IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 11, Issue 2, February 2023

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.165

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165|

|| Volume 11, Issue 2, February 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1102013|

An Text Classification Extracting the Main Topics from Your Dataset

DR.M.GEETHA, SOWNDARRAJAN N

Associate Professor, Department of Computer Applications (MCA), K.S.R. College of Engineering (Autonomous), Tiruchengode, India

Department of Computer Applications (MCA), K.S.R. College of Engineering (Autonomous), Tiruchengode, India

ABSTRACT: The ability to determine the topic of a large set of text documents using relevant keywords is usually regarded as a very tedious task if done by hand. The problem of keyword extraction from conversations, with the goal of using these keywords to fetch, for each short conversation fragment, a small number of potentially relevant documents, which can be recommended to participants, is addressed. A proposed algorithm is to first extract keywords from the conversation which makes use of topic modeling techniques. The keywords which have highest similarity are taken as keywords. Then, a technique to derive multiple topically separated queries from this keyword set, in order to maximize the chances of making at least one related recommendation when using these queries to search over the English Wikipedia. The keywords extracted by the algorithm are highly accurate and fit the cluster topic. Keywords are extracted from documents to classify the documents.

KEYWORDS: keyword extraction, Document recommendation, topic modeling, clustering, text classification.

I. INTRODUCTION

HUMANS are surrounded by an unprecedented wealth of information, available as documents, databases, or multimedia resources. Access to this information is conditioned by the availability of suitable search engines, but even when these are available, users often do not initiate a search, because their current activity does not allow them to do so, or because they are not aware that relevant information is available. We adopt in this paper the perspective of just-in-time retrieval, which answers this shortcoming by spontaneously recommending documents that are related to users' current activities. When these activities are mainly conversational, for instance when users participate in a meeting, their information needs can be modeled as implicit queries that are constructed in the background from the pronounced words, obtained through real-time automatic speech recognition (ASR). These implicit queries are used to retrieve and recommend documents from the Web or a local repository, which users can choose to inspect in more detail if they find them interesting.

The focus of this paper is on formulating implicit queries to a just-in-time-retrieval system for use in meeting rooms. In contrast to explicit spoken queries that can be made in commercial Web search engines, our just-in-time-retrieval system must construct implicit queries from conversational input, which contains a much larger number of words than a query. For instance, in which four people put together a list of items to help them survive in the mountains, pertaining to a variety of domains, such as 'chocolate', 'pistol', or 'lighter'. A comparable research topic is called "automatic term recognition" in the context of computational linguistics and "automatic indexing" or "automatic keyword extraction" in information retrieval research. What would then be the most helpful 3–5 Wikipedia pages to recommend, and how would a system determine them?

Given the potential multiplicity of topics, reinforced by potential ASR errors or speech disfluencies (such as 'whisk' in this example), our goal is to maintain multiple hypotheses about users' information needs, and to present a small sample of recommendations based on the most likely ones. Therefore, we aim at extracting a relevant and diverse set of keywords, cluster them into topic-specific queries ranked by importance, and present users a sample of results from these queries. The topic-based clustering decreases the chances of including ASR errors into the queries, and the diversity of keywords increases the chances that at least one of the recommended documents answers a need for information, or can lead to a useful document when following its hyperlinks. For instance, while a method based on word frequency would retrieve the following Wikipedia pages: 'Light', 'Lighting', and 'Light My Fire' for the above-mentioned fragment, users would prefer a set such as 'Lighter', 'Wool' and 'Chocolate'.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165|

|| Volume 11, Issue 2, February 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1102013|

Relevance and diversity can be enforced at three stages: when extracting the keywords; when building one or several implicit queries; or when re-ranking their results. The first two approaches are the focus of this paper. Our recent experiments with the third one, published separately [1], show that re-ranking of the results of a single implicit query cannot improve users' satisfaction with the recommended documents. Previous methods for formulating implicit queries from text rely on word frequency or TFIDF weights to rank keywords and then select the highest ranking ones [2], [3].In this paper, introduce a novel keyword extraction technique from ASR output, which maximizes the coverage of potential information needs of users and reduces the number of irrelevant words. Once a set of keywords is extracted, it is clustered in order to build several topically-separated queries, which are run independently, offering better precision than a larger, topically-mixed query. Results are finally merged into a ranked set before showing them as recommendations to users.

II. RELATED WORKS

Previous research on application of keyword extraction and classification techniques of data mining briefly reviewed in the following paragraphs.

Jordan et al [10] described Latent Dirichlet Allocation (LDA), a generative probabilistic model for set of discrete data such as text corpora. The results in document modeling, text classification, and collaborative filtering, analyze to mixture of unigrams model and the probabilistic Latent Semantics Analysis model.

Ishizuka et al [4] developed a new keyword extraction algorithm that applies to a single document without applying a corpus. Frequent terms are extracted first, and then a set of co-occurrences among each term and the frequent terms. The degree of bias of the co-occurrence distribution is measured by the $\chi 2$-measure.

Cong wong et al [7] presented a keyword extraction algorithm based on WordNet and PageRank. Pagerank is an algorithm of deciding the importance of vertices in a graph. The result shows that the algorithm is effective and practical.

Shiren Ye et al [2] proposed a Document Concept Lattice (DCL) summarization that indexes the hierarchy of local topics tied to a set of frequent approach and the corresponding sentences consists of these topics. In the procedure of constructing DCL, all sentences in documents are represented by a basket of concepts in base nodes, and frequent concept sets mined from these base nodes will form the derived nodes.

Deana Pennell et al [9] evaluated graph-based approach that measures the importance of a word based on its connection with other words or sentences. The results have shown that the simple unsupervised TFIDF technique performs reasonably well, and the additional information from POS and sentence score helps keyword extraction.

Zhiyuan Liu et al [8] proposed a new graph-based frame-work, Topical PageRank, which incorporates topic information within random walk for keyphrase extraction, thus build a Topical PageRank (TPR) on word graph to measure word relevance with respect to different topics. After that, given the topic distribution of the document, we further measure the ranking scores of words and extract the top ranked ones as keyphrases.

Jeff Bilmes et al [11] designed a class of submodular functions meant for document summarization tasks. These tasks each combine two terms, one which encourages the succinct to be representative of the corpus, and the more which positively rewards diversity. As the corresponding submodular optimization problem can be solved effectively and efficiently.

Ani Nenkova et al [5] illustrated a numerous approaches for identifying important content for automatic text summarization. Topic representation approaches first derive an intermediate representation of the text that taking the topics discussed in the input. Finally, a summary is composed by selecting sentences in a greedy technique, determine the sentences that will go in the summary one by one, or globally optimizing the selection, choosing the best set of sentences to form a summary.

David Harwath et al [3] illustrated Document summarization algorithms are most commonly calculated according to the intrinsic quality of the summaries they produce. The results appear to be correlated with the achievement of an automated topic identification system, and argue that this automated system can move as a low-cost proxy for a human evaluation during the development stages of a summarization system.

Wongkot Sriurai [13] presented a three text categorization algorithms: Naive Bayes (NB), Support Vector Machines (SVM) and Decision tree. The Latent Dirichlet Allocation algorithm is used to cluster the term features into a set of latent topics. From the experimental results, the approach of feature representation with the topic model and using Support Vector Machines to learn the classification model, and yield the best performance.

Menaka [12] described Text Classification algorithm using keyword extraction. Keywords are extracted from documents using TF-IDF and WordNet. Then documents are classified based on extracted keywords using the machine learning algorithms -Naïve Bayes, Decision Tree and k-Nearest Neighbor. The performance analysis of machine learning algorithms for text classification shows that the Decision Tree algorithm gives better results based on prediction accuracy when compared to other two algorithms.

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165|

|| Volume 11, Issue 2, February 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1102013|

III. PROPOSED WORK

A two-stage approach is proposed.

1. Extraction of keywords from the transcript of a conversation fragment for which documents must be recommended, as provided by a conversation fragment.
2. Clustering of the keyword set.
3. Classify the keywords based on k-Nearest Neighbor.

A. Diverse Keyword Extraction

Topic modeling techniques is to build a topical representation of a conversation fragment, and then select content words as keywords by using topical similarity, while also rewarding the coverage of a diverse range of topic. The benefit of diverse keyword extraction is that the coverage of the main topics of the conversation fragment is maximized. Moreover, in order to cover more topic,the proposed algorithm

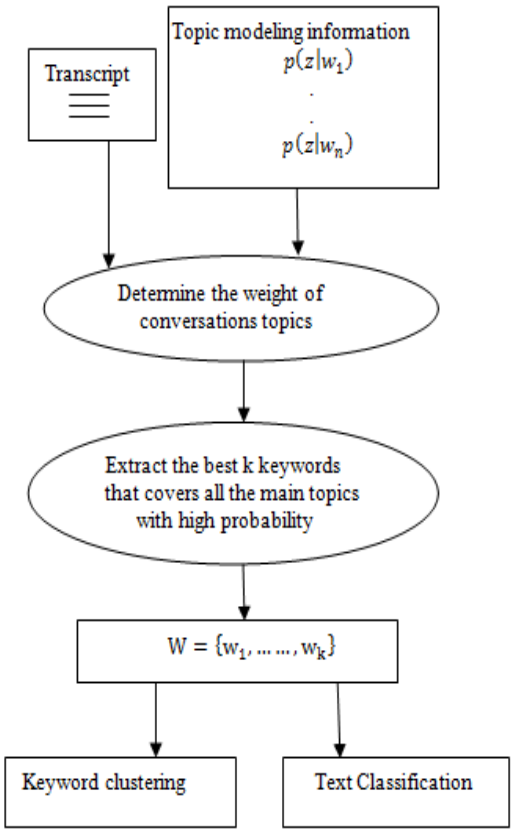

Fig. 1 Keyword based Document Classification

will select a smaller number of keywords from each topic.

The proposed method for diverse keyword extraction proceeds in three steps, represented schematically in Fig. 1

1. A topic model is used to represent the distribution of the abstract topic for each word noted as $p(z|w)$ depicted in Fig. 1
2. These topic models are used to determine weights for the abstract topics in each conversation fragment represented by β_z.
3. Finally, the keyword list $w = \{w_1, \ldots . w_k\}$ which covers a maximum number of the most important topics are selected.

Modeling Topics in Conversations: Topic models such as Latent Dirichlet Allocation (LDA) [10] can be used as topic modeling techniques to determine the distribution over the topic z of each word w, note $p(z|w)$ from a large amount of training documents. LDA implemented in the Mallet toolkit is used in this paper because it does not suffer from the overfitting issue of PLSA.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165|

|| Volume 11, Issue 2, February 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1102013|

$$\beta_z = \frac{1}{N} \sum_{1 \le i \le N} p(z \mid w_i) \qquad (1)$$

where,

β_z = topic weight

N = total no. of words

$p(z \mid w_i)$ = probabilities of word w_i for the topic z

Diverse Keyword Extraction Problem: The objective of the keyword extraction technique with maximal topic coverage. If a conversation fragment mentions t set of topics z, and each word w from the fragment t can evoke a subset of the topics in z, then the goal is to find a subset of k unique words $S \subseteq t$, with $|S| = k$, which maximizes the number of covered topics.

To achieve the goal, the contribution of topic z, with respect to each set of words $S \subseteq t$ of size by summing over all probabilities $p(z \mid w)$ of the words in the set. Afterward, define a reward function for each set S and topic z, to model the contribution of the set S to the topic z. Finally, select one of the sets $S \subseteq t$ which maximize the cumulative reward values over all the topics.

Definition of a Diverse Reward Function: By introducing $r_{S,z}$, the contribution towards topic z of the keyword set S selected from the fragment t :

$$r_{S,z} = \sum_{w \in S} p(z \mid w_i) \qquad (2)$$

Finally, the keyword set $S \subseteq t$, is chosen by maximizing the cumulative reward function over all the topics, formulated as follows:

$$R(S) = \sum_{z \in Z} \beta_z . r_{S,z}^{\lambda} \qquad (3)$$

where,

$R(S)$ is a monotone non-decreasing submodular function

λ is a parameter between 0 and 1

If $\lambda = 1$, the reward function is linear and only measures the topical similarity of words with the main topics of z. However, when $0 \le \lambda \le 1$, as soon as a word is selected from a topic.

B. Keyword Clustering

Document clustering has been widely used in information retrieval system for improving the precision of retrieval results. The goal of clustering is categorize or grouping similar data items together. A common document clustering method is the one that first calculates the similarities between all pair of the documents and then cluster documents together if the similarity values above some threshold. Document clustering has been used in a number of applications. In the information system, it has been used to improve the precision and recall performance, and as an efficient way to find similar documents.

The proposed approach in this paper is first eliminating stopwords in the document and then finds nouns and verbs in the document with using dictionary. These verbs and nouns are help to find the keywords in the document.

Algorithm 1: Diverse Keyword Extraction

Using keyword clustering to cluster documents also can reduce the feature dimensions of the clustering algorithm. This method cluster method cluster documents by joining words that have similar probability distributions among the target words that co-occur that words in verbs and nouns.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165|

|| Volume 11, Issue 2, February 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1102013|

Input: a given text t, a set of topics Z, the number of keywords k

Output: a set of keywords S

$S \leftarrow \varnothing$

While $|S| \leq k$ **do**

$$S \leftarrow S \cup \{\arg\max_{w \in t \setminus S} \left(h\left(w, S\right)\right) \text{ where}$$

$$h\left(w, S\right) = \sum_{z \in Z} \beta_z \left[p\left(z \mid w\right) + r_{S,z} \right]^{\lambda};$$

end

return S

C. Text Classification

Text classification is one of the main functions of machine learning. The task is to assign unlabeled new text document to predefined class. The processing of text classification involves two main problems, first problem is the extraction of feature terms that become efficient keywords in the training phase and then the second is actual classification of the document using these aspect terms in the test phase. Before classifying documents, preprocessing has done. In preprocessing stop words are evacuated and the words are stemmed. Then the term frequency is calculated for each term in a document and also TF-IDF is determined.

Keyword Extraction: Keywords can be considered as condensed version of documents and short forms of their analysis. Keyword extraction is a significant technique for number of text mining related tasks such as document retrieval, webpage retrieval, document clustering and summarization. The main aim of keyword extraction is to extract the keywords with respect to their importance in the text. First step is to select the desired documents and it can be preprocessed.

Stop Words Elimination: Stop words are a part of natural language that does not have so much meaning in a retrieval scheme. The reason that stop-words should be removed from a text is that they make the text look heavier and less relevent for analysts. Removing stop words reduces the dimensionality of term space. The most frequent words are in text documents are prepositions, articles, and pro-nouns etc that does not provide the meaning of the documents. These words are considered as stop words. Example for stop words: the, in, a, an, with, etc. Stop words are eliminated from documents because those words are not treated as keywords in text mining applications.

Stemming: Stemming approach is used to find out the root/stem of a word. Stemming converts words to their stems which incorporates a big deal of language-dependent linguistic knowledge. For example, the words, connection, connects, connected all can be stemmed to the word 'connect'. In the present work, the Porter Stemmer method is used which is the most commonly used algorithm in English.

Term Frequency-Inverse Document Frequency: Term Frequency–Inverse Document Frequency (tf-idf) is a numerical statistic which explain that a word is how important to a document in a collection. Tf-idf is often used as a weighting element in information retrieval and text mining. The value of tf-idf increases proportionally to the number of times a word occur in the document, but is counteracting by the frequency of the word in the corpus.

Term Frequency- Term Frequency (TF) is describing as number of times a term occurs in a document.

$$\text{tf}\left(\text{t},\text{d}\right) = 0.5\frac{0.5 \times \text{f}\left(\text{t},\text{d}\right)}{\max.\text{wordoccurrence}} \tag{4}$$

Where,

f(t,d) denotes the frequency of occurrences of term t

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165|

|| Volume 11, Issue 2, February 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1102013|

Inverse Document Frequency- Inverse Document Frequency (IDF) is a statistical weight used for measuring the importance of a term in a text document collection.

$$\text{idf}\left(t,d\right)=\log\frac{\left|D\right|}{\left(\text{no.ofdoctermst}\right)} \tag{5}$$

IV. EXPERIMENTAL RESULTS

The proposed model is implemented using Intel Pentium i3 processor with RAM capacity of 4 GB. The hard disk capacity is 500 GB. Windows 8.1 operating system is used. The algorithm is implemented in .java net beans.

In this section, the diverse keyword extraction technique, extracts more relevant keywords, which cover more topics. The experiment is done by using conversations and the classification is performed using an open source tool. RapidMiner is an environment for machine learning, predictive analytics, data mining, text mining, and business analytics. This tool is used for research, discipline, education, application development, rapid prototyping, and industrial applications. It provides data mining and machine learning procedures including data loading and transformation, data pre-processing and visualization, modeling, evaluation, and deployment. This tool is written in Java programming language, it uses learning schemes. The proposed work is experimented by using 20 conversations. Conversations are collected manually. The efficient keywords from the fragments are extracted using TF-IDF and WordNet. This keyword extraction process is developed in Java. Then the extracted keywords are stored for clustering and classification.

The efficiency of clustering the keywords is less when compared to classification. Classification comes under supervised learning. The prediction accuracy and the training time are two conditions used to evaluate the performances of the trained models and the prediction accuracy of the each model is compared.

V. CONCLUSION

Text classification is one of the major applications of machine learning. The proposed method use text mining algorithms to extract keywords from journal papers. The WordNet dictionary is used to calculate the semantic distances between the keywords. The extracted keywords are having the highest similarity. Then documents are classified based on extracted keywords using the machine learning algorithms - Naïve Bayes, Decision Tree and k-Nearest Neighbor. The performance analysis of machine learning algorithms for text classification shows that the Decision Tree algorithm gives better results based on prediction accuracy when compared to other two algorithms.

REFERENCES

[1] M. Habibi and A. Popescu-Belis, "Enforcing topic diversity in a document recommender for conversations," in Proc. 25th Int. Conf. Comput. Linguist. (Coling), 2014, pp. 588–599.J.

[2] S. Ye, T.-S. Chua, M.-Y. Kan, and L. Qiu, "Document concept lattice for text understanding and summarization," Inf. Process. Manage., vol. 43, no. 6, pp. 1643–1662, 2007.

[3] D. Harwath and T. J. Hazen, "Topic identification based extrinsic evaluation of summarization techniques applied to conversational speech," in Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP), 2012, pp. 5073–5076.

[4] Y. Matsuo and M. Ishizuka, "Keyword extraction from a single document using word co-occurrence statistical information," Int. J. Artif. Intell. Tools, vol. 13, no. 1, pp. 157–169, 2004.

[5] A. Nenkova and K. McKeown, "A survey of text summarization techniques," in Mining Text Data, C. C. Aggarwal and C. Zhai, Eds. New York, NY, USA: Springer, 2012, ch. 3, pp. 43–76.

[6] T. J. Hazen, "Latent topic modeling for audio corpus summarization," in Proc. 12th Annu. Conf. Int. Speech Commun. Assoc., 2011, pp. 913–916.

[7] J. Wang, J. Liu, and C. Wang, "Keyword extraction based on pagerank," in Proc. Adv. Knowl. Disc. Data Mining (PAKDD), 2007, pp. 857–864.

[8] Z. Liu, W. Huang, Y. Zheng, and M. Sun, "Automatic keyphrase extraction via topic decomposition," in Proc. Conf. Empir. Meth. Nat. Lang. Process. (EMNLP'10), 2010, pp. 366–376.

[9] F. Liu, D. Pennell, F. Liu, and Y. Liu, "Unsupervised approaches for automatic keyword extraction using meeting transcripts," in Proc. Annu. Conf. North Amer. Chap. ACL (HLT-NAACL), 2009, pp. 620–628.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation,"J. Mach. Learn. Res., 2003, vol. 3, pp. 993–1022,.

[11] H. Lin and J. Bilmes, "A class of submodular functions for document summarization," in Proc. 49th Annu. Meeting Assoc. Comput. Linguist. (ACL), Portland, OR, USA, 2011, pp. 510–520.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165|

|| Volume 11, Issue 2, February 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1102013|

[12] N.Radha and S.Menaka, "Text classification using keyword extraction technique", in Proceedings of the Int. Conf. adv. Res. Computer Science and Software Engineering(IJARCSSE), 2013, pp. 734-740.

[13] Wongkot Sriurai, "Improving text classification by using a topical model", in Adv. Computing:An Inernational Journal(ACIJ), 2011, vol. 2, pp. 21-27.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.165

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details