

Context Aware Text Classification Using Keywords

Rengitha R¹, Balu John²

Student, Department of Computer Science and Engineering, GEC Palakkad, Palakkad, India¹

Associate Professor, Department of Computer Science and Engineering, GEC Palakkad, Palakkad, India²

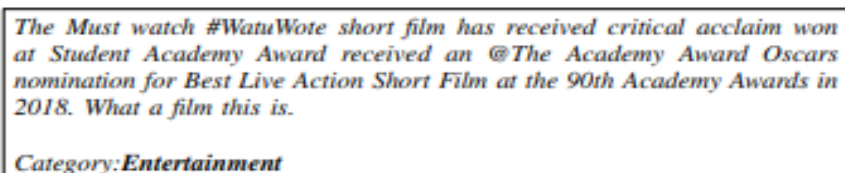
Abstract: Data, is the statistics and facts collected together for reference or analysis. The available data forms like numbers or text, and as facts stored in an individual's mind, or as bits stored in electronic system. Text is the most the basic portrayal of information. Natural Language processing is the emerging field in computer science and text classification is the process of assigning tag or label to the text. It is one of the fundamental task in NLP with many applications such as topic detection, intent detection, sentiment analysis, etc. Text classification can be done based on the different aspects, on which different works have been done. In this work, a novel context aware text classification system has been proposed which is built using a keyword extractor. The keyword extraction is the process of extracting relevant words or phrases from the text. Keyword extraction helps to find out the sense of the important words present in the text, and which subjects are being discussed. The extraction of the keyword is based on semantic knowledge which is available in taxonomies. These keywords are used for classification which is built using deep learning. The results show that the generated result is meaningful and context dependant in most of the cases.

Keywords: Keywords, Taxonomy, Pagerank.

I. INTRODUCTION

Data is often considered to be a distinct piece of information. This piece of information can be extracted from different sources like text, audio, video, images, etc. One of the commonly used format is text. Data analysis is one of the main trends that is happening in computer science field. It is the process of organizing and collecting data to get helpful conclusions from it. The procedure of data analysis utilizes intelligence also, expository thinking to pick up data from the information i.e, analytics. The primary reason for data analysis is to discover importance in data with the goal that the inferred knowledge can be utilized to make appropriate decisions. Natural Language Processing (NLP) is related to text with broad applications such as spam detection, topic labelling, intent detection and sentiment analysis. Text classification is a fundamental step in NLP. It is the process of assigning categories or labels to text according to its content. Some applications of text classification are:

- Sentiment Analysis
- News clustering systems
- Topic Labelling
- Intent Detection
- Question Answering systems
- Language Detection



The Must watch #WatuWote short film has received critical acclaim won at Student Academy Award received an @The Academy Award Oscars nomination for Best Live Action Short Film at the 90th Academy Awards in 2018. What a film this is.

Category:Entertainment

Fig. 1 Examples of text classification

An example of text classification is shown in Fig. 1. There are many ways that are available for text classification. The two broad categories are the classification based on the manual and automatic way. In manual text classification, the human annotator interprets the text and assign the category to that text. This method provides a quality of results but it is time consuming. The second way is the automatic way, the machine itself classifies the text using the techniques like machine learning, rule based approach, and hybrid systems. Proposing a context aware text classification that assigning category to the news articles by identifying the keywords present in the text. The proposed method uses the 5 predefined categories like business, politics, entertainment, education, technology. Keyword identification is done by using background knowledge. These Semantic enriched features help to improve the performance and robustness of the learned classifiers for the texts.



The basic idea of text classification is as shown in Fig. 2. The input to the system is a text, and output is the corresponding label for that text. The label assigned by the classifier, the middle segment is shown in the figure, whose working may vary according to the algorithm or approach used for the implementation. This work is about the classification based on the keyword and analyzing how well could it perform for the given text.

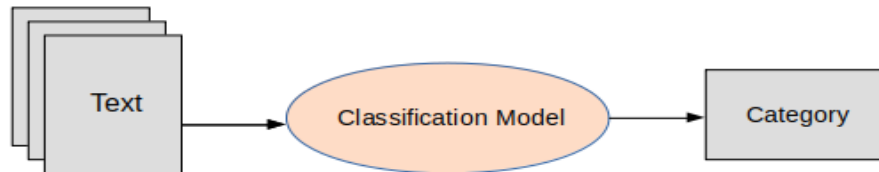


Fig. 2 Examples of text classification

II. RELATED WORK

The technique proposed by Kazuya Shimura et al. [1] is a multi-task learning structure based on the neural network system model, which trains a model to predict the dominant sense for the words in the document. This technique utilizes the highlights gained from dominating faculties are conceivable to segregate the area of the archive and hence improve the general execution of text order. The strategy depends on Centroid-Based Classifier (CBC) because of its hypothetical straightforwardness and computational proficiency proposed by [2] Chuan Liu et al. Classification accuracy of centroid based classifier greatly depends on the data distribution. The model named as Gravitation Model (GM) to solve the class-imbalanced classification problem. The model concentrates on the adjustment of classification hyperplane to reduce the biases inherent in CBC.

The text classification is based on the bag-of-embeddings model [3] by Peng Jin et al., which is firmly identified with the skip-gram model and bag-of-words model. The model integrates context information. It depends on the distributional the closeness between each target word inserting and the unique situation embeddings of words in a context window. Yaw-Huei Chen et al. [4] proposed a latent Dirichlet allocation (LDA) model on a text corpus, can learn the probabilities of topics appeared in a document and the probabilities of words in a topic. The topic distribution for the documents produced by the LDA model can be directly used as features for the classifier. Combining topic information and term frequency information may result in a better feature set and improve the accuracy of the text classifier.

A long-range relationship in the content can be effectively spoken to by utilizing a low-multifaceted nature word-level profound Convolutional Neural System (CNN) design [5]. Whenever it goes deeper in the neural networks the associated complexity also increases. This stance genuine difficulties in practical applications. Rie Johnson and Tong Zhang found a basic system design with which as well as can be expected be acquired by expanding the system profundity without expanding computational expenses by a lot. RadaMihalcea and PaulTarau [6] proposed a method to extract both keyword and sentence extraction. The textrank is the way of representing text in graphical frame-work working based on the voting or recommendation. If one vertex links to another one, it is considered as a vote for that other vertex. The higher the number of votes, which means the higher the importance of the vertex.

Extracting text keywords using WordNet [7], is the model used for extraction relevant words in the text or data. This work is implemented on the WordNet lexical database. It eliminates the affiliation words and for remaining words build a tree with many levels of additional generic terms like superordinate word or lexicographer lexemes. Using custom weights for each tree level and statistical analysis, extract a restricted number of words that are used to define the keywords of a document.

III. DATASET

Two publicly available datasets are used in the system: WordNet and BBC News articles. Here a BBC News dataset, originating from BBC News is used for the entire implementation. This BBC dataset is the collection of news archives related to stories in five topical areas from 2004-2005 from the BBC news site. The topical areas are business, entertainment, politics, sport, tech.

IV. METHODOLOGY

This section gives the system architecture and the overall implementation details of the proposed system. Fig. 3 shows the basic architecture of the proposed system. The overall system architecture consists of two sub-modules:

- Keyword extraction module
- Classifier module

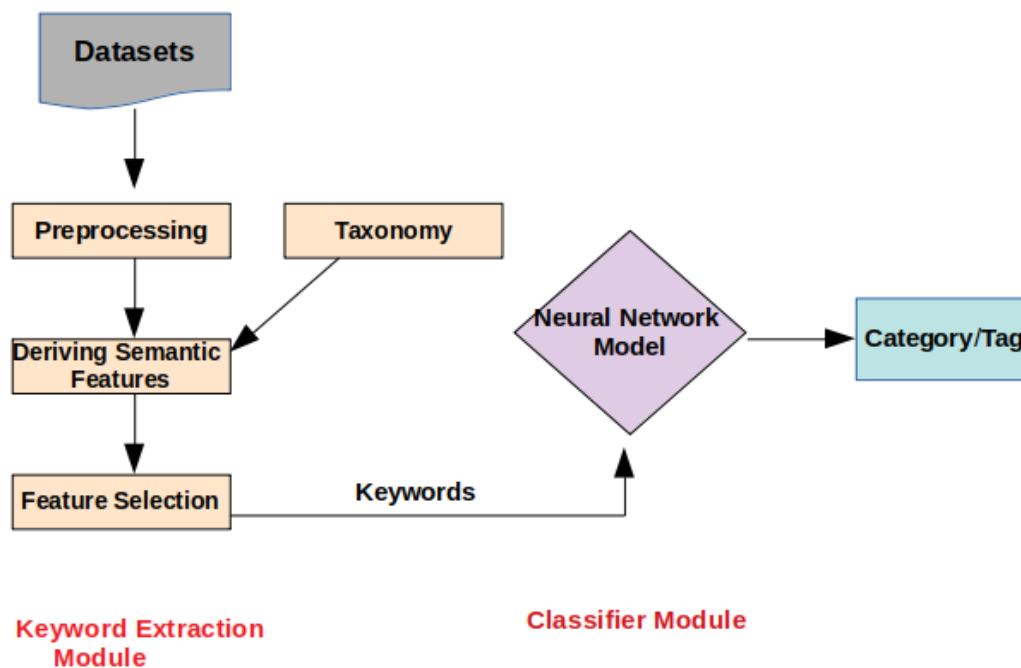


Fig. 3 System Architecture

A. Keyword Extraction Module

The module helps to find out the relevant words or phrases in the text. The important concept is the usage of hypernyms as potential features for semantic feature construction and feature selection for keyword extraction [8]. The steps are,

Load the Dataset and Taxonomy: The chosen dataset is BBC News articles composed of several paragraphs with its corresponding domain as the label. The dataset contains 2225 news documents from the BBC news website corresponding to stories in five topical areas from 2004-2005 and the topical areas are business, entertainment, politics, sport, tech. The WordNet taxonomy [9] is used for the construction of novel features such as background knowledge. WordNet is a lexical database of semantic relations like synonyms, hyponyms, and meronyms between words in more than 200 languages. The structure of WordNet makes it a valuable apparatus for computational semantics and normal language processing.

Preprocessing of Dataset: Preprocessing of data is termed as an important step while dealing with any NLP tasks. Datasets are preprocessed to remove any content that does not have any useful information. Stop word removal is the first preprocessing task that had done. It is a process of removing the words which does not add much meaning to a sentence. For example, the frequently used words like the, in, where, you, are, is, I, he, am, she, and that which has no meaning in actual are known as stop-words. Tokenization is a process of converting the text into tokens that retain all the essential information about the data. The next process is lemmatization, which means converting a word into its meaningful base, or its lemma. That is converting to the ground word. For example, there is no difference between write, wrote, and writing. The purpose of lemmatizing is to narrow everything down to its simplest level it can be [10].

Feature Construction: The proposed method for extracting a keyword from text uses the semantic feature of the text. The semantic feature extraction consists of two steps. First one is Corpus-based taxonomy construction, is created from the dataset. It is a tree like structure with each word is mapped to the hypernym WordNet taxonomy. For example, the word *monkey* one of the WordNet mappings is the *mammal* and further mapped to the *animal*. Finally reaching the most general term *entity*. Second step is Feature construction. The features are the term count calculated for each term during the taxonomy construction.

Feature Selection: feature selection is the method used to reduce the data dimension in predictive analysis. The output of the feature construction is the matrix. If all the hypernyms of the corpus are considered the dimension of the feature matrix will grow to thousands of terms. Here used three method for selection like count-based, mutual information, pagerank[8]. The count-based element determination that takes all hypernyms counts in the corpus-based taxonomy, sorts them in ascending order as indicated by their frequency of occurrence and takes the topmost words. Mutual information aims to explore the relationship between terms and class labels[11]. The selection method pagerank [12] helps to prioritize the hypernyms in the corpus-based taxonomy. The pagerank algorithms take input as a network with a set of starting nodes.



B. Classifier Module

A deep learning-based classifier is implemented here. The sequential model from the Keras is used as a deep learning model. The output from the keyword extraction module is a set of keywords extracted from the corpus. The classifier module intends to classify the text based on the topic.

Data Preparation: It is the some work to be done for the data to be ready for training. The first step is about to split the data into training and test tests. Converts the keyword them to a numbered index. This same procedure used for labels or categories, by using LabelEncoder utility and then convert the labels to a one-hot representation. The one-hot encoding is used to convert labels to integer representation. Normally some algorithms can work with categorical labels directly like a decision tree, there is no need to convert the label. But many algorithms cannot operate on labels directly. So one-hot encoding method is used to convert the label to integer format.

Model Training: Model training is done by using a sequential model. The model is one of the simplest neural networks, which is appropriate for a stack of layers where each layer¹⁹ has exactly one input tensor and one output tensor. Build the model using input, output layer with one hidden layer. The dense layer contains 512 neurons with ReLu as the activation function. The softmax function used the output layer with the number of neurons five, which denotes the topic categories. The dropout ratio is set as 0.5 to prevent the neural network from overfitting.

Hyperparameter Tuning: It is the final stage of the classifier model to tweak some model parameters such as batch size, dropout ratio, network structure, activation function, epoch, and others, improve the accuracy. Randomly changed the parameter value to improve the performance. An epoch is a measure of the number of times all of the vectors are used once to update the weights. Batch size is the number of training examples used in one iteration. Changed the epoch value to 2, 20, 50, 100, and batch size to 32, 64, 128. Tune the parameter based on the training accuracy and tested on the new samples.

V. RESULT AND DISCUSSION

Various experiments are done in the proposed system model to get a system with good accuracy value. The classification accuracy can be manually calculated by randomly giving some inputs to the system and analyzing the responses. For most of the input, it predicted the correct label and, in some cases, got the wrong label. 70% of the data from the corpus used for training and rest for the test. The accuracy of the system is evaluated based on accuracy. The important feature of the system is feature selection. Three feature selection techniques used are count based, mutual information, and pagerank feature selection. Evaluation of the model carried out using these three feature selection methods. From these three features, accuracy order is pagerank, mutual information, count-based feature selection. By using pagerank as feature selection the obtained result is shown in below fig. 4.

Loss	0.187
Accuracy	0.949
F1_score	0.939
Precision	0.959
Recall	0.921

Fig. 4. Evaluation of the system - Pagerank feature selection

VI. CONCLUSION

The idea behind context-aware text classification is detecting keywords and mapping them to the domain. The work is all about keyword extraction and classification. Semantic information is available in taxonomies, this concept is used for keyword extraction. i.e, maps the words from to their hypernym counterparts, which are considered as candidate features and weighted according to a normalized tf-idf metric. Three methods are used to select features like count based, mutual information, pagerank. Deep Learning concept is used for the classifier model building. BBC News corpus is used to train and evaluate the system. The accuracy of the proposed system is 94%, 90%, 88% by feature selection methods pagerank, mutual information, count based respectively. The obtained results show that the system in most cases can produce meaningful results for the given input text. The automatic text classification system helps to reduce the manual labor power required for analyzing large amount of texts. In addition, modification on the system can be made to intake multi-label classifications and can be extended to open domain concepts. Several studies are going on the areas like WSD, embeddings, deep learning advances such as the hierarchical attention networks. The exploration of such options is left for future work.

REFERENCES

- [1]. K. Shimura, J. Li, and F. Fukumoto, "Text categorization by learning predominant sense of words as auxiliary task," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 1109–1119, 2019.
- [2]. C. Liu, W. Wang, G. Tu, Y. Xiang, S. Wang, and F. Lv, "A new centroidbased classification model for text categorization," Knowledge-Based Systems, vol. 136, pp. 15–26, 2017.
- [3]. P. Jin, Y. Zhang, X. Chen, and Y. Xia, "Bag-of-embeddings for text classification.," in IJCAI, vol. 16, pp. 2824–2830, 2016.
- [4]. Y.-H. Chen and S.-F. Li, "Using latent dirichlet allocation to improve text classification performance of support vector machine," in 2016 IEEE Congress on Evolutionary Computation (CEC), pp. 1280–1286, IEEE, 2016.
- [5]. R. Johnson and T. Zhang, "Deep pyramid convolutional neural networks for text categorization," in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 562–570, 2017.
- [6]. D. Zhao, N. Du, Z. Chang, and Y. Li, "Keyword extraction for social media short text," in 2017 14th Web Information Systems and Applications Conference (WISA), pp. 251–256, IEEE, 2017.
- [7]. A. R. Pal and D. Saha, "An approach to automatic text summarization using wordnet," in 2014 IEEE International Advance Computing Conference (IACC), pp. 1169–1173, IEEE, 2014.
- [8]. B. Skrlj, M. Martinc, J. Kralj, N. Lavra, and S. Pollak, "tax2vec: Constructing interpretable features from taxonomies for short text classification," Computer Speech & Language, p. 101104, 2020.
- [9]. S. Scott and S. Matwin, "Text classification using wordnet hypernyms," in Usage of WordNet in Natural Language Processing Systems, 1998.
- [10]. S. Kannan and V. Gurusamy, "Preprocessing techniques for text mining," International Journal of Computer Science & Communication Networks, vol. 5, no. 1, pp. 7–16, 2014.
- [11]. G. Chandrashekar and F. Sahin, "A survey on feature selection methods," Computers & Electrical Engineering, vol. 40, no. 1, pp. 16–28, 2014.
- [12]. J. Kralj, M. Robnik-Sikonja, and N. Lavrac, "Netsdm: Semantic data mining with network analysis.," J. Mach. Learn. Res., vol. 20, no. 32, pp. 1–50, 2019.