Expert Systems With Applications 203 (2022) 117384

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

An unsupervised annotation of Arabic texts using multi-label topic modeling and genetic algorithm

Huda A. Almuzaini, Aqil M. Azmi *

Department of Computer Science, College of Computer & Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

ARTICLE INFO

Keywords:
Arabic corpus
Topic modeling
Multi-label annotation
Genetic algorithm
Latent Dirichlet allocation
Crowdsourcing

ABSTRACT

Every day the world produces an enormous amount of textual data. This unstructured text is of little use unless it is labeled using a combination of categories, keywords, tags. Humans can never annotate such massive data, and with a growing divide between the daily produced data and those annotated, the only alternative is to mechanize it. Automatic annotation process helps in saving resources in terms of time and cost. The process of multi-label annotation involves associating a document with multiple relevant labels. This paper proposes an unsupervised model to annotate corpus using multi-labels automatically. The model is based on multi-label topic modeling and genetic algorithm (GA). Topic modeling is a technique to extract the hidden topics from text, and the GA is used to find the optimal number of topics. We hyper-tuned the parameters of the topic modeling using two different training methods: variational Bayes and Gibbs sampling. The class imbalance in a corpus can affect the result of topic modeling, where the majority class dominates the minority class. We overcome this problem using the partitioning method. Though the proposed model was developed for the Arabic dataset, it is language neutral. We tested our model on three large Arabic corpora and three large English social media datasets. For the Arabic language, our work being the first work that tackles multi-label annotation, we needed a reference to compare our model. For the Arabic corpus, we compared the result of automatic annotation against humans using crowdsourcing (whose labeling was checked for quality). The analysis of the annotation shows an agreement among models (machine vs. human) of 79.30%. Moreover, for the English dataset, the results are quite competitive.

1. Introduction

Multi-label classification and the related problem of multi-output classification are variants of the classification problem where multiple labels are assigned to each instance. The texts in nature can belong to more than one category. For instance, a news article under "politics" category may appear under other categories, e.g., "economics" and "health".

Multi-label text classification is a fundamental task for textual information retrieval, organization, and management. It assumes that each document is associated with one or more categories. In the simplest case, we may consider multi-label classification as a set of binary classification tasks that decides for each label independently whether it should be assigned to the document or not. However, such a binary relevance approach ignores the dependencies between labels (Zhang & Zhou, 2013).

Many studies addressed single- and multi-label classification, typically supervised where the data is manually annotated. For a reliable classification, this approach requires a substantial quantity of annotated data for the training. No doubt, manually building a multi-label training set is much more expensive than a single-label counterpart because the annotator needs to consider every possible category for each document. Being human, the annotator will likely miss some categories while annotating. In addition, many institutions in diverse sectors such as legal, health, media, education need to analyze a considerable volume of data to extract relevant information for specific tasks or organization and classification purposes. Let us not forget the various social texts that need labeling and tagging, such as social bookmarking systems, tweets, and social question and answer sites. Providing automated annotation systems will reduce the need for human intervention, and it will also create opportunities for improved representation of documents for browsing, searching, and making recommendations.

The annotation process is usually performed manually or aided by some tool. For this reason, traditional annotation methods are time-consuming and more suited for application to small corpora. A recent trend has been to utilize Internet crowdsourcing services, which renders these services as a suitable alternative for annotating large-scale

* Corresponding author.
E-mail addresses: hamozeani@imamu.edu.sa (H.A. Almuzaini), aqil@ksu.edu.sa (A.M. Azmi).

https://doi.org/10.1016/j.eswa.2022.117384
Received 12 September 2021; Received in revised form 6 January 2022; Accepted 25 April 2022
Available online 6 May 2022

datasets. However, the quality of these services is not guaranteed and, at the same time challenging to control. Besides, these services are not free, and the cost becomes prohibitive if the volume of the data is high.

Numerous machine learning (ML) techniques were proposed to reduce the cost of the annotation process, including clustering (Patibandla & Veeranjaneyulu, 2018), active learning (Settles, 2009), weakly supervised learning (Chu et al., 2021; Jiang et al., 2021), and semi-supervised learning (Xing et al., 2018; Zhan & Zhang, 2017). Document clustering is a data mining method applied successfully to various tasks. Text clustering is an unsupervised learning method that breaks the text into subgroups that share similar content. This method relies on unlabeled data to generate anonymous groups called clusters. Each cluster is labeled with a cluster name, and any text within that cluster is tagged with the cluster name. Assigning a name to a cluster can be performed manually or automatically (Jo, 2006; Perez et al., 2018; Zha & Li, 2019; Zhu et al., 2009), and the process is known as a single-label annotation.

Multi-label annotation refers to the association of a document with a subset of relevant labels or tags. A significant proportion of textual data from various Internet sources, including news from the BBC and CNN, and social question and answer sites (e.g., Quora, http://www.quora.com), are annotated with several human-assigned labels.

The problem of annotation has been investigated in many languages, particularly in English and to a much lesser extent in Arabic. For the latter, researchers addressed automatic annotation, such as annotating corpora for sentiment analysis tasks using a sentiment lexicon (Guellil et al., 2018, 2020; Imane et al., 2019), and self-training (Kwaik et al., 2020). Interestingly, all existing Arabic studies for automatic annotating the corpus focused on single-label annotation.

Arabic is the liturgical language of 1.8 billion Muslims worldwide, including 427 million native speakers of Arabic. This makes Arabic the fifth most spoken language in the world (Lane, 2021). Moreover, the Arabic language users constitute the fastest-growing language group on the web with regard to the number of Internet users. According to www.internetworldstats.com/stats7.htm, during the twenty years period ending March ’20, Arabic language Internet users grew by 9348%, with the Russian language users being a distant second. Despite the popularity of the Arabic language, it is associated with scarce resources, such as the availability of annotated corpora for research use. It is hoped that labeled data will accelerate the research for the Arabic language in general.

We propose a scheme to multi-label the text given the importance of annotated corpora in any language. Topic model (or modeling) is a technique for discovering the abstract “topics” that occur in a large collection of text. There are several topic modeling algorithms, e.g., Hofmann (1999) and Papadimitriou et al. (2000). Latent Dirichlet allocation (LDA) (Blei et al., 2003), a widely used topic modeling algorithm, is a generalization of Hofmann (1999). This paper proposes a system using LDA combined with a genetic algorithm for self-annotating Arabic text with multi-labels. The model can easily be adapted for other languages, making it language neutral, which we confirm by applying the model on English text. LDA is a probabilistic algorithm that performs soft clustering, allowing the document to belong to more than one cluster with different weights. The challenge is how to extract an optimal number of good-quality topics that are meaningful and clear. This is where the genetic algorithm comes into use. We use the genetic algorithm to optimize the set of labels for each document. Our contributions can be summarized as follows:

- We propose a novel unsupervised model for annotating text. The method combines LDA and genetic algorithm, where a fitness function based on word probabilities is proposed.
- We perform hyperparameter tuning for topic models for two different training methods: Gibbs sampling and variational Bayes.
- We auto-annotate three large Arabic corpora, and the results are compared against crowdsourced humans.
- We show that our model is language agnostic. The model is tested on three large English social media datasets.

The remainder of the paper is organized as follows: Section 2 reviews related work. In Section 3 we present our proposed model. The dataset used is covered in Section 4. Section 5 evaluates our model on Arabic corpus and discusses the result. We describe our crowdsourcing process in Section 6. In Section 7 we test our model on English dataset and discuss its performance on all the datasets used in the study. Finally, we conclude and present ideas for future work in Section 8.

2. Related work

This section presents related work on LDA in general and specifically for the Arabic language. We also examine other works related to automatic annotation.

LDA belongs to a class of research termed multi-label topic modeling, which focuses on clustering data into groups of topics. It is an unsupervised model that uses text corpora to generate several topics, where each topic is associated with a probability distribution of words. It is worth pointing that LDA can be applied in the context of either supervised or semi-supervised learning. There are two common methods for parameter estimation and model training: Gibbs sampling (Griffiths & Steyvers, 2004) and variational Bayes (Blei et al., 2003). Burkhardt and Kramer (2019) suggested that multi-label topic models differ according to three perspectives. The online model (these can be updated and trained with new data), else the model does not support online learning. The topic models may be parametric (specify the number of generated topics), or non-parametric model (automatically adjusts the number of topics). Moreover, the last perspective considers whether label dependencies are modeled. The latter is a crucial feature of multi-label if there is some overlap between the two labels.

Some researchers extended the multi-label topic models to adapt them to different domains, including Polylingual Topic Models used for corpora from multiple languages where the sets of documents should have some loose equivalence, such as Wikipedia articles (Mimno et al., 2009). The Author-Topic Model is used to define the distribution of topics for each author (Rosen-Zvi et al., 2004). In contrast, the Citation Influence Model is used to define the strength of the influence one publication has over another (Dietz et al., 2007). The topic summary provides a short summary of each topic cluster identified in a text (He et al., 2019; Wan & Wang, 2016).

One typical application of LDA is as a method for dimensionality reduction, where each document is represented as a vector of its topics (Brahmi et al., 2012; Jelodar et al., 2019; Pavlinek & Podgorelec, 2017).

We may divide research addressing the Arabic LDA model into two approaches. The first approach applies LDA as a feature selection or dimensionality reduction method before classification, while other approaches use LDA as a clustering algorithm. Brahmi et al. (2012) used LDA as a feature selection method for Arabic text classification. The paper’s main focus was on the effect of stemming on the LDA model with Support Vector Machine (SVM) classifier. The textual content is first stemmed with three stemmer methods: Khoja and Garside (1999), IRIS (Taghva et al., 2005), and a stemmer called BBw. The LDA with Gibbs sampling is applied on the stemmed text, and the resulting topics were fed into the SVM. The experiment was conducted on three datasets collected from Echorouk, Reuters, and Xinhua Internet articles with seven to nine categories. The authors noted that the Khoja stemmer did not perform effectively with the LDA model.

Naili et al. (2017) examined the effect of stemming on an LDA Topic Model. They compared between no stemming, a root-based stemmer and a light stemmer. Experiments were conducted on Alwatan newspaper using six topics. The results demonstrated that light stemming performed optimally with LDA. Their best result was obtained using the LDA setting $\alpha = 0.5$ and $\beta = 0.1$.

The authors in Ayadi et al. (2014) and Zrigui et al. (2012) applied LDA with an SVM classifier for dimensionality reduction. The probability distribution of documents over topics was input to SVM to perform the classification. A variational Bayes algorithm was used to train the LDA model. Before classification, stemming was performed using their in-house stemmer. The experiments were run on 1500 documents in nine categories sourced from three newspapers: Aljazeera, Alnahar, and Alahram. Zrigui et al. (2012) compared the proposed framework using SVM with the Naive Bayes and KNN classifiers. The proposed classifier was shown to outperform the others and resulted in an F-score of 0.91. By contrast, Ayadi et al. (2014) compared LDA with other feature reduction methods, including latent semantic indexing (LSI) and term frequency–inverse document frequency (*tf-idf*). The authors concluded that the LDA model exhibited an improved performance over other reduction methods. Ayadi et al. (2015) compared the performance of LDA with LSI. The experiment was first conducted on the stemmed text, and then the models were used to classify the resultant matrix using SVM. For the experiment, they used 20,000 documents from Open Arabic Tunisian Corpus (OATC) using ten categories. The authors reported that LDA outperformed LSI; however, no significant differences were observed since the classes had a small number of words.

El Bazi and Laachfoubi (2018) applied an extension of LDA, named soft word-class LDA, for extracting features for the task of Arabic named entity recognition (NER).

In Alhawarat and Hegazi (2018) and Kelaiaia and Merouani (2013, 2016) reported that LDA clustering outperforms the k-means algorithm. Kelaiaia and Merouani (2013) used three textual regimes with each model: raw text, cleaned text (stop-words removed), and stemmed text. The experiments were performed on the OSAC dataset with eleven categories. The results showed that the LDA outperforms the k-means in all text versions using both measures, the F-score and the entropy. Alhawarat and Hegazi (2018) proposed an approach based on combining the k-means algorithm and LDA for Arabic clustering. The text was transformed to bag-of-words, and then *tf-idf* weighting was applied. The resulting data were fed to the LDA model. Following this, the k-means algorithm was applied to the resulting topics of the LDA model. The researchers compared the proposed method with another technique based on k-means without the LDA step. The experiments were performed on a primary dataset consisting of MSA news articles with nine categories, and further five datasets: Aljazeera, Alkhaleej, Alwatan, BBC, and CNN, using four to six categories. Five versions of the data were prepared with a range of preprocessing steps and stemming, applicable solely to the primary dataset. Experimental results demonstrated that the combined algorithm outperformed the k-means algorithm. Their best reported result was an F-score of 0.873 when stop-words were removed, and the text was stemmed using Khoja stemmer.

The supervised ML algorithms require training, an expensive venture. Several researchers conducted text categorization using deep learning techniques such as convolutional neural network, long short term memory (Kulkarni et al., 2021; Taware et al., 2021), encoder–decoder model (Xiao et al., 2021), gated recurrent units (Almuzaini & Azmi, 2020; Elnagar et al., 2020), and BERT (Cai et al., 2020). These techniques require a large annotated dataset in order to get good results. In this regard, alternate techniques such as semi-supervised learning are preferred since they require less annotated data for learning. However, the gain in performance associated with the semi-supervised algorithms is not significant with the small set of labeled data. We may enhance the semi-supervised learning by using an automatic annotation method (Zhu et al., 2009), where a small set of labeled documents are used as a seed to label new documents. First, for each category, the annotator extracts a set of keywords from the labeled documents. Next, a category is assigned to the unlabeled document if it has a certain percent of the keywords in that category. The resulting documents are used as the seed for semi-supervised learning. The authors compared their proposed scheme against others, e.g., seeded k-means (Basu et al., 2002), Expectation–Maximization (EM) based Naive Bayes (NB) (Fujino et al., 2005), and the Linear Neighborhood Propagation (Wang & Zhang, 2007). The results confirmed their scheme achieved higher learning accuracy.

Other researchers followed a different path for semi-supervised learning. Pavlinek and Podgorelec (2017) used LDA model for creating a small set of training data. The LDA model was used as a feature extraction method where topic distributions represented the instance. The similarity between the labeled and unlabeled documents was computed to assign a class for the unlabeled documents. The entire dataset was then used for text classification using SVM and NB classifiers. To assess the performance of the proposed method, the document was classified using *tf-idf* features instead of LDA. Similarity measures were then applied to assign a label to the document. The classification accuracy showed that LDA representation yields significant improvement compared with the *tf-idf* method. Co-training was adopted to annotate the corpora using small sets of labeled instances. Co-training is a semi-supervised approach that trains the classifier with at least two views of the data (two distinct features for each instance), followed by combining the classification results. In Xing et al. (2018) and Zhan and Zhang (2017) applied multi-label learning with co-training for multi-label annotation.

Automatic tagging of Arabic texts is rare. Most of the researches focused on sentiment analysis, where a sentence or a word is tagged with labels such as positive, negative, or neutral. Most methods for annotating corpora automatically were based on a sentiment lexicon (Guellil et al., 2018, 2020; Imane et al., 2019). The lexicon was constructed using automatic translation from English to Arabic. This lexicon was used to annotate the corpus with positive and negative tags automatically. Alzanin and Azmi (2019) described an EM-based semi-supervised and unsupervised learning to detect rumors in Arabic tweets, where only “news” tweets were used for the task of rumor detection. The detection mechanism can be considered as a labeling process. The dataset were divided into two groups: 34% for the training, and the remaining was further divided into: 28% as a testing set, and 38% as the unlabeled set for re-learning. The authors used part of the training set for the actual training. Their semi-supervised EM outperformed the Gaussian NB classifier as a base, using a training set not exceeding 60% of the whole training set. They achieved an F-score of 78.6%, while the result varied depending on the initial values for the unlabeled data.

Overall, the use of multi-label topic models as unsupervised learning methods generates a powerful generalizable utility that can solve numerous tasks (Radford et al., 2019). Though of its potential, the LDA is widely used for dimensionality reduction. The LDA as a multi-topic model may be combined with other algorithms to construct a model for diverse NLP functions. To the best of our knowledge, all the existing studies on automatic annotation of Arabic corpora focuses on single-label annotation. We are not aware of any work that tackled multi-label annotation.

3. Our proposed method

This section describes our proposed model for annotating text with multiple relevant labels. The subsequent discussion in this section assumes Arabic dataset, however, the model is language neutral. We confirm this by testing the model on English dataset (see Section 7). The model consists of two parts. A multi-topic model to analyze the corpus, followed by a genetic algorithm (GA) to optimize each document’s best group of labels. Fig. 1 shows our proposed model. Further detail of each element of the model is covered next. Table 1 summarizes the list of symbols used in the subsequent discussion.

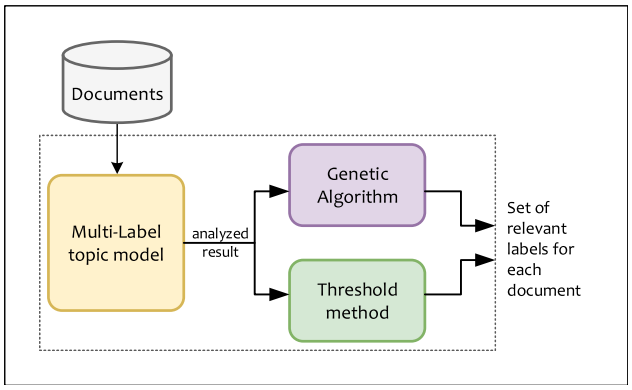


Fig. 1. Our proposed model.

Table 1
List of symbols and their definition.

Symbol	Definition
D	Set of documents in a corpus.
L_i	A set of suggested labels for document d_i.
R_i	A set of relevant labels for document d_i.
q	Number of relevant labels.
CP	Contribution percentage. The probability of the topic occurring in the document.
T_1	First dominant topic with the highest CP.
T_2	Second dominant topic with second highest CP.
CP_{T1}	Contribution percentage of first dominant topic.
θ	Threshold value for the CP.
GA_{L1}	Genetic algorithm (GA) first labeling technique.
GA_{L2}	GA second labeling technique.
V	Represents the population or generation.
v	Represents a single individual ($v \in V$).
S_i	Individual keywords (the candidate labels) + current document d_i words.
$N(w_j, w_k)$	Number of documents in the corpus containing words w_j and w_k.
$N(w_k)$	Number of documents in the corpus containing the word w_k.

3.1. Multi-topic model

Topic modeling is a method for unsupervised classification of documents, similar to clustering on numeric data, which finds some natural groups of items (topics) even if we are unsure what we are looking for. The document can be part of multiple topics, similar to fuzzy (soft) clustering where each data point belongs to more than one cluster. The LDA (Blei et al., 2003) is one of the popular topic modeling methods. LDA is a probabilistic model that requires the corpus to be clustered into a set of topics, where each topic is represented by a set of keywords associated with a probability distribution over words. Let $D = \{d_1, d_2, \ldots, d_n\}$ denote the set of documents in a corpus, where n is the number of documents. For each document $d_i \in D$ the model generates a set of suggested labels L_i using LDA. The suggested labels are the set of keywords from dominant topics T_1 and T_2. Assume these labels are $L_i = \{\ell_1, \ell_2, \ldots, \ell_m\}$, where m is the number of suggested labels. Since we pick the top 10 keywords from both topics, this means m is typically 20.

Our goal is to find a set of relevant labels $R_i \subseteq L_i$ for document d_i, where $q = |R_i|$ is the number of relevant labels. Algorithm 1 details the steps for generating the set of suggested labels L_i.

Next, we pick a subset of the labels from L_i, designating them R_i. We use two different methods to achieve this: the threshold method and the GA.

Threshold method. We choose a threshold value for the contribution percentage (CP), e.g., $\theta = 0.6$. If the CP of the first dominant topic (CP_{T1}) is greater than this value, then assign the first q keywords of that topic as labels for the document. If it is less than the threshold, we then take the first $q/2$ keywords of the first dominant topic and the first $q/2$ keywords from the second dominant topic. This method is simple and yet computationally inexpensive. However, it may not always yield accurate results because other keywords in the suggested set may be more suitable than taking the first q or $q/2$ keywords from T_1.

Algorithm 1: Generating labels for the documents.

Input: Corpus D.
Output: Suggested labels $L_i \; \forall$ document $d_i \in D$.

```
begin
    Create a topic model using the optimal hypertuned parameters
      (Section 5.2). Returns list of topics
    Pick top ten terms (keywords) for each topic
    foreach d_i ∈ D do
        Classify d_i using the topic model and get the contribution
          percentage (CP) for each topic
        Let L_i = keywords from the first (T_1) and second (T_2)
          dominant topics (i.e., topics with the two highest
          contribution percentages)
    end
end
```

Genetic algorithm. In this case, we apply the algorithm to the set of suggested labels L_i as the population. Each individual v from the population V has q genes (i.e., keywords). We propose a fitness function (Eq. (1)) that is based on the coherence measure to evaluate the individual for every iteration. More details to follow.

3.2. Genetic algorithm for generating multiple labels

Once we get the list of suggested labels L_i for the document d_i, we need to annotate it with a set of relevant labels R_i. We propose two types of GA algorithms to accomplish this task, designating them as GA_{L1} and GA_{L2}.

In the GA first labeling technique (GA_{L1}), we choose a threshold value θ for the CP. In case the CP of the first dominant topic $CP_{T1} > \theta$, then we assign the first q keywords of that topic as labels to the document, else the GA proceeds with the set of suggested labels L_i to choose the best q labels that display coherence with the document.

The GA_{L2} is similar to GA_{L1}, except the actions are different. For the case $CP_{T1} > \theta$, we use ten keywords from T_1 (the first dominant topic) as genes to generate the population V, otherwise, we use keywords from T_1 and T_2 represented by L_i.

In both GA techniques, the individuals have q genes, $v = \{\ell_1, \ell_2, \ldots, \ell_q\}$. Each individual represents the candidate labels for the current document. The fitness function ($Fitness$) for evaluating the individual is given by,

$$Fitness = CoherenceScore + reward - penalty, \tag{1}$$

where the coherence score ($CoherenceScore$) for the individual v with the current document d_i is calculated as follows. We start by creating a new list S_i containing the individual v keywords (the candidate labels) and document words. Then, calculating the coherence between all words in the new list using the UMass measure ($UMassScore$) (Mimno et al., 2011),

$$CoherenceScore = \sum_{j>k} UMassScore(w_j, w_k), \tag{2}$$

$$UMassScore(w_j, w_k) = \log\left(\frac{N(w_j, w_k) + \epsilon}{N(w_k)}\right), \tag{3}$$

where w_j, w_k are word pairs of the list S_i; $N(w_j, w_k)$ is the count of documents in the corpus containing the words w_j, w_k; and $N(w_k)$ is the count of documents in the corpus containing the word w_k. We used the UMass measure to calculate the coherence score since it is faster than other measures. A small value ϵ avoids calculating the logarithm

of zero in Eq. (3). Following the suggestion by Stevens et al. (2012), we set $\epsilon = 1E{-}12$.

We then modify the coherence result by either a reward or a penalty. The reward is used (e.g., 0.7) if the individual's keywords do not contain a repeated word (usually due to mutation action). In the case of repeated words, the penalty is applied, and the numerical count of the repeated words specifies it. Then, in each iteration, we change the fitness value for the current solution (which has the highest fitness score) to a smaller value to avoid selecting the same solution again. The two techniques for generating multiple labels GA_{L1} and GA_{L2} are detailed in Algorithms 2 and 3, respectively.

Algorithm 2: GA first labeling technique (GA_{L1}).

Input: Document d_i, keywords of T_1, set of suggested labels L_i, contribution percentage of first dominant topic CP_{T1}.
Output: Relevant labels $R_i = \{\ell_1, \ell_2, \ldots, \ell_q\}$.

```
begin
    if CP_T1 > θ then
        R_i = first q keywords of T_1
    else
        // GA starts
        Genes = L_i
        Randomly create population V from Genes set (each individual v has q genes)
        counter = 0
        while counter < NumGenerations do
            Calculate fitness function for each v
            CurrentSolution = individual v with highest fitness function
            Create a new generation V using the actions: selection, crossover, and mutation
            Assign a very small value to fitness value of CurrentSolution
            counter = counter + 1
        end
        R_i = CurrentSolution
    end
end
```

Algorithm 3: GA_{L2}.

Input: Document d_i, keywords of T_1, L_i, and CP_{T1}.
Output: Relevant labels $R_i = \{\ell_1, \ell_2, \ldots, \ell_q\}$.

```
begin
    if CP_T1 > θ then
        Genes = keywords of T_1
    else
        Genes = L_i
    end
    Create population V randomly from Genes set (q genes per individual v)
    counter = 0
    while counter < NumGenerations do
        Calculate fitness function for each v
        CurrentSolution = individual v with highest fitness function
        Create a new generation V using actions: selection, crossover, and mutation
        Assign a very small value to fitness value of CurrentSolution
        counter = counter + 1
    end
    R_i = CurrentSolution
end
```

4. The datasets

We picked three different Arabic corpora and three large social media datasets in English to test our model.

Table 2
General statistics of the Arabic datasets used: Saudi Press Agency (SPA), Arabic News Texts (ANT) (Chouigui et al., 2018), and Open Source Arabic Corpora (OSAC) (Saad & Ashour, 2010).

	Number of documents		
Category	SPA	ANT	OSAC
Economics	2400	326	3102
Sports	2484	1460	2419
Culture	2094	124	–
Politics	2558	514	–
Society	1466	1087	–
General	2400	–	–
International news	–	1260	–
Local News	–	1260	–
Technology	–	83	–
Astronomy	–	–	557
Cooking	–	–	2373
Education	–	–	3608
Health	–	–	2296
History	–	–	3233
Law	–	–	944
Religion	–	–	3171
Stories	–	–	726
Number of categories	6	8	10
Total # documents	13,402	6,114	22,429
Total # words	2,377,903	794,095	18,183,511
Avg # words/document	177	130	810

The Arabic corpora are: the Saudi Press Agency (SPA) corpus,[1] the Arabic News Texts (ANT) corpus v1.1 (Chouigui et al., 2018), and the Open Source Arabic Corpora (OSAC) dataset (Saad & Ashour, 2010). These datasets were used in many works related to Arabic text categorization, e.g., Almuzaini and Azmi (2020) and El-Alami et al. (2020).

The number of categories varies between the corpora. It is six for SPA, nine for ANT, and for OSAC, it is ten categories. Table 2 summarizes the dataset used in this work. The same subset of ANT corpus (eight instead of nine categories) was used in Almuzaini and Azmi (2020). Only two categories are shared between all three corpora; the rest are shared between two or just unique to a corpus.

For effective evaluation, we decided to assess our proposed model with different datasets over large variable scales. It is worthwhile to note that only the SPA corpus had balanced classes. For example, the "technology" category has only 83 documents in the ANT corpus, whereas "sports" has 1460. The same imbalance occurs with the OSAC corpus, for instance, in the categories of "astronomy" and "history".

For the English dataset, we used three different social media datasets for academic research (Bibsonomy, CiteULike-a, and CiteULike-t). The sharing of references is the object of online-only social bookmarking tools such as Bibsonomy and CiteUlike. Beside references, users can also share and annotate publications. Metadata of the documents such as title and abstract are also available. The original Bibsonomy dataset comprises of 3,794,882 annotations, 868,015 resources, 283,858 distinct tags from 11,103 users (Benz et al., 2010).[2] We used the cleaned version of this dataset from a previous work (Dong et al., 2017), which has a total of 12,101 publications. For cleaning, the authors selected only the documents containing both the title and the abstract.

CiteULike-a and CiteULike-t are benchmark datasets (Wang et al., 2013) having a total of 7386 and 8311 tags, respectively. We used the cleaned versions from Dong et al. (2020), which contain 13,319 and

[1] The official Saudi Press Agency: http://www.spa.gov.sa. We collected a four-month period worth of documents starting Sept. 2018.

[2] Our dataset is based on version "2015-07-01", available at https://www.kde.cs.uni-kassel.de/wp-content/uploads/bibsonomy/. This dataset was accumulated during the period 2005 to July 2015.

Table 3
Statistics for the English datasets used. The cleaned version of the dataset are from: Bibsonomy (Dong et al., 2017), and CiteULike (Dong et al., 2020). The statistics include: number of documents, total number of tags, the average number of tags per document, and the vocabulary size.

Dataset	# Docs	# Tags	Avg tags/doc	Voc. size
Bibsonomy (clean)	12,101	5196	11.59	17,619
CiteULike-a (clean)	13,319	3201	11.60	17,489
CiteULike-t (clean)	24,042	3528	7.68	23,408

24,042 articles, respectively. The cleaning step involves removing tags that occur less than 10 times. Table 3 summarizes the English dataset we used.

5. Evaluation and results on arabic dataset

In this section, we will evaluate our proposed system using the Arabic dataset (Section 4). For testing on the English dataset, see Section 7. For convenience, we will break down this section into several subsections: (i) the preprocessing stage, (ii) fine tuning the parameters, (iii) experimental results and discussion of applying the genetic algorithm for generating multiple labels, and (iv) partitioning experiments for topic extraction.

5.1. Data preprocessing

Preprocessing is critical for the topic model as it affects the resulting topics (Johnsen & Franke, 2019; Schofield et al., 2017). Typically, stemming is an important step in many NLP applications. However, in our case, we do not recommend using stemming since it may result in either a meaningless word or a word with unrelated meaning. For instance, stemming the word قانون meaning "law" results in the word قان, which has a different meaning in Arabic. For this very reason, we could not use these resultant words for annotating the corpora. Algorithm 4 outlines our preprocessing stage.

Algorithm 4: Preprocessing the dataset.

Input: Corpus D
Output: Set of bigrams and trigrams for D

begin
  Filter out digits, non-Arabic words, and special symbols (e.g., !, @, #, $, etc.)
  Remove all the diacritical marking
  Normalize the words (e.g., 'alif' {{أ، إ، آ} → ا)
  Remove definite articles (e.g., ال، وال، بال، كال، لل)
  Remove one or two letter words
  Remove stop-words and unimportant words
  Generate bigrams and trigrams
end

We compiled a set of words that frequently appear in the news, such as acts of speech (e.g., قال، أكد، أضاف), days of the weeks, and words that are related to dates (e.g., ربيع، يوم، عام، موافق). We consider these words as unimportant and thus removed. An n-gram is a sequence of n words. The 2-gram (3-gram) are respectively called bigram (trigram). For instance, "I like drinking tea" has the following bigrams "I like", "like drinking", and "drinking tea"; and two trigrams "I like drinking" and "like drinking tea".

5.2. Fine-tuning the parameters

For the experiments, certain underlying parameters need to be fine-tuned. These values will be used in all future experiments. We experimented with two different algorithms to build and train topic models. The online Variational Bayes and Gibbs Sampling. Our main objective is to derive the optimal combination of parameters for model training. The models are evaluated using coherence scores.

We adopted the C_v coherence measure (Röder et al., 2015), which ranges between 0 and 1. The C_v is based on normalized pointwise mutual information (NPMI), and it is calculated for each word w_i from the top-ranked keywords associated with a topic (Eq. (4)),

$$\mathrm{NPMI}(w_i, w_j)^{\gamma} = \left(\frac{\log \dfrac{P(w_i, w_j) + \epsilon}{P(w_i) \cdot P(w_j)}}{-\log(P(w_i, w_j) + \epsilon)} \right)^{\gamma}. \tag{4}$$

The probabilities are estimated based on word co-occurrence counts among the documents. The word co-occurrence counts are determined using context windows that contain all words located within ±5 tokens around w_i, where w_j is a word from the context of w_i. To avoid a zero logarithm, we add a constant ϵ which is usually set to 1. The parameter γ is used to assign more weight for high NPMI values.

Variational Bayes (VB) is an alternative to Monte Carlo sampling methods, which takes a fully Bayesian approach to statistical inference over complex distributions that are difficult to evaluate directly or sample. We use online VB (Hoffman et al., 2010).

For the VB algorithm, we investigated the following parameters. The number of topics $K = \{6, 7, \ldots, 14\}$, and the Dirichlet parameters Alpha and Beta. We tested Alpha = {0.01, 0.31, 0.61, 0.91, symmetric, asymmetric}, and Beta = {0.01, 0.31, 0.61, 0.91, symmetric}.

The K is the number of topics that may be extracted from the corpus. Although the number of topics is known beforehand, the corpus usually contains more topics than those pre-defined. For example, the "sports" category may include various sub-topics (e.g., "football", "car race", "horse race", etc.), which are also extracted while maintaining a high coherence between topics. Alpha is related to the document-specific topic distribution, whereas Beta is related to the topic-specific word distribution. Symmetric means all the topics have equal probability, which is $1/K$ in the case of Alpha. For Beta, it is all the words are equally likely. The asymmetric value means each topic has a different probability, and it is used for Alpha only.

Gibbs Sampling (GS) (Griffiths & Steyvers, 2004) is a Markov chain Monte Carlo algorithm for obtaining a sequence of observations which are approximated from a specified multivariate probability distribution, when direct sampling is difficult. GS generates a Markov chain of samples, each of which is correlated with nearby samples. As thus, samples from the beginning of the chain are usually discarded.

We used the MALLET.[3] library to build the topic model using the GS algorithm. We investigated two parameters. The number of topics K, and optimize-interval. The latter confers a better fit to the data by allowing some topics to be more prominent than others. For example, optimize every ten iterations. We tested optimize-interval $OI = \{10, 20, 30, 50, 100, 500\}$. We confined the number of topics K to the same values in VB algorithm.

Each of the tuning experiments was performed using linear sensitivity analysis. It involves modifying one parameter at a time while keeping others constant and then analyzing the three corpora in the datasets.

The results of applying the VB and GS algorithms on the datasets are listed in Table 4, where we limit the entries to few top ones based on the coherence C_v. Fig. 2 shows the result of tuning the parameters using the GS algorithm on our dataset. Table 5 lists all the generated topics ($K = 11$) for our dataset using the GS model. A detailed list of the 11 generated topics and their keywords for the SPA corpus using both models is shown in Table 6. The respective C_v score is 0.5993 and 0.6030 for the VB and GS models.

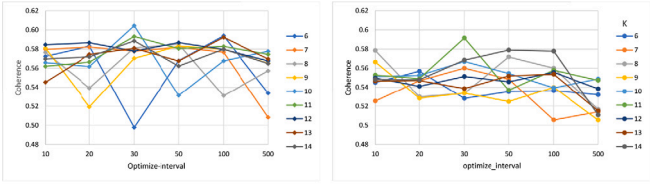
[3] MAchine Learning for LanguagE Toolkit, http://mallet.cs.umass.edu/topics.php.

Table 4
The top four topic models for the individual corpus based on the C_v coherence measure using Variational Bayes (VB) and Gibbs Sampling (GS) algorithms. K is the number of topics, and OI is the optimize-interval.

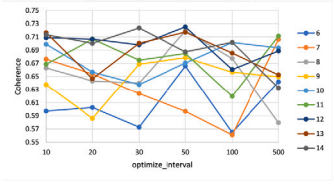
	Variational Bayes				Gibbs Sampling		
Corpus	K	Alpha	Beta	C_v	K	OI	C_v
SPA	6	Symmetric	0.31	0.6121	10	30	0.6045
	14	0.01	0.91	0.6047	11	30	0.6030
	6	0.01	0.31	0.6009	6	100	0.5940
	11	0.01	0.31	0.5993	13	100	0.5923
ANT	11	Asymmetric	0.61	0.5086	11	30	0.5916
	13	Asymmetric	0.61	0.5068	14	50	0.5790
	10	0.01	0.61	0.5065	8	10	0.5786
	10	Asymmetric	0.61	0.4987	14	100	0.5779
OSAC	10	Asymmetric	0.31	0.7730	12	50	0.7251
	11	0.91	0.31	0.7716	14	30	0.7235
	11	Symmetric	0.61	0.7618	8	50	0.7191
	10	0.31	0.31	0.7608	13	50	0.7172

Table 5
List of the generated topics from each corpora using the GS model ($K = 11$ topics, and $OI = 30$). The topics are listed in order of their significance.

	Generated topics		
	SPA	ANT	OSAC
1	Local news	National security	Economics
2	Local sports	Economics	History
3	Economics	Sports	Religion
4	Culture	Sports	Politics
5	Politics	Local news	Religion
6	Events & Activities	Regions	Education
7	International news	Health	Sports
8	Social	International news	Cooking
9	Sports	Culture	Stories
10	Education	Politics	Health
11	Stock market	Law	Astronomy



(a) SPA corpus (b) ANT corpus



(c) OSAC corpus

Fig. 2. Result of hypertuning the parameters for Gibbs sampling algorithm on different corpus in the datasets. We show the coherence measure (C_v) for different values of optimize-interval (OI) and the number of topics (K).

5.3. Variational Bayes or Gibbs sampling—which one?

We experimented with two different algorithms during the parameter tuning (Section 5.2), the VB and the GS models. First, we summarize what we learned from the tuning process, and then we argue which of the two models is a better choice. Briefly, we observed the following in the tuning process:

- We already knew beforehand the number of topics in each corpus of the dataset (Table 2). A model, however, may generate more topics. For example, we have eight categories in the ANT corpus, but we get a higher coherence measure C_v for 11 topics.
- We achieved better performance on the SPA corpus compared to the ANT. This is due to the (almost) balanced distribution of documents over the topics.
- The ANT corpus has an imbalance in the distribution of documents over topics, e.g., "technology" has only 83 documents, while other topics have way more documents. Due to this disparity, keywords related to the dominant topics will cover those in minor topics, resulting in minor topics' disappearance.
- The OSAC corpus had the highest C_v since its categories are picked with little or no overlap (e.g., "astronomy", "cooking", "history", etc.). As with ANT, due to the imbalance, some of the topics might not show up. What is interesting in our experiment is that category "law" is missing, whereas "astronomy" is not (see Tables 2 and 5). Despite the former having more documents (944 vs. 557 documents, respectively). The only explanation for this behavior is the set of keywords for "astronomy" are distinct and rarely shared with other categories, a case which may not hold true for the category "law".
- Both models may generate the same topic label more than once. The topic "local news" appeared twice in the list of topics generated by the VB model (see Table 6). The GS model generated the duplicate topic "sports" (ANT corpus in Table 5).
- For the VB algorithm, the best Beta values for all corpus are (0.31, 0.61), and for Alpha these are (0.01, 0.31, asymmetric).
- For both algorithms, the best choice for the number of topics is between 10 and 12 for any of the corpus.
- For the GS algorithm, we noted: (a) optimize-interval = 500 performed poorly in most of the cases; and (b) $OI \propto 1/K$. The latter means that a higher OI is beneficial when considering fewer topics in most cases.

There is no clear winner. Regarding the coherence score (C_v), the VB did better on the OSAC corpus, while the GB did better on the ANT corpus (see Table 4). On the other hand, the set of topics and the associated keywords generated by the GB model is far better than those by VB (see Table 6), even though their C_v score is very close.

Röder et al. (2015) argued that C_v correlates better with human ratings among the available coherence measures. The value of $C_v \in [0, 1]$, and from our experience, we believe the best score for coherence measure that yields better correlation to humans is a C_v value between 0.5 and 0.6.

In the literature, the C_v score for good models usually does not exceed 0.6, see e.g., Mifrah and Benlahmar (2020), Molavi et al. (2020), Wang and Mengoni (2021) and Zhang et al. (2020). In Ordun et al. (2020) analyzed Covid-19 Twitter data for topics and key terms and features (among others). The study is based on a compiled collection of approximately 23.8 million tweets during the two weeks period starting March 24, 2020. The authors reported low quality of topics associated with high C_v.

The calculation of C_v involves using cosine similarity between the generated vectors word pairs. We show that $C_v > 0.6$ corresponds to having identical words or bigrams, and thus to be avoided. Consider three lists: X, Y, and Z as shown in Table 7. Based on the SPA corpus, the respective C_v score for the three lists is 0.56, 0.65, and 0.72. Regardless of the C_v score, X is the better list.

Several earlier works recommended using GS over VB. The collapsed Gibbs sampler (Griffiths & Steyvers, 2004) is a popular choice for topic model estimation in many applications due to its efficient implementation and results, e.g., Asuncion et al. (2012), Canini et al. (2009), Smola and Narayanamurthy (2010) and Yao et al. (2009). Papanikolaou et al. (2017) studied VB and GS and proved the former model fails when the hypothesis space grows.

As a result, we believe that GB makes a better model to construct the topic modeling than VB. We will try confining $C_v \in [0.5, 0.6]$.

Table 6
List of topics ($K = 11$) and the associated keywords generated from the SPA corpus using both models. Note how each model ranked the topics differently.

	Variational Bayes model (Alpha = 0.1, Beta = 0.31)		Gibbs Sampling model ($OI = 30$)	
Topics	topic rank	Keywords	topic rank	Keywords
Local News	5	امير، رئيس، منطقة، ملك، مجلس، عبدالله، ادارة، فيصل، ملكي _امير، مدير	1	منطقة، امير، رئيس، مجلس، شريفين، مملكة، ملكي _امير، ملك، خادم _حرمين، دكتور
Local Sports	1	بطولة، سعودي، رياضي، رياض، اتحاد، مركز، نادي، رياضة، وطنيبلكرة _قدم	2	بطولة، سعودي، اتحاد، منتخب، رياضي، رياض، رياضة، مركز، سعودية، نادي
Economics			3	مملكة، عمل، سعودية، قطاع، اقتصادي، هيئة، شركة، اعمال، وزارة، مشاريع
Culture			4	عربية، ثقافي، دكتور، مملكة، مركز، ملك، لغة، عالم، كتاب، اسلامية
Politics			5	عربية، رئيس، تعاون، مجلس، مملكة، دول، سياسي، وزير _خارجية، وزراء، بلدين
Events & Activities	3	جمعية، تعليم، جامعة، دورة، برنامج، ادارة، اجتماعي، مدير، مركز، فعاليات	6	معرض، مهرجان، فعاليات، ثقافي، مملكة، وطني، سعودية، تراث، مشاركة، هيئة
International News	6	عربية، رئيس، اتحاد، مملكة، سعودية، امارات، دولية، دولي، عربي، دولة	7	سياسي، امم _متحدة، امن، دولي، يمن، حكومة، دولية، مجلس، يمني، رئيس
Social			8	جمعية، عمل، اجتماعية، تنمية، اجتماعي، مركز، وزارة، مجتمع، خدمات، صحية
Sports	2	منتخب، عاما، اخضر، كاس _اسيا، صين، بنتيجة، اسيا، منتخبات، اخضر _شاب، بمنتخب	9	مركز، فريق، دوري، لكرة _قدم، رياضي، جولة، نقاط، رياض، ملعب، امير
Education			10	تعليم، جامعة، ادارة، عمل، دكتور، مدير، برنامج، ملتقى، برامج، عامة
Stock market			11	اقتصادي، اسهم، بنسبة، مئة، ارتفاع، مستوى، بورصة، مؤشر، تعاملات، مليون _دينار
International Sports	4	تونسي، تونس، تونسية، نجم _ساحلي، صفاقسي، افريقي، ترجي، نادي _بنزرتي، اتحاد _منستيري، ملعب _قابسي		
Sports	7	مركز، لكرة _قدم، فريق، دور، دوري، مباراة، نظيره، فوز، نقاط، جولة		
Security	8	امن، عسكرية، رماية _تكتيكية، قوات، سعودي _لقوى، داخلي، امنية، عقيد، رماية، احتلال		
International Sports	9	نهائي، عندما، صفر، عامي، دوري _انجليزي، فرنسي، احد، دوري _اسباني، استراليا، الماني		
Local News	10	مملكة، عمل، لجنة، جميع، سعودية ، لتحيق، مجلس، مستوى، رئيس، اجل		
Racing	11	نادي _فروسية، جواد، فروسية، مفتوح _درجات، نارية، سيارات _دراجات، انتاج، حفل _سباقه، جبيل، ميدان _فروسية		

Table 7
Example to illustrate that higher coherence values (C_v) do not necessarily correspond to better labels sets.

List	Translated	C_v
X = {معرض، مملكة، مهرجان، فعاليات، ثقافية، تراث، سعودية}	X = {‘exhibition’, ‘kingdom’, ‘festival’, ‘events’, ‘cultural’, ‘heritage’, ‘Saudi’}	0.56
Y = {فعاليات، مهرجان، فعاليات، مشاركة، ثقافي، مهرجان، فعاليات}	Y = {‘events’, ‘festival’, ‘events’, ‘participation’, ‘cultural’, ‘festival’, ‘events’}	0.65
Z = {مهرجان، مهرجان، مهرجان، مهرجان، ثقافي، فعاليات، ثقافي}	Z = {‘festival’, ‘festival’, ‘festival’, ‘festival’, ‘cultural’, ‘events’, ‘cultural’}	0.72

5.4. Using GA to generate multiple labels

In the previous section, we alluded to our decision to use GB for topic modeling. Our main objective is to apply the multi-label annotation to the Arabic corpus using the proposed model. At the outset, we construct the topic model using the optimal set of parameters identified earlier. In turn, we obtained the set of suggested labels L_i for each document $d_i \in D$ (Section 3). The threshold method, which is suitable when resources are limited, was not sufficiently accurate. The other choice is to use GA.

For the GA we set the parameters as follows: (a) the number of relevant labels $q = 6$, and (b) the threshold $\theta = 0.6$. For the SPA corpus, we applied GA_{L1} method, setting the number of generations to 20. The GA_{L2} was not used since it is computationally expensive for large datasets. For the ANT corpus, we applied the GA_{L2} method

Table 8
The average coherence of the relevant labels R_i for each category in the SPA and ANT corpora. A value closer to zero indicates better coherence.

	Average coherence	
Category	SPA	ANT
Economics	−4.2241	−3.7350
Sports	−4.2874	−6.5737
Culture	−3.1086	−4.4188
Politics	−3.9414	−5.3505
Society	−3.9727	−6.0120
General	−3.7961	–
International news	–	−5.4740
Local News	–	−5.5518
Technology	–	−5.5384

Table 9
A sample document from the "culture" category in the SPA corpus and the resulting labels using GA_{L1} method. The CP_{T1} (CP_{T2}) is the contribution percentage (CP) of the top two dominant topics.

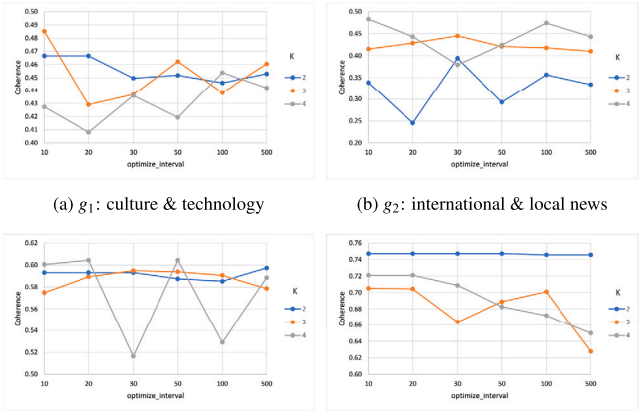
Document	الدمام ١٨ ربيع الآخر ١٤٤٠ هـ الموافق ٢٥ ديسمبر ٢٠١٨ م واس تبنت الشركة السعودية للكهرباء، دعم أكبر مهرجان لذوي الاحتياجات الخاصة وأسرهم، الذي سينطلق للسنة الثانية في المنطقة الشرقية على مدى ثلاثة أيام، حيث يبدأ يوم الخميس المقبل ٢٧ ديسمبر٢٠١٨م، وذلك في الصالة الخضراء بالدمام. وتأتي مبادرة السعودية للكهرباء، في تقديم الدعم اللوجستي للمهرجان الفريد من نوعه، ضمن برامج المسؤولية الاجتماعية، التي تقدمها الشركة لخدمة هذه الفئة الغالية وذويهم، كما تنظم هذا المهرجان، آجمعية همم لأسر ذوي الإعاقة، التي تهتم بخدمتهم، من خلال دعم ورعاية وتوعية الأسر بتنظيم الفعاليات والأنشطة التوعوية والتثقيفية للأسر ورعاية أبنائهم وبناتهم من ذوي الاحتياجات الخاصة. وسيتضمن المهرجان، فعاليات وأنشطة ترفيهية ومسابقات رياضية وثقافية لهم ولأسرهم.
CP_{T1} 0.6533	Keywords (top dominant topic) مركز، وزارة، مجتمع، خدمات، صحية، جمعية، عمل، اجتماعية، تنمية، اجتماعي
CP_{T2} 0.3261	Keywords (second top dominant topic) وطني، سعودية، تراث، مشاركة، هيئة، معرض، مهرجان، فعاليات، ثقافي، مملكة
Coherence −2.7024	R_i labels جمعية، عمل، اجتماعية، تنمية، اجتماعي، مركز

with 20 generations for the first case ($CP_{T1} > \theta$) and 30 generations otherwise.

We calculated the coherence for the relevant labels R_i using Eq. (2) for each document d_i. Table 8 reports the average coherence for each category for the SPA and ANT corpus.

Based on the average coherence score, the experiment revealed that the SPA corpus yields better results when compared to the ANT. The only explanation we have is the ANT corpus has imbalanced classes. We will describe a more efficient solution to annotate a corpus with imbalanced classes in Section 5.5.

We observed a significant correlation between θ and the resultant labels, with a small θ proving detrimental. When θ is small, the GA depends solely on the top dominant topic (T_1) for the keywords. While using keywords from the dominant topics T_1 and T_2 would have improved the resultant labels. An example of resulting labels from a sample document in the SPA corpus using the GA_{L1} method is shown in Table 9. We see the relevant labels are a subset of those in the top dominant topic. Unlike GA_{L2}, the GA_{L1} method ignores those labels generated by the second top dominant topic. We can see that had we combined the keywords from both topics, and it would have improved the resultant label.



(a) g_1: culture & technology (b) g_2: international & local news

(c) g_3: economics & politics (d) g_4: society & sports

Fig. 3. Hypertuning the parameters for all the groups in the ANT corpus with combined categories.

5.5. Partitioning experiments for topic extraction

If the categories of the corpus are known beforehand, and if the number of documents in the categories varies among themselves, we recommend a partitioning approach before topic extraction, to address the problem of imbalance among the categories.

The ANT and OSAC corpus have eight and ten categories, respectively, but they include an unequal number of documents between categories. This causes some topics to dominate others. For example, "law" is one of the categories in the OSAC (Table 2); however, it is missing in the list of generated topics (see Table 5).

Algorithm 5: Partitioning approach for topic extraction.

Input: Set of categories $C = \{c_1, c_2, \ldots\}$ for corpus D.
Output: New set of groups $G = \{g_1, g_2, \ldots\}$.

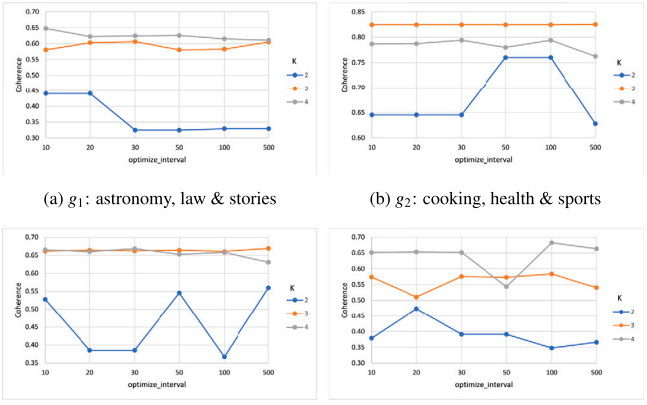
```
begin
    Combine two to three categories c ∈ C of close sizes (in number
      of documents), assigning them to a new group g ∈ G
    foreach g ∈ G do
        Perform hyperparameter tuning (Section 5.2) using the GS
          model to derive the optimal set of parameters
        Apply GA method to generate multiple labels (Section 5.4)
    end
end
```

Partitioning is a known approach to address the imbalance dataset problem. It has been applied successfully in multi-label text classification, e.g., Awasare and Gupta (2017) and Gao et al. (2021). Our objective is to simply lump categories (of sizes that are close to each other) into a sub-dataset or new groups. We then tune the parameters for each of the new groupings. Algorithm 5 details this process.

The set of a new grouping of categories G is shown in Table 10. For example, in the ANT corpus, we combined the categories "culture" and "technology", both of approximately the same size, into one group g_1, having a total of 207 documents. We did not, for instance, group "technology" and "economics" since their sizes vary by much.

Table 10
Combined categories in the ANT and OSAC corpora.

	ANT		OSAC	
Groups	Combined categories	Size	Combined categories	Size
g_1	Culture (124) + Technology (83)	207	Astronomy (557) + Law (944) + Stories (726)	2227
g_2	International news (1260) + Local news (1260)	2520	Cooking (2373) + Health (2296) + Sports (2419)	7088
g_3	Economics (326) + Politics (514)	840	Economics (3102) + Religion (3171) + History (3233)	9506
g_4	Society (1087) + Sports (1460)	2547	Education	3608



(a) g_1: astronomy, law & stories (b) g_2: cooking, health & sports

(c) g_3: economics, religion & history (d) g_4: education

Fig. 4. Hypertuning the parameters for all the groups with combined categories in the OSAC corpus.

Next, we hyper-tune the parameters for the ANT and OSAC corpora after combining the categories. We may now derive an optimal set of parameters for building the model. We explored the following parameters: number of topics $K = \{2, 3, 4\}$, and the $OI = \{10, 20, 30, 50, 100, 500\}$. Figs. 3 and 4 show hyperparameter tuning results for each group with combined categories for both corpora.

The optimal number of topics for the ANT corpus is 3 and 2 for group g_1 (culture and technology) based on the coherence score. For groups g_2 (international and local news) and g_3 (economics and politics) a choice of 3 or 4 topics yields improved results. It is due to the diverse nature of the subjects usually covered in these types of text. For the last group, it is undoubtedly two topics. Moreover, for the OSAC, it is either 3 or 4 topics.

For group g_3 in the ANT corpus (Fig. 3c), we note the combination $K = 2$ and $OI = 500$ yields good coherence, whereas, for $K = 3$ or 4, results were enhanced when lower OI is considered. This agrees with our observation in Section 5.3, where we noted $OI \propto 1/K$.

We finally apply the GA method (Algorithm 5 step 5) to label each group using the previous step' parameters. For the GA method, our parameter setting is: (a) number of relevant labels $q = 6$, and (b) the threshold $\theta = 0.8$. The higher value of the threshold follows our finding in Section 5.4, and this new value is expected to yield more specific results. We decided to use the GA_{L2} method on both corpus with 20 generations.

Table 11 reports the new average coherence for each category in both corpora, the ANT and OSAC, following the process of combining the categories. For the ANT corpus, we note an improvement in the values of average coherence over those reported in Table 8 for each category. The improvement ranges from 5.98% for "economics" category to 45.13% for the "sports" category, where the average coherence was −6.5737 and new value is −3.6073. The improvement over all the categories is 22.45%. Is the improvement statistically significant? Using paired t-test with $\alpha = 0.05$, the one-tailed p-value = 0.00206 showing it is significant. This goes on to show that the partitioning approach did provide a possible solution to the class imbalance problem.

An example of generated labels for a document in group g_1 from the ANT corpus is shown in Table 12. As expected, increasing the threshold θ to 0.8 was beneficial. The resultant labels are satisfactory since they depend on both topics after increasing θ.

Table 11
The average coherence of each category for the resultant labels following the partitioning approach.

	Average coherence	
Category	ANT	OSAC
Economics	−3.5115	−2.5114
Sports	−3.6073	−5.7266
Culture	−4.0650	–
Politics	−3.6187	–
Society	−4.3242	–
International news	−4.3743	–
Local News	−4.5067	–
Technology	−4.3664	–
Astronomy	–	−1.6915
Cooking	–	−2.3107
Education	–	−0.8919
Health	–	−6.3050
History	–	−3.1337
Law	–	−2.9352
Religion	–	−4.3994
Stories	–	−4.0117

Table 12
A sample document in the group g_1 from the ANT corpus and the resulting labels using the GA_{L2} method.

Document	أعلنت وزارة الشؤون الثقافية في بلاغ لها اليوم الاثنين ١٤ نوفمبر ٢٠١٦ أنها وضعت حدا لتكليف إبراهيم لطيف مدير الدورة ٢٧ لأيام قرطاج السينمائية. ودعت الوزارة في ذات البلاغ الهيئة المديرة لتقديم التقريرين المالي والأدبي للدورة المذكورة في أقرب الأجال إلى وزارة الإشراف.
CP_{T1} 0.7664	Keywords (top dominant topic) شركة، مسرحية، خاصة، وزير، وزارة، ثقافة، ثقافية، عربية، انتاج، عمل
CP_{T2} 0.2309	Keywords (second top dominant topic) مهرجان، فيلم، دورة، جائزة، عرض، سينما، مخرج، دولي، سينمائية، افلام
Coherence −3.3942	R_i labels سينمائية، ثقافية، دورة، وزارة، فيلم، ثقافة

For the OSAC corpus, we observed a considerable fluctuation in the coherence. For example, for the category "education", it is close to zero, while for "health", it is near −6. Most of the OSAC categories have a large number of documents which indicates the diversity of topics. Low coherence indicates that the category should be treated alone to achieve better results. The "education" category has the highest coherence since we treated it as a single group.

The nature of the data in the OSAC corpus is problematic for the topic model. The corpus was compiled using more than one source. The sources include BBC Arabic, CNN Arabic, Aljazeera.net, and twenty other Internet resources. The compilers of the OSAC did not bother filtering out the noise. Advertisements at the top of the form were kept. This noise confuses the model regarding the content of the article.

Table 13
The average coherence and similarity measures for the labels generated by our proposed model vs. humans (crowdsourced) for the SPA corpus. In Word2vec similarity, a value of zero means equal, while in cosine similarity, a value of 100 stands for full equality.

	Average coherence		Similarity	
Category	Our system	Humans	Word2vec	Cosine
Economics	−4.2241	−3.4603	0.1815	43.797
Sports	−4.2874	−2.7533	0.1550	53.864
Culture	−3.1086	−3.2759	0.2161	44.748
Politics	−3.9414	−3.2205	0.1721	42.896
Society	−3.9727	−3.3632	0.2963	34.073
General	−3.7961	−3.5651	0.2210	42.070
Average	−3.8884	−3.2731	0.2070	43.575

Removing the advertisement is a challenge since its text varies between documents.

6. Crowdsourcing procedures

We decided to follow this track due to the lack of a reference to evaluate our proposed solution on the Arabic text. Crowdsourcing is a viable option to annotate documents than relying on human experts. However, as we learned the hard way, the quality of the work is not guaranteed.

Many Internet sites offer this service, but few offer the option enabling the user (the consignee) to control the quality of the output. Moreover, the number of Internet sites that support the Arabic language is scarce. We identified two: Click Worker[4] and the Amazon Mechanical Turk (Mturk)[5] while searching for a suitable service.

Click Worker is a German crowdsourcing platform with over 2 million people worldwide working for it. The company offers the option to select the country and the language the users must be familiar with. Unfortunately, Click Worker does not fully support the Arabic, such as the right-to-left text direction. We used this service to annotate 2000 documents from the SPA corpus. As a trial, we published 100 articles and based on the quality of the results; we picked 40 clickworkers. We designed a form with clear instructions. These instructions were presented to the worker before starting the task. The following is the set of instructions: (a) carefully read the article; (b) as you read, identify suitable tags in your mind; (c) go over the suggested labels; (d) you have a limit of 6 labels; (e) choose (suitable) if the labels fit the article, or (not appropriate) if it does not; (f) review your choices and pick the most appropriate 6 labels; (g) write a title for the article without listing your suggested labels; and (h) write another title of the article, making sure it does not include any of the suggested labels.

With every article, a clickworker is given a list of 20 suggested labels to pick from, out of which they have to select only six labels they considered most relevant. We asked the users to write the title in Arabic to ensure that the worker understood the article. It is a sort of quality control when reviewing the results. Following our initial review of the results, we were disappointed to see many documents incorrectly annotated. The Click Worker platform does not provide an option to reject the results prompting the worker to redo the task.

Next, we investigated Amazon's Mturk, for which we created another form. Using the same set of instructions for the Click Worker, but added some constraints to maximize the quality. The total number of workers who finished the task in Mturk was 220.

We calculated the coherence of the labels for all the articles produced through the Mturk crowdsourcing and then computed the average coherence for each category. For the sake of comparison, we report this side by side the average coherence per category for the labels generated by our model (see Table 8). Both averages and the similarity using Word2vec (Mikolov et al., 2013) and cosine are listed in Table 13. Initially, the Word2Vec model was trained, then we used the sent2vec library[6] to calculate the similarity. The cosine similarity was calculated using the formula $A \cdot B / \|A\| \cdot \|B\|$, where the numerator is the dot product of two vectors A and B representing the two sets of generated labels.

We see that our model is close to the human-generated labels for most of the categories through the coherence results. The exception is "economics" and "sports" where humans outdid our model. Overall, the humans did a better job in labeling when compared to our model, in much as such as improving the average coherence by 15.8%. On the other hand, the similarity results using Word2vec model show our model's error rate is 20.7% compared to human labeling. It is worth noting the Word2vec model considers synonyms when calculating the similarity, whereas cosine similarity only considers exact word matching.

7. Evaluation and results on English dataset

The proposed model was originally developed for the Arabic dataset, it is adaptable for other languages.[7] In this section, we test our model on English dataset in order to validate it for other languages. There is no particular reason for picking English other than it has plenty of resources. Besides, English is a kind of lingua franca for the scientific community, the world over. So, the examples presented in this section will be easy to follow. For convenience, this section is divided into (i) experimental setting, (ii) discussion of results, (iii) comparison with other approaches, and (iv) comparing our model's performance on Arabic and English datasets.

7.1. Experimental setting

We will evaluate our proposed model on three datasets in English (see Section 4). The dataset will be divided into mutually exclusive training and testing sets. Similar to Dong et al. (2020), we set aside 10% of the dataset for testing with the rest used for training.[8]

For the preprocessing, we use Algorithm 4 with minor modification. The changes pertain to the English language. In place of steps 3–5, we perform lemmatization using Wordnet Lemmatizer[9] to convert the word to its base form (e.g., "networks" converted to "network").

We need to fine-tune parameters as we did with the Arabic dataset (see Section 5.2). We ran hyperparameter experiments with the GS model on the training set for all the datasets to find the best parameters. Based on the coherence score, the optimal number of topics is 8 for the Bibsonomy corpus and 10 for CiteULike-a. It is 14 for CiteULike-t, the largest of the three datasets. Table 14 shows the list of generated topics from each dataset using the GS model.

We constructed the topic model with the best parameters using the training part for all datasets (steps 1–3 in Algorithm 1). On the training set, document d_i content is represented by the title and the abstract along with the labels of the document (as free text). However, the labels were excluded from the testing set. To generate the labels, we applied steps 4–7 of Algorithm 1, then the GA_{L2} method (Algorithm 3) on the testing set of all the datasets using 20 generations. For the GA method, our parameter setting is: (a) the number of relevant labels $q = 6$, and (b) threshold $\theta = 0.8$.

[4] https://marketplace.clickworker.com.
[5] https://www.mturk.com/.
[6] https://pypi.org/project/sent2vec/.
[7] We would like to thank the reviewers for this suggestion.
[8] As the order of the documents is already randomized, so we selected the last 10% of the documents as testing set.
[9] https://www.nltk.org/_modules/nltk/stem/wordnet.html.

Table 14
List of the generated topics from each dataset using the GS model: Bibsonomy ($K = 8$ topics, and $OI = 30$), CiteULike-a ($K = 10$, $OI = 100$), and CiteULike-t ($K = 14$, $OI = 100$). We list the topics in order of their significance.

	Generated topics		
	Bibsonomy	CiteULike-a	CiteULike-t
1	Education	Software and development	Graph and Networks
2	Software and development	Algorithms	Software and development
3	Human interaction	Information retrieval	Information retrieval
4	Information retrieval	Data mining	Diseases study
5	Algorithm and programming	Brain signaling	Bioinformatics
6	Cerebral palsy	Bioinformatics	Recommendation system
7	Knowledge system	RNA and genes	Technology for learning
8	Bioinformatics	Knowledge system	Brain signaling
9	–	Networks	Classification algorithms
10	–	Evolutionary genetics	Biogeography
11	–	–	Molecular energy
12	–	–	Diseases and genes
13	–	–	Information system
14	–	–	Data analysis

Table 15
The average coherence and similarity measures for the labels generated by our proposed model vs. humans for the testing set of the three English datasets. The size refers to the number of documents in the testing set of a particular dataset.

		Average coherence		Similarity	
Dataset	Size	Our system	Humans	Word2vec	Cosine
Bibsonomy	1211	−2.2034	−3.1792	0.3340	27.75
CiteULike-a	1332	−3.2707	−4.0733	0.4025	15.87
CiteULike-t	2405	−3.6167	−4.3333	0.4524	9.93

7.2. Results and discussion

We calculated the labels coherence for each article produced by our system and then reported the average coherence. For the purpose of comparison, we also list the average coherence for the humans produced labels. Both averages and the similarity using Word2vec and cosine are listed in Table 15. For sample examples of the resulting labels generated by our proposed model, see Tables 16–17.

In Tables 16 and 17, we list all six labels generated by our model. On the other hand, the human labels are much more, 18 and 17, respectively, which we list them all. We would like to share an interesting observation. Many of the labels generated by our model were highly relevant to the document topic, and we were surprised they never showed up in human labels. For instance, in Table 17, only a single label ("brain") is common between both sets, while other labels (e.g., "stimulus" and "neuron") do not appear in human labels even though they are relevant.

In terms of coherence (Table 15), our model consistently outperforms human annotation across all three datasets. The improvement ranged between 16.5% for CiteULike-t dataset and 30.7% for the Bibsonomy dataset. The average improvement is 22%. Is this improvement statistically significant? To answer this question, we use paired t-test with $\alpha = 0.05$ on each of the datasets. The two-tailed p-value is $3.661\text{E}{-155}, 8.183\text{E}{-91}$, and $6.905\text{E}{-132}$ for the datasets Bibsonomy, CiteULike-a, and CiteULike-t, respectively. Based on coherence scores, we can safely conclude that our model significantly improves the annotation performance over humans.

What could be the reason for the low quality of human labels. Human annotators were given a free hand to pick labels of their choice. This resulted in some human labels being irrelevant or useless to the document's contents, e.g. "off" and "down" in Table 16. This was also observed by Dong et al. (2020). Better quality labels are possible if the annotators were given a set of choices to pick from, same as we did in crowdsourcing (Section 6).

Regarding the similarity measures, our model did better labeling the Bibsonomy dataset compared to the other two datasets (see Table 15). We assume this has to do with the CiteULike datasets having certain minor topics that the LDA failed to detect.

It is worth investigating why our model did not do well on the English dataset. The goal of our proposed system (Algorithm 1), was to find the most dominant topic in the documents (e.g. the probability of the topic occurring in the document is over 50%, $CP_{T1} > 0.5$). This is a necessary step in order to identify the relevant labels. We calculated the percentage of the documents in the testing set where $CP_{T1} \leq 0.5$, and discovered that every other document in the CiteULike-a and CiteULike-t datasets fell into the category. For the Bibsonomy, it is was every fifth document (see Table 18). Now, because the averages of CP_{T1} and CP_{T2} are direly low, the LDA distributes this document across different topics rather than identifying the dominant topic. This results in fewer matched labels (tags) in this portion of testing documents. For the datasets CiteULike-a and CiteULike-t it is less than one tag, 0.98 and 0.63 tags, respectively. For such cases, we suggest partitioning, as proposed earlier (Section 5.5) would certainly improve the results, as it might help discover hidden topics. Furthermore, introducing seed words for hidden topics or exploring larger number of topics with LDA might be worthwhile. We leave this as future work to investigate their effect on the proposed system.

7.3. Comparison with other approaches

As we share the same dataset, we decided to compare our approach to recent work on automatic annotation of social text in Dong et al. (2020). The authors devised a model which they refer to as JMAN (joint multi-label attention network). Their method is based on supervised deep learning architecture which uses bidirectional gated recurrent units (Bi-GRU) with attention mechanism.

Table 19 compares the performance of our model and JMAN. We report the performance in terms of accuracy, precision, recall, and F-measure.[10] These are the same metrics used to evaluate JMAN. Briefly, accuracy (Acc) is defined as the percentage of the correctly generated labels over the total number of labels presented. The precision (P) is the percentage of the correctly generated labels over all the generated labels, and recall (R) is the percentage of the correctly generated labels over all the actual labels. The F-score is the harmonic mean of P and R.

On the Bibsonomy and CiteULike-a datasets, JMAN performs better in terms of F-score and accuracy, with roughly 9% and 5%, respectively. However, it is worth noting that our model is completely unsupervised and was able to generate the labels from scratch (making the best use of what it learned from the training set), whereas JMAN is a supervised model which generates labels from a set of predefined tags (multi-label classification).

[10] Because we configured our system to generate 6 labels only for each document, we set the true labels up to 6 for calculating the metrics.

Table 16
A sample document from the Bibsonomy dataset (testing set) and the resulting labels using GA_{L2} method. The CP_{T1} (CP_{T2}) is the contribution percentage (CP) of our model's top two dominant topics. This is followed by the set of labels generated by our model, and the list of labels by human annotators.

Document	Tile: The diversity of epilepsy in adults with severe developmental disabilities age at seizure onset and other prognostic factors. Abstract: There is usually a causal relationship between epilepsy and mental retardation when they coexist. The pathogenetic period of the underlying brain disorder and the time of seizure onset may, however, be widely separated. In 63 institutionalized mentally retarded epilepsy patients, 41 had seizure onset prior to the age of 2, 30 between the age of 2 and 20, and as many as 29 after the age of 20. Whereas uncontrolled epilepsy and cerebral palsy were frequently present in the group of early onset ...
CP_{T1} 0.9682	Keywords (top dominant topic) cerebral_palsy, risk, child, human, factor, cerebral, age, female, male, group
CP_{T2} 0.0102	Keywords (second top dominant topic) education, learning, social, technology, educational, development, information, school, teaching, science
Coherence −1.8863	R_l labels (generated by our model) male, human, cerebral, cerebral_palsy, age, factor
Coherence −3.2494	Labels by humans (all) of, down, male, human, adolescent, adult, age, aged, cerebral, female, factor, mental, middle, prognosis, syndrome, human_factors, middle_aged, age_factors

Table 17
A sample document from the testing set in the CiteULike-a dataset and the resulting labels using GA_{L2} method.

Document	Tile: Modulation of oscillatory neuronal synchronization by selective visual attention. Abstract: In crowded visual scenes, attention is needed to select relevant stimuli. To study the underlying mechanisms, we recorded neurons in cortical area v4 while macaque monkeys attended to behaviorally relevant stimuli and ignored distracters. Neurons activated by the attended stimulus showed increased gamma frequency 35 to 90 hertz synchronization but reduced low frequency 17 hertz synchronization compared with neurons at nearby v4 sites activated by distracters. Because postsynaptic integration times are short ...
CP_{T1} 0.8614	Keywords (top dominant topic) brain, neuron, neural, activity, quantum, visual, response, human, stimulus, signal
CP_{T2} 0.0261	Keywords (second top dominant topic) model, method, algorithm, data, problem, approach, set, analysis, learning, image
Coherence −2.8921	R_l labels (generated by our model) brain, stimulus, neural, neuron, activity, visual
Coherence −3.4607	Labels by humans (all) oscillation, brain, oscillation, monkey, cog_neuro, perception, quals, attention, synchronization, time_oriented_science, cogsci, synchrony, operational_neuro, neuro, gamma, vision, lfp

Table 18
Analysis of the documents in the testing set where $CP_{T1} \leq 0.5$.

			Average		# Matched
Dataset	# Cases	Percentage	CP_{T1}	CP_{T2}	tags
Bibsonomy	255	21.07%	0.4328	0.3257	1.339
CiteULike-a	717	53.83%	0.3973	0.2619	0.975
CiteULike-t	1359	56.51%	0.3870	0.2525	0.628

Table 19
Comparison results of our proposed system and JMAN (Dong et al., 2020) on three social annotation datasets in terms of accuracy (Acc), precision (P), recall (R), and F-score (F_1). All experiments use the same training (90%) and testing (10%) portions. The performance for JMAN are as reported by its authors.

	Our system				JMAN			
Dataset	Acc	P	R	F_1	Acc	P	R	F_1
Bibsonomy	20.2	28.5	30.1	29.1	25.1	58.8	28.6	38.5
CiteULike-a	10.2	16.2	17.1	16.5	13.9	47.4	17.0	25.0
CiteULike-t	6.6	10.5	12.2	11.0	14.5	40.9	17.8	24.8

7.4. Performance comparison of our model on all datasets

We can now discuss our model's performance on the Arabic and the English datasets. When compared to the Arabic datasets, the English datasets produced better results in terms of coherence (Tables 13 and 15). However, the Arabic datasets performed significantly better if we consider the similarity measures. There is a simple reason. For the Arabic, we are comparing two subsets drawn from a single pool; while in English, it is a comparison between two sets drawn from two different pools.

For the Arabic dataset, the LDA created a single set of tags. Our model picks a subset of tags out of this set, while the crowdsource annotators picked their own set of tags. Therefore, the comparison is fair. However, in the instance of the English dataset, the first pool of tags was generated by the LDA, from which our model picked a subset. The second pool is the list of tags by human annotators. Often, pools are dissimilar, as we saw in Table 17, which results in poor similarity.

The low similarity in the English dataset does not imply that the tags are unsuitable for documents, but it does demonstrate that they are not always identical to user-defined tags (see Tables 16–17).

8. Conclusion and future works

One of the most significant impediments of Arabic NLP is the lack of annotated data for research purposes. This study is the first step towards enhancing the availability of annotated corpora. In this work, we propose an unsupervised model that automates multi-label annotation without human intervention. The model relies on LDA and genetic algorithms. We evaluated the model on three large corpora in Arabic. We experimented with hyper-tuning the parameters using two different training methods: variational Bayes (VB) and Gibbs sampling (GS). Experimentally, the GS model outperformed the VB model. Our labeling performs well when the corpus does not suffer from imbalanced categories. Two corpora had the class imbalance problem, where the number of documents between categories varied wildly. We solved this problem by the partitioning approach. In the end, we compared our annotation model against human annotation using crowdsourcing services. We showed it was competitive based on the coherence score, Word2vec, and cosine similarity. Even though our model was originally devised for the Arabic text in particular, it is language neutral. We tested our hypothesis on three datasets in the English language achieving competitive results when compared against a supervised deep learning based approach.

Regarding future research opportunities, we aim to investigate other optimization algorithms in association with the present model. The GA is a population-based metaheuristics algorithm that provides a

sufficiently good solution for the optimization problem. The memetic algorithm (Moscato & Cotta, 2003) extends the genetic algorithm that focuses on local search to obtain an approximate solution. We plan to explore combining the memetic algorithm with a local search approach and seed words to improve the quality and diversity of the labels generated. This may help with the problem of hidden topics that LDA failed to discover. Furthermore, a dynamic set solution may produce a more satisfactory result than using a fixed set of relevant labels. As fitness function plays a critical role in evaluating the solution domain, we will investigate employing different embedding methods as a fitness function to improve the quality of the generated labels.

CRediT authorship contribution statement

Huda A. Almuzaini: Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing – original draft. **Aqil M. Azmi:** Conception and design of study, Analysis and/or interpretation of data, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to express their gratitude and sincere appreciation to Muhamed Alkaoud, for his support in using Amazon Mturk. We would like to thank the Deanship of Scientific Research at King Saud University, Saudi Arabia for funding and supporting this research through DSR Graduate Students Research Support initiative.

References

Alhawarat, M., & Hegazi, M. (2018). Revisiting k-means and topic modeling, a comparison study to cluster arabic documents. *IEEE Access*, *6*, 42740–42749.

Almuzaini, H. A., & Azmi, A. M. (2020). Impact of stemming and word embedding on deep learning-based Arabic text categorization. *IEEE Access*, *8*, 127913–127928.

Alzanin, S. M., & Azmi, A. M. (2019). Rumor detection in Arabic tweets using semi-supervised and unsupervised expectation–maximization. *Knowledge-Based Systems*, *185*, 104945:1–104945:9.

Asuncion, A., Welling, M., Smyth, P., & Teh, Y. W. (2012). On smoothing and inference for topic models. arXiv preprint arXiv:1205.2662.

Awasare, V. K., & Gupta, S. (2017). Classification of imbalanced datasets using partition method and support vector machine. In *Second international conference on electrical, computer and communication technologies* (pp. 1–7).

Ayadi, R., Maraoui, M., & Zrigui, M. (2014). Latent topic model for indexing Arabic documents. *International Journal of Information Retrieval Research (IJIRR)*, *4*(2), 57–72.

Ayadi, R., Maraoui, M., & Zrigui, M. (2015). LDA and LSI as a dimensionality reduction method in Arabic document classification. In *International conference on information and software technologies* (pp. 491–502). Springer.

Basu, S., Banerjee, A., & Mooney, R. (2002). Semi-supervised clustering by seeding. In *Proceedings of 19th international conference on machine learning (ICML-2002)* (pp. 19–26).

Benz, D., Hotho, A., Jäschke, R., Krause, B., Mitzlaff, F., Schmitz, C., & Stumme, G. (2010). The social bookmark and publication management system bibsonomy. *The VLDB Journal*, *19*(6), 849–875.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*(4–5), 993–1022. http://dx.doi.org/10.1016/b978-0-12-411519-4.00006-9.

Brahmi, A., Ech-Cherif, A., & Benyettou, A. (2012). Arabic texts analysis for topic modeling evaluation. *Information Retrieval*, *15*(1), 33–53.

Burkhardt, S., & Kramer, S. (2019). A survey of multi-label topic models. *ACM SIGKDD Explorations Newsletter*, *21*(2), 61–79.

Cai, L., Song, Y., Liu, T., & Zhang, K. (2020). A hybrid BERT model that incorporates label semantics via adjustive attention for multi-label text classification. *IEEE Access*, *8*, 152183–152192.

Canini, K., Shi, L., & Griffiths, T. (2009). Online inference of topics with latent Dirichlet allocation. In *Artificial intelligence and statistics* (pp. 65–72).

Chouigui, A., Khiroun, O. B., & Elayeb, B. (2018). ANT corpus: An Arabic news text collection for textual classification. In *Proceedings of IEEE/ACS international conference on computer systems and applications* (pp. 135–142).

Chu, Z., Stratos, K., & Gimpel, K. (2021). NatCat: Weakly supervised text classification with naturally annotated resource. In *The third conference on automated knowledge base construction*.

Dietz, L., Bickel, S., & Scheffer, T. (2007). Unsupervised prediction of citation influences. In *Proceedings of the 24th international conference on machine learning* (pp. 233–240).

Dong, H., Wang, W., & Frans, C. (2017). Deriving dynamic knowledge from academic social tagging data: a novel research direction. In *iConference 2017 proceedings*. iSchools.

Dong, H., Wang, W., Huang, K., & Coenen, F. (2020). Automated social text annotation with joint multilabel attention networks. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(5), 2224–2238.

El-Alami, F.-Z., El Mahdaouy, A., El Alaoui, S. O., & En-Nahnahi, N. (2020). A deep autoencoder-based representation for Arabic text categorization. *Journal of Information and Communication Technology*, *19*(3), 381–398.

El Bazi, I., & Laachfoubi, N. (2018). Arabic named entity recognition using topic modeling. *International Journal of Intelligent Engineering & Systems*, *11*(1), 229–238.

Elnagar, A., Al-Debsi, R., & Einea, O. (2020). Arabic text classification using deep learning models. *Information Processing & Management*, *57*(1), Article 102121.

Fujino, A., Ueda, N., & Saito, K. (2005). A hybrid generative/discriminative approach to semi-supervised classifier design. In *Proceedings of the 20th national conference on artificial intelligence (AAAI-05)* (pp. 764–769).

Gao, X., He, Y., Zhang, M., Diao, X., Jing, X., Ren, B., & Ji, W. (2021). A multiclass classification using one-versus-all approach with the differential partition sampling ensemble. *Engineering Applications of Artificial Intelligence*, *97*, 104034:1–104034:11.

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences of the USA*, *101*, 5228–5235.

Guellil, I., Adeel, A., Azouaou, F., & Hussain, A. (2018). Sentialg: Automated corpus annotation for algerian sentiment analysis. In *International conference on brain inspired cognitive systems* (pp. 557–567). Springer.

Guellil, I., Azouaou, F., & Chiclana, F. (2020). ArAutoSenti: automatic annotation and new tendencies for sentiment classification of arabic messages. *Social Network Analysis and Mining*, *10*(1), 75.

He, D., Wang, M., Khattak, A. M., Zhang, L., & Gao, W. (2019). Automatic labeling of topic models using graph-based ranking. *IEEE Access*, *7*, 131593–131608.

Hoffman, M., Bach, F. R., & Blei, D. M. (2010). Online learning for latent Dirichlet allocation. In *Advances in neural information processing systems* (pp. 856–864).

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval* (pp. 50–57).

Imane, G., Kareem, D., & Faical, A. (2019). A set of parameters for automatically annotating a sentiment Arabic corpus. *International Journal of Web Information Systems*, *15*(5), 594–615.

Jelodar, H., Wang, Y., Rabbani, M., & Ayobi, S. (2019). Natural language processing via LDA topic model in recommendation systems. arXiv preprint arXiv:1909.09551.

Jiang, H., Miao, Z., Lin, Y., Wang, C., Ni, M., Gao, J., Lu, J., & Shi, G. (2021). Financial news annotation by weakly-supervised hierarchical multi-label learning. In *Proceedings of the second workshop on financial technology and natural language processing* (pp. 1–7).

Jo, T. (2006). *The implementation of dynamic document organization using the integration of text clustering and text categorization* (Ph.D. thesis), University of Ottawa (Canada).

Johnsen, J. W., & Franke, K. (2019). The impact of preprocessing in natural language for open source intelligence and criminal investigation. In *IEEE international conference on big data* (pp. 4248–4254).

Kelaiaia, A., & Merouani, H. (2013). Clustering with probabilistic topic models on Arabic texts: A comparative study of LDA and k-means. In *Modeling approaches and algorithms for advanced computer applications* (pp. 65–74).

Kelaiaia, A., & Merouani, H. F. (2016). Clustering with probabilistic topic models on Arabic texts: A comparative study of LDA and K-means. *International Arab Journal of Information Technology*, *13*(2), 332–338.

Khoja, S., & Garside, R. (1999). *Stemming Arabic text*. Lancaster, UK: Computing Department, Lancaster University.

Kulkarni, A., Mandhane, M., Likhitkar, M., Kshirsagar, G., Jagdale, J., & Joshi, R. (2021). Experimental evaluation of deep learning models for marathi text classification. arXiv preprint arXiv:2101.04899.

Kwaik, K. A., Chatzikyriakidis, S., Dobnik, S., Saad, M., & Johansson, R. (2020). An Arabic tweets sentiment analysis dataset (atsad) using distant supervision and self training. In *Proceedings of the 4th workshop on open-source arabic corpora and processing tools, with a shared task on offensive language detection* (pp. 1–8).

Lane, J. (2021). The 10 most spoken languages in the world. URL: https://www.babbel.com/en/magazine/the-10-most-spoken-languages-in-the-world.

Mifrah, S., & Benlahmar, E. (2020). Topic modeling coherence: A comparative study between LDA and NMF models using COVID'19 corpus. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*, 5756–5761.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. In *First international conference on learning representations (ICLR 2013)*.

Mimno, D., Wallach, H. M., Naradowsky, J., Smith, D. A., & McCallum, A. (2009). Polylingual topic models. In *Proceedings of the 2009 conference on empirical methods in natural language processing (EMNLP-2009)* (pp. 880–889).

Mimno, D., Wallach, H. M., Talley, E., Leenders, M., & McCallum, A. (2011). Optimizing semantic coherence in topic models. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP 2011)* (pp. 262–272).

Molavi, M., Tavakoli, M., & Kismihók, G. (2020). Extracting topics from open educational resources. In *European conference on technology enhanced learning* (pp. 455–460).

Moscato, P., & Cotta, C. (2003). A gentle introduction to memetic algorithms. In *Handbook of metaheuristics* (pp. 105–144). Springer.

Naili, M., Chaibi, A., & Ghézala, H. (2017). Arabic topic identification based on empirical studies of topic models. HAL URL: https://hal.archives-ouvertes.fr/hal-01444574v1.

Ordun, C., Purushotham, S., & Raff, E. (2020). Exploratory analysis of covid-19 tweets using topic modeling, umap, and digraphs. arXiv preprint arXiv:2005.03082.

Papadimitriou, C. H., Raghavan, P., Tamaki, H., & Vempala, S. (2000). Latent semantic indexing: A probabilistic analysis. *Journal of Computer and System Sciences*, *61*(2), 217–235.

Papanikolaou, Y., Foulds, J. R., Rubin, T. N., & Tsoumakas, G. (2017). Dense distributions from sparse samples: Improved Gibbs sampling parameter estimators for LDA. *Journal of Machine Learning Research*, *18*(1), 2058–2115.

Patibandla, R. L., & Veeranjaneyulu, N. (2018). Survey on clustering algorithms for unstructured data. In *Intelligent engineering informatics* (pp. 421–429).

Pavlinek, M., & Podgorelec, V. (2017). Text classification method based on self-training and LDA topic models. *Expert Systems with Applications*, *80*, 83–93.

Perez, F., Lebret, R., & Aberer, K. (2018). Weakly supervised active learning with cluster annotation. arXiv preprint arXiv:1812.11780.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). *Language models are unsupervised multitask learners*: *Technical Report Report 8*, OpenAI Blog.

Röder, M., Both, A., & Hinneburg, A. (2015). Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on web search and data mining* (pp. 399–408).

Rosen-Zvi, M., Griffiths, T., Steyvers, M., & Smyth, P. (2004). The author-topic model for authors and documents. In *Proceedings of the 20th conference on uncertainty in artificial intelligence* (pp. 487—494).

Saad, M. K., & Ashour, W. M. (2010). OSAC: Open source Arabic corpora. In *Proceedings of sixth international symposium on electrical and electronics engineering and computer science (EEECS'10)* (pp. 118–123).

Schofield, A., Magnusson, M., & Mimno, D. (2017). Pulling out the stops: Rethinking stopword removal for topic models. In *Proceedings of the 15th conference of the european chapter of the association for computational linguistics: (Vol 2, Short Papers)* (pp. 432–436).

Settles, B. (2009). *Active learning literature survey*: *Technical Report Computer Sciences Technical Report 1648*, University of Wisconsin-Madison.

Smola, A., & Narayanamurthy, S. (2010). An architecture for parallel topic models. *Proceedings of the VLDB Endowment*, *3*(1–2), 703–710.

Stevens, K., Kegelmeyer, P., Andrzejewski, D., & Buttler, D. (2012). Exploring topic coherence over many models and many topics. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational NLP* (pp. 952–961).

Taghva, K., Elkhoury, R., & Coombs, J. (2005). Arabic stemming without a root dictionary. In *International conference on information technology: Coding and computing (ITCC'05)-Volume II, Vol. 1* (pp. 152–157).

Taware, R., Varat, S., Salunke, G., Gawande, C., Kale, G., Khengare, R., & Joshi, R. (2021). ShufText: A simple black box approach to evaluate the fragility of text classification models. arXiv preprint arXiv:2102.00238.

Wan, X., & Wang, T. (2016). Automatic labeling of topic models using text summaries. In *Proceedings of the 54th annual meeting of the association for computational linguistics (Volume 1: Long Papers)* (pp. 2297–2305).

Wang, H., Chen, B., & Li, W.-J. (2013). Collaborative topic regression with social regularization for tag recommendation. In *Twenty-third international joint conference on artificial intelligence*.

Wang, M., & Mengoni, P. (2021). How pandemic spread in news: Text analysis using topic model. arXiv preprint arXiv:2102.04205.

Wang, F., & Zhang, C. (2007). Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, *20*(1), 55–67.

Xiao, Y., Li, Y., Yuan, J., Guo, S., Xiao, Y., & Li, Z. (2021). History-based attention in Seq2Seq model for multi-label text classification. *Knowledge-Based Systems*, *224*, Article 107094.

Xing, Y., Yu, G., Domeniconi, C., Wang, J., & Zhang, Z. (2018). Multi-label co-training. In *IJCAI'18*, *Proceedings of the 27th international joint conference on artificial intelligence* (pp. 2882–2888).

Yao, L., Mimno, D., & McCallum, A. (2009). Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 937–946).

Zha, D., & Li, C. (2019). Multi-label dataless text classification with topic modeling. *Knowledge and Information Systems*, *61*(1), 137–160.

Zhan, W., & Zhang, M.-L. (2017). Inductive semi-supervised multi-label learning with co-training. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1305–1314).

Zhang, H., Cai, Y., Zhu, B., Zheng, C., Yang, K., Wong, R. C.-W., & Li, Q. (2020). Incorporating concept information into term weighting schemes for topic models. In *International conference on database systems for advanced applications* (pp. 227–244).

Zhang, M.-L., & Zhou, Z.-H. (2013). A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, *26*(8), 1819–1837.

Zhu, Y., Jing, L., & Yu, J. (2009). New labeling strategy for semi-supervised document categorization. In *International conference on knowledge science, engineering and management* (pp. 134–145). Springer.

Zrigui, M., Ayadi, R., Mars, M., & Maraoui, M. (2012). Arabic text classification framework based on latent dirichlet allocation. *Journal of Computing and Information Technology*, *20*(2), 125–140.