N-gram frequency profiles of all the training classes. The comparison is done by calculating Manhattan distance and Dice's measure. [10] presented the N-gram model which can be used to compute the similarity between two strings by counting the number of similar N-grams they share. [11] presented an approach that uses N-gram based on the word and characters. Four basic types have been explored either separated or combined: word, lexical root, root, and N-gram.

This paper is organized as follows: the stemming algorithm will be discussed in section two, while in section three the N-gram concept will be explained. A detailed description of the whole system will be discussed in section four. Finally the results followed by the conclusion will be presented.

Stemmer

Stemmer is an automatic process used to reduce the different morphological forms of words into common root (Stem) to improve the performance of the extraction system [4]. The light stemmer approach that presented by A. Chen and F. Gey [12] will be used. They applied the following rules:

If the word is at least five-character long, remove the first three characters if they are one of the following: وال، لال، سال، اال، مال، وللّ، كال، فال، بال. (like كالأبِ will be أب).

If the word is at least four-character long, remove the first two characters if they are one of the following: وا، ال، فا، كا، ول، وي، وس، سي، لا، وب، وت، وم، للّ، با (like سيأنُ will be أنَّ).

If the word is at least four-character long and begins with و, remove the initial letter و. (like ومالَ will be مالَ).

If the word is at least four-character long and begins with either ب or ل, remove ب or ل.(like ليتمَّ will be يتمَّ).

Recursively strips the following two-character suffixes in the order of presentation if the word is at least four characters long before removing a suffix: ون، ات، ان، ين، تن، تم، كن، كم، هن، يا، ني، وا، ما، نا، هم، ية، ها (like شدَّها will be شدَّ).

Recursively strips the following one-character suffixes in the order of presentation if the word is at least three-character long before removing a suffix: ت، ي، ه، ة. (like حنّتْ will be حنَّ).

N-grams

N-gram is a sub-sequence of N-items in any given sequence, where the grams are characters of words. It is N-character slice of a long string. A word is leading or trailing by spaces and these spaces can represent a sequence of N-grams. The value of N can be chosen for a particular corpus. The word مكتوب can be composed of the following N-grams:

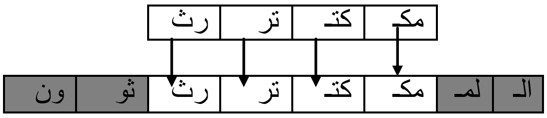
Bi-grams: _مـ ، مكـ ، كتـ ، تو ، وب ، ب_

Tri-grams: _مكـ ، مكتـ ، كتو ، توب ، وب_ ، ب__

Quad-grams:
مكتـ ، مكتو ، كتوب ، توب ، وب__ ، ب___

The advantages of N-gram are that it does not require a preliminary knowledge of the language, does not require predefined rules, and does not require the construction of a database of vocabulary [4].

Arabic nouns and verbs are heavily prefixed and suffixed, and as a result it is possible to have words with different lengths that share same principal concept. Two words are considered similar if they have in common several substring of N-characters, this is done by calculating a coefficient on these two words. The following bi-gram example shows the similarity between the two wordsالمكترثون ، مكترث :



Human languages invariably have some words which occur more frequently than others. One of the most common ways of expressing this idea has become known as Zipf's Law [13], which is stated as follows:-

"The nth most common word in a human language text occurs with a frequency inversely proportional to n"

This has the implication that poems belonging to the same category will have similar N-gram frequency distributions. Figure 1 shows the Tri-gram frequency distribution for a poem belongs to the Ghazal category from our dataset. It clearly shows that the frequencies of the most common Tri-grams are inversely proportional to their ranks.

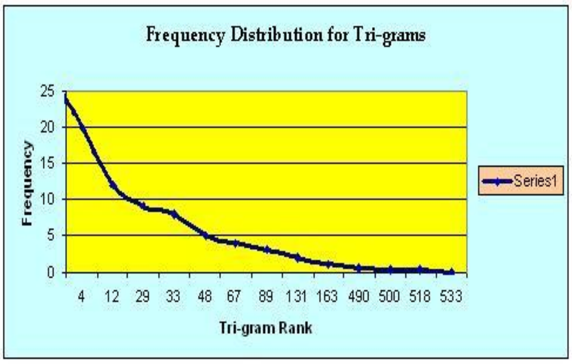


Figure 1: Frequency Distribution of Tri-grams

CAP Categorization

The CAP Categorization compromises three phases: poem pre-processing, frequency profile generation, and poem categorization. The pre-processing phase is to make the poems compact and applicable to train the poem categorization. While the frequency profile generation, the core of the system statistical method, shall be constructed using the compact form of the Arabic poems. Then the poem categorization shall be evaluated by two measures after generating the frequency profiles. The following subsections are devoted to these three phases.

1. CAP Pre-Processing Phase

This phase aims to transform the CAP text file to a form that is suitable for the categorization algorithm. For this work the poems from well known books of Arabic literature have been collected [14][15][16], and managed in collecting them that they cover the eight datasets Hekmah, Fakher, Retha'a, Wasef, Naseeb, Madeh, Ghazal, Heja'a, for each dataset a reference has been given to it in order to distinguish each dataset from the other so for Hekmah the reference Cat1 has been given, Fakher Cat2 and so on.

For each category six verses have been collected from five poetries (that belongs to the same category) and merged them in one text file. A normalization process has been implemented before applying the stem process. This step is essential because any diacritics like (، ’ ، ◌ٍ ، ◌ِ ◌ٌ ، ◌ُ ، ◌ً ، ◌َ) and the variation of the letter forms according to its positions take an important role in such systems.

Stemming process applied to a document is quite different when it is applied to a poem. Any poem is written without using punctuation marks and there are no numbers written in a poem also there is no non Arabic letters in the CAP. This will reduce the process of the normalization phase, the only thing have been removed are the tatweel character "—", stop words which include Arabic prefixes, pronouns and prepositions like (كيف، أين، في، من، علـى، هن، نحـن) and the diacritics. The most important thing to be removed is the conjunction (و) between any two words [12]. Afterwards the following replacements have been done:-

Replacing أ, إ and آ by alif bar ا.
Replacing ى by ي at the end of the words.
Replacing ة by ه at the end of the words.
Replacing the sequence يء by ئ.

After completing the normalization process the light stemmer has been applied according to rules mentioned in section two. The result is a very small size text file that is very reliable for the process of categorization. In this work the several spaces between the poem two halves have been removed and placed with one space only, also the enter key has not been used between the lines of the text file, this will be useful for the second phase.

2. Frequency Profile Generation Phase

The poems represented in the collected eight categories are of a very small size as six verses have been taken from five poems, which mean thirty verses for each category. Twenty verses have been taken for the training process and saved in a text file separated from the rest ten verses which have been used for the testing process. The preprocessing phase that explained in subsection 4.1 is applied to the training and the testing text files. In this phase the N-gram profile must be generated according to the work of [17] shown in figure 2, the flowchart was followed in order to generate the frequency profiles.

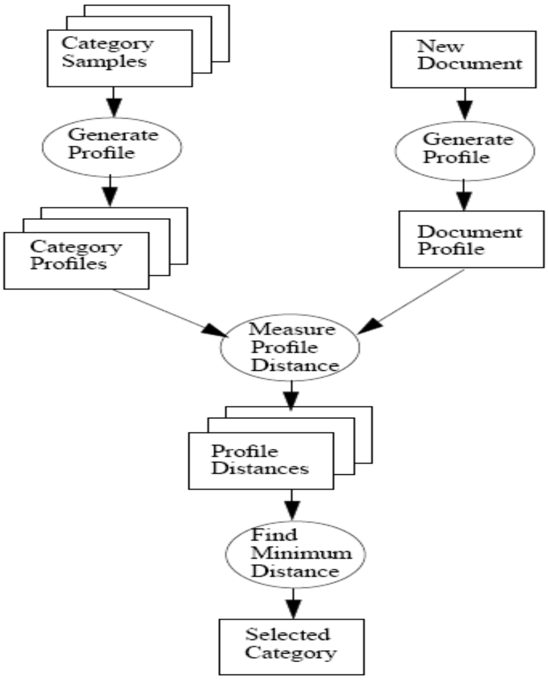


Figure 2: Dataflow for N-gram based text categorization [17].

For all text files that cover the eight categories the N-gram was computed (for N=3 Tri-grams) after scanning every letter. A tri-gram that has a space in the middle, in the beginning and at the end of the word has been omitted. Then inserted into a table to find the counter for the N-gram, and increment it whenever a similar N-gram occurred. From this counter the frequency of occurrence of each N-gram has been computed. The frequencies were saved in a separate file and a descending sort for them has been made (from the most frequent to least frequent). This result is the frequency profile of the text file.

3. CAP Gategorization Phase

After generating the profile of both the training and testing datasets the profile distance has been measured. This measure determines how far out of place an N-gram in the training profile is from its place in the testing profile in terms of similarity and dissimilarity. Two measures have been used the first is the distance or dissimilarity measure called "Manhattan distance". It calculates a rank order statistics for two profiles by measuring the difference in the positions of an N-gram in two different profiles. For each N-gram in the testing profile, a search for the same N-gram in the training category profile has been done to calculate the difference between their positions. For N-grams not found in the training category profile, a maximum value is assigned. After evaluating all N-grams in the testing profile, the sum of the distance measures is computed. Figure 3 shows a sample of how calculating this measure.

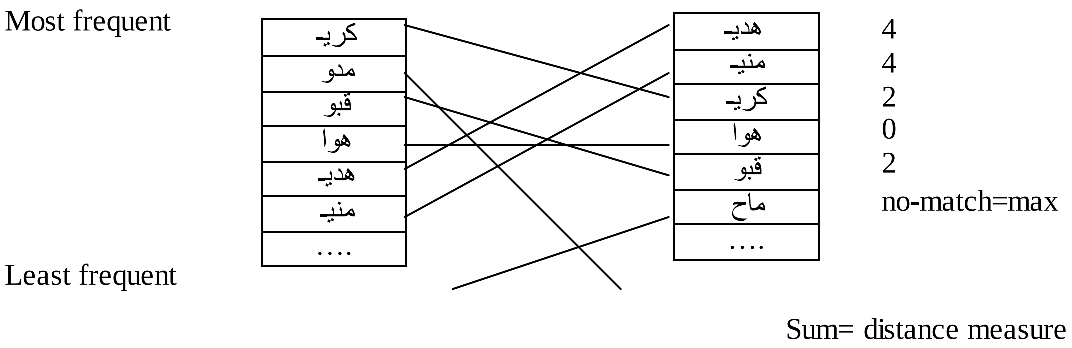


Figure 3: A sample for calculating the distance measure between two profiles

Using the equation (1) for evaluating the distance measure:-

Manhattan $$(P_i, P_j) = \sum_{h=1}^{k} \left| (P_{ih} - P_{jh}) \right| \quad \ldots\ldots\ldots (1)$$

Where Pi, Pj represent two N-gram profiles [18]. After computing the overall distance measure between the testing profiles with all the eight categories training profiles, the category that has the smallest Manhattan distance is chosen to be the category that the testing profile belongs to, and in this step the categorization has been done for this measure.

The second measure as shown in equation (2) is the Dice measure of similarity.

Dice $$(P_i, P_j) = \frac{2|p_i \cap P_j|}{|p_i| + |p_j|} \quad \ldots\ldots\ldots\ldots (2)$$

Where $|P_i|$ is the number of N-grams in profile Pi [18]. In this measure the category with the largest measure is chosen as the category that the testing profile belongs to. The following example calculates the similarity between two words only (and not between two profiles). Suppose there are two words and there is a need to calculate the similarity between them the first word is أستفسارات and the second is أستفسر the bi-gram for them will be as follows:-

سر	فس	تف	ست	أس

ات	را	ار	سا	فس	تف	ست	أس

There are 4 common bi-grams in both words. Similarity measured by Dice's coefficient is calculated as 2C/(A+B), where A and B are the number of unique bi-grams in the pair of words; C is the number of common bi-grams between the pair. The similarity measure for these two words will be: (2*4/(8+5)= 0.615).

Results

In order to evaluate the CAP categorization the recall and precision for both the Manhattan and the Dice measures were calculated. The recall means the proportion of relevant text files retrieved while the precision is the proportion of retrieved text files that are relevant.

Recall = (categories retrieved and relevant)/total relevant categories……..(3)

Precision = (categories retrieved and relevant)/total categories retrieved…….(4)

Table 1&2 shows the computed values of precision and recall for both measures.

Table 1: Recall and Precision using Manhattan measure

Category	Recall	Precision
Cat1 "Hekmah"	0.089	0.46
Cat2 "Fakar"	0.00	0.075
Cat3 "Retha'a"	0.2	0.3
Cat4 "Wasef"	0.294	0.885
Cat5 "Naseeb"	0.011	0.601
Cat6 "Madeh"	0.333	0.667
Cat7 "Ghazal"	0.619	0.931
Cat8 "Heja'a"	0.6	0.896

Table 2: Recall and Precision using Dice's measure

Category	Recall	Precision
Cat1 "Hekmah"	0.122	0.7211
Cat2 "Fakar"	0.00	0.105
Cat3 "Retha'a"	0.402	0.5
Cat4 "Wasef"	0.321	0.388
Cat5 "Naseeb"	0.289	0.3
Cat6 "Madeh"	0.46	0.655
Cat7 "Ghazal"	0.801	0.7
Cat8 "Heja'a"	0.799	0.777

Discussion

In order to have a good information retrieval system this system should retrieve as many poems as possible, which means having a high recall. And it should retrieve very few non-relevant poems, which means having a high precision.

The results for the tri-gram method using the Dice measure exceeded those for the Manhattan. This is due to the nature of the Manhattan measure, and the complex morphological structure of Arabic language.

When a very long poem for each category is used the system will work better as many words will be used that can identify each category separately and more clearly. From the results it is obvious that the poems for Wasef and Naseeb the recall value for them is very low this is due to the nature of such poems. For more explanation, the selection of the poems for category Wasef, the thirty verses were taken for describing a lover, a camel, the nature, the moon, and a dancer six verses for each, wasn't quite enough to find so many tri-grams that are similar with other verses of the same category. Inspite of the diversion in covering this category the problem was with the small dataset being used to achieve this work. Category Fakar has registered the worst values as this kind of poems are not easy to identified even when using a human to do the task of identifying and that reason is due to the duplicate meanings that such category may contain. It is sometimes very close to the category of Madeh and sometimes very close to the category of Wasef. The best values were for the categories of Ghazal and Hija'a.

when implementing the normalization process together with the N-gram alone, (which means without doing the stemming process) the results were really surprising as the values were very close to the first results when using the stemming process (table 1&2). It seems that the normalization process is quite adequate for making any poem compact, especially when omitting the stop words. It is clear that there is kind of fitness between the process of the N-gram and the normalization only when using the CAP, unlike most of the previous works when they used Arabic corpus that got punctuation marks, non Arabic words, and numbers.

The reason of choosing the tri-gram directly instead of other N-grams like the bi-gram was because of the previous works. They recorded better results only when they used the tri-gram.

Conclusion

This paper presented a statistical categorization method for categorizing the Classical Arabic Poetry. The whole work depends on two steps the first is the stemming process, and the second is the N-gram frequency statistics technique. Two measures were used the similarity (the Dice measure) and the dissimilarity (the Manhattan). From the results the Dice's measure is better than the Manhattan measure because it reported higher precision and recall for the same dataset.

References

1. Khreisat, L. **2006**. *Arabic Text Classification Using N-gram Frequency Statistics a Comparative Study*. The International conference on Data Mining Part of the World Congress in Computer Sciences DMIN: 78-82.
2. Larkey, L. S. and Ballesteros, L. and Connel, M. E. **2002**. *Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis*, in Proc. of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 275 – 282.
3. Xu, Jinxi and Croft, W.B. **1996**. Corpus-Based Stemming using Co-occurrence of Word Variants. in ACM TOIS, Jan. 1998, *Computer Science Technical Report* TR96-67, **16(**1):61-81.
4. Al Hajjar A., Hajjar M., Zreik K. **2009**. *Classification of Arabic Information Extraction methods*. Conference ICHSL7, Toulouse, pp.311-317.
5. Adamson George, W. & Boreham, J. **1974**. The use of an association measure based on

character structure to identify semantically related pairs of words and document titles, *Information Storage and Retrieval,* **10**: 253-260.

6. Savoy, J., and Rasolofo, Y. **2002**. *Report on the TREC-11 Experiment: Arabic, Named Page and Topic Distillation Seraches*. TREC-11 Conference, pp 61-73.
7. Xu, J., Fraser, A., and Weischedel, R. **2002**. *Empirical Studies in Strategies for Arabic Retrieval,* SIGIR '02 Tampere Finland, pp 129-137.
8. Suleiman H. Mustafa. **2004**. Character contiguity in N-gram-based word matching: the case for Arabic text searching. *Information Processing and Management*.**41** (4):819-827.
9. Mustafa, H. S., and Al-Radaideh, Q. **2004**. Using N-Grams for Arabic Text Searching *Journal of the American Society for Information Science and Technology,* 55(11), **13(**5): 1002-1007.
10. Ahmed, F. & Nurnberger, A. **27 July 2007**. *N-grams Conflation Approach for Arabic*, ACM SIGIR Conference, Amsterdam, pp 565-577.
11. Sinan, M. & Rammal, M. & Zreik, K. **2008**. *Arabic documents classification using N-gram,* Conference ICHSL6, Toulouse. pp 510-522.
12. Chen, A. & Gey, F. **2002**. "*Building an Arabic stemmer for information retrieval* ", TREC-11 Conference, pp 86-91.
13. Zipf, George K. **1949**." *Human Behavior and the Principle of Least Effort, an Introduction to Human Ecology",* Addison-Wesley, Reading, Mass, p 41.

١٤. جرجي زيدان، **١٩٥٧**. *تاريخ آداب اللغة العربيـة* ، دار الهـلال، ج١ ص ٩٨، ج٢ ص ١٠٥، ج٣ ص ٣٣، ج ٤ ص١٩، ١٠٥.

١٥. د. عمر فروخ، **١٩٧٤**. *المنهاج فـي الأدب العربـي* ، المكتبـــة العصـــرية، ج١، ص ٦٦ ،ج٢ ، ص ٢١٣، ٢١٦، ٢١٩،ج٣، ص ٣٢، ٣٣، ج٤، ص ٣٤,٣٧,٣٩.

١٦. د.هدى شوكت بهنام، **٢٠٠٠**. *مقدمـة القصيدة العربيـة فــي الشـعر الأندلســي*، دار الشـؤون الثقافيــة،ص ٣١١، ٣٢٥.

17. Cavnar, W. & Trenkle, J. **1994**. *N-gram-Based Text Categorization*, In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, pp 250-263.
18. Baeza-Yates, R., & Riberio-Neto, B. **1999**. *Modern Information Retrieval,* Addison Wesley, pp 133,145.