A Hybrid Model for Documents Representation

Dina Mohamed[1], Ayman El-Kilany[2], Hoda M. O. Mokhtar[3]
Faculty of Computers and Artificial Intelligence, Cairo University, Giza, Egypt

***Abstract*—Text representation is a critical issue for exploring the insights behind the text. Many models have been developed to represent the text in defined forms such as numeric vectors where it would be easy to calculate the similarity between the documents using the well-known distance measures. In this paper, we aim to build a model to represent text semantically either in one document or multiple documents using a combination of hierarchical Latent Dirichlet Allocation (hLDA), Word2vec, and Isolation Forest models. The proposed model aims to learn a vector for each document using the relationship between its words' vectors and the hierarchy of topics generated using the hierarchical Latent Dirichlet Allocation model. Then, the isolation forest model is used to represent multiple documents in one representation as one profile to facilitate finding similar documents to the profile. The proposed text representation model outperforms the traditional text representation models when applied to represent scientific papers before performing content-based scientific papers recommendation for researchers.**

***Keywords*—Document representation; latent dirichlet allocation; hierarchical latent dirichlet allocation; Word2vec; Isolation Forest**

I. Introduction

With the rapid growth in the volume of text data and documents over the internet from social media, news articles, scientific papers, and surveys; it becomes a critical issue to find an effective model to represent the text features in the documents before using them in text mining, information retrieval, and recommendation systems. Bag–Of–Words (BOW) model is one of the most popular models for representing documents [1]. It relies on the frequencies of the words within the documents for building the document vector with a fixed length, while it fails to capture the word importance through a collection of documents. Also, BOW doesn't perform well when representing a huge number of documents due to the increasing number of words; which in turn causes a sparse document vector. Term Frequency Inverse Document Frequency (TF-IDF) model has been applied for representing the document as a numeric vector [2, 3]. It measures the importance of the words in a collection of documents, accordingly, the frequent words that appear in many documents such as (if, what, the ...) take a low weight and the rare words that focus on the document purpose take a high weight. TF- IDF model is used in many types of research for information and queries retrieval [4, 5], but it fails to capture the semantics behind words and neglects the order of the words in the documents. With the vital need to capture the semantics of the words to build an effective model for document representation, topic modeling techniques have been proposed for representing documents. Latent Dirichlet Allocation (LDA) is a well-known topic modeling technique [6], in which the documents are represented as a distribution over a set of latent topics that are generated from a set of documents' words. For deeper representation; hierarchical Latent Dirichlet Allocation (hLDA) which is an extension of the LDA model was developed to learn the hierarchical structure for topics from a collection of documents[7]. Recently, word embedding techniques that represent words and documents (e.g. word2vec and doc2vec) as a numeric vector using neural networks were introduced. Word2vec model builds a representation for words as dense vectors depending on the word's context [8].

In this paper, we aim to build a document representation model that exploits the advantages of hierarchical topic modeling (hLDA) and the word2vec model to represent the document as a hierarchical tree of topics. The proposed model starts with building the hierarchical tree of topics with *n* levels from a corpus of a collection of documents. The resulting topics are transformed into numeric vectors using words' vectors resulting from the word2vec model. Then, each document is represented as a hierarchy of topics using the similarity scores between document vector and topic vectors. Once we have a document representation, an isolation forest model is built to represent multiple documents altogether in one representation as a single profile. The resulting profile model is then used to find similar documents to a multiple set of documents that were joint through the profile. The main contributions of this paper are centered around two main points. First, the paper proposes a hierarchy-based representation for a text document that integrates the hierarchical topic modeling and the word2vec model. Second, the paper proposes a unified representation for multiple documents altogether as one profile using the isolation forest model.

We argue that the conjunction between the hierarchical structure of topics and the word vectors would allow a better understanding of the document semantics and consequently a better recommendation. Also, we argue that grouping multiple documents as one profile using the isolation forest model would allow a better representation of the whole set of documents rather than considering each of them individually. To prove our arguments, experiments were conducted using a dataset for scientific papers that contains a set of research papers described by their titles and abstracts, and a set of researchers with their preferred papers. The proposed model was used to represent each paper individually. Then, a subset of each researcher's preferred papers was aggregated as the researcher profile using their representations. Researchers' profiles were used to recommend papers to the researcher where the recommendation results outperform other semantic-based representation models like LDA with word2vec combination [9], and concept-based representation [10].

The paper is organized as follows. In Section II; background about the topic modeling techniques, word2vec model, isolation forest model, and recommendation systems is introduced. Section III discusses the related work while Section IV explains the proposed model. Sections V and VI present the performance evaluation and discussion, respectively. Section VII concludes the paper.

II. Background

A. Topic Modeling

Topic Modeling is an unsupervised machine learning technique that identifies the latent topics behind the text corpus of a set of documents. Latent Dirichlet Allocation (LDA) [6] is one of the main topic modeling methods. It represents the document as a distribution over a mixture of topics with a certain probability, while each topic is represented as a distribution over a mixture of words. Fig. 1 shows an example for the LDA documents representation, at first, the number of topics K is defined, and then each word in the documents is randomly assigned to a topic. The assignment process is repeated until all words are assigned to their correct topics. Finally, the documents are represented as a distribution over a mixture of topics with a certain probability.

The LDA graphical representation model is shown in Fig. 2. It shows two boxes "plates"; the outer one is for representing the collection of documents and the inner is for the words in the document associated with the topics where *M* denotes the documents number in the collection, *N* is the number of words in specific documents, *w* is a specific word in the documents, *z* is the topic assigned to the word, *θ* is the topic distribution for the document, while α and β are the parameters of the Dirichlet, *α* for a document- topic distribution and *β* for word-topic distribution. In this graphical model, the gray node represents the observed variables as the only observed variable is the word *w*, where the other nodes are latent. The arrows are for representing the dependencies between the variables.

The LDA was later extended for discovering the complex structure of the topics. The hierarchical Latent Dirichlet Allocation (hLDA) model is one of the LDA model extensions while the topics are presented in a hierarchical structure [7] using the nested Chinese restaurant process (CRP) [11]. The hLDA model is used to define the topics for a collection of documents and these topics are organized in a hierarchical structure where more general topics appear near the top levels in the hierarchy and more specific topics appear near the leaves. Given a collection of documents and L levels of a hierarchical tree where each node in the tree belongs to a topic, the document is represented as a path from the root node of the tree to the leaf node. Then, a vector of topic proportions θ is identified in addition to the words of each topic.

B. Word2vec

Word2vec is one of the word embedding techniques that aim to map words to numeric vectors to capture the syntactic and semantics regularities behind the words [12]. It helps in natural language processing tasks [13], as it represents the words that have a similar meaning, with a similar representation and similar position in vector so it becomes applicable in finding the relation between the words. The word vectors help answer analogy questions of the form (*a: b as c: d*) where *d* is unknown. For example, the left panel of Fig. 3 illustrates that the relation (man: woman) is as (uncle: aunt) and (king: queen), as it discovers the gender relation between the words [14]. Also, it discovers the singular/plural relations between the words; as illustrated in the right panel of Fig. 3. The words vectors discover the relation between the words through applying algebraic operation between the words vectors for example when subtracting words vectors; vector ("king") – vector ("man") + vector ("woman"), its result is closest to the vector ("queen").

Word2vec learns the distribution of words using neural networks. There are two word2vec models, the Continuous Bag of Words (CBOW) model and the Skip Gram model. CBOW model predicts the current word using its surrounding words in a specific window size. On the other hand, the Skip-gram model predicts the surrounding words using the current word; Fig. 4 illustrates the architecture of the CBOW model and Skip-gram model.

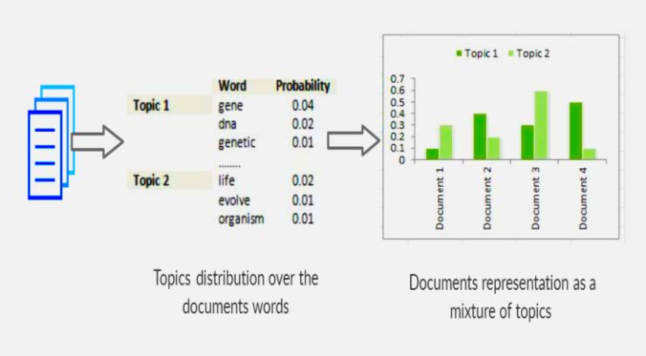

Fig. 1. The LDA Representation for the Documents.

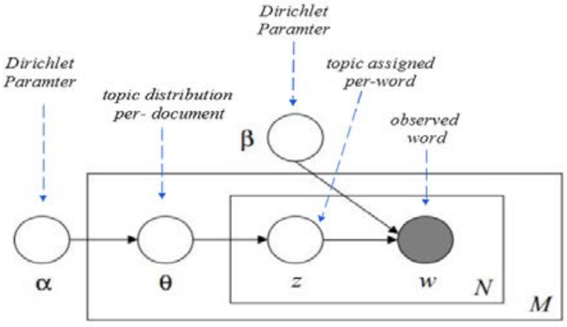

Fig. 2. LDA Graphical Model Representation [6].

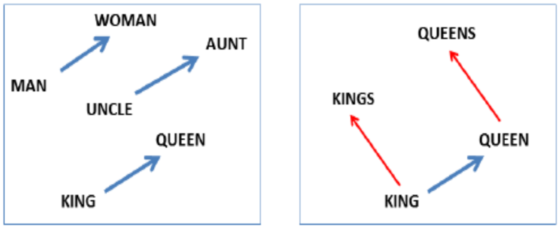

Fig. 3. The Panel shows the different Projection Relation between the Words [14].

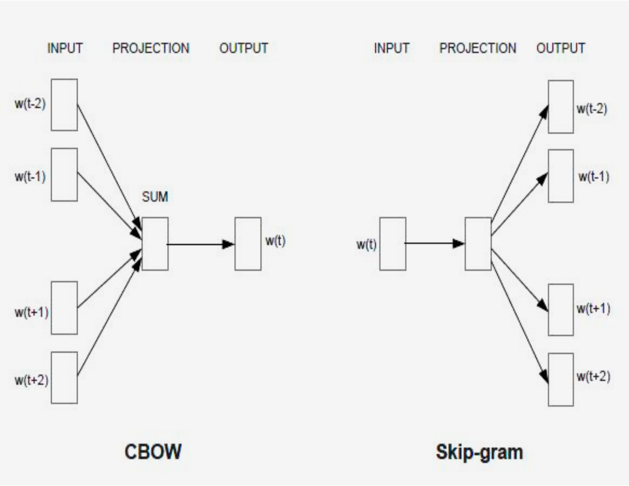

Fig. 4. The CBOW and Skip-Gram Word2vec Models [12].

C. Isolation Forest Model

Isolation forest is a decision-tree algorithm for unsupervised anomaly detection in high-dimensional datasets [15, 16]. In addition, it can be applied for semi-supervised anomaly detection where the dataset has normal instances only. The isolation forest model argues that the normal points are harder to be isolated than the anomalies points. To isolate any point in the data, the isolation forest model randomly selects a feature and a splitting value that lies between the minimum and maximum values for the feature to partition the data and assigns points to each split. Then, the partitioning process is repeated until all the points are isolated. While the normal instances need more partitions to be isolated, the anomaly instances require a low number of partitions. For example, suppose we have a two-dimensional dataset X={x_0, x_i,….x_n}with *n* number of points. Fig. 5 shows that the isolation of the point x_i which is considered to be a normal point would require more partitions, while the anomaly point x_0 would require few numbers of partitions to be isolated.

In the isolation forest model, the repeatedly partitioning for the data points isolation can be represented in a tree structure (isolation tree). The anomalies would have shorter path lengths in the tree since they are easier to isolate than the normal instances. The isolation tree is constructed through repeatedly partitioning for the data points until all points are isolated or the tree reaches the determined height. The anomaly score is calculated using the average of all path lengths for the data points along with all features of the isolation tree where each path length is obtained by counting the number of edges from the root node to the termination node. The anomaly score for a point *x* in a data sample of size n is predicted by using Equation (1) [16].

$$s(x,n) = 2^{\frac{E(h(x))}{c(n)}} \qquad (1)$$

Where *E (h(x))* is the average length of the path *h(x)* across all isolation trees for the point *x*, and *c(n)* is the average length for isolation trees for the given points, which can be calculated as the average path length of the Binary Search Tree (BST) [17]. Fig. 6 illustrates an example of an anomaly detection process using the isolation forest model.

The semi-supervised anomaly detection with the isolation forest model is processed in two stages; the training stage and the testing stage. The isolation trees are built in the training stage while in the testing stage the anomaly score is obtained for each test instance by passing it through the isolation trees to determine the path length in each tree before applying Equation (1).

D. Recommendation Systems

The recommendation systems are used for suggesting items to users according to their interests, as it tries to predict the most appropriate items for the user's needs. There are three main methods for recommending items, Content-based filtering (CBF), Collaborative filtering (CF), and hybrid [19]. The content-based recommendation method analyzes the items' content to represent the interests of the users [20] and recommends similar items for the user's interests. CBF has wildly used for recommending documents and news articles as it depending on analyzing the content of the items and build user profiles. On the other hand, CF recommends the items depending on users who have similar interests with the target user. It predicts the user rates for the unseen items using the rates of his / her correlated users who give a similar rate for the common items. The Hybrid method combines both methods to recommend the items to users.

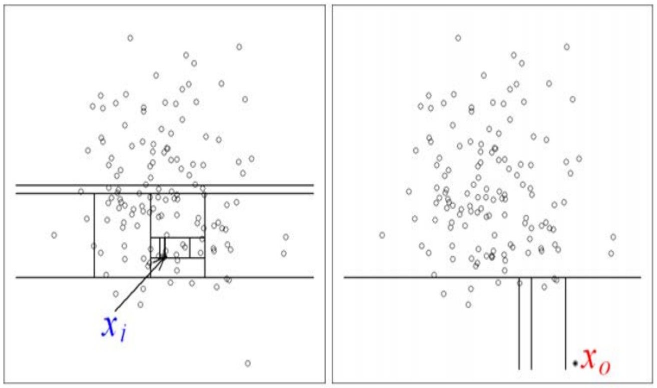

Fig. 5. Isolation for Anomalies Points [15].

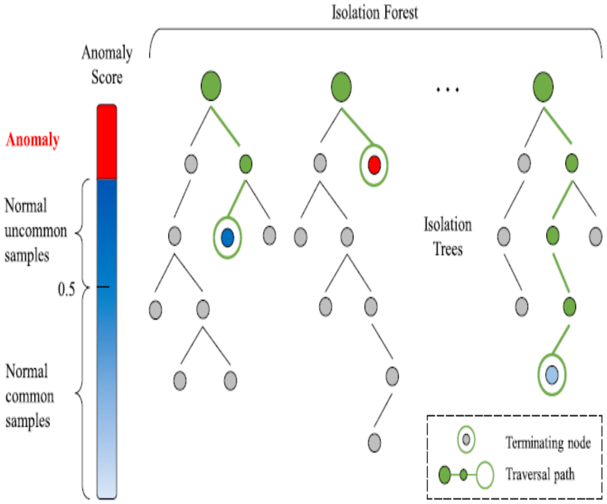

Fig. 6. Anomaly Detection using Isolation Forest [18].

III. Related Work

Text representation has been studied extensively in the literature, where different methods for representing the text through a variety of domains were proposed. The authors in [4] applied TF-IDF to represent the user's query and documents, to return the relevant documents to the user query. Also in [21] the authors designed a framework for document classification and measuring the similarity between the documents using the TF-IDF method and k-Nearest Neighbor (kNN) algorithm. Another semantic-based method of text representation is topic modeling techniques. For example, the authors in [22] utilized the LDA model to extract the concepts from the source code in software datasets to perform concepts analysis and visualization for software code and to find the similarity between the software programs. In [23] the author builds a topic model for contrastive opinion modeling to find opinions depending on a given topic. The authors in [24] introduced an online course recommendation system that recommends courses to students in a college while combining the content-based recommendation and LDA model, where the LDA model was used to discover the topics from the contents of the course and representing courses as a distribution over the topics. In [25] the authors developed the collaborative topic regression (CTR) model to recommend scientific articles to users by combing the collaborative filtering approach with probabilistic topic modeling that analyzes the latent topics in the content of the articles.

The researchers in [26] applied the hierarchical Latent Dirichlet Allocation (hLDA) model to analyze software program text and generate a feature tree to understand the software system, while the tree includes two hierarchies; feature hierarchy and file structure hierarchy. As the feature hierarchy for displaying the features from abstract to detailed levels and the file structure hierarchy for displaying the classes from whole to part. Also, the authors in [27] used the hLDA model to represent the legal documents as a hierarchy to measures the similarity between them before clustering them.

Recently, the word2vec model was widely used for representing the documents. In [28], the authors applied a multi labels classification for news articles where the word2vec model was applied to build a vector for words in news articles to capture the similarity between the words and then use those words vectors as a classification feature. The authors in [10] proposed a model for representing academic articles to recommend them to researchers. This method generates a set of concepts by clustering the word vectors that are learned from the word2vec model where the words with the same semantic meaning will be grouped in one concept. Then, those concepts are used to represent the articles as a distribution over the concepts.

Another research direction that applies a combination of multiple representation models to build an effective representation method to capture the semantics behind the text., For example, the authors in [9] developed a document representation model that combines topics of the LDA model and word vectors of the word2vec model.

IV. Proposed Model

The proposed model aims to build a document representation model that captures the semantics of the text in the documents. The proposed method starts by extracting the latent topics and the hierarchical relation between those topics using the hierarchical Latent Dirichlet Allocation model (hLDA) from the documents corpus. The latent topics are enhanced with the words' vectors generated from the Word2vec model. The document is represented as a feature vector that refers to how the document relates to each topic in the topics hierarchy. Then, the document representation is utilized to represent multiple documents as one profile. Fig. 7 shows the graphical representation model for the proposed model. Our model builds the documents' vectors through main four phases; text processing phase, topics hierarchy construction phase, document representation phase, and profile construction phase. The four phases are described in more detail as follows.

A. Text Processing Phase

In the text processing phase, documents are cleaned for the next phases. Each document is tokenized into words before removing the stop words. Stemming and lemmatization are also performed for each word and they are prepared for the hierarchical topic modeling process. The documents corpus after cleaning are used to train the word2vec model to produce the words' vectors for each word in the documents. The initial document representation vector is calculated as the average of its words' vectors using Equation (2) [9], where each document *d* contains *n* number of words *w* and *v (w)* denotes the word vector.

$$v(d) = \frac{\sum_{i=1}^{n} v(w_i)}{n} \qquad (2)$$

B. Topics Hierarchy Construction Phase

In this phase, the hLDA model uses the documents' corpus returned from the text processing phase to build a hierarchy of topics where each topic is represented by a set of words and their probability for being related to the topic. While the more abstract topics appear near the root node of the hierarchy tree and the more specific topics in the leaves. Fig. 8 illustrates an example for part of the topic hierarchy generated from a collection of research papers. For topics representation in the hierarchy, we used the words' vectors that are generated from the word2vec model to construct a vector for each topic. The topic vector is calculated using Equation (3) [9], where each topic t is represented only with the top m words that have the highest probability associated with the topic. In addition, $\mathrm{p(w_i)}$ refers to the probability of $\mathrm{w_i}$ in topic t and $\mathrm{v(w_i)}$ is the vector of $\mathrm{w_i}$ that is generated by the word2vec model.

$$v(t) = \sum_{i=1}^{m} p(w_i) * v(w_i) \qquad (3)$$

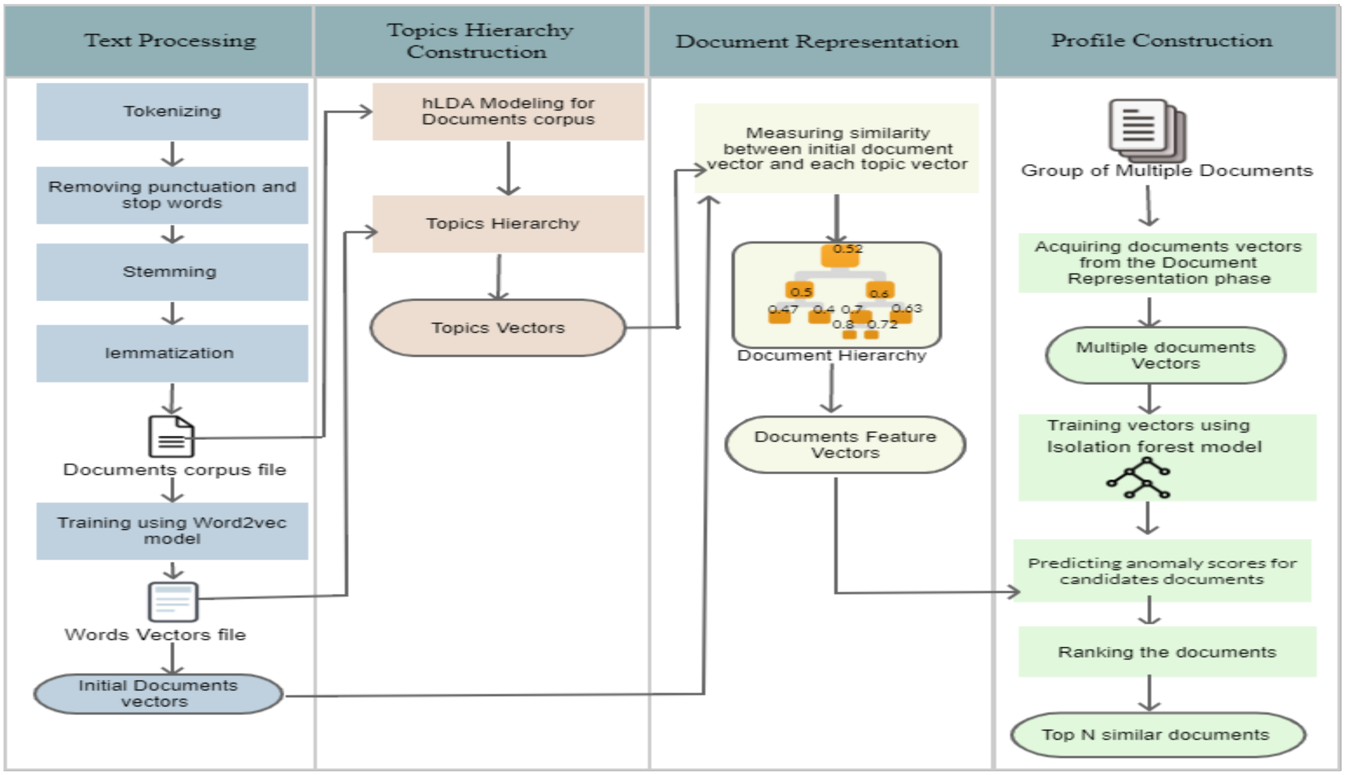
Text Processing
Topics Hierarchy Construction
Document Representation
Profile Construction
Tokenizing
Removing punctuation and stop words
Stemming
lemmatization
Documents corpus file
Training using Word2vec model
Words Vectors file
Initial Documents vectors
hLDA Modeling for Documents corpus
Topics Hierarchy
Topics Vectors
Measuring similarity between initial document vector and each topic vector
0.52
0.5
0.6
0.47
0.4
0.7
0.63
0.8
0.72
Document Hierarchy
Documents Feature Vectors
Group of Multiple Documents
Acquiring documents vectors from the Document Representation phase
Multiple documents Vectors
Training vectors using Isolation forest model
Predicting anomaly scores for candidates documents
Ranking the documents
Top N similar documents

Fig. 7. Graphical Representation of the Proposed Model.

C. Document Representation Phase

This phase uses the initial documents' vectors that were generated in the first phase and the topics vectors of the hierarchy of topics that were generated from the second phase to build the final document representation as a distribution over the hierarchy of topics. In this phase, we build a single feature vector for each document to use in constructing a profile for multi-documents. We calculate the cosine similarity [3] between the initial document vector and each topic vector in the topics hierarchy. The main advantage of using the cosine similarity is that the cosine similarity can measure the similarity between multi-dimensions vectors based on the orientation rather than the magnitude of the vectors, so it is more suitable for the initial documents vectors and the topics vectors which are multi-dimensional vectors. Equation (4) illustrates the calculation of the similarity between the document vector $v(d)$ and the topic vector $v(t)$. Finally, each document is represented as a feature vector of the similarity scores of the topics hierarchy.

$$sim\big(v(d), v(t)\big) = \frac{v(d)*v(t)}{\|v(d)\|.\|v(t)\|} \tag{4}$$

D. Profile Construction Phase

In this phase, we construct a representation for multiple documents where each group of documents is represented as one profile in order to identify the semantics behind those documents. The multiple documents' profile is constructed through the isolation forest model [15] in its semi-supervised setting. The isolation forest model considers the group of documents as a training dataset and predicts the anomaly score for each document in the testing dataset where the anomaly score is obtained as described earlier in the background section. The anomaly score is used to indicate the similarity/dissimilarity between each group of documents in the training data and each document in the testing dataset where the high anomaly score would indicate high dissimilarity. Using the isolation forest model allowed the proposed model to represent the multiple documents as one profile and consequently find the similarity/ dissimilarity between each document in the test dataset and the generated profile using the predicted anomaly score.

V. Performance Evaluation

In order to prove the proposed model's effectiveness, experiments were conducted through a recommendation system where the proposed model was used to represent scientific papers before doing content-based recommendations for the researchers. We used a dataset from CiteUlike [1]; CiteUlike is a site for helping researchers to share scientific papers and finding their preferred papers. This dataset consists of a collection of scientific papers and researchers, while each paper is described by its title and abstract, and each researcher has a list of his/her preferred papers. In the empirical experiments, a random subset of data is selected to measure the performance of the proposed representation model against different representation models. This subset contains approximately 6000 papers and 200 researchers. The proposed model was used to generate vectors' representation for each paper as described in Section IV. First, the text processing phase is performed on the papers. Then, the text generated from the text processing phase is trained using the continuous

[1] www.citulike.org.

bag of words (CBOW) word2vec model through the Gensim[2] library in python with a word vector of size 200. The hLDA model was trained on the paper's text to extract the latent topics and build the topics hierarchy with three levels, which generated 69 topics organized in the hierarchy structure. Fig. 8 shows part of the topics hierarchy that was generated, where each topic is represented by the top 20 words that have the highest probability of being related to the topic. Then, each topic is transformed into its vector representation using the words' vectors generated by the word2vec model. The final vector representation for each document is calculated by getting the similarity between the initial document vector and the topics vectors.

After following the first three phases in the model, a vector for each paper in the dataset is generated with the size of the number of topics in the hierarchy, where each value in the vector represents how the paper is related to the topic. The fourth phase joins each researcher's preferred papers into one profile in order to use it in the recommendation process. The list of the preferred papers for each researcher is divided into two datasets; training and testing, where the training dataset is used for building the profile for the researcher preferences using the isolation forest model [15]. The anomaly score is calculated for each paper in the whole collection of papers to determine the most similar papers to the target profile and how many of them exist in the testing dataset.

The performance evaluation is conducted using the recall evaluation metrics as applied in [29], as the recall function measures the fraction of positive patterns that are correctly recommended [30, 31], while in the dataset we only have the papers that the researchers prefer and there is no information about the papers that aren't preferred by the researchers. The recall is calculated for each researcher, while each researcher has a list of papers that he/she prefers, this list was divided into training and testing datasets. The recommendation system recommends the top N papers for the researcher; N is equal to the length of the testing dataset. The recall [31] is calculated using Equation (5).

$$Recall = \frac{tp}{tp+fn} \tag{5}$$

Whereas *tp* denotes true positive that refers to the number of papers that the researcher prefers from the top N recommend paper and *fn* denotes the false negative that refers to the number of papers that the researcher prefers and not recommended by the recommendation system. The overall recall of the recommendation system is computed by getting the average of all researchers' recall values. While the average recall result is validated with cross-validation technique as the dataset is divided randomly into 5 groups, each time one of the groups takes as a testing dataset and the remaining groups as the training, and the average recall is calculated each time.

We compared our model against the concept-based model [10] and the LDA+Word2vec model [9] given their similarity with the proposed model. Both of concept-based model and the LDA+Word2vec model applied the word2vec model to represent the words. In addition, both combined similar words into different groups in a way or another. The concept-based model generates a set of concepts by clustering the words vectors that are learned from the word2vec model. Then, it uses the generated concepts to represent each document as a distribution over the concepts. On the other hand, the LDA+Word2vec model combines the LDA model and the Word2vec model and acquires the relationship between documents and topics using Euclidean distance. The proposed model, the concept-based model, and the LDA+Word2vec model are used to represent the scientific papers in the recommendation system to recommend papers to researchers depending on their preferred papers. In this experiment, the vector of size 50 was chosen as the number of topics in the LDA+Word2vec model. Also, the number of concepts in the concept-based model was set to 50. Whereas the vector size 50 is chosen based on the experiments that have been conducted for the concept-based model [10] for different sizes of the dataset with different vector sizes (10,30, and 50), while the best results were achieved using the vector of size 50.

The proposed model was compared against the concept-based model and the LDA+Word2vec model in two settings. In the first setting which we call without researcher profile setting, they were all compared without applying phase 4 of the proposed model where each paper vector was observed using different models and cosine similarity was used to find each researcher's most similar papers to his training dataset. In the second setting, each paper vector was observed using different models, and then the profile construction phase was applied to join the researcher training dataset as one profile in order to find the most similar papers to the profile. Both settings generated a list of recommended papers where recall measures were calculated using the number of papers that the researcher prefers from the list of recommended papers. Fig. 9 shows the results of the average recall for the recommendation system with and without building the profile while using different models for document representation.

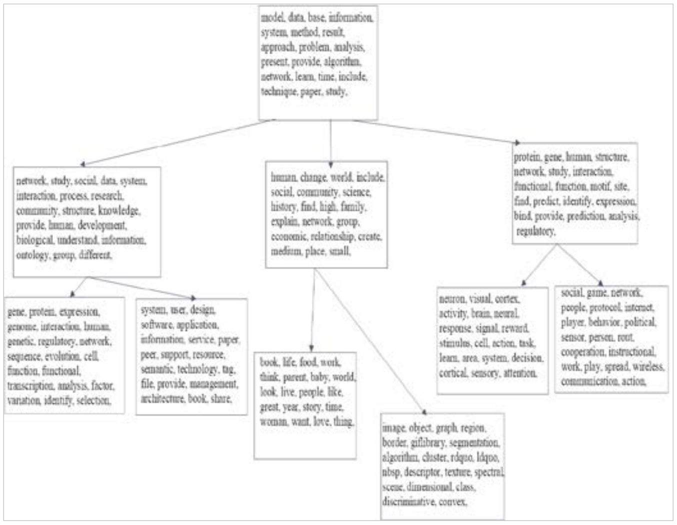
model, data, base, information, system, method, result, approach, problem, analysis, present, provide, algorithm, network, learn, time, include, technique, paper, study,
network, study, social, data, system, interaction, process, research, community, structure, knowledge, provide, human, development, biological, understand, information, ontology, group, different,
human, change, world, include, social, community, science, history, find, high, family, explain, network, group, economic, relationship, create, medium, place, small,
protein, gene, human, structure, network, study, interaction, functional, function, motif, site, find, predict, identify, expression, bind, provide, prediction, analysis, regulatory,
gene, protein, expression, genome, interaction, human, genetic, regulatory, network, sequence, evolution, cell, function, functional, transcription, analysis, factor, variation, identify, selection,
system, user, design, software, application, information, service, paper, peer, support, resource, semantic, technology, tag, file, provide, management, architecture, book, share,
book, life, food, work, think, parent, baby, world, look, live, people, like, great, year, story, time, woman, want, love, thing,
image, object, graph, region, border, giflibrary, segmentation, algorithm, cluster, rdquo, ldquo, nbsp, descriptor, texture, spectral, scene, dimensional, class, discriminative, convex,
neuron, visual, cortex, activity, brain, neural, response, signal, reward, stimulus, cell, action, task, learn, area, system, decision, cortical, sensory, attention,
social, game, network, people, protocol, internet, player, behavior, political, sensor, person, rout, cooperation, instructional, work, play, spread, wireless, communication, action,

Fig. 8. Part of the Topics Hierarchy that Learned from a Collection of Scientific Papers.

[2] https://pypi.org/project/gensim/

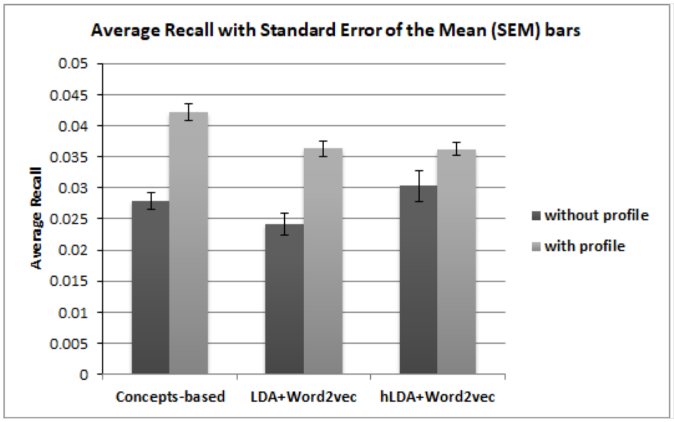

Fig. 9. Profile Effect on Average Recall Value.

Another experiment was conducted for the researchers who prefer a few papers only and the knowledge about their preference is scarce. We selected researchers who have between 10 and 20 preferred papers. The concept-based model, LDA+Word2vec model, and the proposed model were used to represent documents under the same two settings described previously in the previous experiment; once without researcher profile and once with the researcher profile. Fig. 10 shows the average recall results for each way of representation with and without building the profile.

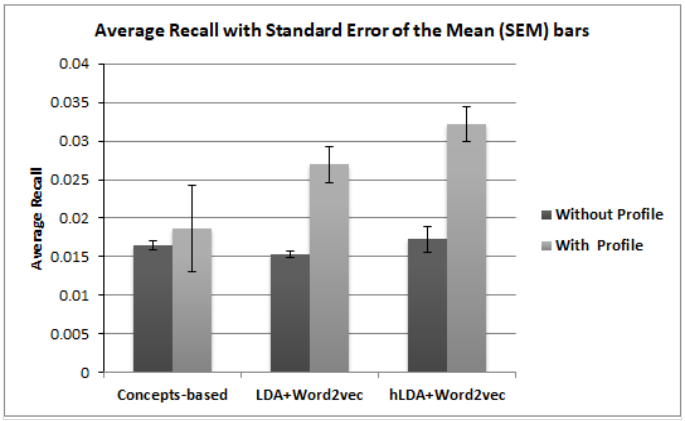

Fig. 10. Average Recall Values for Researchers with Few Preferences.

VI. Discussion

The previous section shows the experimental results of the performance evaluation for the proposed documents representation model against other representation models that exploit the topics and the concepts behind the documents corpus. The results are shown in Fig. 9 illustrate that without building a researcher's profile, the average recall of the proposed hierarchical document representation model is better than both the concept-based model [10] and the LDA+Word2vec model [9]. The proposed model improves the results of the recommendation systems for the dataset of size 6000 papers with 9% from the results of the concept-based model, and with 25% from the results of the (LDA+Word2vec) model. As the proposed model exploits the topics generated from the document and learns the hierarchical relation between those topics, the hierarchy allows the representation of the document to contain a more coarse-grained description of the researcher's preferences rather than a fine-grained description. Such a coarse-grained description allows more diversity while generating recommendations.

In addition, Fig. 9 shows that all document representation models while building researchers' profiles perform better than the same models without building researchers' profiles. Profile construction for multiple documents enhances the results of the recommendation system by 51 % and 50 % for the concept-based model [10] and LDA+Word2vec model [9] respectively rather than the models without building a profile, and with 20% for the hLDA+Word2vec model. It also shows that the concept-based model returns the best recommendation results when building researcher profiles but also it is very close to the performance proposed model. By applying the isolation forest model for the collection of papers that are preferred by the researcher to build one profile for them, it becomes possible to capture the researcher's common interests and to decide the most related papers to the interests of the researcher which are considered as normal behavior. On the other hand, the researcher's insignificant interests would be considered as an anomaly behavior.

Another improvement was achieved by the proposed model while doing recommendations for the researchers who have a few numbers of preferred papers. As shown in Fig. 10, the proposed model outperforms other models for the dataset of the researchers that prefer only from 10 to 20 papers. The proposed model without building a profile performs better than the results of the concept-based model by 5%, and also better than the results of (LDA+word2vec) model by 13%. Those improvements show that the hierarchical representation for the topics was successful in capturing researchers' preferences when only a little information was available about them. This illustrates how the coarse-grained description of researcher preferences can be extremely successful in certain situations. In addition, the results presented in Fig. 10 confirm the advantage of building a profile for all recommendation models where the results of the concept-based model, LDA+Word2vec model, and (hLDA+Word2vec) model are enhanced by 13%, 76%, and 86%, respectively when the profile was used to represent researchers preferred papers.

VII. Conclusion

In this paper, a novel document representation model is proposed. The proposed model combines different representation models into a more effective representation of the documents. More specifically, the model exploits the hLDA model to learn a hierarchy of topics that are generated from documents corpus, combined with the word2vec model to capture the semantics behind the document text. The proposed model introduces a representation for the multiple documents as one profile using the isolation forest model, to facilitate finding the similarity between the multiple groups of documents. The evaluation for the proposed model is conducted through different experiments for recommending scientific papers to researchers against similar methods that apply similar techniques; the concept-based model and the LDA+Word2vec model. The experiments show that the proposed model (hLDA+Word2vec) outperforms the concept-based model with 9%, and the LDA+Word2vec model with 25% as the proposed method exploits the topics behind the documents corpus and the hierarchical relation between them in document representation. In addition, the experiments that are conducted for recommending papers for researchers who like a few numbers of papers show that the representation of papers using the proposed model enhances the recommendation system results from the concept-based model with 5%, and with 13% from the LDA+Word2Vec model. Also, the profile construction for the multiple documents as one profile using the isolation forest model improves the results for the different representation models with 51%, 50%, and 20% for the concept-based model, LDA+Word2vec model, hLDA+Word2vec model, respectively. Therefore, the recommendation system using the proposed model performs better than other methods, especially when using it for constructing a profile.

REFERENCES

[1] Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. J. Doc. (1972).

[2] Dillon, M.: Introduction to modern information retrieval: G. Salton and M. McGill. McGraw-Hill, New York (1983). xv+ 448 pp., $32.95 ISBN 0-07-054484-0, (1983).

[3] Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Inf. Process. Manag. 24, 513–523 (1988).

[4] Ramos, J., others: Using tf-idf to determine word relevance in document queries. In: Proceedings of the first instructional conference on machine learning. pp. 133–142 (2003).

[5] Manning, C.D., Raghavan, P., Schütze, H.: Scoring, term weighting and the vector space model. Introd. to Inf. Retr. 100, 2–4 (2008).

[6] Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022 (2003).

[7] Blei, D.M., Griffiths, T.L., Jordan, M.I., Tenenbaum, J.B.: Hierarchical topic models and the nested Chinese restaurant process. Adv. Neural Inf. Process. Syst. (2004).

[8] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013).

[9] Wang, Z., Ma, L., Zhang, Y.: A hybrid document feature extraction method using latent Dirichlet allocation and word2vec. In: 2016 IEEE First International Conference on Data Science in Cyberspace (DSC). pp. 98–103 (2016).

[10] Mohamed, D., El-Kilany, A., Mokhtar, H.M.O.: Academic Articles Recommendation Using Concept-Based Representation. In: Proceedings of SAI Intelligent Systems Conference. pp. 733–744 (2020).

[11] Blei, D.M., Griffiths, T.L., Jordan, M.I.: The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. J. ACM. 57, (2010). https://doi.org/10.1145/1667053.1667056.

[12] Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv Prepr. arXiv1301.3781. (2013).

[13] Collobert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th international conference on Machine learning. pp. 160–167 (2008).

[14] Mikolov, T., Yih, W., Zweig, G.: Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies. pp. 746–751 (2013).

[15] Liu, F.T., Ting, K.M., Zhou, Z.-H.: Isolation forest. In: 2008 Eighth IEEE International Conference on Data Mining. pp. 413–422 (2008).

[16] Liu, F.T., Ting, K.M., Zhou, Z.-H.: Isolation-based anomaly detection. ACM Trans. Knowl. Discov. from Data. 6, 1–39 (2012).

[17] Preiss, B.R.: Data structures and algorithms. John Wiley & Sons, Inc. (1999).

[18] Chen, H., Ma, H., Chu, X., Xue, D.: Anomaly detection and critical attributes identification for products with multiple operating conditions based on isolation forest. Adv. Eng. Informatics. 46, 101139 (2020). https://doi.org/https://doi.org/10.1016/j.aei.2020.101139.

[19] Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Recommender systems handbook. pp. 1–35. Springer (2011).

[20] Pazzani, M.J., Billsus, D.: Content-based recommendation systems. In: The adaptive web. pp. 325–341. Springer (2007).

[21] Trstenjak, B., Mikac, S., Donko, D.: KNN with TF-IDF based framework for text categorization. Procedia Eng. 69, 1356–1364 (2014).

[22] Linstead, E., Rigor, P., Bajracharya, S., Lopes, C., Baldi, P.: Mining concepts from code with probabilistic topic models. In: Proceedings of the twenty-second IEEE/ACM international conference on Automated software engineering. pp. 461–464 (2007).

[23] Fang, Y., Si, L., Somasundaram, N., Yu, Z.: Mining contrastive opinions on political texts using cross-perspective topic model. In: Proceedings of the fifth ACM international conference on Web search and data mining. pp. 63–72 (2012).

[24] Apaza, R.G., Cervantes, E.V., Quispe, L.C., Luna, J.O.: Online Courses Recommendation based on LDA. In: SIMBig. pp. 42–48 (2014).

[25] Wang, C., Blei, D.M.: Collaborative topic modeling for recommending scientific articles. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 448–456 (2011).

[26] Sun, X., Liu, X., Duan, Y., Li, B.: Using hierarchical latent dirichlet allocation to construct feature tree for program comprehension. Sci. Program. 2017, (2017).

[27] Venkatesh, R.K.: Legal documents clustering and summarization using hierarchical latent Dirichlet allocation. IAES Int. J. Artif. Intell. 2, (2013).

[28] Rahmawati, D., Khodra, M.L.: Word2vec semantic representation in multilabel classification for Indonesian news article. In: 2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA). pp. 1–6 (2016).

[29] Li, Y., Yang, M., Zhang, Z.M.: Scientific articles recommendation. In: Proceedings of the 22nd ACM international conference on Conference on information & knowledge management. pp. 1147–1156 (2013).

[30] Manning, C., Schutze, H.: Foundations of statistical natural language processing. MIT press (1999).

[31] Baeza-Yates, R., Ribeiro-Neto, B., others: Modern information retrieval. ACM press New York (1999).