International Journal of Information Science and Management
Vol. 23, No. 4, 2025, 167-188
DOI: https://doi.org/10.22034/ijism.2025.1990502.1057 / DOR: https://dor.isc.ac/dor/20.1001.1.20088302.2025.23.4.8.8

Original Research

Automated Fuzzy Weighted Multi-Semantic Label Extraction of Persian News

Sahar Esmaeili Shayan
Ph. D. Candidate, Department of Management, Faculty of Social Sciences and Economics, Alzahra University, Tehran, Iran.
shayansahar3@gmail.com
ORCID iD: https://orcid.org/0000-0002-5543-5218

Neda Abdolvand
Associate Prof., Department of Management, Faculty of Social Sciences and Economics, Alzahra University, Tehran, Iran.
Corresponding Author: n.abdolvand@alzahra.ac.ir
ORCID iD: https://orcid.org/0000-0003-3623-1284

Saeedeh Rajaei Harandi
MSc, Department of Management, Faculty of Social Sciences and Economics, Alzahra University, Tehran, Iran.
sa.rajaeeharandi@gmail.com
ORCID iD: https://orcid.org/0000-0001-9710-3644

Received: 26 February 2023
Reviewed: 13 March 2023
Accepted: 26 August 2025

Abstract

The vast number of online text documents and Semantic Web trends has increased researchers' interest in semantic multi-label extraction. Research on the semantic extraction of multiple labels in Western and Eastern European languages is already well established. The challenge of machine reading and web-based knowledge extraction requires a scalable system to extract diverse information from large and heterogeneous collections. Hence, this study developed the multi-semantic fuzzy weight labeling system using natural language processing and supervised deep learning techniques. A long short-term memory (LSTM) was used for the extraction of labels, and the LSTM2 introduced by Yan, Wang, Gao, Zhang, Yang & Yin (2018) was used for the extraction of the label weights. To assess the degree of belonging of each document to each label, the resulting weights were modified according to their appearance in the document's subject or in the Meta section of the web page, and the weights were normalized and fuzzified. Finally, the C-means fuzzy clustering algorithm was applied to the documents to assign each data point a degree of membership in relevant clusters. According to the results, the model's accuracy was 59.8%, indicating that the extraction of weighted key phrases and the semantic labeling of the text could be improved through supervised methods.

Keywords: Text Mining, Multi-Label Extraction, Persian Natural Language Processing, LSTM.

Introduction

The development of high-speed Internet networks coupled with inexpensive data storage has led to a boom in electronic articles, with thousands of text documents now being published on the Internet every day. News represents a significant portion of big data (Zha & Li, 2019). The vast number of text documents requires efficient classification and retrieval. This highlights

the need for simpler text search methods and big data management techniques (Za'in, Pratama, Lughofer, & Anavatti, 2017; Gattiker, Hamada, Higdon, Schonlau, & Welch, 2016; Altınel, Ganiz, & Diri, 2015). Text clustering and classification are essential in the automated processing of this volume of text data (Za'in et al., 2017).

Clustering is primarily performed using two approaches: classic and fuzzy. With classic clustering, each object belongs to a single cluster (Za'in et al., 2017). Still, in fuzzy clustering, the nature of the objects is such that each object can be placed in different clusters according to the definition of the membership function in each cluster and with varying degrees of membership (Xie, Deng, Xia, Zhao, Wang & Gao, 2023; Za'in et al., 2017). Like clustering, labeling is a big data management technique used to identify the content and meaning of text (Gattiker et al., 2016). It also enables users to define, classify, and access text that contains the desired semantic content (Dang, Kalender, Toklu, & Hampel, 2017; Sun, Sun, & Cheng, 2016). In this case, the class of labels is selected from a set of hundreds, thousands, or even more (Liu, Chang, Wu & Yang, 2017). Human labeling has several limitations, including the potential for labeling text based on user preferences, the lack of necessary semantic alignment between the label and the document content, and the presence of semantic ambiguity in the labeled text. Therefore, using automated labeling can overcome some of these problems and improve the documents' clustering (Dang et al., 2017).

Additionally, text data, such as documents and web pages, are often annotated with multiple labels. Hence, the use of broad, data-driven approaches, such as multi-label extraction methods, seems necessary. Multi-label classification can be viewed as a subset of multi-class prediction, where the goal is to predict a set of labels associated with a given input (Altınel, Ganiz & Diri, 2015). Key multi-label extraction methods include: Text classification, a supervised training method in which the semantic class is static and pre-defined (Roy, Das, Kundu & Nasipuri, 2017), topic modeling, an unsupervised training method that uses a hidden variable to infer text meaning in two primary forms from a set of topics (distribution of terminological polynomials), and assignment of issues to documents (distribution of topic polynomials (Mei, Shen & Zhai, 2014). The weakness of this method lies in the creation of synthetic and mixed themes (Sorodoc, Lau, Aletras & Baldwin, 2017; Chopra, Auli & Rush, 2016). Text Document Abstract is an unsupervised training method for creating a text document subject in two forms: structured and unstructured document abstract. Significant disadvantages of this method are the difficulty of implementation due to the need to define the final standard form of the topics and the need for general knowledge to complete the abstract (Liu, Li, Zheng, & Sun, 2009). Research on the semantic extraction of multiple labels in Western and Eastern European languages is well advanced (W3Techs, 2023). The nature of the Persian language is complex, as it includes words with separate components and phrasal verbs, making it challenging for many text classification systems to be applied in the Persian language (Ahmadi, Tabandeh, & Gholampour, 2016). Therefore, previous studies on the automatic classification of Persian texts are limited. The efforts of Persian text mining researchers have been more focused on comparing Persian key phrase extraction methods (Lazemi, Ebrahimpour-Komleh & Noroozi, 2019), the semantic labeling of Persian sentences (Kingma & Ba, 2014; Kou, Li & Baldwin, 2015), and even attempts to solve the fuzzy classification problem in text documents have been mostly limited to English. Efforts at fuzzy clustering of text documents and Persian messages are also unsatisfactory.

The use of the Persian language by internet users and the production of electronic text

documents in this language have increased significantly, with Persian users reaching around 2% in 2019 (Ahmadi et al., 2016; Lazemi et al., 2019). Additionally, the Persian language has the highest usage among Internet users in the Middle East, surpassing Turkish by three ranks and Arabic by eight ranks. Among the top 10 internet languages, Persian content and the number of pages with Persian content on the internet ranked 8th in 2019. As W3Techs' website shows, Persian ranks third among the fastest-growing content languages on the Internet (Ahmadi et al., 2016). Additionally, approximately 1.7% of web pages on the internet are in Persian (Yu, Li, Zhao, Zhang, & Wu, 2015). This growth reflects the growing popularity of this language on the World Wide Web. Given the increasing role of the Persian language in the preparation of electronic texts and the importance of semantic labeling in Persian texts for improving clustering results, the question arises: how to annotate messages in Persian with machine learning capabilities and automatic fuzzy weighting?

Persian text documents on the internet are often unstructured and can convey diverse types of information. Thus, each message conveys multiple meanings (Zha & Li, 2019). Therefore, this study aims to extract and weight semantic labels from Persian text documents using natural language processing tools in conjunction with text mining and machine learning techniques. Furthermore, given the limited use of the Persian language in studies conducted in this area, this study aims to expand the use of Persian as a non-English language. The results of the survey can be used to understand the multiple meanings of the text document and achieve various natural language processing goals.

The study proceeds with a literature review, followed by an explanation of the research method in detail. Finally, the discussion, as well as the conclusions and future proposals, are commented on.

Literature Review

Natural Language Processing (NLP) is a computational approach concerned with giving computers the ability to understand natural language (spoken and written) in the same way a human might be based on a range of theories and technologies (Khurana, Koli, Khatter & Singh, 2023; Schoene, Basinas, van Tongeren, & Ananiadou, 2022; Sullivan & Keith, 2019). It employs algorithmic approaches grounded in statistical techniques to infer the semantic meaning of textual data (Leeson, Resnick, Alexander, & Rovers, 2019). In NLP and machine learning, a topic model is a type of statistical graphical model that helps discover abstract "topics" that appear in a set of documents. The topic modeling technique is often employed in the text mining pipeline to uncover the hidden semantic structures within the text body (Lin, 2022). Semantic structure prediction is the task of identifying goals, labeling their contexts or meanings, and labeling all their reasoning errors in a sentence (semantic labeling) (Swayamdipta, Thomson, Lee, Zettlemoyer, Dyer & Smith, 2018). Semantic labels are words or phrases that indicate the subject of the text and help users to define, classify, and access text that contains the desired semantic content (Za'in et al., 2017; Dang et al., 2017). Multi-semantic labeling is a fundamental way of extracting, managing, and organizing text documents. This method assumes that each text document belongs to one or more categories, and therefore, each document can have multiple semantic labels (Pawar, Phansalkar, Sharma, Sahu, Ang, & Lim, 2023; Roy, Das, Kundu, & Nasipuri, 2017).

The purpose of multi-label text classification is to assign the most appropriate labels that accurately explain their meaning. Multi-label classification is a subset of multi-class prediction

that aims to assign multiple labels to a single input (Altınel et al., 2015; Roy et al., 2017). Two main approaches to solving various labeling problems are surface learning and deep learning (Sun et al., 2016). The surface learning approach aims to map the document in a semantic space with fewer dimensions. The key disadvantage of this approach is presenting documents based on the number of words in the text document and ignoring the order of words that appear in the electronic document, which significantly affects the overall message of the document (Yacob & Semere, 2021; Pan, Swaroop, Immer, Eschenhagen, Turner & Khan, 2020). However, the deep learning approach is notable for its ability to remember the order of words in a text document. It is therefore better suited to analyze and understand the meaning of texts (Kim & Lee, 2014).

The essential methods attributed to the deep learning models associated with multi-label classification include CNN (Rajabi, Sahebari & Thomas, 2022), recurrent neural networks (RNN) (Zheng & Zheng, 2019; Zhang, Lee & Radev, 2016), recurrent convolutional neural networks (RCNN) (Johnson & Zhang, 2014), and Bow Convolutional Neural Network (Bow-CNN) (Wang, Chen, Hao, Peng & Hu, 2019). CNNs' struggle to preserve the semantic sequence of words and sentences led to the development of a specific type of RNN called LSTMs (Tai, Socher, & Manning, 2015). LSTM is a type of RNN developed to overcome two issues, including vanishing and exploding gradients, when applying the recurrent neural networks to train long-distance correlations in sequences (Xiong, Jin, Liu, Cai & Xiao, 2023; Tai et al., 2015). Multi-label document classification using the LSTM model can address the challenges in representing documents for the automatic and semi-automatic selection of classifiers for each document class. The LSTM model can also be used to score documents using a label sequence and threshold selection for the labels (Soloshenko, Orlova, Rozaliev & Zaboleeva-Zotova, 2015).

The essential methods attributed to the deep learning models associated with multi-label classification include CNN (Rajabi et al., 2022), recurrent neural networks (RNN) (Zheng & Zheng, 2019; Zhang et al., 2016), recurrent convolutional neural networks (RCNN) (Johnson & Zhang, 2014), and Bow Convolutional Neural Network (Bow-CNN) (Wang et al., 2019). The shortcomings of CNN in maintaining the semantic sequence of researchers' words and sentences led to the development of a specific type of RNN called LSTMs (Tai et al., 2015). LSTM is a type of RNN developed to overcome two issues, including vanishing and exploding gradients, when applying the recurrent neural networks to train long-distance correlations in sequences (Xiong et al., 2023; Tai et al., 2015). Multi-label document classification using the LSTM model can address the challenges in representing documents for the automatic and semi-automatic selection of classifiers for each document class. The LSTM model can also be used to score documents by utilizing a label sequence and selecting thresholds for the labels (Soloshenko et al., 2015).

Some studies, primarily focused on English and non-Persian texts, labeled text documents using different dictionaries for each subject. They then rated each word in the dictionary, provided a correlation graph of the terms, and calculated the overall score (Altınel et al., 2015). In another study, an entity was considered in the document. The degree of popularity of the given entity was measured based on the text document's views, and then the semantic label was assigned to the document (Ashtiani & Raahmei, 2023). For example, Khuntia and Gupta (2023) used the word embedding technique and LSTM to classify Indian headlines. His proposed method overcomes the need for data labeling by utilizing a BERT sentence encoder. In another

study, Azarafza, Feizi-Derakhshi, and Shendi (2020) proposed a new method for automatically extracting key phrases from Persian scientific documents, utilizing the power of incident-based and statistical approaches. Their results showed that their method works better than similar work on Persian texts. Another study by Davari, Mahdian, Akhavanpour, and Daneshpour (2020) proposed a deep-learning-based method for extracting features of Persian texts, classifying texts, and identifying subjects in collections. Their results showed that the classification of Persian news texts worked well. Dastgheib, Koleini, and Rasti (2020) also proposed a hybrid method combining structured correspondence learning (SCL) and convolutional neural networks (CNN). Their results showed that combining SCL and CNN can improve sentiment classification results by more than 10%. Masoudian, Derhami, and Zarifzadeh (2019) also proposed a "local classifier on each parent node" approach to classify partially tagged documents hierarchically. Their results indicated that the proposed approach performed acceptably.

Another study by Dastgheib and Koleini (2019) on improving Persian text classification using latent semantic spaces demonstrated that LSI semantic spaces can achieve better performance in terms of computation time and classification accuracy. The study by Sharifi and Mahdavi (2019) on unsupervised methods for Persian key phrase extraction, despite promising results, proposed improving the evaluation phase by using specialist-preferred labels for better semantic key phrase extraction. In another study, Farahani, Fatemi, and Ghorbani (2019) proposed a supervised approach to extract keywords from Persian documents. They used lexical chains and focused on the relationship between words. Their results showed that using semantic labels would improve keyword extraction performance. The study by Lazemi et al. (2019) also proposed a hybrid method for automatically extracting keywords from Persian documents using the Support Vector Machine (SVM) algorithm. In addition, an extensive corpus was developed to evaluate the proposed method. Their results demonstrated the good performance of their proposed method in recognizing keywords from Persian documents. This study suffers from a lack of attention to semantic correlation in the body of text, despite excellent key phrase extraction results. They addressed this scarcity in their article as concept extraction. In another study, Ghasemi and Jadidinejad (2018) applied a deep convolutional neural network to Persian characters, achieving higher precision (0.49) compared to other character-based representations. Some researchers also focused on extracting semantic features or solving classification challenges in Persian using CNN, LSTM, and hybrid models (Dastgheib et al., 2020; Davari et al., 2020).

Recent advancements in hierarchical and multi-label text classification have shown promise in addressing complex labeling structures in diverse domains. Cheng and Shi (2025) proposed a label-aware dual graph attention network for hierarchical multi-label classification in tourism datasets, achieving notable improvements in accuracy and semantic alignment of labels. Similarly, Chen, Zou, Cheng, Liu, and Xie (2024) developed deep neural networks to interpret semantic content in online course reviews, demonstrating the efficacy of neural attention mechanisms in extracting context-aware labels. Tarekegn, Ullah, and Cheikh (2024) offered a comprehensive survey of deep learning models tailored for multi-label learning, identifying architectural trends and challenges. Additionally, Li, Liu, Wang, and Chen (2024) explored incomplete multi-label learning problems, proposing robust solutions for scenarios with missing label data.

Aghighi and Bashiri (2025) specifically evaluated deep learning models on Persian texts

and found that CNN architectures outperformed LSTM and RNN models in terms of classification stability and performance, using the Hamshahri dataset. These recent developments suggest the importance of model structure and data preparation strategies in enhancing multi-label classification performance, particularly in low-resource languages such as Persian.

Based on studies, research on semantic text labeling has focused on the discovery and use of emotional keywords. That is, concrete words that express the emotional state of the speaker. Using emotional keywords is the most direct way to detect user emotions from text input, and several methods using selected emotional keywords have been proposed. However, all keyword-based systems suffer from the following problems: ambiguity in the definition of emotional keywords; emotion recognition from sentences without emotional keywords, and especially without semantic and syntactic information for emotion recognition (Wu, Chuang, & Lin, 2006). In addition to keyword-based approaches, some researchers used other clues to textual data, such as pragmatic intention, plausibility of text content, and paragraph structure (Cejuela et al., 2017).

Additionally, the importance of semantic labeling in Persian texts and its application to enhance the clustering results of text documents has been overlooked. Additionally, efforts have been made to cluster documents in a fuzzy manner; however, the messages in Persian are not satisfactory. Therefore, this study aims to address the problem of extracting multi-semantic labels from Persian electronic texts using a deep learning approach. This approach will extract the subject from the semantic blocks of text documents, investigate the general meaning of the document, and facilitate clustering to improve results.

Materials and Methods

This study aims to develop a fuzzy-weighted multi-semantic labeling system using natural language processing and supervised deep learning techniques. Since standard datasets in Persian are either nonexistent or very limited, this study utilized real and practical data from www.kookmobile.com. The dataset consists of two multi-label datasets containing 1106 Persian-language articles collected by crawling on the Persian technology-related content site. LSTM was used to extract labels, and another LSTM called LSTM2 (introduced by Yan et al., 2018) was used to extract the weight of labels. LSTM is a form of RNN architecture that addresses the challenges of manual or automatic classifiers in classification, as well as the difficulties in representing documents for the automatic and semi-automatic selection of classifiers for each document class. The LSTM model can also score documents by generating label sequences and applying threshold-based classification (Yan et al., 2018; Altınel et al., 2015).

To assess the degree of belonging of each document to each label, the resulting weights were modified according to their appearance in the document's subject or in the Meta section of the web page, and the weights were normalized and fuzzified. Finally, the C-means fuzzy clustering algorithm was implemented in the documents to assign membership to all data in each cluster. A key difference between classical clustering and fuzzy clustering is that in the fuzzy approach, each object can be divided into different clusters based on the definition of the membership function in each cluster, with varying degrees of membership (Cejuela et al., 2017). Since the results of clustering algorithms in natural language processing and text mining are not always consistent (Meesad & Li, 2014), using multi-semantic labels for Persian texts is

expected to enhance the clustering results.

The Python programming language was used to carry out the research phases. The study framework is shown in Figure 1.

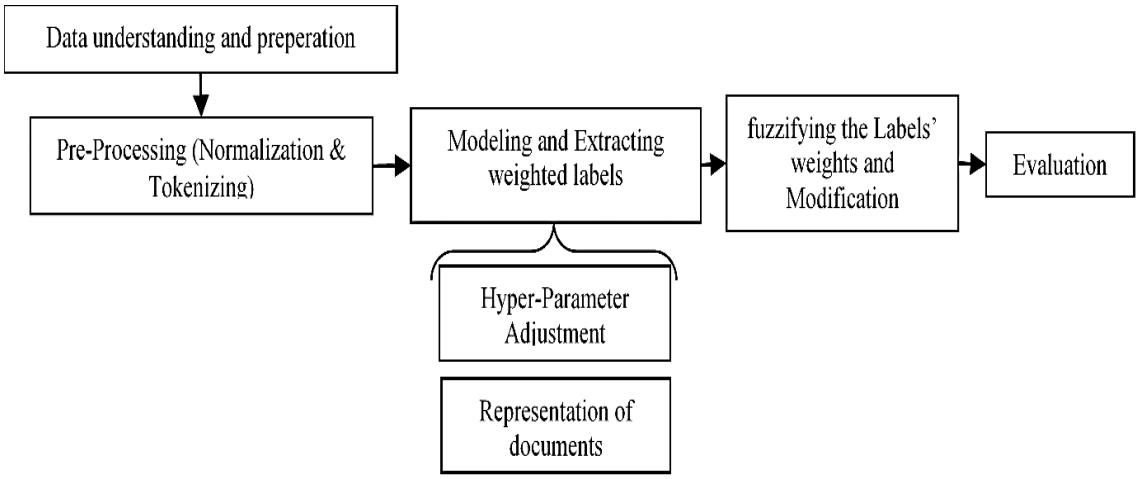

Figure 1: The framework of the study

Data analysis

Data understanding and preparation

The dataset used in this study includes the full-text document, the URL of each document, the meta description of the web page, the subject of the document, and the specialized labels www.kookmobile.com developed by WordPress CMS using an HTTP crawler. Keywords from the Meta labels provided in the HTML page were used as labels for training the model. Some articles did not include the relevant keywords or used all keywords in the article title and description, so they were excluded from the modeling and scoring phase. First, all articles on the target page, a technology news page, were crawled. For this purpose, a complete list of site articles was created based on the XML sitemap, which was manually compiled from the original sitemap. After preparing the crawler code, each article was parsed, and its text was stored in the text file after retrieval. The crawler code was built using the BeautifulSoup framework. After running this code on the landing page, 726 items were returned. An expert performed the tracking of news-related labels through the above technique. Subsequently, the phases of pre-processing (normalization), text processing, and model modeling and evaluation were carried out to extract, classify, and fuzzify the semantic labels. The number of document labels ranged from 2 to 7, and the total number of classes was 726. Table 1 indicates some dataset statistics.

Table 1
Dataset details

Dataset	Number of documents	Number of verified documents (with specialist labels)	Min and max words per document	Number of all tokens in the dataset	The average number of sentences per document	Number of multi-tokens phrases in constructed dictionary	Number of all single token and multi-tokens	Min and max labels per document	Number of classes(labels)	Type of label context
Electronic Text Document	1146	726	450< >1500	80001	36	240	5836	>2 <7	240	Technological

Pre-processing (normalization and tokenization)

Text is unstructured data, so preprocessing is required to transform unstructured text data into a structured form. This was done through normalization. Text normalization is a text preprocessing method that refines the text. When normalizing text, all language tags are first removed from the text, followed by the removal of HTML tags. The next step in the text normalization phase involves removing words that play no fundamental role in shaping or changing the meaning of the sentence. This way, after the text normalization phase, the text is reduced, and only words with morphological and syntactic roles remain in the file. To normalize the text, all downloaded files were first uploaded in a loop, and 726 files were processed into Hazm normalizer and shared on GitHub. The output was then placed in a regular directory.

Due to the processing of Persian text and the encoding of various characters in Persian text documents, the normalization stage of Persian text, such as removing language markers from the text and cleaning up text within HTML tags, is challenging. Therefore, the encoding of various characters within punctuation was taken into account. For example, there are always multi-token terms; space is a separator between Persian terms. For example, in “تعمیرات تلفن همراه را به ما بسپارید” (Leave the mobile phone repairs to us), the phrase “تلفن همراه” may be spelled in two forms of “تلفنهمراه” and “تلفن همراه”, both of which are correct; but writing the second is common. In this example, considering the space as a delimiter, the word “تلفن همراه” has been split into two tokens “تلفن" and “همراه” while the token should read “تلفن همراه”. To solve this problem, a dictionary of these terms was formed, and the presence of these terms in the sentence was checked to replace them with an appropriate space. In the next step, words that played no role in the sentences were removed.

The text is then tokenized. This step separates the remaining components of the normalization step to form a bag of words. To tokenize the existing text, the output of the previous step was first split into individual tokens via the Hazm library. Then each generated item was saved with its own ID. For this purpose, the texts were divided into two levels: word

and sentence. The role of each word in the sentence in terms of verb, subject, and object was determined by two functions, Chunker and POS-Tagger, and the Dependency-Parser function from the Hazm library.

Some words may have the same meaning, but only their properties are different. Additionally, Persian vocabulary was often derived from words related to other concepts, such as anatomical terms. For example, in the word "تعمیرکاران", the word "تعمیر" ends with "کار", which in turn ends with the noun "ان" (repairmen). Hence, the words were derived from their root. This was performed using a toolkit introduced by Paik (2013), a freely available source toolkit that includes core activities such as normalization, tokenization, and POS tagging. The result of this step was a text in which the related words were matched to their roots.

The normalized documents were then read from the normalized dictionary and indexed into the TextBlob library to calculate the TF-IDF weight for each word and assign the weights to each term based on the term frequency and the inverse document frequency. The most important terms have a higher score (Chollet, 2017).

Modeling and weighted labels extraction

At this phase, an LSTM network was used to extract labels. Due to the shortcomings of traditional text classification methods, a new recurrent ranking framework (LSTM2) introduced by Yan et al. (2018) was employed to extract the weights of labels and simultaneously rank them (Figure 2). This model has six hidden layers and consists of two steps: RepLSTM, a method for document representation, and RankLSTM, a method for weighting the classification of documents.

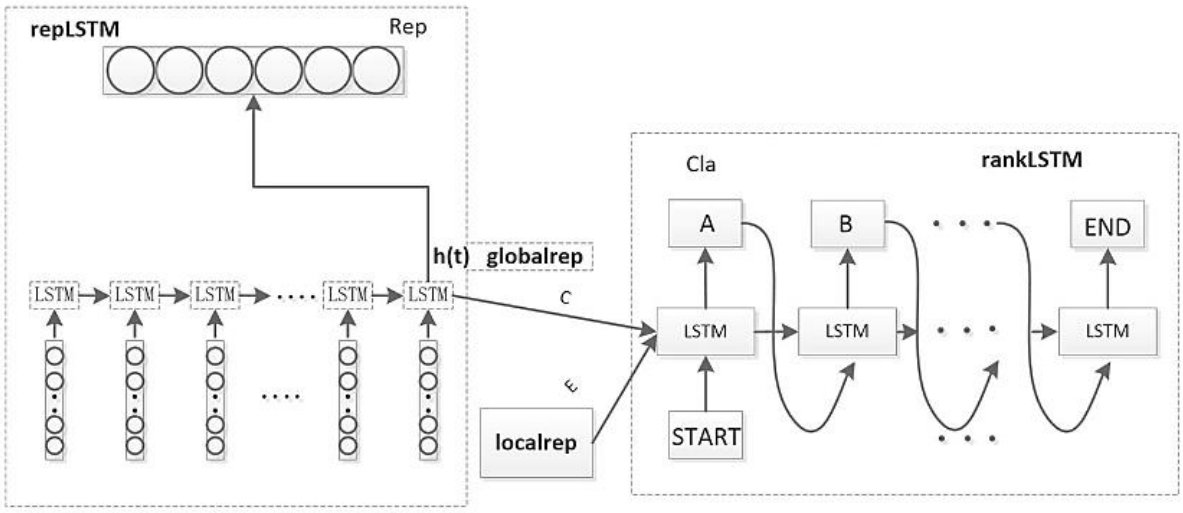

Figure 2: *Proposed LSTM2 model (Yan et al., 2018)*

The LSTM model with six hidden layers and an initial training rate of 0.01 was used to model the label rendering and classification processes. The primary issue with using this model is overtraining, which yields impractical results. To avoid this problem, the hyperparameters of the LSTM model were tuned by selecting and adjusting the activation, cost, and optimization function, as well as other functional modeling parameters. For initial weighting, a cross-library context layer was employed to examine the relationship between words and sentences, as well

as the conceptual connections between paragraphs, based on the weight and importance of each word in the text (Kingma & Ba, 2014).

The cost and optimization functions, which are the most critical parameters for implementing the model, have been selected and fitted. The labeling process used the binary cross-entropy cost function based on Equation 3.1 to update the model weights:

$$L(q) = -\frac{1}{N}\sum_{i=1}^{N} \quad Y_i * log\ log\ (P(Y_i)) + (1 - Y_i\ ) *\ log\ \ (1 - P(Y_i) \qquad (1)$$

Where Y_i is the label and index i is the number of data (726 items), ranging from 1 to N (1 for actual predicted values and 0 for false predicted values), and P (Y_i) is the probability of a correct prediction. Entries: These values are used to repeat all entries whose number is N. If the label is guessed correctly (Y = 1), the value of log P (Y_i) is added to the final cost function. If the label is not guessed correctly (y = 0), the value of log (p (1)) that represents the probability of a model error is added to the cost function.

Establishing training cycles for modeling is an experimental process, so modeling was stopped after the lowest cost was reached (approximately 0.019); this number of training cycles effectively trained the network. Other hyperparameters have been adjusted based on best practices. In the training process, 20% of the dataset was used as test data and 80% of it as training data. The model used for data training and weighted label creation incorporates both the representation process (RepLSTM) and the label ranking process (Rank LSTM), which were trained separately. The RepLSTM process is based on supervised training and consists of three levels: document entry, word presentation, and labeling. The document input utilizes the Word Embedding (WE) method for each word, employing word2vec (Deng, Tur, He, & Hakkani-Tur, 2012). WE is a deep learning approach that lacks knowledge of the syntactic and grammatical structure, meaning, and morphology of words, phrases and sentences (Mikolov, Yih & Zweig, 2013). The repLSTM entry is based on representing WE features. Therefore, it can be applied to any language (Deng et al., 2012). Thus, the first layer of the model consisted of embedded words with a length of 100 and a depth of 50.

The second layer uses a 50% dropout layer. The third layer contains the LSTM model, and the fourth layer uses batch normalization with a size of 128. The fifth layer was used to preserve meaning and takes context into account. In the sixth layer, a 50% drop layer was used to prevent overfitting of the model. The output of this process contains 726 text documents that are injected into the RankLSTM process to extract weighted labels. The RankLSTM stage is identical to the RepLSTM model, whose output comprises 599 keywords and weighted labels derived from document training.

In the word representation process, the model is optimized to minimize the error between the input words and the vector resulting from the implementation of the word embedding model. Additionally, each label is utilized in supervised training. The Softmax function was implemented at the output layer as a network activation function to produce an output as a percentage of the likelihood of each label in each text document. Softmax is described as producing class probabilities rather than percentages or fuzzy numbers. The output of the first process, which were the public labels of each document, served as the input to the second process, RankLSTM, which ranked the public labels extracted from the first process. The weights were trained through supervisor training and through the introduction of labels (2). A linear matrix relation (C, E) was used to integrate two sets of output attributes, localRep and globalRep, through Equation 2 (Yan et al., 2018).

$$H_{end} = C * globalRep + E * LocalRep \quad (2)$$

Where, Hend is the input to the RankLSTM network. If the weight matrix (E) is 0, the network considers only global features.

End-to-end learning is an effective method for training a model to provide explicit word representation instructions during the classification process (Kou et al., 2015). The RankLSTM stage achieves this by updating the effect of the word representation (as input) on the predicted labels (as output), rather than simply updating the word representation itself. Del would overfit after a few training cycles; therefore, the training process should stop once the model stops overfitting. Until then, Adam was used to mitigate noise problems, and two blending and sorting modes were configured to prevent overfitting of the model. However, due to overfitting in the initial training courses, 50% of the trained data was discarded, and the template model was identified in the newly entered data

Setting training cycles for modeling is an experimental process; therefore, modeling was stopped once the lowest cost (approximately 0.019) was reached. The number of training cycles effectively trained the network. Adding dropout layers to some of the network's hidden layers helps avoid two common neural network training problems: overfitting and underfitting.

Labels' modification and fuzzifying the weights

The candidate's weight should be adjusted based on its relevance to reflect the meaning behind the text better. The weight assigned to each label indicates the importance of the labels in the document. Like the probability values, these values range from 1 to 100. The closer the probability of extracting the document label is to 100, the more relevant the semantic domain of a text document is for this label. The following factors were considered to be effective factors for label weight:

- Letter trigram that uses the superposition of ternary word vectors formed from the words of the text to determine the weight coefficients of the labels (Brin & Page, 1998).

- PageRank, which uses the page rank of the text document and the graph of the links in the text as coefficients (Saaty, 1988).

- Topic Overlap, which considers the overlap between the number of words in the topic and the labels to apply the label coefficient (Chopra et al., 2016).

It is therefore necessary to modify the weights obtained from the network. For this purpose, the initially crawled dataset included the URL of the webpage to determine the importance of the extracted labels and their occurrence in the URL. Then the topics of the articles were considered, and string n-grams were created. Matching n-grams were extracted from the subject. Then the extracted labels were listed. Finally, the web page's meta description was checked to list the meta description key phrases and extracted labels.

To identify the impact of each factor on semantic document comprehension, Multi-criteria Decision Making (MCDM) was used to compare pairs of factors. Fifteen search engine experts completed the questionnaires. The inconsistency of the MCDM model was reported as 0.08, which is less than 0.1, demonstrating the reliability of the results. The coefficients used to correct label weight come directly from the MCDM model. Finally, the labels were fuzzified to determine how much each label contributes to the overall meaning of a document. To do this,

Saaty's method was used, dividing each weight by the total weight of the items in the collection such that the sum of the weights of each item in the collection equals one and the weights are normalized (Equation 3) (Hadifar & Momtazi, 2018).

$$n_{ij} = \frac{x_{ij}}{\sum_{k=1}^{m} x_{ij}} \quad (3)$$

Where n_{ij} is the fuzzy weight of each item, and m is the total number of items. With this technique, the weight of each label was normalized, and the labels were represented in a fuzzy manner.

Evaluation

At this point, the network performance and output labels were evaluated. The precision, recall, and F1-score were used to assess the network performance (Table 2).

Table 2

Evaluation of the proposed LSTM model

Phase	Criterion Evaluation	Criterion value
Extracting semantic labels (Input = full-text articles)	F1-score	59.8%
	Precision	42.7%
	Recall	96.8%

As indicated in Table 2, the F1-score value of Persian data is 59.8%. The comparison of model output labels with model training labels of text document #2 is indicated in Table 3.

Table 3

Comparison of model output and model training labels

Text Document Title: آموزش بکاپ اطلاعات از گوشیهای اندروید با شیوه آسان (Training Backup Android phone information simply and understandably for everyone)				
Document Number	Model Training Labels for This Document	Output Labels With Main Text Input	Weight of Labels	
			Before Normalization and making Fuzzy	After Normalization and making Fuzzy
#2	1- آموزش (Training) 2- اندروید (Android)	1- گوشی های اندروید (Adroid Phones) 2- بکاپ (Backup) 3- اطلاعات (Information)	0.82 0.72 0.94	33% 29% 38%

Based on Table 3 in Text Document #2, which discusses Android phones, backups, and data, the model was accurately used to label the document. After entering the normalized text, the model output contained three labels "گوشی‌های اندروید، بکاپ، اطلاعات" (Android phones, backup, and information). However, the content provider specialist had marked two labels in the document. The model results included one- and two-word keyword phrases, whereas the expert labels consisted only of one-word keyword phrases. Regarding the semantics of the document,

it is evident that the most subjective trend is related to "بکاپ اطلاعات گوشی های اندروید" (Android phone information backup), which is more closely aligned with the document's semantics. The model output specifies more relevant and meaningful labels for the electronic document. The relative weights, on the other hand, indicate that the full text of the document is about information rather than Android phones and backups.

Fuzzy clustering of Persian documents (an example)

At this stage, for fuzzy clustering of Persian Documents, the skyfuzzy library was used. To use this library, the following code must be added to the fuzzy code section of the project:

```
from __future__ import division,
print_function
import numpy as np
import matplotlib.pyplot as plt
                import skfuzzy as fuzz
```

Other parameters required to run the fuzzy clustering model on the study dataset are set as shown in Table 4.

Table 4
Values of fuzzy clustering function parameters

Parameter	Value
Centers	Final_labels
Alldata	True
Ncenters	20
Error	0.005
Maxiter	100
Init	None
Cluster_membership	• Documents with a keyword weight of more than 0.9 • Documents with keyword similarity greater than 0.8

Then, the partition coefficient function was used to determine the correct number of clusters (Equation 1). The Partition Coefficient measures the degree of fuzziness of the final partition cluster using the fuzzy partitioning matrix. This function has values between 1/c for a set of completely fuzzy objects and 4 for complex clusters. The higher this value, the better the partition results (Bezdek, 1981).

$$V_{PC}(U) = \frac{1}{n}(\sum_{k=1}^{n}\sum_{i=1}^{c} u_{ik}^{2}) \qquad (4)$$

In which V_{PC} is the value of the partition coefficient, n is the cluster centers, U is the membership degree, and k is the optimal number of clusters (Bezdek, 1981). Based on the results, the highest value of the partition coefficient function was 0.74 for 20 clusters. Therefore, 20 cluster centers were chosen for clustering. Then, the fuzzy clustering C-Means (FCM)

algorithm was implemented. In the FCM clustering method, a membership value was assigned based on the similarity of the data sample to the center of the cluster. Membership scores range from 0 to 1, with higher similarity indicating a higher membership score (Li et al., 2017). A visualization of the clusters created using the optimal number of cluster centers is shown in Figure 3. As shown in Figure 3, the highest level of compression between clusters was observed at cluster center 20.

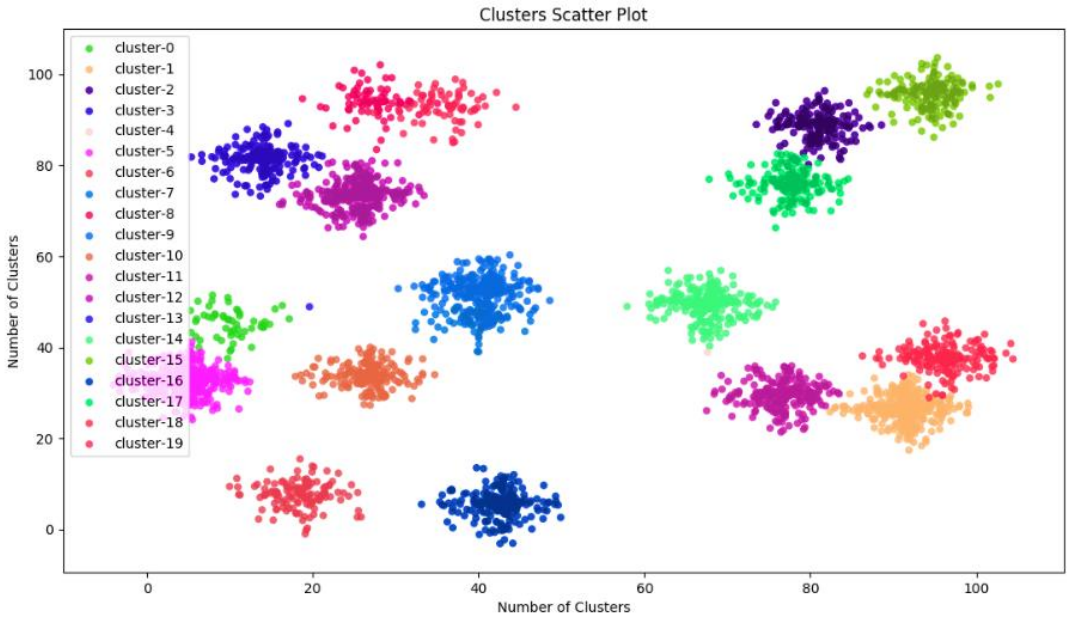

Figure 3: Clusters' visualization

Discussion

This study developed a novel system for multi-semantic labeling of Persian electronic texts using supervised deep learning techniques. The key contribution lies in the implementation of LSTM and an enhanced model, LSTM2 (Yan et al., 2018), for label extraction and weighting, respectively. This approach departs from existing shallow methods that rely primarily on syntactic features by employing semantic role labeling tailored to the intricacies of Persian.

The paper details the development of a multi-semantic fuzzy weight labeling system using natural language processing and supervised deep learning techniques, specifically LSTM models. The study's results indicated an accuracy rate of 59.8% for the model. This suggests that the extraction of weighted key phrases and the semantic labeling of text can be enhanced using supervised methods. The accuracy reflects the model's capability to deal with the complexity and variability of Persian text data, demonstrating a significant step forward in the semantic processing of Persian news articles.

The system achieved an F1-score of 59.8%, with a high recall rate of 96.8% and a relatively lower precision of 42.7%. This result indicates that while the model effectively identifies a large portion of the relevant labels, it requires further optimization to reduce false positives. The system also introduced a fuzzification step for label weights, leveraging document features such as metadata and subject headings, which enriched the semantic granularity of the label associations. Moreover, the integration of word2vec representations for the first time in Persian multi-label document clustering stands as a novel contribution, providing nuanced similarity metrics.

Due to the scarcity of standardized datasets for Persian, real-world web content was used, providing a practical testing ground while also presenting challenges for benchmarking. Therefore, this study will serve as a basis for expanding the application of the Persian language in this field. The results of the study showed that LSTM2 modeling could enhance the extraction of multiple labels from Persian electronic texts and reveal the semantic blocks that comprise a text document.

The proposed system was benchmarked against prior works and recent research to evaluate its novelty and practical advantages. Compared to English-language models, particularly the original implementation of LSTM2 by Yan et al. (2018), which reported an accuracy of 67% for English documents, the current system underperformed. However, it is crucial to note that the English dataset used in Yan et al.'s work did not incorporate fuzzy weighting, and differences in language structure, resources, and dataset quality likely contributed to this performance gap. The proposed system outperforms recent advances in multi-label classification of Persian texts. For instance, Aghighi and Bashiri (2025) reported strong results using CNN models for Persian document classification, whereas our LSTM2-based model leverages label ranking and fuzzification to handle multilabel ambiguity effectively. Cheng and Shi (2025) demonstrated the utility of hierarchical architectures with dual graph attention mechanisms, which can be extended to Persian datasets to enhance understanding of label hierarchies. The semantic modeling approach used by Chen et al. (2024) provides a foundation for integrating sentiment and subjectivity into label semantics, which complements the semantic weight calibration performed in this study.

Furthermore, the survey by Tarekegn et al. (2024) underscores the significance of tailored deep learning frameworks for multi-label learning, which supports our system's dual-model architecture (RepLSTM and RankLSTM). Similarly, Li et al. (2024) highlighted the challenges of incomplete labeling, a common issue in Persian corpora, affirming the need for robust weight normalization and fuzzy logic as implemented in our system. Khuntia and Gupta (2023) leveraged BERT encoders for headline classification, which, while effective, is language-specific and lacks the fuzzy membership advantage in clustering used here. The value of this study is also evident when considering specific Persian NLP applications. For example, Azarafza et al. (2020) focused on keyphrase extraction from Persian texts using incident-based and statistical models, whereas this work addresses semantic blocks via deep learning and fuzzy weightings. Dastgheib et al. (2020) and Davari et al. (2020) applied hybrid CNN or LSTM architectures to sentiment classification and topic modeling; however, they did not implement a multi-label framework or utilize document structure to calibrate semantic weights, as done here. While models such as those proposed by Masoudian et al. (2019) and Dastgheib and Koleini (2019) achieved higher accuracies (approximately 82%–91% and 84%, respectively), their work focused on binary or specific-topic classification (e.g., sports or politics), using domain-specific datasets. In contrast, the current study addresses a broader and more semantically ambiguous task across various topics, utilizing a multi-label setup.

This system outperformed several Persian models: Ghasemi and Jadidinejad (2018), whose character-based CNN achieved approximately 49% accuracy. This positions the present work as a methodological advancement, especially given that no standardized Persian dataset exists and the model was tested on real-world, heterogeneous web data. It is well noted that these accuracies have been obtained using different datasets, indicating how the characteristics of these datasets affect the models' accuracy. Although communication systems can detect users'

emotional states using other communication methods, the diversity and complexity of language make it challenging for researchers to identify emotional states solely from textual data. Emotion recognition is crucial for specific text-based communication tools, such as dialogue systems, which are a type of human-machine communication system that utilizes only text input and output. Recognizing the emotional states of users enables the chat system to tailor the response accordingly (Wu et al., 2006). Thus, the results of this study can be used to understand the multiple meanings of a text document and achieve various natural language processing goals.

Conclusion

In conclusion, this study makes a significant contribution to the field of Persian natural language processing by developing a fuzzy-weighted multi-semantic labeling system. The model, designed with supervised deep learning techniques and enhanced by semantic fuzzification, addresses the unique challenges posed by the Persian language and its low-resource setting. By integrating advanced methods such as LSTM2 and word2vec representations, the system demonstrates a novel approach to multi-label classification that captures both the semantic richness and ambiguity of Persian electronic texts.

Theoretically, the study contributes to Persian NLP by extending deep learning frameworks to manage semantic fuzziness and multilabel complexity in an underdeveloped area in low-resource languages. By merging supervised LSTM architectures with fuzzy logic and decision-making criteria, the research develops a combined method that measures the likelihood of a document belonging to multiple topic groups. This model advances the field of semantic web applications for Persian by moving beyond deterministic or rule-based keyword extraction. The use of context-aware word embeddings (word2vec) in tandem with deep sequential models underscores a shift toward meaning-based, rather than lexical, document representations in Persian. From a practical standpoint, the system developed in this study can benefit Persian-language digital content providers, search engines, and recommender systems. Multi-label semantic tagging enhances the granularity and relevance of search results, supporting improved content retrieval and classification across various domains, including news portals, academic repositories, and e-commerce platforms.

The fuzzified label weights offer an additional benefit by providing a ranked semantic view of document content. This feature is particularly valuable in scenarios where users' expectations vary and documents are inherently multi-thematic. It enhances user retention by increasing the relevance of served content and can directly influence metrics tied to monetization and SEO optimization. The approach also supports emotion-aware applications by aligning with NLP models designed for user sentiment extraction from textual input, thereby contributing to human-machine interaction platforms, such as chatbots. Despite its contributions, the study has several limitations. The lack of a standardized Persian dataset for multi-label classification poses a significant constraint on model benchmarking and generalization. Although the current dataset was derived from real-world web content, its topical and stylistic variability likely influenced the model's performance. The LSTM2 model also suffered from overfitting during training, necessitating the use of dropout and early stopping mechanisms to ensure generalizability.

Future research should consider expanding the dataset and applying cross-domain validation techniques to enhance the accuracy of the results. Additionally, exploring Generative

Adversarial Networks (GANs) may offer advantages in generating and augmenting synthetic data for Persian texts. A comparative evaluation with other foundational techniques, such as Latent Dirichlet Allocation (LDA), PositionRank, PKE, or standard MLC baselines, would further contextualize the model's strengths and weaknesses. Another promising direction involves enhancing the label weighting mechanism by integrating advanced attention-based transformers or hierarchical attention networks, which could better capture document-level semantics. Furthermore, investigating emotion recognition and sentiment detection layers on top of the multi-label classifier could broaden the scope of practical applications.

Acknowledgment

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. In addition, the authors declare that they have no competing interests.

References

Ahmadi, P., Tabandeh, M., & Gholampour, I. (2016, May). Persian text classification based on topic models. In *2016, the 24th Iranian Conference on Electrical Engineering (ICEE)* (pp. 86-91). IEEE. https://doi.org/10.1109/IranianCEE.2016.7585495

Altınel, B., Ganiz, M. C. & Diri, B. (2015). A corpus-based semantic kernel for text classification by using the meaning values of terms. *Engineering Applications of Artificial Intelligence*, 43, 54-66. https://doi.org/10.1016/j.engappai.2015.03.015

Aghighi, R. & Bashiri, H. (2025). Text classification of Persian documents with deep learning. In *Advanced Interdisciplinary Applications of Deep Learning for Data Science* (pp. 143–170). IGI Global. https://doi.org/10.4018/979-8-3693-4759-1.ch006

Ashtiani, M. N. & Raahemi, B. (2023). News-based intelligent prediction of financial markets using text mining and machine learning: A systematic literature review. *Expert Systems with Applications*, 217, 119509. https://doi.org/10.1016/j.eswa.2023.119509

Azarafza, M., Feizi-Derakhshi, M. R. & Shendi, M. B. (2020, March). TextRank-based microblogs keyword extraction method for the Persian language. In *Proceedings of the 3rd International Congress on Science and Engineering (Hamburg, Germany)*. Retrieved from https://www.researchgate.net/publication/338533840_TextRank-based_Microblogs_Keyword_Extraction_Method_for_Persian_Language/link/5e19d387a6fdcc28376b9525/download?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19

Bezdek, J. C. (1981). *Pattern recognition with fuzzy objective function algorithms*. Plenum Press. https://doi.org/10.1007/978-1-4757-0450-1

Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7), 107–117. https://doi.org/10.1016/S0169-7552(98)00110-X

Cejuela, J. M., Bojchevski, A., Uhlig, C., Bekmukhametov, R., Kumar Karn, S., Mahmuti, S., Baghudana, A., Dubey, A., Satagopam, V. P., & Rost, B. (2017). nala: Text mining natural language mutation mentions. *Bioinformatics*, 33(12), 1852–1858. https://doi.org/10.1093/bioinformatics/btx083

Chen, X., Zou, D., Cheng, G., Liu, Y. & Xie, H. (2024). Deep neural networks for the automatic understanding of the semantic content of online course reviews. *Education and Information Technologies*, 29(4), 3953–3991. https://doi.org/10.1007/s10639-023-11980-6

Cheng, Q. & Shi, W. (2025). Hierarchical multi-label text classification of tourism resources using a label-aware dual graph attention network. *Information Processing & Management*, 62(1), 103952. https://doi.org/10.1016/j.ipm.2024.103952

Chollet, F. (2017). *Deep learning with Python.* Simon and Schuster.

Chopra, S., Auli, M. & Rush, A. M. (2016, June). Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 conference of the North American chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 93-98). https://doi.org/10.18653/v1/N16-1012

Dang, J., Kalender, M., Toklu, C. & Hampel, K. (2017). *Semantic search tool for document tagging, indexing, and search*. US Patent 9,684,683.

Dastgheib, M. B. & Koleini, S. (2019). Persian text classification enhancement by latent semantic space. *International Journal of Information Science and Management (IJISM)*, 17(1), 33-46. Retrieved from https://ijism.isc.ac/article_698289_45b04d125578f24c0ca1b3f1e06e346e.pdf

Dastgheib, M. B., Koleini, S., & Rasti, F. (2020). The application of deep learning in Persian document sentiment analysis. *International Journal of Information Science and Management (IJISM)*, 18(1), 1-15. https://dor.org/20.1001.1.20088302.2020.18.1.1.0

Davari, N., Mahdian, M., Akhavanpour, A. & Daneshpour, N. (2020, August). Persian Document Classification Using Deep Learning Methods. In *202, the 28th Iranian Conference on Electrical Engineering (ICEE)* (pp. 1-5). IEEE. https://doi.org/10.1109/ICEE50131.2020.9260650

Deng, L., Tur, G., He, X. & Hakkani-Tur, D. (2012, December). Use of kernel deep convex networks and end-to-end learning for spoken language understanding. In *2012, IEEE Spoken Language Technology Workshop (SLT)* (pp. 210-215). IEEE. https://doi.org/10.1109/SLT.2012.6424224

Farahani, B. D., Fatemi, S. O., & Ghorbani, M. (2019, April). Automatic keyphrase extraction from Persian scientific documents using semantic relations. In *2019, the 27th Iranian Conference on Electrical Engineering (ICEE)* (pp. 1972-1978). IEEE. https://doi.org/10.1109/iraniancee.2019.8786696

Gattiker, J. R., Hamada, M. S., Higdon, D. M., Schonlau, M. & Welch, W. J. (2016). Using a Gaussian process as a nonparametric regression model. *Quality and Reliability Engineering International*, 32(2), 673-680. https://doi.org/10.1002/qre.1782

Ghasemi, S. & Jadidinejad, A. H. (2018, April). Persian text classification via character-level convolutional neural networks. In *2018, the 8th Conference of AI & Robotics and the 10th RoboCup Iran Open International Symposium (IRANOPEN)* (pp. 1-6). IEEE. https://doi.org/10.1109/RIOS.2018.8406623

Hadifar, A. & Momtazi, S. (2018). The impact of corpus domain on word representation: a study on Persian word embeddings. *Language Resources and Evaluation*, 52(4), 997-1019. https://doi.org/10.1007/s10579-018-9419-x

Johnson, R. & Zhang, T. (2014). Effective use of word order for text categorization with convolutional neural networks. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 103–112, Denver, Colorado. Association for Computational Linguistics* (pp. 103-112). Association for Computational Linguistics. https://doi.org/10.3115/v1/N15-1011

Kim, J. & Lee, M. (2014, November). Robust lane detection based on a convolutional neural network and random sample consensus. In *International Conference on Neural Information Processing* (pp. 454-461). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-12637-1_57

Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. https://doi.org/10.48550/arXiv.1412.6980

Khuntia, M. & Gupta, D. (2023). Indian news headlines classification using word embedding techniques and an LSTM Model. *Procedia Computer Science*, 218, 899-907. https://doi.org/10.1016/j.procs.2023.01.070

Khurana, D., Koli, A., Khatter, K. & Singh, S. (2023). Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*, 82(3), 3713-3744. https://doi.org/10.1007/s11042-022-13428-4

Kou, W., Li, F. & Baldwin, T. (2015). Automatic labelling of topic models using word vectors and letter trigram vectors. In *Information Retrieval Technology: 11th Asia Information Retrieval Societies Conference, AIRS 2015* (pp. 253–264). Springer. https://doi.org/10.1007/978-3-319-28940-3_20

Lazemi, S., Ebrahimpour-Komleh, H. & Noroozi, N. (2019). PAKE: A supervised approach for Persian automatic keyword extraction using statistical features. *SN Applied Sciences*, 1,1574. https://doi.org/10.1007/s42452-019-1627-5

Leeson, W., Resnick, A., Alexander, D. & Rovers, J. (2019). Natural Language Processing (NLP) in Qualitative Public Health Research: A Proof of Concept Study. *International Journal of Qualitative Methods*, 18. https://doi.org/10.1177/1609406919887021

Li, X., Liu, J., Wang, X. & Chen, S. (2024). A survey on incomplete multi-label learning: Recent advances and future trends. https://doi.org/10.48550/arXiv.2406.06119

Lin, B. (2022). Knowledge management system with NLP-assisted annotations: A brief survey and outlook. https://doi.org/10.48550/arXiv.2206.07304

Liu, J., Chang, W. C., Wu, Y. & Yang, Y. (2017, August). Deep learning for extreme multi-label text classification. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 115-124). https://doi.org/10.1145/3077136.3080834

Liu, Z., Li, P., Zheng, Y. & Sun, M. (2009, August). Clustering to find exemplar terms for key phrase extraction. In *Proceedings of the 2009 conference on empirical methods in natural language processing* (pp. 257-266). Retrieved from https://aclanthology.org/D09-1027.pdf

Masoudian, S., Derhami, V. & Zarifzadeh, S. (2019, April). Hierarchical Persian text categorization in the absence of labeled data. In *2019, the 27th Iranian Conference on Electrical Engineering (ICEE)* (pp. 1951-1955). IEEE.

Meesad, P. & Li, J. (2014, December). Stock trend prediction relying on text mining and sentiment analysis with tweets. In *2014, the 4th World Congress on Information and Communication Technologies (WICT 2014)* (pp. 257-262). IEEE. https://doi.org/10.1109/wict.2014.7077275

Mei, Q., Shen, X. & Zhai, C. (2007, August). Automatic labeling of multinomial topic models. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 490-499). https://doi.org/10.1145/1281192.1281246

Mikolov, T., Yih, W. T. & Zweig, G. (2013, June). 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). Retrieved from https://aclanthology.org/N13-1090.pdf

Paik, J. H. (2013, July). A novel TF-IDF weighting scheme for effective ranking. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 343-352). https://doi.org/10.1145/2484028.2484070

Pan, P., Swaroop, S., Immer, A., Eschenhagen, R., Turner, R., & Khan, M. E. E. (2020). Continual deep learning by functional regularisation of memorable past. *Advances in Neural Information Processing Systems*, 33, 4453-4464.

Pawar, D., Phansalkar, S., Sharma, A., Sahu, G. K., Ang, C. K. & Lim, W. H. (2023). Survey on the biomedical text summarization techniques with an emphasis on databases, techniques, semantic approaches, classification techniques, and similarity measures. *Sustain.* 15(5), 4216. https://doi.org/10.3390/su15054216

Roy, S., Das, N., Kundu, M. & Nasipuri, M. (2017). Handwritten isolated Bangla compound character recognition: A new benchmark using a novel deep learning approach. *Pattern Recognition Letters*, 90, 15-21. https://doi.org/10.1016/j.patrec.2017.03.004

Rajabi, E., Sahebari, M. & Thomas, T. (2022). Analyzing systemic lupus erythematosus publications using neural network–based multi-label classification algorithms. *Lupus*, 31(7), 820-827. https://doi.org/10.1177/09612033221093548

Saaty, T. L. (1988). What is the analytic hierarchy process? In Mitra, G., Greenberg, H. J., Lootsma, F. A., Rijkaert, M. J., Zimmermann, H. J. (eds) *Mathematical Models for Decision Support*. NATO ASI Series, vol 48. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-83555-1_5

Schoene, A. M., Basinas, I., van Tongeren, M. & Ananiadou, S. (2022). A Narrative Literature Review of Natural Language Processing Applied to the Occupational Exposome. *International Journal of Environmental Research and Public Health*, 19(14), 8544. https://doi.org/10.3390/ijerph19148544

Sharifi, A. & Mahdavi, M. A. (2019). Supervised approach for keyword extraction from Persian documents using lexical chains. *Signal and Data Processing*, 15(4), 95-110. http://dx.doi.org/10.29252/jsdp.15.4.95 [in Persian]

Soloshenko, A. N., Orlova, Y. A., Rozaliev, V. L. & Zaboleeva-Zotova, A. V. (2015). Establishing the semantic similarity of the cluster documents and extracting key entities in the problem of the semantic analysis of news texts. *Modern Applied Science*, 9(5), 246-268. http://dx.doi.org/10.5539/mas.v9n5p246

Sorodoc, I., Lau, J. H., Aletras, N. & Baldwin, T. (2017, April). Multimodal topic labelling. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers* (pp. 701-706). Retrieved from https://aclanthology.org/E17-2111.pdf

Sullivan, F. R. & Keith, P. K. (2019). Exploring the potential of natural language processing to support microgenetic analysis of collaborative learning discussions. *British Journal of Educational Technology*, 50(6), 3047–3063. https://doi.org/10.1111/bjet.12875

Sun, Y., Sun, H. & Cheng, R. (2016, April). Fast and semantic measurements on collaborative tagging quality. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 363-375). Cham: Springer International Publishing. Cham: Springer International Publishing. http://dx.doi.org/10.1007%2F978-3-319-31750-2_29

Swayamdipta, S., Thomson, S., Lee, K., Zettlemoyer, L., Dyer, C., & Smith, N. A. (2018). Syntactic scaffolds for semantic structures. *arXiv preprint arXiv:1808.10485*

Tai, K. S., Socher, R. & Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers),* (pp. 1556-1566). Beijing, China. Association for Computational Linguistics. https://doi.org/10.3115/v1/P15-1150

Tarekegn, A. N., Ullah, M. & Cheikh, F. A. (2024). Deep learning for multi-label learning: A comprehensive survey. https://doi.org/10.48550/arXiv.2401.16549

W3Techs (2023). Usage statistics of content languages for websites. Retrieved from https://w3techs.com/technologies/overview/content_language

Wang, J., Chen, Y., Hao, S., Peng, X. & Hu, L. (2019). Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 119, 3-11. https://doi.org/10.1016/j.patrec.2018.02.010

Wu, C. H., Chuang, Z. J. & Lin, Y. C. (2006). Emotion recognition from text using semantic labels and separable mixture models. *ACM transactions on Asian language information processing (TALIP)*, 5(2), 165-183. https://doi.org/10.1145/1165255.1165259

Xie, J., Deng, Q., Xia, S., Zhao, Y., Wang, G., & Gao, X. (2023). Research on an Efficient Fuzzy Clustering Method Based on Local Fuzzy Granules. https://doi.org/10.48550/arXiv.2303.03590

Xiong, H., Jin, K., Liu, J., Cai, J. & Xiao, L. (2023, May). Deep learning-based image text processing research. In *202, the IEEE 9th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing (HPSC), and IEEE Intl Conference on Intelligent Data and Security (IDS)* (pp. 163-168). IEEE. https://doi.org/10.1109/BigDataSecurity-HPSC-IDS58521.2023.00037

Yacob, F. & Semere, D. (2021). A multilayer shallow learning approach to variation prediction and variation source identification in multistage machining processes. *Journal of Intelligent Manufacturing*, 32(4), 1173-1187. https://doi.org/10.1007/s10845-020-01649-z

Yan, Y., Wang, Y., Gao, W. C., Zhang, B. W., Yang, C. & Yin, X. C. (2018). LSTM^2: Multi-label ranking for document classification. *Neural Processing Letters*, 47, 117-138. https://doi.org/10.1007/s11063-017-9636-0

Yu, S., Li, X., Zhao, X., Zhang, Z. & Wu, F. (2015). Tracking news article evolution by dense subgraph learning. *Neurocomputing*, 168, 1076–1084. https://doi.org/10.1016/j.neucom.2015.05.077

Za'in, C., Pratama, M., Lughofer, E. & Anavatti, S. G. (2017). Evolving type-2 web news mining. *Applied Soft Computing*, 54, 200-220. https://doi.org/10.1016/J.ASOC.2016.11.034

Zha, D. & Li, C. (2019). Multi-label dataless text classification with topic modeling. *Knowledge and Information Systems*, 61(1), 137-160. https://doi.org/10.1007/s10115-018-1280-0

Zhang, R., Lee, H. & Radev, D. (2016). Dependency-sensitive convolutional neural networks for modeling sentences and documents. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, (pp.1512–1521). San Diego, California. Association for Computational Linguistics. https://doi.org/10.18653/v1/N16-1177

Zheng, J. & Zheng, L. (2019). A hybrid bidirectional recurrent convolutional neural network attention-based model for text classification. IEEE Access, 7, 106673-106685. https://doi.org/10.1109/ACCESS.2019.2932619