Arabic Text Classification Methods: Systematic Literature Review of Primary Studies

Waleed Alabbas, Haider M. al-Khateeb and Ali Mansour
Institute for Research in Applicable Computing (IRAC)
University of Bedfordshire, Luton, United Kingdom
waleed.alabbas@study.beds.ac.uk, haider.alkhateeb@beds.ac.uk, ali.mansour@beds.ac.uk

Abstract— Recent research on Big Data proposed and evaluated a number of advanced techniques to gain meaningful information from the complex and large volume of data available on the World Wide Web. To achieve accurate text analysis, a process is usually initiated with a Text Classification (TC) method. Reviewing the very recent literature in this area shows that most studies are focused on English (and other scripts) while attempts on classifying Arabic texts remain relatively very limited. Hence, we intend to contribute the first Systematic Literature Review (SLR) utilizing a search protocol strictly to summarize key characteristics of the different TC techniques and methods used to classify Arabic text, this work also aims to identify and share a scientific evidence of the gap in current literature to help suggesting areas for further research. Our SLR explicitly investigates empirical evidence as a decision factor to include studies, then conclude which classifier produced more accurate results. Further, our findings could be updated more frequently, while Tweets are retrievable in real-time. In general, Text classifiers can be categorized into two models: Generative and Discriminative. For instance Naïve Bayes (NB) is an example of a generative model that will first try to estimate parameters from \( p(x | y) \) and \( p(y) \) from the training data, then calculates \( p(y | x) \) by using Bayes theorem. Where \( p(x | y) \) stands for a conditional probability of \( x \) given \( y \) is true. It is called “generative” since we can generate new samples by sampling from the learned joint distribution \( p(x, y) \). In contrast, a discriminative model estimates parameters of \( p(x | y) \) directly from the training data without assuming anything about the input distribution \( p(x) \), such models include Support Vector Machines (SVM), Neural Networks and Decision Trees [2, 3]. SVM is considered a non-probabilistic binary linear classifier, it can be used for both classification or regression. For a given set of training samples, the SVM model is representation of these samples as mapped points in space, isolated by a gap to distinguish the different categories. Likewise, Decision Trees can be used as a predictive model. Their structure includes leaves to represent classes (target values) and branches to represent conjunctions of features. However, in complex classification tasks, trees could fail to generalize from the training data (overfitting) or correctly illustrate a concept.

Furthermore, these two approaches can be combined to create a hybrid model, known as Generative-Discriminative Pairs (CDP). It is a relation between a generative model and a discriminative model where one can be directly transformed to the other [3]. Examples include the Discriminative Hidden Markov Model (D-HMM) [4] and the pair of Naïve Bayes together with Logistic Regression, in which a model is trained by optimizing a combination of the generative and discriminative log likelihood functions to classify text. CDP can have many advantages to address practical challenges. [5] developed a hybrid model that can switch between generative and discriminative algorithms systematically as a subtask of the learning process, this has allowed them to achieve better results while discovering rare categories in a given dataset.

While discriminative classifiers often outperformance their generative counterparts in accuracy, generative models have several advantages. It is assumed they are easier to classify data and could achieve better accuracy when the training data is limited [3]. However, a generative approach produces a probability density model over all variables in a system and manipulate it to compute classification. While the overall design of generative models has the advantage of being more complete by definition, it can be wasteful and non-robust [6]. A discriminative approach makes no clear attempt to model the underlying distributions of the features in a system and is only interested in optimizing a mapping from the inputs to the required class. As such, learning (not modelling) is the focus of discriminative approaches which often lack flexible modelling, its techniques could feel like black-boxes where the relationships between variables are not as explicit as in generative models [6].

Although TC remains an active research area with novel techniques designed and tested on English scripts [7], there seems to be very little work done on Arabic text. With the absence of a Systematic Literature Review (SLR) based on a
comprehensive search protocol and quality assessment, it is not possible to determine the research gap for Arabic text, this has become one of the objectives for this study. For instance, it is important to conclude better performing classifiers and which text pre-processing and Dimensionality Reduction Techniques (DRT) [8] were proven more effective for Arabic.

Arabic is the 5th widely used language in the world. It is officially used in 24 countries, the mother tongue for more than 422 million persons and the second language for almost another 250 million. Arabic has 28 letters and the orientation of writing is from right to left. Its script has a unique shape, marks, diacritics, Style (font), numerals, distinctive letters and none distinctive letters [9]. Noaman and Al-Ghuribi, [10] discussed its complex morphology and how words could have different meaning within a given context. Arabic is highly inflectional and derivational [11], it does not use capitalization for proper nouns which is a very useful input when classifying English documents. Arabic synonyms are widespread [12]. The majority of words have a tri-letter root, while the rest have a quad-letter root, penta-letter root or hexa-letter root [13].

Recent publications into this growing area of research include Fawaz and AbuZein’s work [14] to enhance classifier’s performance on Arabic text using cosine similarity and latent semantic indexing, the effect of preprocessing on Arabic document categorization by Ayedh et al.[15] and others [16-18]

The remaining of this paper covers the methodology in Second 2 which also discusses the research questions of this study, protocol used and finally the data extraction strategy. Section 3 contains SLR results analysis and discussion of key findings from the included primary studies. Finally, conclusions are written in Section IV.

II. METHODOLOGY

The research method is based on the SLR guidelines for the discipline of computer engineering as proposed by Kitchenham and Charter [19]. Key phases we followed are demonstrated in Fig 1, we share further reflection on each within the consequent sections. In general, we have identified the problem statement, research questions and fundamental aspects of the review protocol as part of the Planning phase. To mitigate subjectivity, we enforced a role that each of these phases is initiated after a full evaluation and approval of the previous one. The Search Strategy, consisted of the study selection criteria, procedure, unified search string and study quality assessment. The third phase is mainly concerned with the development of our Data Extraction strategy. And the final phase of the systematic review involved data synthesis and critical analysis.

A. Research questions

The main aim of this study can be achieved through answering the research questions define and discussed below:

RQ1. What TC models have been applied on Arabic text and supported by an empirical evidence to estimate their accuracy? And which models performed better on Arabic text?

RQ2. What characteristics can be identified to describe corporuses, techniques and algorithms used that can affect accuracy for these TC models?

The term ‘models’ used in the questions above could be used interchangeably with ‘methods’ and ‘techniques’. Answering RQ1 helps to conclude a list of all relevant TC methods within the scope and requirement of this study, while RQ2 investigates their key characteristics. RQ1 helps to research the accuracy of their implementation and therefore reliability in a real life application. Both RQ1 and RQ2 help to identify the gap in current literature and suggest areas for further investigation.

To frame these research questions effectively, PICOC criteria (Population, Intervention, Comparison, Outcome, and Context) [19, 20] were applied from viewpoint of software engineering as follows:

<table>
<thead>
<tr>
<th>Population</th>
<th>Text Classification Models.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intervention</td>
<td>Generative and Hybrid models.</td>
</tr>
<tr>
<td>Comparison</td>
<td>Discriminative models.</td>
</tr>
<tr>
<td>Outcomes</td>
<td>Accuracy of the models analysed.</td>
</tr>
<tr>
<td>Context</td>
<td>Academic research.</td>
</tr>
</tbody>
</table>

B. Data sources and search strategy

Pioneer database sources for software engineering research publications have been used as shown in Table 1. This study begun in January 2015 and therefore considered publications up to that date. Searching keywords were defined to include the following key terms and synonyms constructed with logical operators to return the best possible search outcome:

('Arabic text' OR 'Arabic script') AND ('classification' OR 'Classifier' OR 'categorization' OR 'categorisation')

This search string was adapted to the built-in options of each database from Table 1 to filter and refine results. Further, grey literature was considered in our search strategy together with a snow balling approach (reference of references) where any paper collected by our search criteria can manually lead to another reference from within its bibliography.

<table>
<thead>
<tr>
<th>Database</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEEExplore</td>
<td><a href="http://ieeexplore.ieee.org">http://ieeexplore.ieee.org</a></td>
</tr>
<tr>
<td>ACM Digital library</td>
<td><a href="http://dl.acm.org">http://dl.acm.org</a></td>
</tr>
<tr>
<td>CiteSeerX library</td>
<td><a href="http://citeseerx.ist.psu.edu/index">http://citeseerx.ist.psu.edu/index</a></td>
</tr>
<tr>
<td>Science Direct</td>
<td><a href="http://www.sciencedirect.com">http://www.sciencedirect.com</a></td>
</tr>
<tr>
<td>Springer</td>
<td><a href="http://link.springer.com">http://link.springer.com</a></td>
</tr>
<tr>
<td>Academic Search Elite</td>
<td><a href="https://www.elsevierhost.com/">https://www.elsevierhost.com/</a></td>
</tr>
<tr>
<td>DOAJ</td>
<td><a href="https://doi.org/">https://doi.org/</a></td>
</tr>
<tr>
<td>Web of Knowledge</td>
<td><a href="http://www.webofknowledge.com">http://www.webofknowledge.com</a></td>
</tr>
<tr>
<td>Scopus</td>
<td><a href="http://www.scopus.com/">http://www.scopus.com/</a></td>
</tr>
<tr>
<td>Google scholar</td>
<td><a href="http://scholar.google.co.uk">http://scholar.google.co.uk</a></td>
</tr>
</tbody>
</table>

Figure 1 – Main stages followed in this SLR.

C. Study selection criteria

In this step we apply rigorous inclusion and exclusion criteria to ensure valuable and relevant information in response to our defined research questions. These criteria were enforced after reading the title, abstract and then full text of the articles as demonstrated in the study selection procedure shown in (2.4.) For instance, [21] was excluded because it does not report the method’s accuracy and [22] was not a primary study.

Inclusion criteria:
- Must be a primary study reporting on TC models
- Must include analysis and empirical evidence.

Exclusion criteria:
- Publication is not peer reviewed.
- Arabic is not the language used to test the TC model.

D. Study selection procedure

The selection of the primary studies was examined by all authors. Four different phases show how the selection procedure was implemented as illustrated in Fig. 2:

Phase 0 – Keywords-based filtering.

In this phase, the search string was applied to the ten scholarly databases shown in Table 1. This has yielded a total of 1464 articles which were included in the next phase.

Phase 1 – Title, indexing keywords and abstract-based filtering.

In this phase, titles were examined against the inclusion and exclusion criteria. Articles deemed to be of any relevance were directly included in the next phase. In conclusion, 863 articles were discarded and 365 articles were included.

Phase 2 – Full text-based filtering.

This was the final stage where the reviewers discussed and resolved disagreements regarding the relevance of the articles to the study. A total of 192 articles were identified to be duplicates downloaded from different databases and were therefore discarded. Upon reconsideration of the inclusion and exclusion criteria, 125 articles were excluded for different reasons; for instance, [23] was not peer reviewed, [24] did not include an empirical study and [25] did not satisfy a number of the quality assessment criteria shown in (2.5). The final set of primary study had a total of 48 remaining articles.

E. Study quality assessment

Included papers had to satisfy a quality assessment designed as a measure to determine if a given paper is suitable to address our research questions. The following checklist had to be met with affirmative answers:
- Was the number of training and testing data identified?
- Were the pre-processing techniques used in the study clearly described and their selection justified?
- Were the classifiers used in study clearly described?
- Is there comparison with other approaches?
- Were the performance measures fully defined?

F. Data extraction strategy

Data extracted from the studies, were tabulated and comprised the following characteristics: year of publication, number of learning and testing documents, features selection approaches, classification algorithm and accuracy.

III. RESULTS ANALYSIS AND DISCUSSION

A. Primary studies

There were a total number of 48 included studies in the form of journal articles and conference proceedings published between 2006 and 2014. Our analysis shows most articles were published within the last 5 years which was an early indication that Arabic TC is an active research area and started to evolve very recently. More details are demonstrated in Table 2.

<table>
<thead>
<tr>
<th>Source</th>
<th>06</th>
<th>07</th>
<th>08</th>
<th>09</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>J</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>7</td>
<td>25</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td>0</td>
<td>23</td>
</tr>
<tr>
<td>%</td>
<td>4</td>
<td>4</td>
<td>7</td>
<td>10</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

B. Key focus areas

We found that primary studies can be classified by their main focus area into four domains: TC algorithms, Features Selection (FS), Stemming Techniques (ST) and Term Weighting (TW). The majority of work was on TC algorithms as shown in Table 3.

<table>
<thead>
<tr>
<th>Focus area</th>
<th>%</th>
<th>Studies</th>
</tr>
</thead>
<tbody>
<tr>
<td>TC algorithms</td>
<td>56</td>
<td>[11, 26-51]</td>
</tr>
<tr>
<td>FS</td>
<td>25</td>
<td>[12, 52-62]</td>
</tr>
<tr>
<td>ST</td>
<td>13</td>
<td>[63-68]</td>
</tr>
<tr>
<td>TW</td>
<td>6</td>
<td>[13, 69, 70]</td>
</tr>
</tbody>
</table>

Each of these focus areas are discussed in details within Sections III.D, III.E and III.F.
C. Data collection (Corpus)

Collecting data to create a suitable dataset is the first step in text classification studies. Whilst there are several free benchmarking datasets for English used for TC purposes: the 20 Newsgroup contains around 20,000 texts distributed almost evenly into 20 classes; Reuters-21578 contains 21,578 texts belonging to 17 classes; and RCV1 (Reuters Corpus Volume 1), contains 806,791 texts classified into four main classes [26]. Unfortunately, the case is different for Arabic. There seems to be no free benchmarking dataset identified from the included studies for Arabic TC. For most research, authors collect data to build their own datasets, mostly from online formal websites and news articles. Table 4 describes the datasets used in each study. It also shows the language model selected, whether it is classical Arabic (also known as Qur'anic) which could also include old poem and religious scripts; modern Arabic currently used in formal press and government communications; colloquial Arabic as in informal local dialects; or a mixture of these. It has also been noted that some papers do not seem to describe their datasets enough which makes it difficult to classify their datasets. Such works usually do not publish their data for other researchers to utilize. Consequently, the confidence in the results derived from such experimental studies is not satisfactory enough. The performance of the adopted data mining approaches is biased to such data sets and could be ambiguous.

<table>
<thead>
<tr>
<th>Models</th>
<th>Corpus</th>
<th>Studies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classic</td>
<td>Quran</td>
<td>[49]</td>
</tr>
<tr>
<td></td>
<td>Religious scripts</td>
<td>[38, 45, 55, 57]</td>
</tr>
<tr>
<td></td>
<td>Old books</td>
<td>[31]</td>
</tr>
<tr>
<td>Modern</td>
<td>Websites</td>
<td>[13, 27, 29, 53, 60, 61, 63, 64, 66, 67]</td>
</tr>
<tr>
<td></td>
<td>News articles</td>
<td>[11, 12, 28, 32, 39, 42-44, 46, 47, 50-52, 54, 59, 65, 68-70]</td>
</tr>
<tr>
<td>Colloquial</td>
<td>User Reviews</td>
<td>[48]</td>
</tr>
<tr>
<td>Hybrid</td>
<td></td>
<td>[26, 30, 34, 41]</td>
</tr>
<tr>
<td>unknown</td>
<td></td>
<td>[33, 35-37, 40, 56, 58, 62]</td>
</tr>
</tbody>
</table>

Results show that most work is conducted on the modern language with a single study [48] covering informal (colloquial) writing, this is an interesting finding because it recovers a huge and critical technology gap, informal Arabic is people use on social media, especially Twitter. Arabic dialects vary from one Arab country to another and could also slightly vary between cities and towns.

With regards to the size of datasets, they ranged from 119 documents divided into three classes [12] to 17,652 documents divided into six classes [30]. The vast majority of studies measures the size by the number of documents rather than word count. This detail given an indication on the size but it remains a challenge to have an accurate statistical comparison between the different datasets used.

D. Text pre-processing and dimensionality reduction techniques

Pre-processing is a trial to improve text classification by removing worthless data. It may include the removal of numbers, punctuation (e.g. hyphens) and stop-words (e.g. prepositions and pronouns). Due to its writing style, Arabic requires careful strategies at this stage to normalize writing forms and removing diacritics.

A number of dimensionality reduction techniques are also used to reduce the number of terms included for analysis (classification); high dimensionality data do not satisfy the requirements of TC methods to produce reasonably accurate outcome and are therefore considered problematic [71]. Included studies identified the use of two reduction techniques, namely: Stemming and Feature Selection.

1) Stemming

Stemming is a technique to reduce the high dimensionality of the feature space in text classification. Several Stemming approaches exist for the Arabic language each produces a different set of roots. These are identified in Table 5 and discussed in further details below.

Root-based stemming (Lexical) is based on removing all attached prefixes and suffixes in an attempt to extract the root of a given Arabic surface word. An example of this approach is the Khoja stemmer [72]. Its core-function works by mapping words into their root patterns. Root patterns in Arabic are three, four, five, or six-letter patterns. More than 80% of the Arabic words can be mapped into three-letter root pattern, reducing a word to its root pattern could decreases the number of words from hundreds of thousands to as little as 4,749 as in [69].

<table>
<thead>
<tr>
<th>Stemmer</th>
<th>Studies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root-based stemming</td>
<td>[12, 34, 35, 39, 48, 65, 69]</td>
</tr>
<tr>
<td>Light stemming</td>
<td>[27, 29, 43, 45, 47, 55, 60, 64, 66]</td>
</tr>
<tr>
<td>Statistical stemmer</td>
<td>[49, 61, 62]</td>
</tr>
<tr>
<td>Hybrid</td>
<td>[63]</td>
</tr>
</tbody>
</table>

Light Stemming does not attempt to give the linguistic root pattern for the word, instead, its main focus is to remove the most frequent suffixes and prefixes. There are different types of Light Stemming and many studies have considered this approach (Table 5). The literature in general gives an argument that light stemming allows remarkably good information retrieval, [73] discuss this in further details.

Statistical stemmer (character level N-Gram), N-Gram is a set of N consecutive characters extracted from a word. The main idea behind this approach is that, similar words will have a high proportion of N-Gram in common. This was also tested where a number of 3-grams is applied on the following string: “text classification”, the output is: “tex”, “ext”, “xt_”, “t_c”, “_cl “, “ cla”, “las”, “ass”, and so on [63]. Each of these strings will then be compared against the output of another string to measure and determine the level of similarity between the two.

A hybrid approach was also tested where a number of stemming techniques are used together in an attempt to improve the process. For example [63] proposed a hybrid method incorporating Khoja stemmer, light stemmer and N-Gram. Results were promising with an improvement in the overall accuracy. Likewise [69] used root extraction by assigning weights and ranks to the letters that constitute a
word. However, they mention that roots are semantically weak in the meaning that several words can be mapped onto the same root.

Nonetheless, in some cases stemming techniques could decrease the performance of the classifier used. Kanaan et al., [44] observed this behavior when light stemming was used with the Rocchio and NB algorithms. Likewise, Al-Kabi et al., [67] conducted an experiment and concluded that Khoja stemmer did not improve the classification accuracy for NB, SVM (SOM) and decision tree (J48).

2) Feature selection

Some reduction methods utilize features (terms) selection to reduce dimensionality. These statistical techniques work at the term level, as such, when 3-gram is utilized; text is split into chunks of 3 terms (words rather than characters). Table 6 demonstrates which FS techniques was used by each study.

Table 6 – Feature Selection Techniques

<table>
<thead>
<tr>
<th>FS Techniques</th>
<th>Studies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chi-square</td>
<td>[29, 31, 41, 47, 52, 54, 56, 58, 61]</td>
</tr>
<tr>
<td>Term Frequency</td>
<td>[43, 59]</td>
</tr>
<tr>
<td>Document Frequency</td>
<td>[70]</td>
</tr>
<tr>
<td>Information Gain</td>
<td>[30]</td>
</tr>
<tr>
<td>N-gram</td>
<td>[13, 48, 68]</td>
</tr>
<tr>
<td>Hybrid</td>
<td>[26]</td>
</tr>
</tbody>
</table>

Most studies applied Chi-square (CHI) while there was a single study [26] attempting a hybrid approach in which the authors applied Document Frequency and Galavotti, Sebastiano, Simi (GSS).

E. Feature representation (term weighting)

TC algorithms require that text features are formatted before they can be interpreted by the specified classifier, this process is also referred to as term weighting because each term is entered together with a weight value. Included papers show the most used technique is the Term Frequency-Inverse Document Frequency (TF-IDF) as in [27, 32, 37, 40, 43, 45, 48, 51, 53, 55, 57, 58, 60-62, 67]. It is a statistical method to indicate the significance of a word within a given corpus. This utilization of the technique is justified assuming the authors wanted to weight terms while considering its significance across all documents rather than a single one. Although, in [58] a simpler but more limited method has also been used to conclude a Boolean value of zero or one, a term can be described to be either important or not important. Whilst in TF-IDF, for a given term, a bigger TF-IDF value indicates a more frequent word. As such, data can be represented as a matrix with n rows and m columns wherein the rows correspond to the texts in the training data, and the columns correspond to the selected feature. The value of each cell in this matrix represents the weight of the feature in the text.

F. Classification algorithms and accuracy

Each study used their very own corpus and different experiment conditions in terms of their training and testing procedure, pre-processing and DRT. Hence, it is not feasible to statistically compare accuracy values (cross studies). However, when we analyze the outcome of different studies, there is evidence that the Support Vector Machine (SVM) classifier (a discriminative model) outperforms other classifiers with the exception of two studies reporting in favor of the C5.0 Decision Tree Algorithm, and one study on k-NN. This outcome is demonstrated in Table 7.

Table 7 – Studies investigating accuracy. Accuracy values for each study have been reported in the following format: [study] (accuracy for the preeminent classifier – accuracy for the first method in comparison, accuracy for the second method, …)

<table>
<thead>
<tr>
<th>Preeminent Classifier</th>
<th>Compared with</th>
<th>Studies (and accuracy values)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>NB</td>
<td><a href="0.805-0.755">26</a>, <a href="0.778-0.74">36</a>, <a href="0.954-0.884">38</a>, <a href="0.778-0.74">42</a>, <a href="0.9241-0.8949">61</a>, <a href="0.8638-0.7741">65</a></td>
</tr>
<tr>
<td>k-NN</td>
<td><a href="0.827-0.448">46</a>, <a href="0.585-0.563">51</a></td>
<td><a href="0.956-0.94">27</a>, [28](0.611-0.585, 0.601), [51](0.914-0.845, 0.727)</td>
</tr>
<tr>
<td>NB</td>
<td>AN</td>
<td>[56](0.9141-0.8778, 0.7581, 0.7472)</td>
</tr>
<tr>
<td>J48, ROCHIO</td>
<td>NB</td>
<td>[33](0.968-0.8942, 0.8507), [37](0.9608-0.9048, 0.856), [67](0.896-0.753, 0.835)</td>
</tr>
<tr>
<td>J48, k-NN</td>
<td>NB</td>
<td>[47](0.98-0.856, 0.967, 0.799)</td>
</tr>
<tr>
<td>k-NN, ANN, RACHIO</td>
<td></td>
<td>[48](0.611-0.585, 0.601)</td>
</tr>
<tr>
<td>k-NN, SVM</td>
<td></td>
<td>[48](0.857-0.824, 0.646)</td>
</tr>
<tr>
<td>Decision-tree (C5.0)</td>
<td>SVM, ANN</td>
<td>[30](0.8443-0.761, 0.7566, 0.6378)</td>
</tr>
<tr>
<td>k-NN</td>
<td>SVM, NB</td>
<td>[48](0.666-0.598, 0.563)</td>
</tr>
</tbody>
</table>

While all included studies have also reported the accuracy of their classifiers, Table 7 includes only those attempted to conduct experiments on multiple algorithms within a controlled environment for comparison purposes.

Results show that generative models remain an option when the amount of training is relatively small and could therefore be faster, both algorithms that reportedly outperformed other models are discriminative (SVM and C5.0). SVM is a supervised learning algorithm, with an appropriate kernel, the algorithm can function competently whether or not the data is linearly separable. It is widely used even with text of high-dimensionality. However, its disadvantage can be summarized to be the algorithms complexity, interpretability and memory requirements [74]. However, not all discriminative models performed better, the K-Nearest Neighbor (k-NN) Classifier is an exemplar for this case. It is discriminative because it models the conditional probability of data belonging to a given class. k-NN computes the similarities between a new sample and the training samples previously stored in a dataset. The most K similar ones are then listed in a descending order. Finally, the new sample takes the class label that belongs to the majority of these K neighbors [43]. It should therefore not be preferred for text categorization [74]. Nonetheless, although C5.0 Decision Tree algorithm outperformed SVM, the later outperformed another Decision
Tree algorithm; J48 while many other remain untested in the literature.

As mentioned earlier, the remaining set of the included studies did not conduct a comparison between classifiers, they have instead investigated other factors. For instance, [63] used NB with different stemmer techniques and found that a hybrid method gives more accurate results if compared to a root-based stemmer, light stemmer or n-gram (statistical stemming). Likewise, [47] used SVM with different stemming techniques, however the study reports very minor effect on accuracy.

IV. CONCLUSION

More work need to be done on Arabic text analysis as it can be applied to solve real-world problems such as automating procedures, building intelligence and mitigating cybercrime [75]. Future work on TC techniques for Arabic text should ideally consider using a corpus that is available online for download; this will enable comparative experiments by other researchers and conclude robust facts with regards to the accuracy and speed of the different algorithms and techniques available. Additionally, datasets should be described thoroughly in the papers, sharing the word count to describe the size is the right approach rather than the number of documents collected.

Implementing a hybrid Stemming and/or Feature Selection approach could improve the accuracy as few studies suggest. Majority of papers report on using root-based stemming, light stemming and Chi-square, therefore more research is needed to investigate the opportunities and threats for adopting hybrid Dimensionality Reduction Techniques on Arabic text during both: Stemming and Feature Selection.

Not all discriminative algorithms outperform the accuracy of generative models; NB outperformed k-NN, however both preeminent algorithms from the included studies were discriminative; SVM and C5.0. Additionally, no work has been found by our search protocol to compare with a hybrid model; Generative-Discriminative Pairs (CDP).

Further, TF-IDF was used in the vast majority of papers but there was little discussion and justification for adopting this statistical method, it is very critical that new research realize this limitation in current literature, lack of details was a key reason to exclude some papers in our protocol mainly because they have failed to describe their training datasets and report the accuracy of the utilized algorithms.

REFERENCES