COMP5423 Internet Infrastructure and Protocols

Survey on Text Classification

Written Report

**Group Members & Contribution**

|  |  |
| --- | --- |
| Student ID | Name |
| 19074889G | WONG CHI HANG |

Table of Contents

[Task Definition 3](#_Toc70795421)

[Single Label Classification 3](#_Toc70795422)

[Multi-Label Classification 3](#_Toc70795423)

[Challenges 4](#_Toc70795424)

[Typical methods 5](#_Toc70795425)

[Preprocess 5](#_Toc70795426)

[Shallow Learning Model 6](#_Toc70795427)

[Deep Learning Model 8](#_Toc70795428)

[Evaluation 9](#_Toc70795429)

[State-Of-The-Arts 11](#_Toc70795430)

[Application scenarios 12](#_Toc70795431)

[Email Filtering 12](#_Toc70795432)

[News Filtering 12](#_Toc70795433)

[Conclusion 13](#_Toc70795434)

[Reference 14](#_Toc70795435)

Task Definition

Text classification is a task to classify a text into different categories. The text can be a paragraph or a short sentence such as news title or email subject. [4] Text classification can be categorized as below types.

Single Label Classification

In this classification, a text is categorized to a single label. For example, a text can be categorized as “good”, “bad” or “average” etc. For a special situation that only two classes available for the text to be categorized, it is called binary classification. For example, categorize if an email is a spam or not.

Multi-Label Classification

In this classification, a text is categorized to a multi-label. For example, a tweet may contain several tags. For a special situation that the text is classified into a defined hierarchy of output categories, it is called hierarchical classification. For a text from a company’s memo can be categorized to multi-label that related to: “Person”, “Employee” and “Manager”.

Diagram

Description automatically generated

Diagram 1.1

Challenges

Text classification is one of the applications in Natural Language Processing, it faces some common challenges as other applications in NLP do such as contextual words, synonyms, ambiguity including lexical, semantic and syntactic ambiguity.

1. contextual words and phrase

Same words in different context may have different meaning. For example, the car is running but its gas is running out soon.

1. Synonyms

Same idea can be expressed by different words or ways such as big, large, huge, jumbo, titanic.

1. Irony and sarcasm

Sometimes the meaning of human’s writing or speaking is opposite to the wording they use. Human can distinguish it by the situation or intonation, however, it’s hard for computer. For example, a boss fired his staff and let security guards send him out of office, the staff roar at his boss: “Thank You!”.

1. Lexical ambiguity

A single word can be used as a verb, noun or adjective. For example, book and show

1. Semantic ambiguity

One sentence can be interpreted more than one meaning. For example, “A boy saw a man in the park.” can be interpreted as both the boy and the man are in the park or the man in the park, but the boy is not.

In addition, there are some more challenges for text classification. For short text classification, the challenges come from the natural of itself - short text [10]:

1. Sparseness

Since a short text may only contain few words to dozen words, the features can be extracted from the text is very limited.

1. Immediacy

Short text usually sent out in very short times like conversation in instant messenger such as whatsapp, wechat and telegram. Therefore, the text is short, but the quantity is large.

1. Noises and imbalanced distribution

Short text usually contains numerous of misspellings, non-standard terms and noise.

Typical methods

According to the research paper “A Survey on Text Classification: From Shallow to Deep Learning” [3], typical methods show as the figure 3.1 below. In this section, the processes including preprocess, shallow learning, deep learning and evaluation will be further discussed.

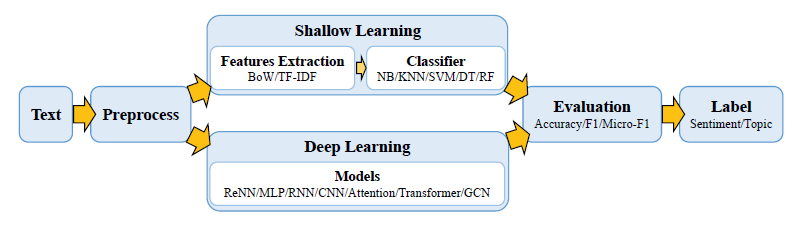


Fig 3.1 Flowchart of the text classification with classic methods [3]

Preprocess

During preprocessing, tokenization, stop words removal, capitalization, slang and abbreviation conversion, noise removal, spelling correction, stemming and lemmatization will be performed [5].

Tokenization

It is a process to break a sentence or a passage into words, phrases, symbols or other meaningful elements. The converted elements are called tokens. Usually, it is rule-based that the sentence or passage is converted to token according to a set of rules.

Stop Words Removal

There are some words frequently appear but do not contain important meaning in classification algorithms. For example, in English, “a”, “about”, “above”, “across”, “after”,“afterwards”, “again” etc. It is a common technique to remove those words from texts and documents.

Capitalization

For some situation, a word may be capitalized for some reasons. For example, the first word in an English sentence. However, from computer’s point of view, it is treading as different words such as “Apple” and “apple” since “A” and “a” are different character to computer. To deal with the inconsistent capitalization, the common approach is to reduce every letter to lower case.

Slang and Abbreviation

Slang is a subset of the language used in informal language that combines several words to form a special meaning which may different from the words original meaning such as “break a leg”, “go banana” etc. Abbreviation is a shorten form of word or phrase that mostly contain first letters of the words like the F.B.I is an abbreviation of “the Federal Bureau of Investigation”. To deal with these words,

Noise Removal

Special character and punctuation are usually not necessary for text classification. And also some text or document may contain a hyper link that start with http:// which is useless and could be a noise for classification and should be removed in preprocessing.

Spelling Correction

It is optional to have a spell checking since the typo are usually found on informal text or data source. It is rarely found in formal document.

Stemming

Stemming is a procedure that removing prefixes or suffixes from a word. For example, “studies” to “studi”, “studying” to “study”.

Lemmatization

This is a process to return the base or dictionary form of a word usually by removing its inflectional endings. For example, “cats” to “cat”, “ponies” to “poni”.

Shallow Learning Model

After preprocessing, the processed data can be fed to either shallow learning model or deep learning model. There are two main components in shallow learning model: feature extraction and classifier.

Feature Extraction

In this procedure, the text or document is represented by selected features. The features are then feed to the classifier for classification. Following are commonly used features for text classification.

1. Bag of Words (BOW)

This is the most basic model that put all the words into a bag and count the frequency of every word. For example, there are two sentences:

Sentence1 = “I go to school by bus.”

Sentence2 = “I go to swim every day.”

BoW = {“I”:2, “go”:2, “to”:2, “school”:1, “by”:1, “bus”:1, “swim”:1, “every”:1, “day”:1}

1. Count Vectorization (One-hot-Encoding)

This idea of this method is creating a vector which dimensions as same as the number of unique words in the corpora. Using the two previous sentences as example:

The unique words are: {“I”, “go”, “to”, “school”, “by”, “bus”, “swim”, “every”, “day”}

The count vector of that two sentences will be:

Sentence 1 = [ 1, 1, 1, 1, 1, 1, 0, 0, 0]

Sentence 2 = [ 1, 1, 1, 0, 0, 0, 1, 1, 1]

1. Term frequency-inverse document frequency (TF-IDF)

This is a weighting scheme calculated by term frequency and inversed document frequency. The term frequency refers to the frequency of a term occur within a document. Document frequency refers to the frequency of a term occur in a document across the entire collection. In here, the document frequency is used inversely.

1. Word-to-Vector (word2vec)

Word2vec is a neural probabilistic model using a statistical computation method to calculate representing vector from a text corpus. There are two model architectures for word2vec: continuous bag-of-words (CBOW) and continuous skip-gram. In the continuous bag-of-words architecture, the model learns an embedding by predicting the current word from a window of surrounding context words and ignoring the order of context words. In continuous skip-gram architecture, the model uses the current word to predict the surrounding window of context words.

Classifier

The classifier takes the extracted features as input, then automatically analyze it and assign a set of pre-defined class based on its content. Typical classifiers can be categorized on the approach that it based as below [3]:

1. PGM-based method

Probabilistic graphical models (PGMs) including widely used model Naïve Bayes and HMM are combinations of probability theory and graph theory to express the conditional dependencies among features in graphs.

1. KNN-based method

This method relies on K-Nearest Neighbors (KNN) algorithm to classify an unlabeled sample by its K-nearest labeled samples. For example, for k = 3, if 3 nearest labeled samples to an unlabeled sample are labeled as {red, red, blue}, then the unlabeled sample will be labeled as “red”.

1. SVM-based method

This method uses support vector machines algorithm to construct a hyperplane or set of hyperplanes in a high or infinite dimensional space for classification. The optimal hyperplane is the hyperplane with maximum margin which are gaps between the instances of the given classes.

1. DT-based method

Decision Tree (DT) based method is a flow-chart-like tree structure learning method, in which non-leaf nodes denotes a test on an attribute and each branch represents an outcome of the test, and each leaf node holds a class label.

Deep Learning Model

Deep learning model is another model that taking pre-processed text as input, then giving labeled output by using deep learning approach. Different from shallow learning model, there is no feature extraction procedures in deep learning model, the features are learnt by the model algorithm itself. Typical deep learning model approaches are categorized by their algorithm used as below:

1. ReNN-based model

The recursive neural network (ReNN) automatically learns from the text to predict labels. In ReNN, all nodes are combined into parent nodes using a weight matrix which is shared across the whole model. The vector of the parent node has the same dimensional as its leaf node vector.

1. MLP-based model

The multilayer perceptron (MLP) based model uses a simple neural network structure that contains an input layer, a hidden layer with an activation function in all nodes and an output layer. In MLP based model, each node connects with a certain weight.

1. RNN-based model

The recurrent neural network (RNN) language model learns historical information and consider the location information among all words. In RNN, each word is represented by a specific vector using a word embedding, then it is fed to the RNN cells. The output vector of the RNN cells have the same dimension with the input vector and are fed to the next hidden layer. The parameters in RNN are shared across different parts of the model and each input word has the same weights. The predicted label is the output of last hidden layer.

1. CNN-based model

The convolutional neural network (CNNs) was proposed for image classification to extract features from pictures by using convolving filters. To use it in text classification, the text is required to represent by vector which is similar to the image representation vectors. Then, the vector matrix is fed into the convolutional layers which contain several filers. After the convolutional layer, the outputs are then goes through the pooling layer and the result is concatenated to obtain the final vector representation which is used to predict the category.

Evaluation

To evaluate the result, the approaches for single-label and multi-label classification are slightly different. Accuracy, Error Rate, Precision, Recall and F1 score are the mostly used approaches. The notations below are used in the formula.

|  |  |
| --- | --- |
| Notations | Descriptions |
| TP | true positive |
| FP | false positive |
| TN | true negative |
| FN | false negative |
|  | true positive of the t th label on a test |
|  | false positive of the t th label on a test |
|  | true negative of the t th label on a test |
|  | false negative of the t th label on a test |
| S | label set of all samples |
| Q | the number of predicted labels on each text |

The following are commonly used to evaluate a single-label classification.

1. The formula of Accuracy, Error Rate, Precision, Recall and F1

1. Micro-F1 and Macro-F1

In multi-label classification, a text can be assigned more than one label, therefore, the evaluation metrics designed for single-label classification are not suitable to evaluate result of multi-label classification. Therefore, other evaluation metrics like Micro-F1 and Macro-F1 are introduced.

, where ,

, where ,

State-Of-The-Arts

Refer to the below table from the research paper [3], there is a trend of using attention, trans and GNN based model for state-of-the-arts approach.

Table

Description automatically generated

1. Attention-based model

To overcome the problem of CNN and RNN in text classification that they are not intuitive enough for poor interpretability, especially in classification errors. The attention-based methods are introduced. The hierarchical attention network (HAN) is one of attention-based models. HAN includes two encoders and two levels of attention layers. It aggregates essential words into sentence vectors then aggregates vital sentence vectors into text vectors.

1. Transformer-based model

Trnasformer-based models use unsupervised methods to mine semantic knowledge and then construct pre-training targets. Famous Trnasformer-based models are including ELMo, OpenAI GPT and BERT.

1. GNN-based model

Graph neural networks (GNN) turns text classification into a graph node classification task. First, the input texts and the words in the text are defined as nodes to construct a structural graph. The graph nodes are connected by edges. There two kind of edges: document-word edges and word-word edges. The words and texts are represented in the hidden layer. Famous GNN -based models for text classification are including text graph convolutional network (TextGCN), heterogeneous graph attention network (HGAT) and MAGNET.

Application scenarios

As the beginning of the report, text classification is defined as a task that assigns a pre-defined label to an unlabeled text. By this operation, we can identify what the text is related to or what kind of information it contains without checking its content by human. It can aid people to save a lot of time and efforts. Therefore, the technology of text classification is suitable for the application helping people to deal with high volume of information. Two application scenarios are listed below as examples:

1. Email Spam Filtering

By using text classification with the email subject and the content, an email can be classified as spam or not. Traditionally, spam email was filtered based on numbers of rules or factors, such as sender’s email address, suspicious phrases, malicious attachment or hyperlinks. Text classification techniques can greatly increase the accuracy of spam filtering.

1. Fake news and inappropriate content detection

This is the responsibility for social media or forum keeping their platform not being abused to spread fake news or inappropriate content. For some famous social media like Facebook, Instagram and twitter, there may be over million new posts per hour. It is impossible to check it one by one manually. Text classification can help with this scenario.

Conclusion

In this report, the whole process for text classification have been covered from pre-processing to evaluation. Also, the task definition and its application scenarios are also included. This can give the reader a clear picture about text classification. Moreover, this study can assist new learners who are interested in text classification to implement the model from shallow learning model to deep learning model and state-of-the-art approaches.

Reference

Microsoft. (2019). *New-SelfSignedCertificate.* [online] Available at: https://docs.microsoft.com/en-us/powershell/module/pkiclient/new-selfsignedcertificate?view=win10-ps [Accessed 13 May. 2020].

Computing for geeks. (2019). *Install and Configure OpenSSH Server on Windows Server 2019.* [online] Available at: https://computingforgeeks.com/install-and-configure-openssh-server-on-windows-server [Accessed 13 May. 2020].

Microsoft. (2019). *OpenSSH Server Configuration for Windows 10 1809 and Server 2019.* [online] Available at: https://docs.microsoft.com/en-us/windows-server/administration/openssh/openssh\_server\_configuration [Accessed 13 May. 2020].

server-world.info. (2019). *OpenSSH : SFTP only + Chroot.* [online] Available at: https://www.server-world.info/en/note?os=Ubuntu\_18.04&p=ssh&f=5 [Accessed 13 May. 2020].

How to forge. (2019). *How to install and configure VSFTPD.* [online] Available at: https://www.howtoforge.com/tutorial/how-to-install-and-configure-vsftpd/ [Accessed 13 May. 2020].