IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 26, 2021, accepted February 8, 2021, date of publication February 12, 2021, date of current version March 1, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3059312

A Review of Text-Based Recommendation Systems

SAFIA KANWAL, SIDRA NAWAZ, MUHAMMAD KAMRAN MALIK, AND ZUBAIR NAWAZ
Punjab University College of Information Technology (PUCIT), Lahore 54590, Pakistan

Corresponding author: Safia Kanwal (safiakanwal2006@gmail.com)

ABSTRACT Many websites over the Internet are producing a variety of textual data; such as news, research articles, ebooks, personal blogs, and user reviews. In these websites, the textual data is so large that the process of finding pertinent information by a user often becomes cumbersome. To overcome this issue, "Text-based Recommendation Systems (RS)" are being developed. They are the systems with the capability to find the relevant information in a minimal time using text as the primary feature. There exist several techniques to build and evaluate such systems. And though a good number of surveys compile the general attributes of recommendation systems, there is still a lack of comprehensive literature review about the text-based recommendation systems. In this paper, we present a review of the latest studies on text-based RS. We have conducted this survey by collecting literature from preeminent digital repositories, that was published during the period 2010-2020. This survey mainly covers the four major aspects of the textual based recommendation systems used in the reviewed literature. The aspects are datasets, feature extraction techniques, computational approaches, and evaluation metrics. As benchmark datasets carry a vital role in any research, publicly available datasets are extensively reviewed in this paper. Moreover, for text-based RS many proprietary datasets are also used, which are not available in the public. But we have consolidated all the attributes of these publically available and proprietary datasets to familiarize these attributes to new researchers. Furthermore, the feature extraction methods from the text are briefed and their usage in the construction of text-based RS are discussed. Later, various computational approaches that use these features are also discussed. To evaluate these systems, some evaluation metrics are adopted. We have presented an overview of these evaluation metrics and diagramed them according to their popularity. The survey concludes that Word Embedding is the widely used feature selection technique in the latest research. The survey also deduces that hybridization of text features with other features enhance the recommendation accuracy. The study highlights the fact that most of the work is on English textual data, and News recommendation is the most popular domain.

INDEX TERMS Recommendation systems, review of recommendation system, text-based recommendation system.

I. INTRODUCTION

The advancements in digital technology, especially after the introduction of smartphones, have exploded online data tremendously. Social sites such as Facebook and Twitter are significant sources of data generation. Moreover, question answering sites, such as Quora and Stack overflow, are also adding data swiftly in this pool. Furthermore, the number of publications is increasing enormously over the last few years and so the trend of personal blogging. The advent of digitalisation has also affected the life of users in both good and bad ways. The good part is that the data is easily and instantly available. Whereas, the bad part is that the abundance of data has made it difficult for users to find the most relevant and required information in a short time. To address this problem of finding the most relevant information from the overloaded pool of data, recommendation systems have been developed [1]. It is a unique set of tools and techniques that give suggestions to a user about specific items that can be of their interest. A recommendation system keeps track of a customer's profile and based on their interest, suggests a product or a service [2]. These suggestions can occur in any domain ranging from web-service to use in software to a news article to read [3]–[5]. The problem of recommendation is twofold, i.e.,(i) estimating the value of prediction for an item (ii) ranking these items by their prediction value. And there are various types of approaches to achieve this task. Most popular among them are Collaborative filtering(CF) and

The associate editor coordinating the review of this manuscript and approving it for publication was Emre Koyuncu.

Content-based filtering (CB) [1]. In CF, the recommendation system suggests a new item to a user that is consumed by similar users. Whereas, in the case of CB, the recommendation system recommends a new item based on their content and attributes. In other words, the recommended item will have similar attributes to the previously consumed items by the same user. However, recently, both these techniques are used in a hybrid fashion and have shown promising results. As the data can be of any type, such as; images, text, numbers, etc; The recommendation system could be built for any type of data. Among the available types of data, the textual data constitutes a considerable portion. For Instance, news, research articles, blogs, and different kinds of reports are the primary sources of this textual content. The abundance of textual data poses similar challenges, as discussed earlier for data in general. Therefore, researchers have also worked on recommendation systems for textual data to find the most relevant text. Some domains of textual recommendation are news recommendation, article/blog recommendation, book/movie recommendation etc. Researchers are working on these textual based domains [6] over the last few decades. Each domain suffers from its own sets of challenges [7]. Hence, somewhat different techniques are used in each domain. So it is highly essential to have an overview of all such work. Recently, deep learning models have replaced traditional algorithms in almost every domain of computer sciences. Likewise, the field of recommendation systems has also adopted this new trend. There is a wide variety of neural network architectures that are used for textual recommendations [8]–[10]. This survey also presents an overview of the latest techniques employed in the case of textual recommendation systems. Moreover, the evaluation mechanisms for textual recommendation systems are also compiled in this article. Mostly traditional evaluation metrics used for RS [11], are employed from the field of information retrieval. But some unaccustomed metrics like specificity and diversity are also being used in textual RS [12].

A. RESEARCH OBJECTIVES

Following are the key objectives of this study.

- One of the primary research objectives is to lay a foundation of subsequent studies on text-based Recommendation Systems, which, has become an important research paradigm. The article will highlight the achievements made in the context of textual data recommendations.
- To develop an understanding of how to extract useful features using different techniques for building a text-based recommendation system
- To compile evaluation metrics used in the evaluation of text-based recommendation systems.
- To give an overview of attributes found in the datasets used in textual RS.

The motivation behind this study is to give a jump start to the researchers, who want to pursue their research in the field of textual recommendation systems. That's why the study not only covers the feature selection techniques used in textual recommendation systems but also gives an overview of the various datasets used in this domain. The rest of this paper is organised as follows. Section II presents a detailed methodology to conduct this study. In section III, existing surveys about text-based RS are discussed. The next section talks about the various methods and sources used in literature to extract features from textual data. In section V, the algorithmic approaches for recommendations are compiled in detailed. Section VI gives the overview of the datasets in terms of their attributes and volume. Section VII, briefs the evaluation metrics used in textual RS. And in the last Section VIII overall summary of this work is presented the findings are discussed.

II. METHODOLOGY

For this study, a literature review is conducted as a primary task. The process involved the searching for relevant research papers. For this purpose keywords and phrases are used on famous repositories including ACM, IEEE, DBLP and Google Scholar. Firstly, the identified keywords are used along with their synonyms to find the relevant literature. The basic set of keywords are *"content-based recommendation system", "News recommendation", "textual recommendation", "user reviews recommendation", "document recommendation", "articles recommendation"*. Secondly, the papers which were based on ranking techniques or using other content such as video and images were omitted. Afterwards, the papers were selected on the following criteria orderly:

- Only Publications from 2010-2020 are considered.
- To overcome the language barrier, only English literature is considered
- Only those studies are included that are either recommending textual content or textual content is used as the auxiliary source to recommend other items.
- The studies which recommend the textual item but doesn't consider their attributes instead use rating or other features for recommendation are not included in the study.

After acquiring the relevant literature; selected articles were studied in detail, and the following information is extracted for all the selected articles.

- The approach of feature selection and algorithms were listed.
- Later, the attributes of the datasets used are collected from their source publications.
- Afterwards, all the research findings against this study are consolidated to present the current state in this domain.

Currently, deep learning based approaches are being widely used to solve the problem of textual recommendation system. Therefore, this survey also discusses the details about deep learning techniques used in textual recommendation systems.

The complete process of filtering is given in Figure1.

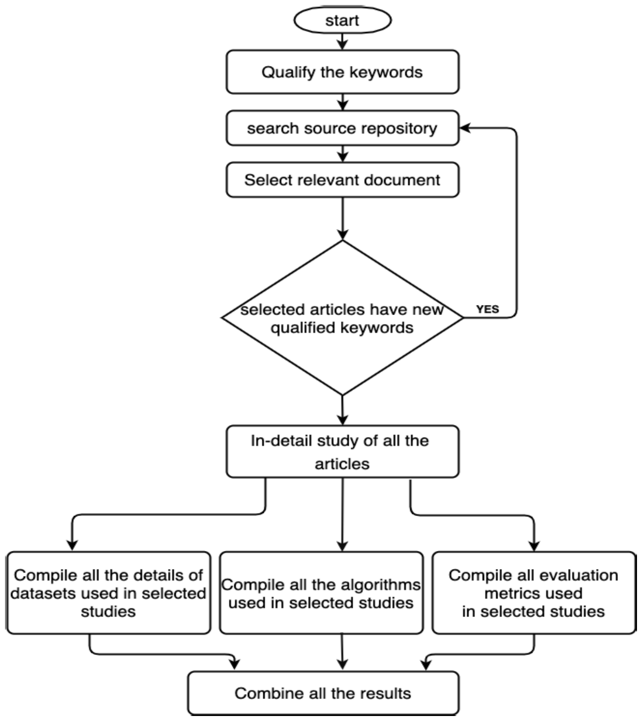


FIGURE 1. Detailed methodology.

III. EXISTING SURVEYS

Many surveys are covering the different aspects of Recommendation Systems. Some of them are general, and some are specific to a particular subject; like Chen *et al.* [13] summarizes all the Recommendation Systems that use user reviews.

In another survey [14] recent trends in content-based RS are reviewed,, where they presented a brief history of content-based recommendation systems; the latest trends were described in terms of data and algorithms. In terms of data, the survey argued that usage of Link Open Data(LOD) is being employed nowadays to get the extra meta-data features of an item. Moreover, more content is gathered from a forum, user reviews, and tags; they categorised it as User Generated Data (UGD), visual and multimedia features, and heterogeneous information networks are also discussed as a source of taking content. In the terms of algorithms, the following approaches were highlighted; Meta-path based; in which a path is described corresponding to a relation between two entities, new metadata encoding; word and doc embeddings to identify latent features of the text, and Deep learning techniques.

Another study [15] specifically focused on those RS, that employed ontologies for the development of e-learning RS. At the start of the survey, it briefly discussed all the possible types of RS. After providing a detailed description of ontologies and e-learner systems, the writer summarises all the papers in hand in terms of ontologies used, ontology representation language, and recommended learning resources.

In another survey [13], a detailed study was carried on about the Recommendation System entirely based upon user reviews(UR) or their efficiency has been improved using UR. In the first part of the survey, the general introduction of RS was given including its basic techniques (Content-based, Rating-based collaborative filtering, Preference-based product ranking). The second part of the survey discussed the elements (Frequent terms, Review topics, Feature opinions, Contextual opinions, Comparative opinions, Review emotions, Review helpfulness) of the reviews used in RS. In the next two sections of the survey, all those studies were discussed in detail who used UR as user profiling and product profiling for recommendations, and in the last, they talked about the practical implication of their findings for five dimensions: data condition, new users, algorithm improvement, profile building, and product domain. There is one such recent study that captures the details about text mining techniques in different recommendation systems [16].

As recently the trend is shifted towards deep learning even in the field of RS and almost every one using it to build the RS for its proven, and promising results. A survey om deep-learning techniques used in RS [17] compiled all these studies. Although the paper's whole sole focus is deep learning techniques and it doesn't directly mention the usage of text-based RS, but it does includes all the work done on textual data using DL approach. Deep-learning techniques used in RS are also compiled in a study [18] presented by Batmaz *et al.* It briefly describes the development techniques, whereas puts more emphasis on the issues and challenges that can be addressed by using DL. Moreover, the domains in which these models are adopted are also presented in a structured form.

There are some domain-specific studies as well, like the study presented by [7] Beel *et al.*covers all the studies about research paper recommendations in the period of 1998 - 2013 and claims that there are more than 200 articles published about this particular topic. It doesn't give any insight into the details of algorithms used to extract textual features. Similarly, the study [19] presented by Mozhgan *et al.* is focused about on News Recommendations only. Although the text-based approaches covered in this is very concise and short, yet it does give an overview of the trend people working on News RS. The study gives an outline of Algorithmic approaches, challenges, and evaluation metrics of News RS. The shortened summary of popular and publicly available data sets are also included in this work. Another Study [20] enlists the challenges and methods of the news domain, however, it hardly mentions any particular design on pure textual data, rather it's more general.

A. SUMMARY OF SURVEYS

As described above, there are a variety of surveys and state of the art studies that compile and present the previous work from a different perspective. Here our main focus of studying all those which are based on textual data and talks about textual data recommendation in one way or other, partially

TABLE 1. Summary of existing surveys.

Study	Major Focus	Coverage	SP	AS	ST	AT
[13]	User Reviews	not specified	N	N	C	C
[14]	Content base	2000-2019	N	N	N	P
[15]	Ontology based RS for e-learning systems	2005-2014	C	C	C	P
[17]	Deep Learning Techniques used in RS	2005-2014	C	C	P	P
[18]	challenges and remedies of RS using Deep Learning	2013-2017	P	C	C	P
[19]	News Recommendation Systems	2006-2015	C	C	C	P
[16]	Text mining techniques in recommendation system	not specified	N	C	C	C

or completely. We ignored those studies which were more general.

SP: Survey Process (Methodology)

AS: Articles Selection, articles are searched and gathered with some strategy

ST:Summary Table: Summary of analysis is presented

AT: About Text Only

C: Completely / Yes

P: Partial

N: Not present at all

IV. FEATURE SELECTION APPROACHES

In textual recommendation systems, various Natural Language Processing (NLP) techniques are employed to extract the useful features from the text. These techniques range from simple keywords-based techniques; where keywords are used as features, to the complex deep learning techniques, where a complex neural network is built to extract the hidden, latent features in the text. In this section, these feature extraction techniques are discussed in detail.

A. KEY WORD BASED

In the simplest form of text-based features, keywords are extracted from the text to present an item or user's profile. After parsing and cleaning data; the process of keyword extraction follows the necessary steps of NLP like tokenisation, stop word removal and stemming. For instance, words like "computing", "computer" and "computers" could all be saved as their root word i.e. "compute". The keywords are then converted to numerics using the following methods:

1) TF-IDF

There are various approaches to present these keywords/terms. One fundamental way is the Vector Space Model (VSM). In VSM, an item's content is presented as an m-dimensional vector where m is the total numbers of terms extracted from the text. Each position in the vector specifies the weight of a term corresponding to that item or user and is calculated with a basic TF-IDF weighting scheme.

$$wi = tf_i.log[\frac{n}{If_i}] \tag{1}$$

where tf_i is total occurrences in the description of an item

$n =$ total no of items in the collection

$If_i =$ number of items whose description contains term t at least once

Although this is a fundamental technique of content-based filtering, and researchers have been using it for quite a long time. Yet it is equally popular in recent studies as well.

Given the title and abstract of articles, a Jain *et al.* [21] used tf-idf technique for profiling and SVD to reduce matrix sparsity. News recommendation systems discussed in studies [22], [23] are built using tf-idf techniques as well. In these studies, the scientists created a user profile based on various topics as keywords. A vector space model based on keywords was maintained for each topic separately. The score of keywords shows the interest rate of users for a particular news item. Another study [24] is using the tf-idf similarity to recommend the social tags. TF-IDF is also used as a base technique for recommendation along with other factors of manipulation; for example MAPS [25] used it as a base along with time and distance information to recommend the Point of Interest-based on the previous history of check-ins, in location-based social networks.

An interesting variation of TF-IDF is $TF - ID_uF$ [26]. In $TF - ID_uF$ the considered document frequencies are the ones that are present only in a user's personal documents instead of full corpus. If t is the term to weight, tf(t) is the frequency of given term t in the documents, the document collection for a user is c_u and the number of documents is N_u then the formula to compute the $TF - ID_uF$ would be as follow

$$TF - ID_uF = tf(t) * log\frac{N_u}{n_u} \tag{2}$$

where n_u is the number of documents in c_u that contain t

2) BM25TF-IDF

BM25 stands for "Best Match 25". It is the 25th iteration of tweaking the relevance computation and has its ground in probabilistic information retrieval.[1] BM25IDF is similar to IDF but they added 1 to the value, before taking the log, which makes it impossible to compute a negative value. And Tf is modified by considering the length of the document. The length L is used with the relevance of the average document

[1]https://opensourceconnections.com/blog/2015/10/16/bm25-the-next-generation-of-lucene-relevation/

length of a given corpus. It also has a tweaking factor of k whose by default value is set to 1.2, and a higher value of k causes TF to take longer to reach saturation.

$$IDF * ((k+1) * tf)/(k * (1.0-b+b * (|d|/avgDl))+tf) \tag{3}$$

The same approach is used social tags for the evaluation of different information retrieval models for Recommendation task [24].

B. LETTER BASED

This technique is an extension of the bag of words(BOW) technique, But instead of words, letters, or combinations of letters(bi-gram, tri-gram) are taken as input features. The primary advantage of this technique is to limit the vocabulary size, which becomes infinite in the case of the BOW. In a study [27], researchers proposed word hashing technique in which tri-grams of words are used as a feature instead of a full word. So, for example, a word "good" would be represented as /go,goo,odd,od/. With the experimentation, they have further reduced the data set of the vocabulary of 40k to the token size of 10306 and vocabulary of 500k to the token size of 30621. Later couple of studies [28], [29] adopted their technique in their deep learning models.

C. SEMANTIC FEATURES

The recommendation systems that are built on keywords-only features, as discussed earlier, give a good start for recommendations. However, they are not capable of exploiting the semantic attributes of text; which could be an essential source to understand user behaviour and recommend useful items. In this section, we will summarize the efforts made by researchers to explore this domain.

1) WordNet - ConceptBased

Synset Frequency – Inverse Document Frequency (SF-IDF) works the same as TF-IDF but instead of term, [30] used a synset in which two words sharing their meaning would be considered one term. This work is extended by incorporating the relationships like (Member meronym, Attribute, Domain of this synset - Region, Cause, Derivationally related form) among different sets of synonyms [31]. NLP techniques are used to extract all the synsets from the unread news items and the concepts that represent the semantical relationship among these synsets are included to extend this set.

This search is further extended by incorporating name entity recognition. User profile and unread news are used to construct a vector, that contains, all possible pairs of named entities. A similarity score is computed by taking the average of normalised similarity scores of SF-IDF+ [32].

2) KNOWLEDGE/Ontology-BASED

Usage of ontologies is still a good source of semantics features as in this scheme. A vector of features is constructed on concepts like entities, attributes of entities, named-entities, or a combination of a few or all. This technique is called Concept Frequency - Inverse Document Frequency (CF-IDF) [33] is based on Word-Net that extracts the concepts based on domain ontologies. To obtain these concept ontologies, various NLP techniques such as tokenisation, part-of-speech tagging, and word sense disambiguation are used. Each news item is represented as a concept vector, and its cosine similarity with the user's concept vector is computed for the recommendation. CF-IDF+ [34] is an extension of CF-IDF and it identifies whether a concept is a class or an individual before computing the related concepts. These related concepts are computed by multiplying the weights of original concepts with the weight of relationships among them. For more semantic interpretation, NER information is incorporated [35].

All the work mentioned above is a series of efforts put in the path of the news recommendation system and is implemented in the Hermes News Portal which has two extensions, Athena [36] and Ceryx [30]. Instead of using the Hermes news portal, Abel *et al.* [37] gathers a large number of Twitter data and make user profiles on three different types; hash tag-based, entity-based, and topic modeling-based. Similarly for news recommendation, Ferdous and Ali [38] employed the semantic-web pipeline and categorized the entities using Gazetteer. Entities appearing in the title are called Concepts.

Obeid *et al.* [39] proposed a system to recommend majors, universities, and employment. In this case, ontologies are of three types; higher education institution, students, and employment. Advertisements recommendation system proposed in a study [40] have interest ontologies that are comprised of the main concepts and the relationships among them. Another interesting study [41] compared the effectiveness of four distinct knowledge-based strategies; Tagme, Explicit Semantic Analysis(ESA), Babelfy and Distributional Lesk-Word Sense Disambiguation and Entity Linking (DL-WSDEL). They exploit different knowledge sources to build concept-based representations and to provide cross-lingual recommendations.

D. CHI-SQUARE FEATURE SELECTION

In feature selection, one's aim is to find the features which are highly dependent on the response. chi-squared test is the statistical test to measure the effectiveness of categorical data.This methodology is adopted to enhance effectiveness of RS for recommending computer science publications [42].

E. TOPIC MODELING

Topics refer to a word or couple of words used as heading to specific text. Sometimes data is given, such as that couples of lines are related to one topic or in some other scenarios, "Topic Modeling" algorithms are employed to form such arrangements of data. The vector of features is then represented as these topics. Following are such scenarios. The random walk with restart algorithm is used by Tang *et al.* [43], to find the association of words to the defined topics to find cross-domain publication recommendations.

Another study [44] used Latent Dirichlet allocation (LDA) based model to find the estimated topical unexpectedness. They have achieved this by considering the following factors into their model; "Word Specificity", "Topic Dissimilarity" and "Limitation of Small Topics' Influence" He and Tan [45] solves the problem of recommending interesting topics in SINA blogs, in case a user has not checked it timely. They used a k-score technique on a semantic network to extract the topic of interest. They then used these topics to compute to build up the user-influence model. Moreover number of followings, followers, authentications, likes, forwards, comments, and tweets are used as indexes for factor analysis. Afterward, Z-score is used to normalise this data. Finally, the Tweet Recommendation model in SINA blogs combines the tweet popularity score and tweet authority score.

F. EMBEDDING

Word Embedding is a technique of presenting words into vectors of real numbers. It is first introduced by Google as Word2vec [46]. After that Facebook's fastText [47] and Stanford University's Glove [48] has become as established techniques in this regard. A similar concept has been adopted by a few other studies [49], [50] to represent a series of words in the form of vectors. In these studies, techniques proposed make a single vector of a sentence or a document.

In the case of the Recommendation System, this has been used in different forms. Ozsoy [51] has used the Word2vec algorithm to form location-based data into the vector space form, then recommendations are made based on similarities. Samarinas and Zafeiriou [52] also used trained fastext word embeddings along with other deep learning techniques for news recommendations. Similarly, in the other studies a whole document or article is embedded and then its similarity with others is found [4], [53]–[57]. Moreover, It is used as base line model of feature extraction before feeding data to neural network [13], [58]–[61]. Furthermore a knowledge graph of entities are also converted in to embeddings to further use it in deep-learning settings. TransE [62], TransH [63], TransR [64], TransD [65] and TranSparse [66] are different algorithmic techniques to convert a knowledge graph in to embedding representation. These algorithms for conversion of Knowledge Graph (KG) are later use in recommendation task [3], [5], [67]–[69].

Event2Vec is the news recommendation technique proposed by Setty and Hose [5]. This model represented news events in a network of events, entities, event types, and additional information. To learn latent features and network embeddings from this network, random biased walks like Breath-first search and depth-first search are used. Using these network embeddings, the similarity between these events is computed and events that are richly connected in the network have more relevance. As compared to the Node2Vec technique that does not differentiate between the entity node and event node in the network, this model has a better performance.

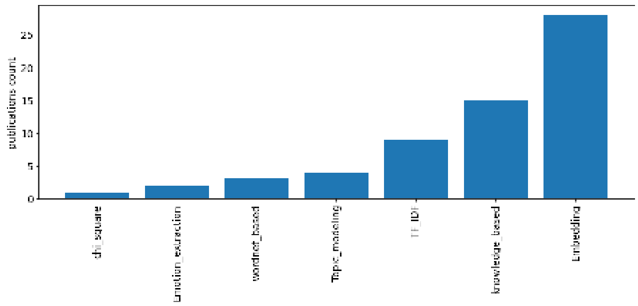


FIGURE 2. Feature selection approaches.

G. EMOTIONS AND ASPECTS EXTRACTIONS

When it comes to the rating data like movies and product recommendations, where mostly collaborative techniques are used. Rating data contains user biases and many people may have different scales of rating. For example, user A might have rated a movie three because he liked this movie and the other person B have rated the same movie three because he didn't like it much. So in this case sentiments, emotions, and other opinion aspects may add a beneficial role. For instance sentiment analysis is used to recommend a movie using movie reviews [51].A survey by Chen *et al.* [13] also concludes such techniques where this type of information is extract from user reviews to build up a recommender system.

H. FEATURE SELECTION USING DEEP LEARNING TECHNIQUES

Deep-learning is a set of techniques in which a neural network is formed with multiple layers and various settings. These hidden layers allow determining more complicated feature transformations due to the number of trainable weights and biases in the network. Recently Deep-learning techniques are extensively employed in the area of natural language processing. Recommendation Systems are now also built using these techniques. Details are in the next section.

I. SUMMARY

The figure 2 summarizes the utilization of these techniques in various studies. It is evident that feature extraction using Embedding are most popular. embedding techniques like doc2Vec, event2vec, fastext and word2Vec are used when there is a large amount of text and instead of representing the words as tokens, which are converted into vectors of real numbers. Embedding reduce the sparse vectors into low-dimensional real numbered vectors; therefore, these are widely used in a large amount of data such as text. Moreover, their intrinsic property of keeping similar items closer, are helpful in recommendation in terms of finding similar items.

Semantics features techniques like wordnet are used to extract the semantic attributes of text and topic modelling is useful when there are keywords, and some of these words can be used as a topic or heading. Recommendation systems also analyze the psychological and emotional attributes of users, and sentiment feature extraction is a technique to extract these type of features. These techniques are helpful when text is used as an auxiliary material along with the other information to boost the recommendation accuracy.

V. COMPUTATIONAL APPROACHES

In the last few years, an immense amount of work has been done on the recommendation systems(RS). Big companies like YouTube, Google, and Amazon [70] are continuously improving their recommendation systems by employing the latest techniques. While text shares a significant portion of data available over the internet, researchers are exploiting it to enhance the quality of RS. Recommendation systems, built for news or academic journals are usually entirely text-based and referred to as Content-based systems. In contrast, product suggestion systems such as Amazon and Netflix utilise item attributes and user reviews for this particular purpose. Here we will discuss the different approaches that make use of text; in one form or other, to make a recommendation.

A. PROFILE SIMILARITY APPROACH

In this particular setting the user profile and item profile is first generated using different feature selection techniques as described above. These profiles are then compared to find the best suited recommendations. There are a multiple ways of this comparison, though the cosine similarity is the most famous one. Here is a brief detail about all such similarity measures used in literature.

1) COSINE SIMILARITY

Cosine similarity measures the similarity of two documents irrespective of their sizes. Mathematically, every document is projected as a vector in multi-dimension space, and then the cosine angle is computed between two vectors. The smaller angles refers to the higher similarity

$$\cos(a, b) = \frac{ab}{\|a\|\|b\|} = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} (a_i)^2}\sqrt{\sum_{i=1}^{n} (b_i)^2}} \tag{4}$$

Most of the time in the content-based recommendation, cosine similarity is applied on a feature vector to find the most similar item. The approach is adopted in a wide range of domains, including, finding the similarity of profiles for recommending social tags [24] to recommending news [22]

Table 3 presents a detailed list of studies in which this approach is utilized.

2) LINKED DATA SEMANTIC DISTANCE

This approach is introduced in a study [71] to find the similarity of two resources based upon a property; that is two resources are more similar if there are a higher number of linked resources via a property.

3) EUCLIDEAN DISTANCE

Euclidean distance is the most common mathematical definition of distance between two points and objects; generally, when the distance is referred to as measure, it is actually meant Euclidean distance.

$$= \sqrt{\sum_{i=1}^{n} (a_i - b_i)^2} \tag{5}$$

The studies which use this approach are mentioned in Table 3.

4) JACCARD SIMILARITY

It is a simple yet intuitive measure of the similarity between the two sets. In the case of documents, it is defined as the collection of common words between two sets over the union of different words in two documents. Jaccard Similarity = number of common words in A and B / number of unique words in A and B

$$JaccardSimilarity = A \cap B / A \cup B \tag{6}$$

In the selected literature, a couple of studies [72], [73] used this measure to find similar objects to user interest.

B. MACHINE LEARNING-BASED APPROACHES

In recent years, Machine Learning is a useful tool used to solve learning problems in the field of Computer Science. The research used this approach to recommend the textual content as well. In one such study [42], softmax regression, a machine learning technique is used to recommend the journal and conferences to researchers to publish their papers by using the text of their abstract. In this study, first tf-idf is used to extract the feature set, after that Chi-square feature selection method is used for the selection of the most useful features. Then softmax regression is used for training and recommendation purposes. There are 14,012 records containing title, abstract, author, and link of papers. To ensure the records in the dataset are correct, 20 percent of abstracts from each journal and conference are checked manually. Two-thirds of all abstracts are used as training samples, and the rest is used for testing. In the experiment, papers published in 2013 and 2014 are considered. They have also published their work as a web-service.[2]

Konstantin Bauman et.al developed the Sentiment Utility Logistic Model (SULM). It is different in a way that it not only recommends the item but also finds the most relevant aspect of that item with respect to the user. It computes two things, first, the sentiments about the aspects from the reviews, and second, it finds the ratings of the reviews. Then, it finds, using Stochastic Gradient descent, such a coefficient theta that fits sentiments and ratings both.

C. NEURAL NETWORK-BASED RECOMMENDATION SYSTEMS

The earlier discussed approaches require features that are extracted manually. Artificial neural networks are an evolutionary invention in Machine learning. Taking inspiration from human neurons, researchers build an approach

[2] http://www.keaml.cn/prs/

to learn tasks using layers of the neural network without spoon-feeding features to the machine. Currently, neural networks are built with extensive layers and are known as deep neural networks and its learning technique is called deep learning. Information processing tasks such as text analysis and speech recognition have acquired excellent efficiency due to deep learning or hierarchical neural networks [74]. They are also found useful in textual recommendation system. Multi layered perceptron(MLP), Convolution Neural Network(CNN), Recurrent Neural Network(RNN) and Attention networks are some commonly used types of neural networks in the case of textual recommendation systems. The detailed description of these techniques is given below.

1) NEURAL NETWORK EMBEDDINGS

Embeddings play an essential role in machine translations. Recommendation systems that work on textual data, like news recommender, mostly use embedding techniques for a deep understanding of the text. Embedding maps discrete variables to continuous variables in the form of a vector. Neural Network embeddings can find the nearest neighbours in the embedding space based on the user interest. Chandra et.al used ''title'' and ''abstract'' sections to extract textual features and projected all the documents in such a way that a document is closed to its referenced papers. This projection of the whole corpus is embedded once as a ''documents embedding'' and later used for query documents. The query document is projected on the same vector space to find the nearest neighbours for candidate references. Afterward, Chandra *et al.* used a trained neural network on textual fields of documents to find the score of each candidate returned in phase 1. The input to this neural network is a pair (dq, di) and output is the score for each pair. In the end, the documents are ranked based on this score.

Word2Vec is a model that is used to create word embeddings. In a study [52] a modified content-based method is introduced. It combines the coverage score, the popularity factor, and the fact that if a news is click-bait or not, for the final recommendations. For content recommendations, first, the data is cleaned, then the entities are extracted from the piece of news articles, such that each entity has a weight based on their frequency. Finally, a graph is built using the entity pairs. The weighted score of an entity is calculated by the Page-Rank algorithm using this graph. They combined this score with the score they got by trained word embeddings and used it in the semantic vector representation. They used the fast-text model to train these embeddings. The similarity score for this section is computed by calculating the cosine similarity of the user's semantic vector with a candidate news article. Moreover, the classification model is trained using Bi-LSTM to detect click-bait and predict popularity. Furthermore, the coverage score cv(a) is computed which is based on the number of articles that cover the same news story (related articles in the same cluster) and the quality of their source.

Instead of using Word2vec, a study [54] proposed the usage of the Doc2Vec technique to generate embeddings on a cross-lingual dataset of uncategorized news. These embeddings are then used for content-based filtering to make recommendations. The result was then compared with LDA and LSA, and it was proven that Doc2Vec outperforms both of these topic modeling techniques in terms of accuracy. As the user data was missing, they proposed to recommend news on based on the similarity of items in a fuzzy fashion as described in another paper [75].

For research paper recommendations, embeddings are used to build a citation network. VOPRec is a technique that learns structural vectors using Struc2vec algorithm and textual vectors using Doc2vec algorithm, which is the embeddings technique. To build a weighted citation network for the recommendation, m-nearest text-based, and n-nearest structured based neighbours are connected [4]. In another such study [8], embeddings are used from textual data to extract the text-based features as well as embedding based on the features gathered from the link open data(LOD).

2) MLP

Multilayer perceptron neural networks(MLP) have at least 3 layers(input, hidden, and output) and each neuron has a non-linear activation function except for the input node. A study [76] used MLP to analyse the multi-behaviour data by correlating predictions of each behaviour type in a waterfall manner and thus capturing ordinal relations between these behaviours. To rank a course in a particular class by recognising patterns in data, another study [77] used MLP. Moreover, Chen *et al.* [78] proposed a novel location-aware topic model based news recommendation system built on multilayered perceptron (MLP). They used every Wikipedia concept as a Topic and to overcome the issues of sparsity, high dimensionality, and redundancy; they propose DL based model called deep semantic analysis (DSA), which utilises deep neural networks to map the (Wikipedia-concept-based) topic space to an abstract, dense, and low dimensional feature space, where the localised similarities between the users and their target news are maximised, and those with irrelevant news are minimised.

3) CNN

Historically, Convolution Neural Networks(CNN) are most effective in the field of image processing. Recently, it is also found useful in recommender systems as well. For example in a study [79]], researchers used text features to recommend learning sources. An LDA is applied to text to get several topics, that are made input to the CNN model, which outputs the corresponding latent factor model L1. In another study [58] a convolution neural network-based news recommender system is proposed which mainly constitutes three modules, i.e., news encoder, user encoder, and click predictor. The purpose of the news encoder is to learn news representation. It takes the title of news as its input; generates its embedding and feeds these embedding to CNN. After that, an attention model learns the essential words in the title. The user-encoder module also uses the attention mechanism to determine the user representation using its browsed news. The click predictor is to predict the probability of a user clicking a piece of candidate news. To train the whole model; the concept of negative sampling is used. All the news articles clicked by users are considered as positive, and K negative samples are taken in the same impression, which was not clicked by the user. And in this way, the news click prediction probability becomes a classification task where the loss function to compute news recommendation is the negative log-likelihood of all positive samples. This work is extended by jointly training news encoder module for topic classification task as well [59]. Xuejian Wang et.el introduced a combined learning model in which CNNs are used to learn articles representation and attention mechanismis introduced to cater to the diverse variance of editors' behavior [80].

4) RNN

Recurrent neural networks (RNN) are derived from MLP and have their internal memory. In the recommendation task, LSTM and BiLSTM are two commonly used recurrent neural networks. Usage of GRUs(Gated Recurrent Unit) based RNN model is more effective in learning expressive aggregation of user history [81]. In a recent study [81] GRUs based RNN model is used for the news recommender system of Yahoo! News. They proved that usage of GRUs is more effective in learning of expressive aggregation of user history. The distributed representation of news articles was learned using an auto-encoder in the form of embeddings. There is a significant improvement in results using this approach when compared to a word-based approach. Jizhou Huang et. al presented a multitask model that shared semantic query representation with entity recommendation and ranking [82]. This model also handles in-session and ambiguous entities using the previous search log of a real-time search engine for context analysis. For the semantic understanding of queries, BiLSTM was used. They used a neural network to map related entities and initial query into the same vector so that semantic relevance would be determined using the similarity function.

Another study [83] discussed three memory-based approaches for recommending entities. These domains are document-based, query-based, and session-based which are used in a collaborative filtering fashion to find the similarities between entities. They followed a document-based approach that observed a user's previous record to determine the relevancy of an entity to the session by comparing its similarity with all entities of the same session. The query-based approach further personalises recommendations by finding relevant bodies of the queries from all other previous sessions related to the current session. The session-based approach uses Co-occurrence similarity and Centroid session similarity to find sessions similar to the current session and recommend entities from those sessions. RNN is used to add temporal effectiveness in user profiling [28].

A novel contextual session-based recommendation was proposed in a study to recommend news and referred to as CHAMELEON [10]. It is composed of two modules the Article Content Representation (ACR) and the Next- Article Recommendation (NAR). The input of the ACR is the textual data of the news article and its meta-data features. The module is trained for the news category classification task. The internal learned embeddings during this process are used in the NAR module afterward. NAR module recommends the next article in the active session. The input of the module is learned news embedding (from the ACR module) of the last viewed article, the contextual property of news(popularity, recency), and the user's geolocation information. All these inputs are combined into the user's embeddings. The positive samples are formed by maximising the similarity between Predicted Next-Article Embedding and the User Embedding.

5) AUTO ENCODERS

Recommendation systems have high dimension data and autoencoders are neural networks that are used in dimensionality reduction of data for unsupervised learning. Autoencoders can also be used in learning embeddings of high dimensional data. For example [68] used textual, visual, and structural information and built three types of embeddings using state of the art deep learning techniques. For textual data, they used Stacked denoising autoencoders [9] to learn embeddings. Afterward, they combined all these embeddings and built a matrix, and applied the collaborative filtering technique on this embedding matrix for recommendations. Stacked Denoising autoencoders can be combined with the Stacked Denoising Autoencoders Embedding (SADE) model as the basic component to learn robust and effective representations [84]. Given a poor quality description by service developers and mashup queries it is hard to suggest useful web services. The problem is catered by [84] by developing "DLTSR"(A Deep Learning Framework for Recommendations of Long-tail Web Services) with Stacked Denoising Auto-encoders as the main building block. where they used SADE as a basic component. Moreover, to guide the learning of representations, knowledge from usages in the hot service side is imposed as a regularisation on the output of SADE.

6) DEEP RE-INFORCEMENT Learning(DRL)

As autoencoders were invented by combining neural networks with unsupervised learning, deep learning neural networks are also be combined with reinforcement learning to make Deep Reinforcement Learning(DRL). A deep reinforcement learning platform for news recommendation, DRN is to tackle the challenges of dynamic changes in news articles, incorporating user's feedback, and the challenge of increasing useful diversity in recommendations. [85].

7) DEEP LEARNING-BASED KNOWLEDGE GRAPHS RECOMMENDATION SYSTEMS

Deep learning neural networks can also be used to extract complicated relations from existing knowledge graphs (KG) like DBpedia and Wikidata to further use in the task of recommendation. Such as in [86], they transformed the plain category structure of DBpedia into hierarchical taxonomy as preprocessing. In this way, all entities were mapped into a hierarchy. After linking these entities to hierarchical categories using deep learning techniques, explicit entity ratings given by the users were extended to categorical hierarchy. They used two spreading functions, Bell Log and Intersect booster, to score entities according to their hierarchical categories. Using a content-based recommendation approach, high ranked entities are added to the top-n list. A study [87] proposed CTransR-CF algorithm based on knowledge graph embedding, which is fused in collaboration filtering technique to recommend Top-N predictions. They used movie-lens knowledge graph data set provided by "The Movie Database (TMDb)" website.[3] Another study [88] developed an RNN based knowledge graph embeddings; in which semantic path mining has been done by adopting two strategies, i.e. they only considered user-item paths, and they limit the path of the edges to 3,5, and 7. To learn the path representation and entity distribution, Attention-Gated Hidden Layer and Embedding Layer are added in the model architecture. The model is learned by minimising the Binary Cross Entropy between the observed and estimated ratings. To recommend an item, the score at the test time is computed by taking the inner product of the user's and item's corresponding embeddings.

Microsoft Knowledge graph was also used to extract the neighbouring entities of an entity [67]. Upon which researchers used deep learning techniques to recommend a news to the user. They have also used embeddings to build a knowledge graph of entities. When a news comes in, they extract the entities in the news and also the neighbouring entities in the KG. They combined this knowledge-level news representation with word-level representation to design a component knowledge- aware convolution neural networks (KCNN) and generated a knowledge-aware embedding vector. They then used an attention module to automatically match candidate news to each piece of clicked news and aggregated the user's history with different weights. For CTR prediction, candidate news' embedding and user's embedding are processed through a deep neural network (DNN). Unlike other deep learning recommendation model, KCNN is specialised in news recommendations. For experimentation, they used data from Bing news and Microsoft Satori knowledge graph to extract data.

Instead of only using explicit ratings, Vagliano et.al [89] used user reviews and comments for recommending. They proposed a model Sem- RevRec to discover annotated and hidden entities from textual data which consists of reviews of music, books, and movies from IMDB, LibraryThing, and Amazon respectively. They used two semantic annotators AIDA and DBpedia Spotlight for rendering entities from their context. AIDA showed better accuracy in independent comparison. After finding annotated entities, they used SPARQL queries to retrieve relevant entities from two knowledge bases, DBpedia, and Wikidata, for they both have Linked data. After that, they used these annotated and initial entities for recommendation and ranking. After comparing SemRevRec model with Random guess, Item KNN, and Bayesian Personalized Ranking (BPR), they concluded that it outperforms them.

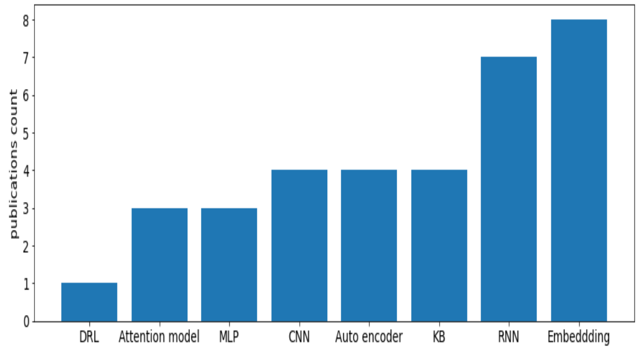


FIGURE 3. Neural network techniques in the recommendation system.

D. SUMMARY OF ALGORITHMIC APPROACHES

In the end, we summarize the publication count of the various neural network technique used in the papers selected by us in Figure 3 based on the criteria discussed earlier. After analysis, it is easy to perceive that majority of recommendation systems are RNNs and embeddings based. RNN architectures like LSTM(Long short-term memory) and BiLSTM are used for sequential learning problems, and in textual data, sequence learning process provides an in- depth semantic analysis of textual documents. Recent recommendation systems are based upon textual and graphical data instead of just utilising user-generated ratings, RNN architectures like GRU persist copious information for analysing these types of data. RNNs and embeddings are the best choices for text-based recommendations according to our study. We have already discussed that profile similarity measures are used to compare two vectors to find the most similar item in recommendation using mathematical formulas like cosine similarity and Euclidean distance. In contrast, neural networks use embedding space to find the nearest neighbours. Their impact on recommendation literature can be visualised in the Figure 4

The evident result of this observation is that usage of neural network techniques in recommendations have been almost 50% more than profile similarity approaches in the last 2-3 years. Neural network embeddings and RNNs pro-vide a systematic way to deal with content-based recommendations. Now, traditional recommenders that used ratings instead of reviews and based upon only collaborative filtering techniques are not common. A lot of researchers have developed hybrid content-based and collaborative filtering recommender systems that use neural networks instead of profile similarity approaches.

[3]https://www. themoviedb.org/

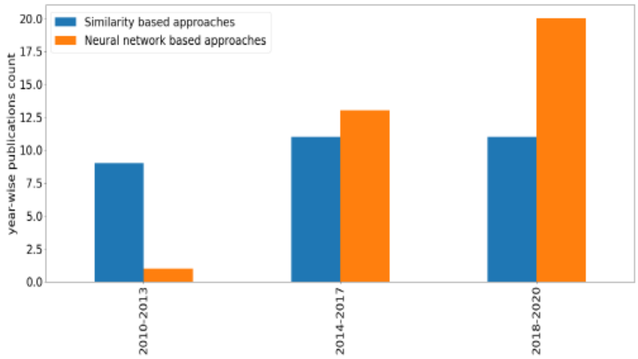


FIGURE 4. Comparison of similarity-based approaches and neural network-based approaches.

VI. DATA SETS

Dataset is the primary ingredient for building a recommendation system. The main components for the recommendation system dataset are items, users, and feedback history. Users are the people who have interacted with a few items previously, and similar new items are recommended to them in the future based upon their feedback history. The feedback can be both explicit and implicit. When a user gives some ratings to a particular item, this is an example of explicit feedback. When a user reads the news, it is taken as positive feedback and this is an example of implicit feedback. For textual recommendation systems, there is a vast spectrum of domains that constitute textual content; like, books, news, and academic journals. And several academic services have published the dataset of these domains for the ease of researchers like Yelp [90], Docear [6], Book-Crossing [91] and Plista [92]. In the case of a textual recommendation system, mostly the system is built upon different features, which are extracted from the textual data items. The textual features are extracted directly from such data items. The detailed study of this subject has revealed that there are also a few cases in which the actual dataset doesn't contain any textual item such as movie in the Movie lens dataset. Still, the researchers have used auxiliary textual data, such as, movie reviews or concepts of movies for their recommendation. In one such other example [41], Narducci *et al.* used Wikipedia data to recommend movies. In some other scenarios [86], [89], semantic analysis on reviews of products using textual content-based techniques are used to understands the user emotions about the product. These techniques enhanced the effectiveness of recommendations. Hence, text-based recommendation techniques are not only used in the case where items are pure text like news recommendations. But, they are also used in other domain recommendation systems where items are not text. There are some other insights which we have extracted after the analysis of different studies. We have concluded that the researchers have used both public and proprietary datasets for their experiments. Even for textual recommendation item, a few datasets have also included a few non-textual features [93]. For example,the publisher id is given in datasets Plista and Globo.

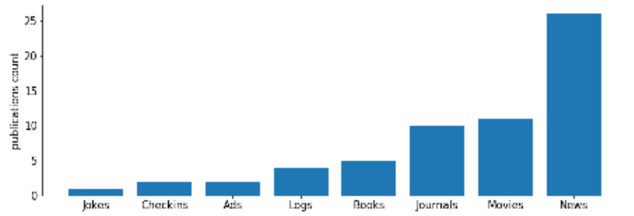


FIGURE 5. Domain wise data-set distribution.

Another insight extracted from study is that minimum number of users were 1000 in publicly available datasets. These insights are compiled in table 2 to give the overview of features and in table 2 to outline the volumes, availability and language of different publicly available datasets. In fig5 the contribution of different domain in the dataset is shown.

The following Table 2 summarizes the key characteristics of all the data-set discussed above.

VII. EVALUATION METRICS

Evaluation metrics are employed in every research to measure the effectiveness of work. In a textual recommendation system, the problem of recommendation can be of two types, if seen from a user's perspective. Either a user will consume the item, in other words, the system will or will not predict the item to the user. Or the system will predict how many ratings will be given to the particular item by the user. Usually, the recommendation system can be categorized into three types. It's a classification problem when an item is recommended, and It's a regression problem when the item ratings are predicted. And, it becomes a ranking problem, when rated items are ranked and we evaluate them according to their order. The evaluation mechanism differs for each type of problem. Hence there are separate evaluation metrics for each type of problem. For example, Mean Absolute Error (MAE), Mean Squared Error(MSE), Rooted mean square error (RMSE) are evaluation metrics used in case of regression; Accuracy, Precision, Recall, F-Measure, Specificity, Area Under the curve(AUC), Hit Rate(HR), Click Through Rate(CTR), Specificity, Average Click Rate(ACR), Macro Recall(MR) are used in case of classification and Mean Reciprocal Rank(MRR), Mean Average Precision(MAP), Precision@ K, Recall@k, F-Measure@k, Average Reciprocal Hit-Rank (ARHR), Discounted Cumulative Gain (DCG) are used in case of a ranking problem. Besides the ones discussed earlier, some other aspects that could be checked to measure the performance of the RS are; how is the user experience about overall system, does a user found something surprisingly good, or how diverse are the recommended items with one another. These aspects are measured with the following metrics in the presented literature; Diversity, Serendipity, Entropy-Based Novelty(EBN), Coverage, and Number of Satisfied Users (NSU). Here is a brief description of all these metrics used in the studied literature. The in-depth discussion on this topic is out of the scope of this particular

TABLE 2. Summary of the results.

Domain	Textual Attributes	Other Attributes	Dataset	Volume(# of items×# of users)	Language	Publicly Available
News	Topic Category	Number of words per title	MSN News [58], [59]	42,255 × 10,000	English	No
	Canonical URL, Reference page URL, Title, Category	Word Count, Published Time, User's City, User's Region, User's Country, OS, Device Type, Time of event	Adressa [94]	11207 × 561,733	Norwegian	Yes
	Article Embeddings	Category Id, word count of given article, publisher Id	Globo [95]	3M × 80M	Portuguese	Yes
	News Article	-	Sina News [96]	28,737 × 1127	Chinese	No
	None-Collaborative	Article Id , User ID	XMU News [93]	13,564 × 163,627	Chinese	No
	Article title	Published date	The examiner [97]	3.09M × 21,000	English	Yes
	Article title, article text, URL, Image Url	Publisher Id, creation time, user's OS, user's browser,user's device, user's location, user's widget,user's language	plista [92]	84,210,795 × 1,095,323	German	Yes
	Tweets text,Hash tags, Topics,Entities	Time	Twitter-News [37]	77,544 × 1, 619	English	No
	Topic category, Sub category,Url resources	User's age, Gender, Location	Yahoo News [98]	11,915 × 7,642	English	Yes
	News, Categories, Concept	User-stock relationship, User's news relationships,Stock-related relationships, News-related relationships	GF Securities data [69]	64,881 × 3800	Chinese	No
	Headlines, Entity name,Category, Topic category	Provider	Custom Data [85]	1,355,344 ×541,337	-	No
	Category, SubCategory, Title, Abstract, Title Entities, Abstract Entities, URl	Time	MIND [99]	161,013 × 1 million	English	Yes
Publications / Journals or conferences	Publication title, Subdomain	Author, Co-author	AMiner dataset [100]	1,932,442 × 1,436,990	English	Yes
	Url of pdf documents, Citation	User's age, Gender and Date of registration, Usage intensity of docear	Docear's [6]	9.4M × 21,439	English	Yes
	Scientific documents, Citation	Author information	OpenCorpus [53]	6.9M × 8.3M	English	No
Books	Book title,Book-Author,	Year-Of-Publication, Publisher, Cover image url	Book-crossing [91]	271379 × 278858	English	Yes
	Description from Wikipedia, Entities from Wikidata and DBpedia knowledge bases	Ratings, User ID, Book ID	DBbook [56], [57]	2362 × 5095	English	Yes
	Reviews	Helpfulness votes, Flags, Rating	LibraryThing [101]	337,561 × 73,882	English	Yes
	Reviews, User's queries	User's info, Click record, Rating	IntentBooks [68]	18,475 ×92,564	English	No
Checkins	Location	Checkins	Checkins [102]	1,385,223 × 11,326	English	No
	Reviews, Business location	Checkins	Yelp [90]	229,907 × 43,873	English	Yes
Movies	Reviews, Description from IMDb, Entities from knowledge bases	Ratings, User ID, Movie ID	Movielens [103]	27,000 × 138,000	English	Yes

TABLE 2. *(Continued.)* **Summary of the results.**

Domain	Textual Attributes	Other Attributes	Dataset	Volume(# of items×# of users)	Language	Publicly Available
Query logs	Queries, Entities	Click record, User's info, Session	Commercial data [83]	6M × 4M	English	No
Jokes	Jokes	Rating , User's info	Jester [104]	150 × 79,681	English	Yes
Others	Tags	Book marked urls of webpages	Delicious [24]	84,005 × 1,000	English	Yes
	Tags	User's info, Artist Info	Last.fm [24]	50,202 × 1,000	English	No
	Tweets	User's Id, Like and dislike percentage, Artist Info	Tweets [45]	606×-	Chinese	No
	Question and answers, Topic	User's info, Rating	Q&A document [105]	606×-	Chinese	No
	Reviews, Product's description	User's info, Rating	Amazon [61]	1,569,973 × 3,035,045	English	No

study. For detailed insights about evaluation metrics, [106] can be explored.

A. MEAN ABSOLUTE ERROR(MAE)

If P is the predicted rating and R is the actual rating of an item, the error is the difference between these ratings. MAE is the average of absolute error values [107].

$$MAE = Average(|P - R|)labeleq \tag{7}$$

$$[MAE = \frac{\sum_{ratings} |P - R|}{\text{Number of ratings}} \tag{8}$$

B. MEAN SQUARE ERROR(MSE)

Mean absolute error(MSE) is computed by taking an average of squared error values over all ratings.

$$MSE = \frac{\sum_{ratings} (P - R)^2}{\text{Number of ratings}} \tag{9}$$

C. ROOT MEAN SQUARE ERROR(RMSE)

RMSE is actually a square root of MSE [107].

$$RMSE = \sqrt{MSE} \tag{10}$$

D. ACCURACY

In the recommendation system, accuracy computes how accurate the recommendations are made. The accuracy in terms of True Positive (TP), False Positive(FP), True Negative(TN) and False Negative(FN) is defined as

$$\frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where, positive items are those that are being recommended and the items which are in recommendation list are negative. Positive items which are according to the interests of users are True positive(TP) and remaining items in the list are False Positive(FP). Similarly, the negative items that should have been recommended but are not recommended are False Negative(FN) and those negative items which should not have been in recommendation list are True Negative(TN).

E. PRECISION, RECALL AND F-MEASURE

These are decision support metrics. These metrics are used to see whether a certain recommender made a good decision in choosing good items and avoiding bad items.

1) PRECISION (P)

It is the percentage of selected items that are relevant to the user [107]. Precision measure how much recommended items are relevant to users interest.

$$Precision = P = \frac{N_{rs}}{N_s} \tag{12}$$

N_{rs}: Number of selected and relevant items N_s: Number of selected items In the term of TP, TN, FP and FN precision is defined as:

$$\frac{TP}{TP + FP} \tag{13}$$

2) RECALL (R)

It is the percentage of relevant items that are selected [107]. The Recall is to ensure that all relevant items that may be useful for users are present in the recommendation list. It is also called sensitivity.

$$Recall = R = \frac{N_{rs}}{N_r} \tag{14}$$

N_r: Number of relevant items In the term of TP, TN, FP and FN recall is defined as:

$$\frac{TP}{TP + FN} \tag{15}$$

3) F-MEASURE

To balance precision and recall F-measure is used. It is used for ensuring that the recommended list has the only relevant item and all possible relevant items are present in this list [107].

$$\text{F-Measure} = \frac{2PR}{P + R} \tag{16}$$

F. SPECIFICITY

Specificity tells us the negative results we have encountered while making recommendations. It basically tells about fall-out score. In the terms of TN(True Negative), FP(False Positive), TP(True Positive and FN(False Negative) specificity is given as:

$$Specificity = \frac{TN}{(FP + TN)} \tag{17}$$

G. AREA UNDER THE CURVE (AUC)

AUC score gives the area under the Receiver Operating Characteristic (ROC) curve. ROC curve is obtained by plotting sensitivity/recall values against specificity/fallout. Recall values are plotted alongside the y-axis and the x-axis contains fallout values. A higher AUC score means a better recommendation system. For a perfect recommendation system, the ROC curve would move towards 1.0 recall and 0.0 fallout. After all relevant items are retrieved, it would go straight right towards 1.0 fallout.

H. HitRate

HitRate is a count of how many correct predictions a particular recommender system has achieved. There are different types of hit rate, Total Hit Rate (THR), Availability Hit Rate (AHR), Outage Hit Rate (OHR) and Coverage Area Accuracy (CAA).

$$HitRate = HR = \frac{\text{Number of hits}}{n} \tag{18}$$

Several hits means that number of items that are in the recommendation list are being returned or rated by users. Hit rate 1.0 means that all desired items are being recommended and 0 means that no desired item is being recommended.

I. CLICK THROUGH RATE (CTR)

Click-Through Rate(CTR) is the percentage at which the recommended item is being clicked by any random user. In simpler words, it is a ratio of the total number of clicks on item and the number of times this item is displayed to the user. CTR prediction is the process of predicting the possibility that the items on the recommendation list will be clicked.

$$CTR = \frac{\text{Number of clicks}}{N_t} * 100 \tag{19}$$

N_t: Number of the times an item is shown

J. AVERAGE CLICK RATE(ACR)

ACR is used in articles recommendation where R_u is article recommended to user u and A_u are actual articles [108]. Click is counted when the user reads the recommended article.

$$ACR = \frac{\Sigma_{u \in U} |R_u \cap A_u|}{|U|} \tag{20}$$

K. P@k,R@k,F/F1@K

The problem with Precision, Recall, and F-Measure is that they cover the all the items. However, in a real time system, the user is mostly interested in top k items. To deal with this problem, top-k settings are introduced in which top-k items from the recommendation list are used. P@k is the precision of top-k items that are good, R@k is the recall of top-k good items and similarly, F/F1@k is the F-measure of top-k items that are in the recommendation list.

$$P@k = \frac{N_{r@k}}{k} \tag{21}$$

$$R@k = \frac{N_{r@k}}{N_r} \tag{22}$$

$$\text{F@k} = \frac{2P@kR@k}{P@k + R@k} \tag{23}$$

L. MEAN RECIPROCAL RANK (MRR)

The rank-based metrics decide where to put the items in the recommendation list. In other words, its purpose is to decide the relative preference of items [107]. The items which the user likes should be at a higher level in the list than the items the user dislikes so that the user finds its required items quickly. Items that are at the top of the list have a higher rank. If item A is at 2nd position in recommendation list of relevant items, then its rank will be 2, and reciprocal rank will be 1/2. The mean reciprocal rank is the mean of all items reciprocal rank.

$$\text{MRR(O,U)} = \frac{1}{|U|} \Sigma_{u \in U} \frac{1}{k_u} \tag{24}$$

M. MEAN AVERAGE PRECISION (MAP/MAP@k)

Precision and Recall doesn't cater to the relevance factor of recommended items, hence they all appear to be of equal interest to the user. MAPs take care of the ordering of these items. [109]

MAP = sum(ratings of recommended items)/N recommended items

N. AVERAGE RECIPROCAL HIT-RANK (ARHR)

HitRate does not give any information about the position of an item in the top-k list. To evaluate whether the desired item is present in the top-k list and if this item is at a particular position Average Reciprocal Hit-Rank is used. It gives more weight age to the hit of position one than position two. If h is the hit count at position p for the given top-n list then the ARHR for n users would be [110]

$$ARHR = \frac{1}{n} \Sigma_{i=1}^{h} \frac{1}{p_i} \tag{25}$$

O. DISCOUNTED CUMULATIVE GAIN (DCG)

In the recommendation list, top-n relevant items should have more value. To achieve this goal, the weights of relevant items are heavy as compared to irrelevant items. DCG is used to emphasize the items that are most relevant in the list. Its purpose is to measure the value or utility of an item at each position. The utility is basically a rating value given to the item by the user.

$$DCG = \Sigma_i \frac{r_i}{disc(i)} \tag{26}$$

$$disc(i) = \begin{cases} 1, & i \leq 2 \\ log_2\, i, & i > 2 \end{cases} \quad (27)$$

i is the position of the item in a list and r_i is the relevance score.

In some studies DCG is used in its normalized formed, and reffered as Normalized Discount cumulative gain(NDCG) [96].

P. DIVERSITY

Diversity in recommender system refers to difference between items in the top-n list. It can be only applied to the top-n recommender list to know how items are distinct from each other. Pairwise similarity matrix contains similarity values of items with each other. Higher similarity means low diversity. Diversification is an approach to remove too similar items from the list [96], [108].

$$Diversity = 1 - \frac{\Sigma_{i,j \in R(u), i \neq j} sim(i,j)}{\frac{1}{2}|R(u)|(|R(u)| - 1)} \quad (28)$$

R(u) is an item recommendation list.

Q. SERENDIPITY

A general definition of serendipity is a surprise or unexpected results that are rather delightful. In the recommendation system, it refers to recommended items that users least expected in their recommendation list but these items turn out to be highly liked afterward. Serendipity in the recommender system is measured by dividing bits of delights by the total number of recommended items. And these bits of delight are measured by subtracting the primitive score of an item from the prediction score and multiplying it to its relevance score.

$$\frac{1}{N} \Sigma_{i=1}^{N} max(Pr(s_i) - Prim(s_i), 0) * isrel(s_i) \quad (29)$$

So serendipity is the sum of obviousness of item multiply by relevance score for all items, and all of these are divided by the total number of the selected item. A primitive estimate is necessary to measure serendipity. One primitive estimate is the overall popularity or obviousness of the item. The relevance score is usually zero or one. It is the divergence from the traditional prediction of items [111].

Here $s_i(i = N)$ denote the i-th ranked item

Pr is the estimated score from the prediction model

Prim is the estimated score from primitive prediction model

isrel is relevant score

R. ENTROPY-BASED NOVELTY (EBN)

Popularity is the evaluation metrics that can be measured by the percentage of users who rate a particular item. Novelty is the opposite of popularity which refers to the percentage of items that are not known [11]. Here EBN refers to the capability of a system to recommend the items relevant to the user but not much common among other users or less popular items. Entropy is actually a measure of uncertainty instead of similarity.

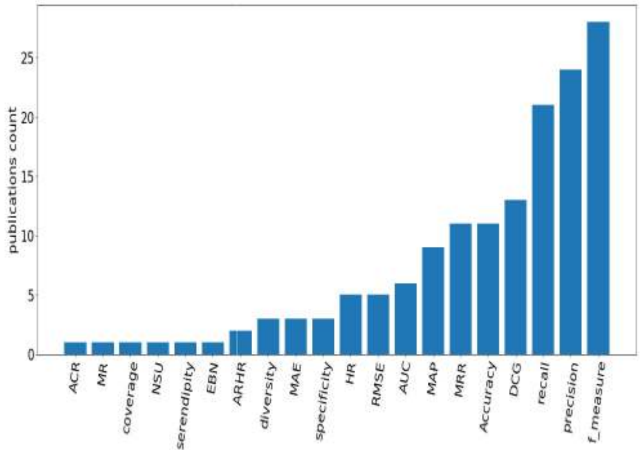


FIGURE 6. Recommendation system evaluation metrics.

S. COVERAGE

Coverage represents the percentage of items/users that are recommendable by the recommendation system from a set of items/users. The system is not being able to predict some items to the users because of the lack of user interaction with the items, generally known as the cold start problem. There are two types of coverage, item Coverage and user coverage [11]. Item coverage is computed as follow:

$$ItemCoverage = n/N * 100 \quad (30)$$

where n = number of recommendable items

N = total number of items likewise user coverage represents the set of users to whom recommendation were made from all potential users.

T. NUMBER OF SATISFIED USERS(NSU)

It is defined in [108] as:

$$NSU = \Sigma_{u \in U} \mathbb{I}(|R_u \cap A_u| > 0) \quad (31)$$

$\mathbb{I}$ is predicate whose value is 1 if the prediction is true otherwise false.

U. OVERALL SUMMARY OF EVALUATION METRICS

The above section summarized the details about the metrics used for recommendation in the studies. The Figure 6 summarises the usage of these metrics in the selected literature. From the figure it is obvious that most popular evaluation metrics are *"Precision","Recall" and "F-measure"*. In some studies all three are reported together while in some others "precision" is reported and some studies reported "Recall". Though these are standard data mining metrics that give a good insight into the correctness of the researcher's approach to solve the in-hand problem. It is equally effective for the problem of recommendation as well.

VIII. OVERALL SUMMARY

The following table gives a summary of chosen papers on textual data recommendation. More than 60 studies are concluded here. The first column is the reference of the study, and the second column is about the dataset being used in this particular study. The names of the dataset are similar to the ones mentioned in section VI. Several studies their proprietary dataset, for such studies, the domain to which this data belongs is mentioned. For the case of news articles scraped from a certain website, only the domain name ''News'' is mentioned in the respective column. For movie recommendations, the source from where the extra-textual information is taken is also mentioned along with the dataset.

- News (recommending news)
- Twitter-News (recommending news using Twitter data)
- Google News (news recommendation using news records form Google news portal)
- Movielens(Wikipedia/IMDb/DBpedia/TMDb/ Satori knowledge base) (using description, textual information, entities and reviews of movies from specified source for concept extraction)
- Publications/ Journals or conferences (recommending relevant articles and research papers)
- Advertisement (ad recommendation using advertisement data)
- Query logs (entity recommendation using browsing record, search logs and query data)
- Checkins (recommending locations to visit/check-in)
- Q&A document (recommending Q&A documents for knowledge management system)

The third column refers to the feature selection or how the profiling was done. The fourth and fifth columns are about how the system was evaluated; what metrics were used; and what was their score. The sixth column is about the algorithms used in each study. And the last column is about the actual problem addressed in that particular work. The identified problems are classified into the following classes.

The identified problems are classified in following classes.

BS: Better System

The recommendation system on this domain already exists with the available dataset to the researchers. They have improved the Accuracy / score of the system

NDND: New Domain with the new dataset:

The recommendation data was not available earlier on the given platform. Researchers have formed the RS and validated their work by applying some of the baseline/other methods to justify their system as the better one.

Analysis:

Datasets and techniques already exist but in a complicated form. After removing complexity, they analysed the performance of recommendation system and stated whether these changes were harmful to the performance or not.

IX. FINDINGS AND INSIGHT

This section will present the findings of the detailed literature review done in the previous section. The results of the work would be presented in the next subsection. The last part will conclude this study.

This section enlists the findings of the overall work.

- Nowadays, the text-based recommender system doesn't follow one specific theoretical approach; such as collaborative filtering or standard content-based filtering. Instead, the process of recommendation involves a combination of a variety of techniques from different domains like NLP and deep learning.
- Though Online Evaluation gives a real-time flavor, Offline Evaluation is the one that is a practically established choice. With a wide variety of given metrics, researchers are most interested in calculating the right prediction. Or how right their approach can predict? In other words, they are most interested to know the Recall of the developed system.
- The literature suggests recommending the diverse yet interesting recommendation that can happily surprise the user. But, still, most of the system is evaluated in terms of how similar it can predict.
- The textual data is used in two ways for recommendations. First, it is used directly in the domains, where the actual content constitutes text such as news and research papers. Second, when used indirectly, where it is used as auxiliary data to enhance the recommendation accuracy such as user's reviews about a product in product recommendation or entity information to recommend movies
- Generally, the researchers build recommender-systems for their platforms, and hence their data is not publicly available. Even the publicly available dataset is sometimes in an abstract form like in Globo [95]], therefore, users are restricted to use the given embeddings and will be unable to exploit the raw data to extract more features.
- Extracting useful text-based features such as keyword-based, letter-based, concept-based, topics, named entities, semantic features using embeddings and latent hidden features derived from deep learning techniques is a crucial and essential step in a text-based recommender system.
- Deep-learning-based techniques most of the time rely on their frame-work for the extraction of features. Still, if the other aspects such as popularity or extracted entities are combined with it, it will boost the recommendation accuracy.
- Among the given deep learning-based techniques, Embeddings and RNNs are frequently used ones. Because Embeddings are a proven way to extract semantic features from text and it also an excellent method for dimensionality reduction. RNNs are famous for their sequence-based learning, hence they are a compelling choice for profiling and session base recommendation, which is required in the case of a news recommendation system.
- Generally, deep learning techniques are used to enhance the accuracy of recommendations for semantic analysis. Recently, the researchers working on the Hermes news portal [35] achieved an F-measure of 60% by adding additional information of named-entities from

TABLE 3. Overall summary.

Study	Dataset	Profiling/ Modeling	Evaluation Metrics	Score	Algorithmic Technique	Problem Address
[23]	News	TF-IDF	Minimize distance	0.18	Minimizing distance	Proposes a cyclic process to improve user profile automatically by inferring the feedback ***NDND***
[24]	Delicious, Last.fm	Social tags, TF-based, TF-cosine, TF-IDF, BM25-based similarity, BM25 cosine-based similarity	Precision@N MAP DCG	0.364 0.145 0.390	Similairty	Evaluated a few content-base models on profiles which were presented in terms of social tags ***BS***
[26]	Docear's RS platform dataset	TF, TF-IDF, TF-IDuF	CTR	4.06% 5.09% 5.14%	Cosine similarity	Proposes a new technique for profiling ***BS***
[29]	Cross-domain collaboration	Letter-based features, Multi layered perceptron	Precision@k MAP Recall@k	40.0 41.2 69.8	Cosine similarity	General Deep learning framework for matching item features and user features in cross domain environment ***NDND***
[28]	News	Letter-based features, RNN(LSTM)	Precision@k Accuracy MAP MRR	0.245 0.814 0.099 0.397	Similairty	Deep learning for temporal news Recommendations ***NDND***
[112]	News	title ,body,entities (BERT embeddings)	Accuracy	0.704	Similairty	A news competition winner by microsoft on MIND dataset***BS***
[43]	Aminer	Letter-based features, Cross-domain topic learning	Precision@k MAP Recall@k ARHR	37.7 40.6 35.6 14.3	Random walk	Cross-domain topic collaboration ***BS***
[113]	Publications/ Journals or conferences	TF-IDF, Topic specificity	Accuracy	0.47	Correlation coefficient	Used entropy measure and LSA for ontology-based recommender ***NDND***
[37]	Twitter-News	Hash tags, Entity-based, Topic modeling	MRR	0.1	Distance	Comparison of profiling methods ***NDND***
[30]	Hermes News Portal	Sysnet	Accuracy Specificity F1-measure	80.1% 94.7% 46.8%	Cosine similarity	Usage of semantics instead of words only to increase Accuracy ***BS***
[33]	Hermes News Portal	Concept-based	Accuracy Recall F1-measure	+4.2% +24.0% +19.1%	Similarity	Usage of semantics ontologies and compare with TF-IDF ***BS***
[34]	Hermes News Portal	Concept-based, Related concepts	Accuracy Recall F1-measure	0.85072	Similarity	Usage of semantics ontologies and compare them with TF-IDF ***BS***
[31]	Hermes News Portal	Inter-synset semantic relationships	Accuracy Specificity F1-measure	79.75% 93.13% 59.73%	Cosine similarity	Usage of semantics ontologies and compare them with TF-IDF ***BS***
[32]	Hermes News Portal	Incorporated NER	Accuracy Specificity F1-measure	81% 91% 58%	Cosine similarity	Usage of Name Entity along with SF-IDF+ ***BS***
[35]	Hermes News Portal	Incorporated NER	F1-measure	60%	Cosine similarity	Usage of Name Entity along with CF-IDF+ ***BS***
[41]	DBbook, Movielens (Wikipedia)	Concepts incorporated from knowledge sources	F1-measure	58.2% 54.7%	Cosine similarity	Cross-lingual recommendation ***BS***
[40]	Advertisement	Concepts incorporated from knowledge sources	Precision	78%	Cosine similarity	Usage of ontologies for profile generation in ads recommendation ***NDND***

TABLE 3. ***(Continued.)*** **Overall summary.**

Study	Dataset	Profiling/ Modeling	Evaluation Metrics	Score	Algorithmic Technique	Problem Address
[79]	Book-Crossing	LDA for topic modeling	MAE RMSE	2.6032 3.38	CNN, Cost function minimization	Using CNN for content-based recommendation ***BS***
[42]	Publications/ Journals or conferences	Chi-square feature selection	Accuracy F1-measure	35.03 0.18	Softmax regression, Cost function minimization	RS for computer science and technology articles ***NDND***
[45]	Tweets	k-score to extract topics	Recall Precision	76.73% 87.50%	Linear regression, k-core analysis	SINA micro-blog tweets recommendation ***NDND***
[56]	Movielens (Wikipedia), DBbook	Word embedings	F1-measure@k	0.5757	LSI, Random Indexing, Word2Vec	Comparing LSI, Random Indexing and Word2Vec for learning word embeddings ***BS***
[53]	Publications/ Journals or conferences	NNSelect, NNRank	MRR F1-measure	0.771 0.329	Cosine similarity	Content-based citation recommendation for query documents ***BS***
[4]	Publications/ Journals or conferences	Graph learning, Textual and structural vectorization	Precision Recall F1-measure NDCG	90.30% 51.42% 75.14% 84.69%	Similarity, Doc2vec, Struc2vec	Paper recommendation by learning vector representation ***NDND***
[67]	News	Knowledge graph embedding	AUC F1-measure	65.7±1.1 68.8±1.4	Softmax, KCNN	CTR prediction for news recommendation using Knowledge-aware CNN ***NDND***
[78]	Twitter-News	Geographical topic-based semantic modeling, ELSA	MAP@k Recall@k F@k Precision@k	0.5662 0.9768 0.9403 1.1162	Cosine similarity	Personalized news recommendation using location information ***NDND***
[54]	News	LSA, LDA and doc2vec document embeddings	Accuracy	91.0%	doc2vec, Softmax	News recommendation using paragraph vectors and comparison with LDA and LSA ***NDND***
[89]	Movielens (IMDb), LibraryThing, Last.fm	Semantic annotation, Ranking functions	Precision Recall nDCG EBN diversity	0.0857 0.0561 0.0686 0.7820 0.2431	LDSD measure	Semantic annotation using reviews and linked data ***NDND***
[86]	Movielens (DBpedia)	Taxonomic structure with multiple inheritance in KB	Recall	0.24	Bell Log, Intersect booster, Reverse spreading	Utilizing hierarchical structure of KB instead of its item taxonomies ***NDND***
[8]	Movielens (Wikipedia)	BRNNs, LOD-based features	F1-measure@k	0.654	Logistic regression	Content-based recommendation by learning joint representation of the items ***NDND***
[87]	Movielens (TMDb)	Knowledge graph embedding, semantic information maintenance	Recall Precision F1-measure	0.25 0.07 0.09	Pearson correlation, Similarity	Collaborative filtering recommendation using semantic information inherent in the item ***BS***
[10]	Globo News	Article Content Representation(ACR), Next-Article Recommendation (NAR) module, LSTM	HR@k MRR@k	0.72 0.51	Softmax, Cosine similarity	Using a CNN to extract textual features from news articles ***BS***
[59]	MSN News	Context capturing via CNNs	AUC MRR nDCG@k	0.6289 0.3315 0.4392	CNN, Softmax	Topic-aware news recommendation ***NDND***
[68]	Movielens (Satori knowledge base), IntentBooks	Heterogeneous network embedding, Deep learning embedding	MAP@K Recall@K	0.07 0.5	Bayesian TransR, Bayesian SDAE, Bayesian SCAE	Hybrid recommendation by integrating collaborative filtering and KB ***BS***
[80]	Publications/ Journals or conferences	Attention based modeling, NNLM	AUC F1-measure Precision Recall	0.853 0.317 0.258 0.45	Maximal distance	Dynamic attention deep model for non-explicit selection criteria ***NDND***

TABLE 3. *(Continued.)* **Overall summary.**

Study	Dataset	Profiling/ Modeling	Evaluation Metrics	Score	Algorithmic Technique	Problem Address
[85]	News	DQN-based reinforcement learning	CTR Precision@k nDCG	0.0113 0.0149 0.0492	DDQN	Online personalized news recommendation by modeling dynamic news features and user preferences ***NDND***
[88]	Movielens (IMDb), Yelp	Embedding-based, learning path saliency using SPM	Precision@k MRR	0.1396 0.3056	Semantic path mining, RNN, Max pooling	Usng RNN for recommendation by automatically capturing entity relations encoded in KGs ***BS***
[58]	MSN News	News encoding, User encoding, Click prediction	AUC MRR nDCG@k	0.6243 0.3321 0.4380	CNN, Softmax	News recommendation by learning title using CNN ***NDND***
[84]	Query logs	Probabilistic graphical model, Mashup-service pair	Recall@k	<0.8	SDAE, Matrix factorization	Using SDAE to deal with unsatisfactory quality of description given by service developers and mashup queries ***NDND***
[82]	Query logs	Bi-LSTM, GBDT, LTR	nDCG@k	0.2834	Cosine similarity	Context modeling for improving entity recommendation ***NDND***
[83]	Query logs	Query-based, Session-based translation model	MRR Precision@k F@k	0.0936 0.0626 0.0447	Argmax , Co-occurrence similarity, Centroid session similarity	Domain-agnostic method for entity suggestions by exploiting query log data ***NDND***
[5]	News	Network embeddings	Precision@k MRR	0.899 0.487	Softmax , Random walk	Learning latent feature in event network for news events ***NDND***
[114]	Yelp	Sentiment extraction	Precision@k AUC	0.862 0.745	Stochastic Gradient Descent, Logistic function	Recommending both item and positive aspects to enhance user experience ***BS***
[115]	News	Hadoop framework	Accuracy F1-measure- Measure@k	0.95	Neighborhood similarity	News recommendation by supporting valuable neighbors' patterns and change of user preferences over time and domain ***NDND***
[108]	Plista	LDA clustering, NER for entity extraction	ACR MR NSU	<0.5 <0.75 <0.95	Distance, 2-Approximation algorithms, MAXMIN algorithm	Diversify news recommendation when users are concerned about privacy and location ***NDND***
[116]	News	SVM-Rank, SVM-Perf, Bag of words	nDCG MAP	0.270 0.266	Similarity, TF-IDF	News recommendation using real time user data ***NDND***
[117]	Publications/ Journals or conferences	CTR initialization, LDA, Hierarchical Bayesian model	AUC HRk@k	<0.11 <0.75	Diffmax, Softmax	Comment worthy article recommendation using content-driven user profiles ***NDND***
[25]	Checkins	Topic-sensitive PageRank	Precision Recall F1-measure	0.35400 0.03100 0.05700	Distance	Analyzing check-in information to recommend potential check-in locations ***NDND***
[27]	Query logs	Word-hashing	NDCG@k	0.498	Softmax	Extended previous deep architecture latent semantic models ***NDND***
[118]	News	Vector-based article representation, TF-IDF	F1-measure Precision Recall	0.182 0.165 0.202	Similarity	Overview of article vectors for similarity search and real content-based recommendation ***NDND***
[119]	Adressa	Long-term and short-term interest modeling	AUC F1-measure	78.62% 81.01%	Attention-based LSTM	Introduced GNewsRec to model heterogeneous user-news-topic graph ***BS***
[120]	Yahoo! movies	DSAG, MAG	MSE	0.47 0.53	Argmax probability	Used advice givers for predictions by weighing reviews ***BS***
[121]	Advertisements	TF-IDF	MAP@k Recall F1-measure	85.6% 81% 79.2%	Cosine similarity	Used shared ontology model to recommend advertisements on social platform ***NDND***
[105]	Q&A document	Biterm topic model (BTM)	Precision Recall F1-measure	0.793 0.900 0.843	Max similarity	Combined CB, CF and complementary-based recommendation to obtained comprehensive knowledge from limited Q&A documents ***NDND***
[122]	News	CTR	MAP MRR	0.4319 0.4501	LambdaMart	Understanding user fatigue factors in recommender systems ***NDND***

TABLE 3. ***(Continued.)*** **Overall summary.**

Study	Dataset	Profiling/ Modeling	Evaluation Metrics	Score	Algorithmic Technique	Problem Address
[93]	News	Sparsity	RMSE	0.727273	Regularization, SVD	Analyzing the performance of regularization methods in SVD ***NDND***
[123]	News	Multi-tier architecture	Precision Recall F1-measure	87.7 61.5 72.3	Nearest neighbor algorithm, Naïve Bayesian classifier	Presented an adaptive system to learn users' interest from news stories ***NDND***
[73]	News	Time ordered measure,	Precision Recall	>0.2	Jaccard similarity, Time-dependent similarity	Using time sequence characteristic of user behaviors in news recommendation ***NDND***
[124]	News	Temporal and Non-Temporal Scoring Function	BinHR ARHR	>0.8	UKNN, Tag UKNN, TimeTag UKNN, Item-KNN	Recommending articles to active users using latent-tag-vectors ***NDND***
[96]	News	Bisearch optimization, Coherence estimation	F1-measure Diversity NDCG	0.5278	Bipartite graph	Recommending news by helping users better understand the news story chain building ***NDND***
[44]	Publications/ Journals or conferences	LDA	Mean Surprise Rating	0.48 0.25 0.16 < 0.01 < 0.01	Serendipity, Cosine similarity, Unexpectedness, Diversity, Topic dissimilarity	Built an unexpectedness model to recommend serendipitous news articles ***NDND***
[51]	Checkins	Word Embedding(W2V)	Precision NDCG HitRate Coverage	0.119 0.169 0.618 1	Cosine similarity	Used W2V models (skipgram and CBOW) for recommendation of next check-in venue (location) on Location Based Social Networks (LBSNs) ***BS***
[57]	Movielens (Wikipedia), DBbook	Word embedings	F1-measure@k	0.6260	W2V, RI , LSI	Performing experimental evaluation to compare the performance of word embeddings on different datas ***BS***
[55]	Plista	Matrix factorization model, Neural network	HR NDCG	0.768 0.538	Logistic function	Using neural network for news recommendation ***NDND***
[61]	Amazon, Yelp	Hierarchical LSTM	MAE RMSE	0.9256 1.2503	ReLU function	Using hierarchical and symmetrical RNNs and apply these to reviews ***NDND***
[60]	Adressa Last.fm	Factor embedding, Item embedding	MRR Precision Recall F1-measure	0.357 0.336 0.346 0.202	Session-based CNN, GRU, Softmax function	Using dynamic attention-integrated neural network for session based recommendation ***NDND***
[69]	News	Graph embeddings, Node2vec, Bi-LSTM, CRF	RPI rate Precision@k Recall@k	0.397297 0.1920 0.1560	Euclidean distance, Cosine similarity	Using knowledge graph for financial news recommendation ***NDND***
[72]	News	Table-based similarity dependent CRP	Precision@k Recall@K F1-measure@K nDCG@K	0.38 0.38 0.43 0.56	Jaccard similarity	Incorporated trend ***NDND***
[125]	Google News	Click distribution of categories(topics)	CTR	< 1.5	Bayesian law	Incorporated user interest in Google News ***NDND***
[126]	News	Clustering, Topic modeling , Named entities	F1-measure Diversity	0.5015 0.6930	Cosine similarity	Used multiple features ***NDND***

concept-based knowledge bases. This is the highest realised F-measure on this news portal so far.

- It is evident from this study that researchers tend to develop their RS for their own platform/self- generated data and validate their approach by implementing the preexisting baseline algorithms. There are some studies which utilize publicly available dataset or have taken the part of some recommendation system competition, and there are just a few studies which actually address any challenge or issue of RS.

X. CONCLUSION

Text-based recommendation systems have become more prevalent in the last decade because the internet is generating the bulk of textual data every day over different websites. This study aims to explore text-based recommendation literature and summarize critical approaches to provide a single platform for the understanding of new researchers. The survey covers four main aspects of a text-based recommendation system. First, what are the fundamental techniques of feature extraction used in text-based recommendation system? Second, proprietary and publicly available datasets and their details. Third, how such systems are evaluated what are the most frequently used evaluation metrics, and lastly what algorithmic approaches are opted to formulate the problem. After surveying an immense amount of text-based recommendation literature, its comprehensive summary is presented in Table 3.Datasets and metrics that are involved in evaluating a recommender are specified in their respective column. A detailed description of evaluation metrics and volume(number of item number of users) of datasets are stated in the Section VII and VI of this study, respectively.

REFERENCES

[1] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl.-Based Syst.*, vol. 46, pp. 109–132, Jul. 2013.

[2] Y. Wang, N. Stash, L. Aroyo, L. Hollink, and G. Schreiber, "Semantic relations for content-based recommendations," in *Proc. 5th Int. Conf. Knowl. Capture (K-CAP)*, 2009, pp. 209–210.

[3] Z. Cao, X. Qiao, S. Jiang, and X. Zhang, "An efficient knowledge-graph-based Web service recommendation algorithm," *Symmetry*, vol. 11, no. 3, p. 392, Mar. 2019.

[4] X. Kong, M. Mao, W. Wang, J. Liu, and B. Xu, "VOPRec: Vector representation learning of papers with text information and structural identity for recommendation," *IEEE Trans. Emerg. Topics Comput.*, early access, Apr. 26, 2019, doi: 10.1109/TETC.2018.2830698.

[5] V. Setty and K. Hose, "Event2Vec: Neural embeddings for news events," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, New York, NY, USA, 2018, pp. 1013–1016.

[6] J. Beel, S. Langer, B. Gipp, and A. Nürnberger, "The architecture and datasets of docear's research paper recommender system," *D-Lib Mag.*, vol. 20, no. 11/12, 2014.

[7] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Research-paper recommender systems: A literature survey," *Int. J. Digit. Libraries*, vol. 17, no. 4, pp. 305–338, Nov. 2016.

[8] C. Musto, T. Franza, G. Semeraro, M. de Gemmis, and P. Lops, "Deep content-based recommender systems exploiting recurrent neural networks and linked open data," in *Proc. Adjunct Publication 26th Conf. User Modeling, Adaptation Personalization*, New York, NY, USA, 2018, pp. 239–244.

[9] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. 12, pp. 3371–3408, 2010.

[10] G. de Souza Pereira Moreira, F. Ferreira, and A. M. da Cunha, "News session-based recommendations using deep neural networks," in *Proc. 3rd Workshop Deep Learn. Rec. Syst.*, New York, NY, USA, 2018, pp. 15–23.

[11] G. Shani and A. Gunawardana, *Evaluating Recommendation Systems*. Boston, MA, USA: Springer, 2011, pp. 257–297.

[12] D. Jannach, L. Lerche, I. Kamehkhosh, and M. Jugovac, "What recommenders recommend: An analysis of recommendation biases and possible countermeasures," *User Model. User-Adapted Interact.*, vol. 25, no. 5, pp. 427–491, Dec. 2015.

[13] L. Chen, G. Chen, and F. Wang, "Recommender systems based on user reviews: The state of the art," *User Model. User-Adapted Interact.*, vol. 25, no. 2, pp. 99–154, Jun. 2015.

[14] P. Lops, D. Jannach, C. Musto, T. Bogers, and M. Koolen, "Trends in content-based recommendation," *User Model. User-Adapted Interact.*, vol. 29, pp. 239–249, Mar. 2019.

[15] J. K. Tarus, Z. Niu, and G. Mustafa, "Knowledge-based recommendation: A review of ontology-based recommender systems for e-learning," *Artif. Intell. Rev.*, vol. 50, no. 1, pp. 21–48, Jun. 2018.

[16] Y. Betancourt and S. Ilarri, "Use of text mining techniques for recommender systems," in *Proc. 22nd Int. Conf. Enterprise Inf. Syst.*, vol. 1, 2020, pp. 780–787.

[17] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 5:1–5:38, 2019.

[18] Z. Batmaz, A. Yurekli, A. Bilge, and C. Kaleli, "A review on deep learning for recommender systems: Challenges and remedies," *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 1–37, Jun. 2019.

[19] M. Karimi, D. Jannach, and M. Jugovac, "News recommender systems—Survey and roads ahead," *Inf. Process. Manage.*, vol. 54, no. 6, pp. 1203–1227, Nov. 2018.

[20] Ö. Özgöbek, J. A. Gulla, and R. C. Erdur, "A survey on challenges and methods in news recommendation," in *Proc. WEBIST*, 2014, pp. 278–285.

[21] S. Jain, H. Khangarot, and S. Singh, "Journal recommendation system using content-based filtering," in *Recent Developments in Machine Learning and Data Analytics*, J. Kalita, V. E. Balas, S. Borah, and R. Pradhan, Eds. Singapore: Springer, 2019, pp. 99–108.

[22] J.-W. Ahn, P. Brusilovsky, J. Grady, D. He, and S. Y. Syn, "Open user profiles for adaptive news systems: Help or harm?" in *Proc. 16th Int. Conf. World Wide Web*, New York, NY, USA, 2007, pp. 11–20.

[23] A. Moreno, L. Marin, D. Isern, and D. Perello, "Dynamic learning of keyword-based preferences for news recommendation," in *Proc. IEEE/WIC/ACM Int. Joint Conf. Web Intell. (WI) Intell. Agent Technol. (IAT)*, Aug. 2014, pp. 347–354.

[24] I. Cantador, A. Bellogín, and D. Vallet, "Content-based recommendation in social tagging systems," in *Proc. 4th ACM Conf. Rec. Syst.*, New York, NY, USA, 2010, pp. 237–240.

[25] R. Baral and T. Li, "Maps: A multi aspect personalized poi recommender system," in *Proc. 10th ACM Conf. Rec. Syst.*, New York, NY, USA, 2016, pp. 281–284.

[26] J. Beel, S. Langer, and B. Gipp, "TF-IDUF: A novel term-weighting scheme for user modeling based on users' personal document colletions," 2017.

[27] P.-S. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. Heck, "Learning deep structured semantic models for Web search using clickthrough data," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manage.*, New York, NY, USA, 2013, pp. 2333–2338.

[28] Y. Song, A. M. Elkahky, and X. He, "Multi-rate deep learning for temporal recommendation," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, New York, NY, USA, 2016, pp. 909–912.

[29] A. M. Elkahky, Y. Song, and X. He, "A multi-view deep learning approach for cross domain user modeling in recommendation systems," in *Proc. 24th Int. Conf. World Wide Web*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2015, pp. 278–288.

[30] M. Capelle, F. Frasincar, M. Moerland, and F. Hogenboom, "Semantics-based news recommendation," in *Proc. 2nd Int. Conf. Web Intell., Mining Semantics*, New York, NY, USA, 2012, pp. 27:1–27:9.

[31] M. Moerland, F. Hogenboom, M. Capelle, and F. Frasincar, "Semantics-based news recommendation with SF-IDF+," in *Proc. 3rd Int. Conf. Web Intell., Mining Semantics*, New York, NY, USA, 2013, pp. 22:1–22:8.

[32] M. Capelle, M. Moerland, F. Hogenboom, F. Frasincar, and D. Vandic, "Bing-SF-IDF+: A hybrid semantics-driven news recommender," in *Proc. 30th Annu. ACM Symp. Appl. Comput.*, New York, NY, USA, 2015, pp. 732–739.

[33] F. Goossen, W. IJntema, F. Frasincar, F. Hogenboom, and U. Kaymak, "News personalization using the CF-IDF semantic recommender," in *Proc. Int. Conf. Web Intell., Mining Semantics*, New York, NY, USA, 2011, pp. 10:1–10:12.

[34] E. de Koning, F. Hogenboom, and F. Frasincar, "News recommendation with CF-IDF+," in *Advanced Information Systems Engineering*, J. Krogstie and H. A. Reijers, Eds. Cham, Switzerland: Springer, 2018, pp. 170–184.

[35] E. Brocken, A. Hartveld, E. de Koning, T. van Noort, F. Hogenboom, F. Frasincar, and T. Robal, "Bing-CF-IDF+: A semantics-driven news recommender system," in *Advanced Information Systems Engineering*, P. Giorgini and B. Weber, Eds. Cham, Switzerland: Springer, 2019, pp. 32–47.

[36] W. IJntema, F. Goossen, F. Frasincar, and F. Hogenboom, "Ontology-based news recommendation," in *Proc. EDBT/ICDT Workshops*, New York, NY, USA, 2010, pp. 16:1–16:6.

[37] F. Abel, Q. Gao, G.-J. Houben, and K. Tao, "Analyzing user modeling on Twitter for personalized news recommendations," in *User Modeling, Adaption and Personalization*, J. A. Konstan, R. Conejo, J. L. Marzo, and N. Oliver, Eds. Berlin, Germany: Springer, 2011, pp. 1–12.

[38] S. N. Ferdous and M. M. Ali, "A semantic content based recommendation system for cross-lingual news," in *Proc. IEEE Int. Conf. Imag., Vis. Pattern Recognit. (icIVPR)*, 2017, pp. 1–6.

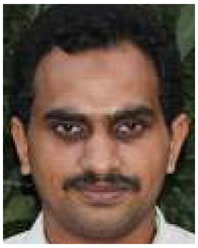
[39] C. Obeid, I. Lahoud, H. El Khoury, and P.-A. Champin, "Ontology-based recommender system in higher education," in *Proc. Companion Web Conf. (WWW)*, 2018, pp. 1031–1034.

[40] F. García-Sánchez, J. A. García-Díaz, J. M. Gómez-Berbís, and R. Valencia-García, "Ontology-based advertisement recommendation in social networks," in *Proc. 15th Int. Symp. Distrib. Comput. Artif. Intell.*, F. De La Prieta, S. Omatu, and A. Fernández-Caballero, Eds. Cham, Switzerland: Springer, 2019, pp. 36–44.
[41] F. Narducci, P. Basile, C. Musto, P. Lops, A. Caputo, M. de Gemmis, L. Iaquinta, and G. Semeraro, "Concept-based item representations for a cross-lingual content-based recommendation process," *Inf. Sci.*, vol. 374, pp. 15–31, Dec. 2016.
[42] D. Wang, Y. Liang, D. Xu, X. Feng, and R. Guan, "A content-based recommender system for computer science publications," *Knowl.-Based Syst.*, vol. 157, pp. 1–9, Oct. 2018.
[43] J. Tang, S. Wu, J. Sun, and H. Su, "Cross-domain collaboration recommendation," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, 2012, pp. 1285–1293.
[44] M. Jenders, T. Lindhauer, G. Kasneci, R. Krestel, and F. Naumann, "A serendipity model for news recommendation," in *KI 2015: Advances in Artificial Intelligence*, S. Hölldobler, R. Peñaloza, and S. Rudolph, Eds. Cham, Switzerland: Springer, 2015, pp. 111–123.
[45] Y. He and J. Tan, "Study on SINA micro-blog personalized recommendation based on semantic network," *Expert Syst. Appl.*, vol. 42, no. 10, pp. 4797–4804, Jun. 2015.
[46] Y. Goldberg and O. Levy, "Word2Vec explained: Deriving Mikolov's negative-sampling word-embedding method," *CoRR*, vol. abs/1402.3722, 2014. [Online]. Available: http://arxiv.org/abs/1402.3722
[47] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. Comput. Linguistics*, vol. 5, pp. 135–146, 2017. [Online]. Available: https://www.aclweb.org/anthology/Q17-1010
[48] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1532–1543.
[49] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. 31st Int. Conf. Int. Conf. Mach. Learn. (ICML)*, vol. 32. JMLR.org, 2014, pp. II–1188–II–1196.
[50] M. Pagliardini, P. Gupta, and M. Jaggi, "Unsupervised learning of sentence embeddings using compositional n-gram features," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1. New Orleans, LA, USA: Association for Computational Linguistics, Jun. 2018, pp. 528–540. [Online]. Available: https://www.aclweb.org/anthology/N18-1049
[51] M. G. Ozsoy, "From word embeddings to item recommendation," *CoRR*, vol. abs/1601.01356, 2016. [Online]. Available: http://arxiv.org/abs/1601.01356
[52] C. Samarinas and S. Zafeiriou, "Personalized high quality news recommendations using word embeddings and text classification models," EasyChair Preprint 1254, 2019.
[53] C. Bhagavatula, S. Feldman, R. Power, and W. Ammar, "Content-based citation recommendation," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1. New Orleans, LA, USA: Association for Computational Linguistics, Jun. 2018, pp. 238–251. [Online]. Available: https://www.aclweb.org/anthology/N18-1022
[54] R. N. Nandi, M. M. A. Zaman, T. Al Muntasir, S. H. Sumit, T. Sourov, and M. J.-U. Rahman, "Bangla news recommendation using doc2vec," in *Proc. Int. Conf. Bangla Speech Lang. Process. (ICBSLP)*, Sep. 2018, pp. 1–5.
[55] D. Khattar, V. Kumar, M. Gupta, and V. Varma, "Neural content-collaborative filtering for news recommendation," in *Proc. NewsIR@ECIR*, 2018, pp. 45–50.
[56] C. Musto, G. Semeraro, M. de Gemmis, and P. Lops, "Learning word embeddings from wikipedia for content-based recommender systems," in *Advances in Information Retrieval*, N. Ferro, F. Crestani, M.-F. Moens, J. Mothe, F. Silvestri, G. M. Di Nunzio, C. Hauff, and G. Silvello, Eds. Cham, Switzerland: Springer, 2015, pp. 729–734.
[57] C. Musto, G. Semeraro, M. Degemmis, and P. Lops, "Word embedding techniques for content-based recommender systems: An empirical evaluation," in *Proc. RecSys Posters*, 2015.
[58] C. Wu, F. Wu, M. An, J. Huang, Y. Huang, and X. Xie, "NPA: Neural news recommendation with personalized attention," *CoRR*, 2019.
[59] C. Wu, F. Wu, M. An, H. Yongfeng, and X. Xie, "Neural news recommendation with topic-aware news representation," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*. Florence, Italy: Association for Computational Linguistics, 2019, pp. 1154–1159.
[60] L. Zhang, P. Liu, and J. A. Gulla, "Dynamic attention-integrated neural network for session-based news recommendation," *Mach. Learn.*, vol. 108, no. 10, pp. 1851–1875, Oct. 2019.
[61] J.-D. Zhang and C.-Y. Chow, "SEMA: Deeply learning semantic meanings and temporal dynamics for recommendations," *IEEE Access*, vol. 6, pp. 54106–54116, 2018.
[62] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Advances in Neural Information Processing Systems*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Red Hook, NY, USA: Curran Associates, 2013, pp. 2787–2795.
[63] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proc. AAAI*, 2014, pp. 1112–1119.
[64] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 2181–2187.
[65] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process.* Beijing, China: Association for Computational Linguistics, 2015, pp. 687–696.
[66] G. Ji, K. Liu, S. He, and J. Zhao, "Knowledge graph completion with adaptive sparse transfer matrix," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, p. 985–991.
[67] H. Wang, F. Zhang, X. Xie, and M. Guo, "DKN: Deep knowledge-aware network for news recommendation," in *Proc. 27th Int. Conf. World Wide Web (WWW)*, 2018, pp. 1835–1844.
[68] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, "Collaborative knowledge base embedding for recommender systems," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, 2016, pp. 353–362.
[69] J. Ren, J. Long, and Z. Xu, "Financial news recommendation based on graph embeddings," *Decis. Support Syst.*, vol. 125, Oct. 2019, Art. no. 113115.
[70] B. Smith and G. Linden, "Two decades of recommender systems at Amazon.com," *IEEE Internet Comput.*, vol. 21, no. 3, pp. 12–18, May 2017.
[71] G. Piao and J. G. Breslin, "Measuring semantic distance for linked open data-enabled recommender systems," in *Proc. 31st Annu. ACM Symp. Appl. Comput.*, New York, NY, USA, 2016, pp. 315–320.
[72] R. C. Bagher, H. Hassanpour, and H. Mashayekhi, "User trends modeling for a content-based recommender system," *Expert Syst. Appl.*, vol. 87, pp. 209–219, Nov. 2017.
[73] Y. Xiao, P. Ai, C.-H. Hsu, H. Wang, and X. Jiao, "Time-ordered collaborative filtering for news recommendation," *China Commun.*, vol. 12, no. 12, pp. 53–62, Dec. 2015.
[74] L. Deng and D. Yu, "Deep learning: Methods and applications," *Found. Trends Signal Process.*, vol. 3, no. 3, pp. 197–387, 2014.
[75] M. Adnan, M. Chowdury, I. Taz, T. Ahmed, and R. M. Rahman, "Content based news recommendation system based on fuzzy logic," in *Proc. Int. Conf. Informat., Electron. Vis. (ICIEV)*, Los Alamitos, CA, USA, 2014, pp. 1–6.
[76] C. Gao, X. He, D. Gan, X. Chen, F. Feng, Y. Li, T.-S. Chua, and D. Jin, "Neural multi-task recommendation from multi-behavior data," in *Proc. IEEE 35th Int. Conf. Data Eng. (ICDE)*, Apr. 2019, pp. 1554–1557.
[77] V. N. Marivate, G. Ssali, and T. Marwala, "An intelligent multi-agent recommender system for human capacity building," in *Proc. 14th IEEE Medit. Electrotech. Conf. (MELECON)*, May 2008, pp. 909–915.
[78] C. Chen, X. Meng, Z. Xu, and T. Lukasiewicz, "Location-aware personalized news recommendation with deep semantic analysis," *IEEE Access*, vol. 5, pp. 1624–1638, 2017.
[79] J. Shu, X. Shen, H. Liu, B. Yi, and Z. Zhang, "A content-based recommendation algorithm for learning resources," *Multimedia Syst.*, vol. 24, no. 2, pp. 163–173, Mar. 2018.
[80] X. Wang, L. Yu, K. Ren, G. Tao, W. Zhang, Y. Yu, and J. Wang, "Dynamic attention deep model for article recommendation by learning human editors' demonstration," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, 2017, pp. 2051–2059.
[81] S. Okura, Y. Tagami, S. Ono, and A. Tajima, "Embedding-based news recommendation for millions of users," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, New York, NY, USA, 2017, pp. 1933–1942.
[82] J. Huang, W. Zhang, Y. Sun, H. Wang, and T. Liu, "Improving entity recommendation with search log and multi-task learning," in *Proc. 27th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2018, pp. 4107–4114.
[83] I. Fernández-Tobías and R. Blanco, "Memory-based recommendations of entities for Web search users," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, New York, NY, USA, 2016, pp. 35–44.

[84] B. Bai, Y. Fan, W. Tan, and J. Zhang, "DLTSR: A deep learning framework for recommendations of long-tail Web services," *IEEE Trans. Services Comput.*, vol. 13, no. 1, pp. 73–85, Jan. 2020.
[85] G. Zheng, F. Zhang, Z. Zheng, Y. Xiang, N. J. Yuan, X. Xie, and Z. Li, "DRN: A deep reinforcement learning framework for news recommendation," in *Proc. World Wide Web Conf.* Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2018, pp. 167–176.
[86] S. K. Cheekula, P. Kapanipathi, D. Doran, P. Jain, and A. P. Sheth, "Entity recommendations using hierarchical knowledge bases," in *Proc. KNOW@LOD*, 2015, pp. 1–13.
[87] M. Zhu, D.-S. Zhen, R. Tao, Y.-Q. Shi, X.-Y. Feng, and Q. Wang, "Top-N collaborative filtering recommendation algorithm based on knowledge graph embedding," in *Knowledge Management in Organizations*, L. Uden, I.-H. Ting, and J. M. Corchado, Eds. Cham, Switzerland: Springer, 2019, pp. 122–134.
[88] Z. Sun, J. Yang, J. Zhang, A. Bozzon, L.-K. Huang, and C. Xu, "Recurrent knowledge graph embedding for effective recommendation," in *Proc. 12th ACM Conf. Rec. Syst.*, New York, NY, USA, 2018, pp. 297–305.
[89] I. Vagliano, D. Monti, A. Scherp, and M. Morisio, "Content recommendation through semantic annotation of user reviews and linked data—An extended technical report," 2017, *arXiv:1709.09973*. [Online]. Available: https://arxiv.org/abs/1709.09973
[90] *RecSys2013: Yelp Business Rating Prediction*. Accessed: Sep. 15, 2020. [Online]. Available: https://www.kaggle.com/c/yelp-recsys-2013/data?select=yelptestset.zip
[91] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, "Improving recommendation lists through topic diversification," in *Proc. 14th Int. Conf. World Wide Web (WWW)*, 2005, pp. 22–32.
[92] B. Kille, F. Hopfgartner, T. Brodt, and T. Heintz, "The plista dataset," in *Proc. Int. News Rec. Syst. Workshop Challenge*, New York, NY, USA, 2013, pp. 16–23.
[93] Y. Ji, W. Hong, Y. Shangguan, H. Wang, and J. Ma, "Regularized singular value decomposition in news recommendation system," in *Proc. 11th Int. Conf. Comput. Sci. Educ. (ICCSE)*, Aug. 2016, pp. 621–626.
[94] J. A. Gulla, L. Zhang, P. Liu, O. Özgöbek, and X. Su, "The adressa dataset for news recommendation," in *Proc. Int. Conf. Web Intell.*, New York, NY, USA, 2017, pp. 1042–1048.
[95] G. Moreira. *News Portal User Interactions by Globo.com*. Accessed: Sep. 15, 2020. [Online]. Available: https://www.kaggle.com/gspmoreira/news-portal-user-interactions-by-globocom
[96] W. Gu, S. Dong, and M. Chen, "Personalized news recommendation based on articles chain building," *Neural Comput. Appl.*, vol. 27, no. 5, pp. 1263–1272, Jul. 2016.
[97] Rohk. *The Examiner—Spam Clickbait Catalog*. Accessed: Sep. 15, 2020. [Online]. Available: https://www.kaggle.com/therohk/examine-the-examiner
[98] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proc. 19th Int. Conf. World Wide Web (WWW)*, 2010, pp. 661–670.
[99] F. Wu, Y. Qiao, J.-H. Chen, C. Wu, T. Qi, J. Lian, D. Liu, X. Xie, J. Gao, W. Wu, and M. Zhou, "MIND: A large-scale dataset for news recommendation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, Jul. 2020, pp. 3597–3606. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.331
[100] Rohk. *Aminer*. Accessed: Sep. 15, 2020. [Online]. Available: https://www.aminer.org/
[101] C. Cai, R. He, and J. McAuley, "SPMC: Socially-aware personalized Markov chains for sparse sequential recommendation," 2017, *arXiv:1708.04497*. [Online]. Available: https://arxiv.org/abs/1708.04497
[102] H. Gao, J. Tang, and H. Liu, "Exploring social-historical ties on location-based social networks," in *Proc. 6th Int. AAAI Conf. Weblogs Social Media (ICWSM)*, 2012, pp. 1–8.
[103] GroupLens. *MovieLens*. Accessed: Sep. 15, 2020. [Online]. Available: https://grouplens.org/datasets/movielens/
[104] K. Goldberg. *Jester Datasets for Recommender Systems and Collaborative Filtering Research*. Accessed: Sep. 15, 2020. [Online]. Available: http://eigentaste.berkeley.edu/dataset/
[105] M. Li, Y. Li, W. Lou, and L. Chen, "A hybrid recommendation system for Q&A documents," *Expert Syst. Appl.*, vol. 144, Apr. 2020, Art. no. 113088.
[106] A. Gunawardana and G. Shani, *Evaluating Recommender Systems*. Boston, MA, USA: Springer, 2015, pp. 265–308.
[107] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, Jan. 2004.
[108] M. S. Desarkar and N. Shinde, "Diversification in news recommendation for privacy concerned users," in *Proc. Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2014, pp. 135–141.
[109] K. Bondarenko. *Precision and Recall in Recommender Systems. And Some Metrics Stuff*. Accessed: Sep. 15, 2020. [Online]. Available: https://bond-kirill-alexandrovich.medium.com/precision-and-recall-in-recommender-systems-and-some-metrics-stuff-ca2ad385c5f8
[110] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 143–177, Jan. 2004.
[111] T. Murakami, K. Mori, and R. Orihara, "Metrics for evaluating the serendipity of recommendation lists," in *New Frontiers in Artificial Intelligence*, K. Satoh, A. Inokuchi, K. Nagao, and T. Kawamura, Eds. Berlin, Germany: Springer, 2008, pp. 40–46.
[112] H. Cheng. *MIND News Recommendation Technical Report*. Accessed: Sep. 15, 2020. [Online]. Available: https://msnews.github.io/assets/doc/1.pdf
[113] G. R. Chughtai, J. Lee, M. Shahzadi, A. Kabir, and M. A. S. Hassan, "An efficient ontology-based topic-specific article recommendation model for best-fit reviewers," *Scientometrics*, vol. 122, no. 1, pp. 249–265, Jan. 2020.
[114] K. Bauman, B. Liu, and A. Tuzhilin, "Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, 2017, pp. 717–725.
[115] K. Saranya and G. S. Sadhasivam, "A personalized online news recommendation system," *Int. J. Comput. Appl.*, vol. 57, no. 18, pp. 1–9, 2012.
[116] A. Gershman, T. Wolfe, E. Fink, and J. G. Carbonell, "News personalization using support vector machines," Tech. Rep., 2011.
[117] T. Bansal, M. Das, and C. Bhattacharyya, "Content driven user profiling for comment-worthy recommendations of news and blog articles," in *Proc. 9th ACM Conf. Rec. Syst. (RecSys)*, New York, NY, USA, 2015, pp. 195–202.
[118] M. Kompan and M. Bielikova, "Content-based news recommendation," in *Proc. Int. Conf. Electron. Commerce Web Technol.*, 2010, pp. 61–72.
[119] L. Hu, C. Li, C. Shi, C. Yang, and C. Shao, "Graph neural news recommendation with long-term and short-term interest modeling," *Inf. Process. Manage.*, vol. 57, no. 2, Mar. 2020, Art. no. 102142.
[120] S. Rosenthal, M. M. Veloso, and A. K. Dey, "Online selection of mediated and domain-specific predictions for improved recommender systems," in *Proc. 7th Workshop Intell. Techn. Web Personalization Recommender Syst. (ITWP), 21st Int. Joint Conf. Artif. Intell. (IJCAI)*, in CEUR Workshop Proceedings, vol. 528. Pasadena, CA, USA: CEUR-WS.org, Jul. 2009.
[121] F. García-Sánchez, R. Colomo-Palacios, and R. Valencia-García, "A social-semantic recommender system for advertisements," *Inf. Process. Manage.*, vol. 57, no. 2, Mar. 2020, Art. no. 102153.
[122] H. Ma, X. Liu, and Z. Shen, "User fatigue in online news recommendation," in *Proc. 25th Int. Conf. World Wide Web*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2016, pp. 1363–1372.
[123] N. T. Huy, D. T. Chuan, and V. A. Nguyen, "Implicit feedback mechanism to manage user profile applied in vietnamese news recommender system," *Int. J. Comput. Commun. Eng.*, vol. 5, no. 4, pp. 276–285, 2015.
[124] N. Muralidhar, H. Rangwala, and E.-H.-S. Han, "Recommending temporally relevant news content from implicit feedback data," in *Proc. IEEE 27th Int. Conf. Tools With Artif. Intell. (ICTAI)*, Nov. 2015, pp. 689–696.
[125] J. Liu, P. Dolan, and E. R. Pedersen, "Personalized news recommendation based on click behavior," in *Proc. 15th Int. Conf. Intell. User Interfaces*. New York, NY, USA: Association for Computing Machinery, 2010, p. 31–40.
[126] L. Li, D. Wang, T. Li, D. Knox, and B. Padmanabhan, "Scene: A scalable two-stage personalized news recommendation system," in *Proc. 34th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.* New York, NY, USA: Association for Computing Machinery, 2011, pp. 125–134.

SAFIA KANWAL is currently pursuing the Ph.D. degree with the Punjab University College of Information Technology (PUCIT). She has been over five years of software development experience. She has been associated with the University of Gujrat. Her research interests include natural language processing, machine learning, and recommender systems.

SIDRA NAWAZ is currently pursuing the M.Phil. degree with the Punjab University College of Information Technology (PUCIT). Her research interests include natural language processing, machine learning, and data science.



MUHAMMAD KAMRAN MALIK received the master's degree from NU-Fast and the Ph.D. degree from the Punjab University College of Information Technology (PUCIT). He is currently an Associate Professor with the University of the Punjab, Lahore. He closely works with industry as a Consultant. He has published several articles in reputed journals, such as ACM *TALLIP* and *Scientometrics*. He holds a patent. His research interests include machine learning and natural language processing.

ZUBAIR NAWAZ received the Ph.D. degree in computer engineering from the Delft University of Technology, in 2011. He is currently an Assistant Professor with the Punjab University College of Information Technology (PUCIT). He is working in the field of data science. His research interests include data science, machine learning, high-performance computing, compiler optimization, and scientific computing.