Madhav Sharma, 2022, 10:1
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

Text categorization Technique for Document retrieval using Natural language Processing

Asst. Prof. Madhav Sharma, Asst. Prof. Vijay Mohan Shrimal
Department of Computer Science & Engineering,
Jagannath University,
Jaipur, India

Abstract- The problem for modern information technologies and web-based services is to choose, filter, and preserve ever-increasing amounts of textual data to which frequent access is necessary. Text Categorization is a subtask of Information Retrieval that allows users to browse the collection of texts related to their own interests more easily by navigating across category hierarchies. This paradigm is extremely beneficial not only for retrieving and filtering information, but also for developing user-driven web services. Because of the large quantity of documents involved in the aforementioned applications, automatic data classification is required. In most statistical Machine learning models, the bag-of-words representation is utilized to train the goal classification function. Single words from the documents are used as features to learn the statistical models. Typical natural language structures such as morphology, syntax, and semantic are completely ignored in the creation of the classification function. On the other hand, the semantic information provided by Text Categorization models has yet to be used in the most important natural language applications. Information extraction, question/answering, and text summarization should all include category information since it aids in the selection of domain knowledge that language applications typically use in their processing.

Keywords- Application, Filtering, Information Retrieval, Text Categorization.

I. INTRODUCTION

The problem for modern information technologies and web-based services is to choose, filter, and preserve ever-increasing amounts of textual data to which frequent access is necessary. Because it contains several approaches that aid correct information retrieval and, as a result, user satisfaction, Information Retrieval (IR) is regarded as a good methodology for the automated management of information/knowledge.

The classification of electronic documents into generic categories (e.g., sport, politics, religion, etc.) is an intriguing way to improve the performance of IR systems, because it allows users to more easily browse a collection of documents relevant to their specific interests: (a) Users can more easily browse a collection of documents relevant to their specific interests.

(b) Sophisticated IR models can benefit from the data that has been classified. The creation of textual texts, for example, is done using the contents of the document. The major areas of interest are indicated by a preliminary categorization stage. As a result, Text Categorization (TC) is playing an increasingly important role not only in retrieval and filtering, but also in the development of user-driven online services.

II. EFFICIENT MODELS FOR AUTOMATED TC

Text categorization is the process of assigning documents to preset categories. In the domains of information retrieval and machine learning, it's a hot topic. A variety of supervised learning algorithms have been used to handle this problem. A model for the classification problem is shown below. Create a judgment function that can determine the correct

text classes, such as D 2 C. Build a decision function that can decide the correct classes for texts, i.e. D 2 C, given a set of user interests stated in classes (i.e. topics/subtopics labels), C = C1,..., C|C|, and a variety of existing papers previously categorized in these classes (i.e. training set).

As a result, the decision function is tasked with categorizing freshly received documents (d D) into one (or more) classes (es) based on their content.

1. Designing a Text Classifier:

The design of general text classifiers foresees a set of tasks universally recognized by the research community:

1.1 Features design: in this phase the following pre-processing steps are carried out: –Corpus processing, filtering, and formatting all the documents belonging to the corpus.

1.2 Gathering relevant information: In conventional methodologies, words are frequently used as basic units of information. A stop list is used to remove function terms in this case (that exhibit similar frequencies over all classes). This section considers the linguistic information that distinguishes a document (and its class). To construct features that are more complicated than basic words, structured patterns (i.e. multiple word expressions) or lexical information can be used (e.g., word senses).

1.3 Normalization: A famous method used here is word stemming, which involves deleting frequent suffixes from words. Stems are the words that have been stemmed. Normalization refers to the activity of lemmatization in linguistic analysis (i.e. words and/or complex nominal) (i.e. detection of the base form of rich morphological categories, such as nouns or verbs1).

1.4 Feature selection: which aims to exclude non-informative phrases from documents in order to improve categorization and reduce computing complexity. 2, information gain, or document frequency are common selection criteria.

1.5 Feature Weighting: features assume usually different roles in documents, i.e. they are more or less representative. Different weights are associated to features via different, often diverging, models.

1.6 Similarity estimation is modeled via operations in spaces of features. This can be carried out between pairs of documents or between more complex combinations of features (e.g., profiles as the combination of features coming from different representative documents). Usually quantitative models (i.e. metrics) are adopted for this.

1.7 Inference: similarity among document/profile representations activates the target classification decision. Assignment of an incoming document to a target class is based on a decision function over similarity scores. Different criteria (i.e. purely heuristics or probability-driven rules) are used in this task.

1.8 Testing: The accuracy of the classifier is evaluated using a set of relabeled texts (the test-set) that were not used during the learning phase (training-set). The labels of the classifier are compared to the ones that are correct. The gap between the human choice (provided by the training data) and the underlying classification system is usually indicated by one or more numerical scores at the end of this stage.

2. Profile-based Text Classifier:

Profile-based classifiers define each target class using a profile, which is commonly a vector of weighted phrases (Ci). These vectors are produced from articles that were previously categorized and used to train the system under Ci. The categorization procedure starts with a comparison of the incoming document d to the different profiles (one for each class).

Early profile-based classifiers, for example, used the Vector Space Model [Salton and Buckley, 1988] to define similarity. It's worth noting that this method's main advantages are its computational efficiency and ease of use.

The development of a profile-based classifier requires a specialization of some phases:

2.1 Features weighting, or the construction of synthetic profiles, can be divided into two steps: – the creation of a d representation for documents d The features f retrieved from d are defined by d. The weights of those features are called components. - the creation of a visual representation

2.2 In profile-based classifiers, similarity estimate is always done between unknown (i.e. unclassified) documents d and the previously determined profiles (C I The space determined by the characteristics (i.e. weighted elements of vectors d and C I is commonly used to establish similarity

2.3 Inference: Over the similarity scores, a decision function is frequently applied. Probability, fixed, and proportional threshold are the most commonly used inference methods. These are known as P cut (a threshold for each class exists and is used to determine whether or not a document belongs to it the best k-ranked classes are assigned to each document) and in cut (a threshold for each class exists and is used to determine whether or not a document belongs to it the best k-ranked classes are assigned to each document) (the test-set documents are assigned to the classes proportionally to their size).

III. NLP FOR TEXT RETRIEVAL

Basic retrieval models that use simple stems for indexing can benefit from NLP representations, whereas advanced statistical retrieval models do not benefit from NLP. A more critical examination is carried out. With the use of context-based indexing or metadata, information retrieval is described as the process of accessing and getting the most suitable information from text depending on a specific query submitted by the user.

Information retrieval (IR) is a software programmed that manages the organization, storage, retrieval, and evaluation of data from document repositories, particularly textual data. The system helps users find the information they need, but it does not provide specific solutions to the inquiries. It informs the existence and location of documents that might consist of the required information. The documents that satisfy user's requirement are called relevant documents. Classical Problem in Information Retrieval (IR) System.

The primary purpose of IR research is to create a model for retrieving data from document repositories. The ad-hoc retrieval problem is a classic topic in the IR system that we will examine here.

In ad-hoc retrieval, the user must type a natural-language query that defines the information needed. The IR system will then return the necessary papers pertaining to the requested information. For example, let's say we're looking for something on the Internet, and it returns some specific pages that are pertinent to our search, but it also returns some non-relevant pages. This is due to a difficulty with ad-hoc retrieval.

IV. ASPECTS OF AD-HOC RETRIEVAL

Some features of ad-hoc retrieval that are explored in IR research are as follows:

- How can users improve the initial formulation of a query with the help of relevancy feedback?
- How to implement database merging, i.e., how to combine results from many text databases into a single result set?
- How do you deal with data that is partially corrupted? Which models are suitable for the task?

1. Information Retrieval (IR) Model:

Models are employed mathematically to comprehend real-world occurrences in a wide range of scientific fields. A model for information retrieval predicts and explains what information a user would find useful in response to a query.

The IR model is a pattern that describes the above-mentioned components of the retrieval process and is made up of the following pieces:

- A model for documents.
- A model for queries.
- A matching function that compares queries to documents.

Mathematically, a retrieval model consists of –

- D – Representation for documents.
- R – Representation for queries.
- F – The modeling framework for D, Q along with relationship between them.
- R (q,di) – A similarity function which orders the documents with respect to the query. It is also called ranking.[2,3]

2. Types of Information Retrieval (IR) Model:

An information model (IR) model can be classified into the following three models –

2.1 Classical IR Model: It is the most basic and fundamental IR model. This paradigm is built on mathematical concepts that are easily recognized and understood. The three most frequent IR models are Boolean, Vector, and Probabilistic.

2.2 Non-Classical IR Model: This approach is the polar opposite of the traditional IR model. The underpinnings of such IR models include similarity, probability, and Boolean operations. Non-classical IR models include information logic models, situation theory models, and interaction models.

2.3 Alternative IR Model: It is the application of particular techniques from different domains to improve the classical IR model. Alternative IR models include cluster models, fuzzy models, and latent semantic indexing (LSI) models.

3. NLP: How Does NLP Fit into the AI World?

- Now that we have a basic understanding of AI, machine learning, and deep learning, let's return to our original question. Is Natural Language Processing (NLP) considered AI, Machine Learning, or Deep Learning?
- Artificial intelligence (AI), natural language processing (NLP), and machine learning (ML) are all phrases that are regularly thrown around. Their crazy love, on the other hand, has logic to it.
- Machine learning is a subset of natural language processing, yet both are included in the larger category of artificial intelligence.
- Artificial intelligence (AI) and computational linguistics are combined in Natural Language Processing (NLP) to allow computers and people to communicate in a natural way.
- By allowing a computer to analyze and interpret what a user said, NLP strives to bridge the gap between computers and people (input voice recognition). This task has proven to be quite challenging.
- To communicate with humans, a programmed must understand syntax (grammar), semantics (word meaning), and morphology (tense) (conversation). Because recalling such a large number of rules can be scary, previous attempts at NLP have had mixed outcomes. As a new system was implemented, NLP gradually improved, shifting from a strict rule-based computer programming methodology to a pattern-learning based computer programming methodology. In 2011, Siri made her debut on the iPhone. In 2012, it was revealed that using graphical processing units (GPU) could boost the performance of digital neural networks and natural language processing (NLP). [3]
- NLP helps computers understand unstructured content by applying AI and machine learning to construct derivations and provide context to language, similar to how human brains develop derivations and provide context to language. It's a programmed for detecting and evaluating unstructured data's hidden "signals."
- Organizations would have a greater understanding of how the public perceives their products, services, and brand, as well as their competitors' perspectives.
- Google has now launched its own neural-net-based engine for eight languages, effectively reducing the quality gap between its previous system and a human translator and reigniting interest in the method. If given the right data, computers can already produce an eerie echo of human speech. In recent years, deep learning architectures and algorithms have made substantial advances in fields such as image recognition and audio processing.
- Their application to Natural Language Processing (NLP) was first lackluster, but it has subsequently progressed significantly, producing state-of-the-art results for a variety of popular NLP tasks. In tasks including named entity recognition (NER), part of speech (POS) identification, and sentiment analysis, neural network models have outperformed traditional approaches. The progress of machine translation is maybe the most astonishing.

4. Semantic representation:

Text Categorization and Word Sense Disambiguation are areas of language processing that have recently received a great deal of attention. This is because of the impact they have on harnessing the ever-growing textual information posted on the Internet or other on-line document collections.

In the study we report in this thesis, we tried to see if the accuracy of TC could be improved when more sophisticated linguistic representations based on word meanings would also be available.

5. Extraction of basic NLP-features:

The entire complexity of the language processor is determined by the processing models for two language levels: morph syntactic and grammatical recognition, as well as the lexical resources utilized. For expressing legal linguistic derivations, morphological recognition is the process of discovering the canonical lemma associated with a text unit, which is often done utilizing massive dictionaries and generative (rule-based) components. These techniques are frequently optimized for (almost) linear pattern matching algorithms and do not have a high level of complexity. [1]

V. TERMINOLOGY EXTRACTION

The Terminology Extraction requires the following phases: Candidate head of terminological

expressions selection; before the target complex nominal grammar can be applied, the candidate head of terminological expressions must be chosen. Statistical filters based on T F IDF are researched for this purpose. This phase can be completed in O(m M log(m M)) time given M documents and m the maximum amount of words per document.

In the Grammar Application, all windows of n words around the head are taken into account. The algorithm tries to apply the target challenging nominal grammar in such windows. All sub-sequences of words surrounding the head that fit the grammar are stored in a database. The amount of subsequences to parse is less than the number of times each word appears in the text. The grammar can be implemented over a lengthy period of time because the number of window words is considered to be constant. By keeping the word window size constant, the length of possible expressions is limited, but it allows for a linear extraction technique. [4]

Statistical filtering; after the processing of all documents in the target category, statistical filter are applied to select the most suitable terminological expressions. Even this phase requires linear times.

VI. CONCLUSION

Single words from the documents are used as features to learn the statistical models. Typical natural language structures such as morphology, syntax, and semantic are completely ignored in the creation of the classification function. On the other hand, the semantic information provided by Text Categorization models has yet to be used in the most important natural language applications.

Information extraction, question/answering, and text summarization should all include category information since it aids in the selection of domain knowledge that language applications typically use in their processing.

REFERENCES

[1] R. Barzilay and M. Elhadad. Using lexical chains for text summarization. In In Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97), ACL, Madrid, 1997, 1997.

[2] R. Basili and A. Moschitti. A robust model for intelligent text classification. In Proceedings of the thirteenth IEEE International Conference on Tools with Artificial Intelligence, November 7-9, 2001 Dallas, Texas, 2001.

[3] Roberto Basili and Alessandro Moschitti. Intelligent NLP-driven text classification. International Journal on Artificial Intelligence Tools, Vol. 11, No. 3, 2002.

[4] Roberto Basili and Fabio Massimo Zanzotto. Parsing engineering and empirical robustness. Natural Language Engineering, to appear, 2002.

[5] Roberto Basili, Alessandro Moschitti, Maria Teresa Pazienza, and Fabio Massimo Zanzotto. Personalizing web publishing via information extraction. Special Issue on Advances in Natural Language Processing, IEEE Intelligent System, to appear, 2003.

[6] Ron Bekkerman, Ran El-Yaniv, Naftali Tishby, and Yoad Winter. On feature distributional clustering for text categorization. In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pages 146–153. ACM Press, 2001.