NAMED ENTITIY RECOGNITION and ATTRIBUTE EXTRACTION using MACHINE LEARNING.

Mr.Ankush Hutke, Shubham Jain, Hemil Doshi,Hiba Momin

[1] *Mr.Ankush Hutke, Information Technology, Rajiv Gandhi Institute of Technology, Maharashtra, India*
Shubham Jain, Information Technology, Rajiv Gandhi Institute of Technology, Maharashtra, India
Hemil Doshi, Information Technology, Rajiv Gandhi Institute of Technology, Maharashtra, India
Hiba Momin, Information Technology, Rajiv Gandhi Institute of Technology, Maharashtra, India

ABSTRACT

Named-entity recognition (NER) is a subtask of information extraction that seeks to locate and classify named entities in text into pre-defined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc.
In focused NER, once the entities are recognised we further aim at finding the most important named entities among all the others in a document, which we refer to as focused named entity recognition. We implement this using a classifier approach, i.e. Naïve Bayes classification, and we show that these focused named entities are useful for many natural language processing applications, such as document summarization, search result ranking, and entity detection and tracking.
Attribute extraction on the other hand, involves automatic selection of attributes in your data (such as columns in tabular data) that are most relevant to the predictive problem you are working on.
Many researchers had proposed rule based or statistic based approaches to deal with the extraction task in a variety of application areas. Here we try to implement an approach to extract the entities' attributes from unstructured text corpus Our goal can be twofold in this respect, firstly we can aim at simply organizing information so that it is useful to people, or put it in a semantically precise form to make further inferences using algorithms. In the current market scenario, big data is at the crowning point of the latest technology. Big data involves not just collection but manipulation in a way that we can develop prescriptive and predictive models from it, and extract patterns that prove of value to the customers. Machine learning involves the task of providing the computer with a decision making capability, basically a computer mimics the human mind and develops intelligent behavior. Within machine learning algorithms, we have the task of natural language processing. NLP has the subtask of Information extraction, and many other applications like named entity recognition, speech recognition, optical character recognition, word sense disambiguation, etc. Reading about this further motivated us to research in this field, and implement systems that can be used for these applications. We chose this project with an interest to learn about the applications of machine learning algorithms, and implement a system for entity recognition and attribute extraction.

Keyword : - *NER,AE and NBC etc*

1. INTRODUCTION

The world has entered the era of big data. How to deal with massive text effectively and efficiently has become an urgent problem in front of us. With the rapid growth of online electronic documents, many technologies have been developed to deal with the enormous amount of information, such as automatic summarization, topic detection and

tracking, and information retrieval. Information extraction involves finding and understanding limited but relevant parts of texts, this is done from many pieces of text. Based on this structured representation of the relevant information is created.

A key task is to identify the main topics of a document, where topics can be represented by words, sentences, concepts, and named entities. Entity refers to the independent existence of things. Each entity has its own characteristics, that is, different entities have their own specific attributes, and can be distinguished from other entities. The name of Entity often represents species, which have the same nature as other nouns. Generally, people specify a name for each entity, which is also known as Named Entity, NE. The entities with same category generally have similar attributes, but they are different in the value of the property. Different types of entities generally have different properties. Any concrete or abstract object can be called an entity. Different from other applications of information extraction, a user`s interest can also be defined as an entity, such as people, institutions, products, etc. Moreover, things that appear in the corpus, can all be defined as an entity. Entities with different types have different attributes and information characteristics.

Our definition of focused named entities is mainly concerned with Who and What. Therefore it is almost self-evident that the concept of focused named entity is important for document understanding and automatic information extraction. Moreover, we shall illustrate that focused named entities can be used in other text processing tasks as well. For example, we can rank search results by giving more weights to focused named entities. We define focused named entities as named entities that are most relevant to the main topic of a news article. Encouraged by this study, we further investigated the machine learning approach to this problem, which is the focus. We discuss various issues encountered in the process of building a machine learning based system, and show that our method can achieve near human performance. Entity Attribute Extraction is another important technology in the field of natural language processing, and the informations obtained can be not only provided to the users directly, but also used as the basis of building intelligent query and data mining.

As an important aspect of information extraction, entity attribute can be used to define a new entity, conduct entity mining and other practical applications. The main purpose of the study on information extraction is to get the structured information from natural language text. The task of entity attribute extraction is to let computer fetch the attributes and their values by itself. While entities with same category generally have roughly the same attribute information structure, but the value of each attribute will be different. For example, there are general attributes of a people entity: full name, occupation, work units, mail, telephone, hobbies, etc.; the attributes of organization/unit entity: the name of institution/unit, address, department, responsible person, the nature of services, etc. And the typical attributes of a product entities: product name, manufacturer, product function, art, price, brand, characteristics, and so on. In recent years, with the rapid development of search engine technology, searching has become more and more intelligent. The search engine has evolved from the "keywords search" to "SNS Search" and "Entity search". Entity Search is more complicated than the keywords Search. Although the traditional keyword search has developed well, the results provided by the search engine can help users find the information, but in fact for the "Search Engine" system itself, it does not understand the meaning of the search. The primary focus on Entity search is not the "key words" but the object, such as people, institutions, organizations, etc. We hope that a conversion from keywords to entity can help search engine understand and organize search results from a more subtle point of view.

1.1 Problem Definition

Our task is to create a system that automatically selects the focused named entities from a set of all entities in a document. Entity Search needs an entity-related information database. The information database not only includes massive entity information but also the relevant attributes which can accurately describe the entity. The construction of the Entity Database needs long-term accumulation and the relevant data mining technology. This project provides an entity attribute extraction method, once the topical entities are discovered, to conduct the entities and syntax analysis, and the extraction of entity attribute with deep learning method. This method can be used in the task of entity attributes.

.

1.2 Scope and Limitations

In this project we studied the problem of focused named entity recognition. We gave examples to illustrate that focused named entities are useful for many natural language processing applications. The task can be converted into a binary classification problem. We focused on designing linguistic features, and compared the performance

of three machine learning algorithms. Our results show that the machine learning approach can achieve near human-level accuracy. Because our system is trainable and features we use are language independent, it is easy for us to build a similar classification model for other languages. Our method can also be generalized to related tasks such as finding important words and noun-phrases in a document.

In the future, we can integrate focused named entity recognition into real applications, such as information retrieval, automatic summarization, and topic detection and tracking, so that we can further study and evaluate its influences to these systems.

Introduction related your research work Introduction related your research work Introduction related your research work Introduction related your research work Introduction related your research work Introduction related your research work Introduction related your research work Introduction related your research work Introduction related your research work Introduction related your research work Introduction related your research work .

2. Overview of Proposed Systems

Process:

Step 1: The first step of collecting data is done by web scraping. Web scraping (web harvesting or web data extraction) is a computer software technique of extracting information from websites. This is accomplished by either directly implementing the Hypertext Transfer Protocol (on which the Web is based), or embedding a web browser.

Step 2: Since the corpus is a huge chunk of plain text obtained from the Internet, or probably some other source which provides any computer readable document, it contains noisy text. At first, we must process the corpus to get pure unstructured text.

Step 3: Locating authority information associated with the words by comparing the words with the authority list which has the most common uses of the words, this task enables us to extract named entities and based on historical data we also predict named topical entities.

Step 4: After detecting the named entities we extract their attributes through Attribute Extraction(AE).

Step 5: In this step the generated list of entities and attributes can be used for various NER applications.

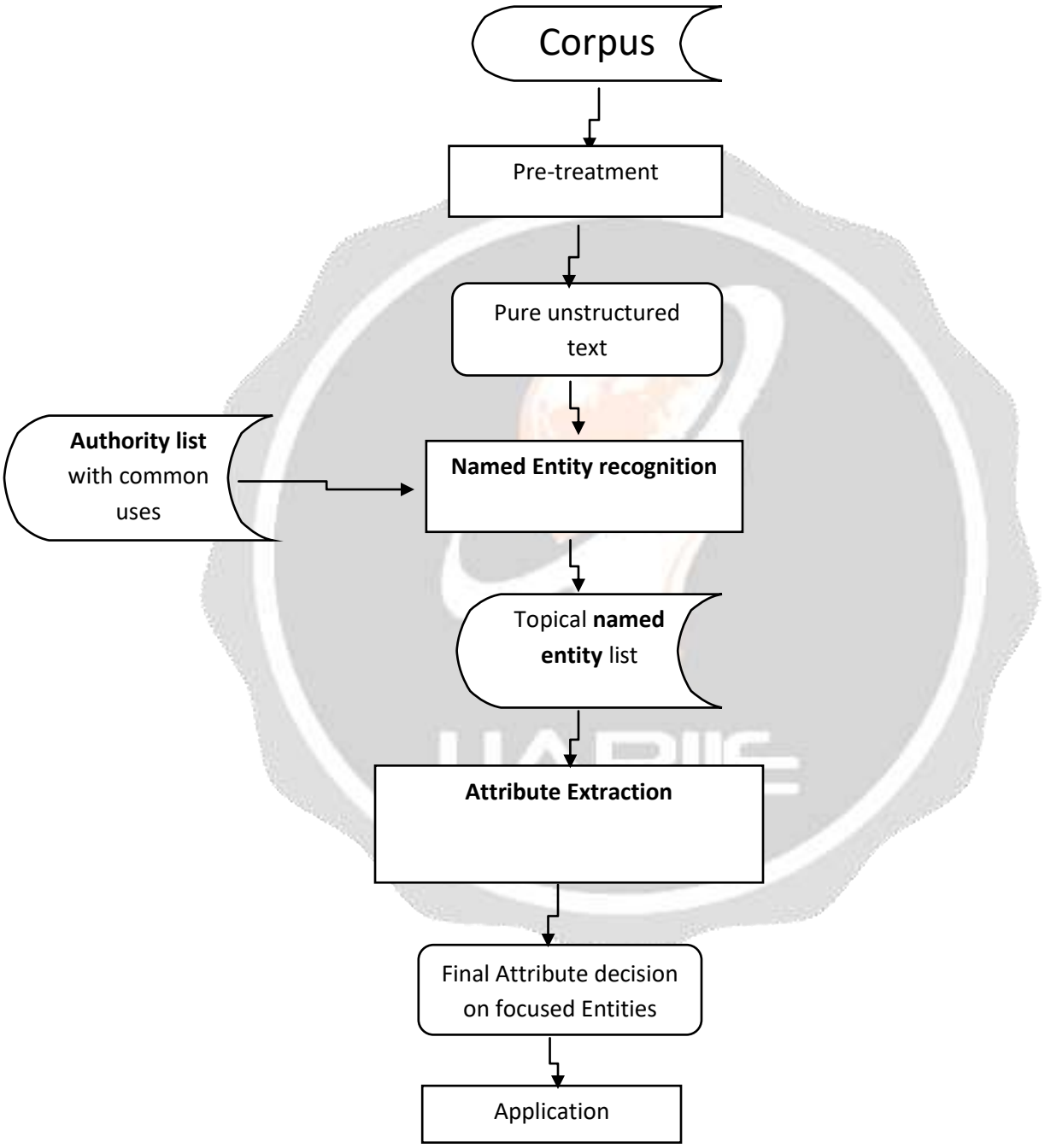

Fig -1: Proposed System

2.1 Architecture of Book Recommendation System

The system architecture is as follows:

The user uses the user interface to enter a list of URLs of websites from where he/she wants to extract data about a particular book or a particular author.
After this,the URLs are sent to the Web Scraper Module that scrapes these sites using appropriate commands.Web Scraping is done using Beautiful Soup.
The scraper generates a file of scraped data from the specified sites.This file is then forwarded to the classification tool that uses Sentiment Analysis to classify the content as either positive or negative.Sentiment Analysis id done using Naïve Bayes Classification as the central tool

Only the positive content is forwarded to the Recommender System.The Recommender System is the heart of the architecture that implements the two main modules of the system i.e., Named Entity Recognizer (NER) and Attribute Extracter (AE). The NER module is implemented using python programming and NLTK (Natural Language Tool Kit).Using this,the module creates a list of tagged entities.

These entities are sent to the Attribute Extracter module,that again uses python tools to extract attributes of the generated entities.The final list of books with the same author,same genre,same rating etc are then recommended to the user.

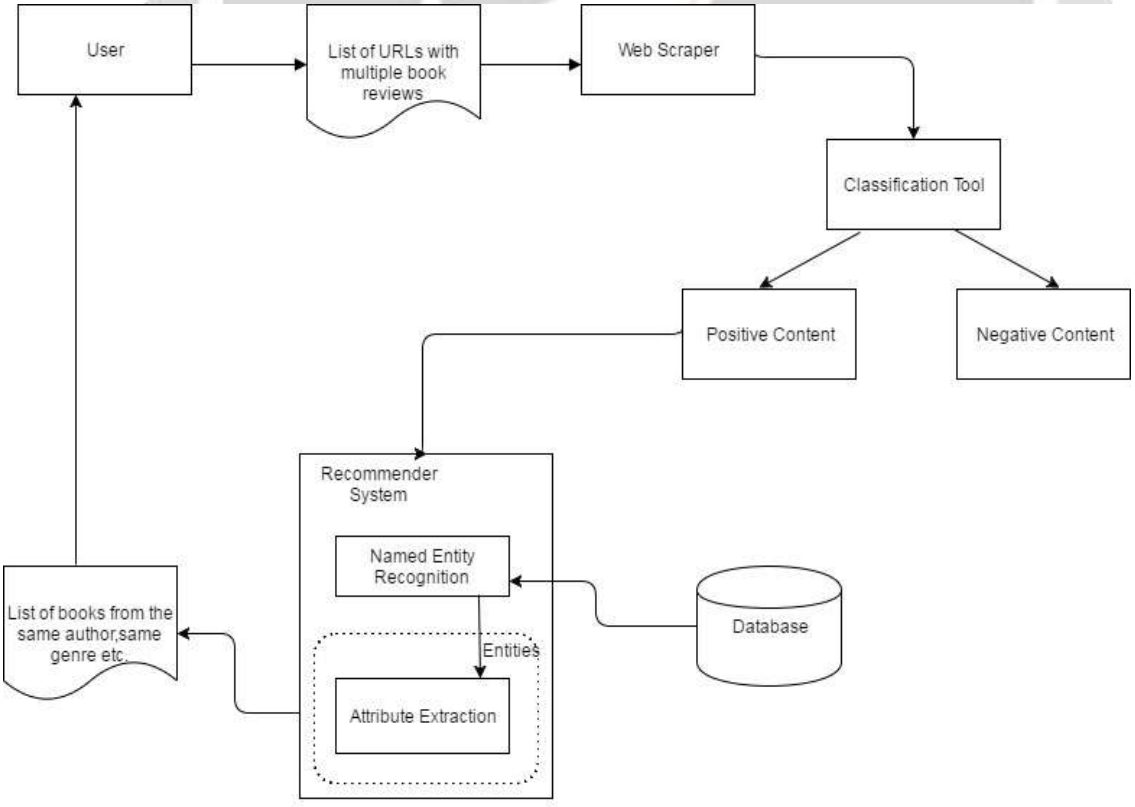

Fig -2: Architecture of System

3. MODULAR IMPLEMENTATION

Modular design or "modularity in design" is a design approach that subdivides a system into smaller parts called modules or skids that can be independently created and then used in different systems. A modular system can be characterized by functional portioning into discrete, scalable, reusable methods, rigorous use of well-defined modular interfaces and making use of industry standards for interfaces. Modular design is an attempt to combine the advantages of standardized (high volume normally equals to low manufacturing costs) within those of customization**.**Our system has four main modules

3.1 Module 1: User Interface

The User Interface of our Book Recommendation System is created using Flask. Flask is a python framework used to create user friendly web applications. The UI of the system is simple in design and is user friendly. The first page of the UI displays a Start Now button which on clicking initializes the recommendation process in the background.
Our UI consists of 3 main pages.
The first page of the UI is a user friendly interface that simply provides an option of starting the recommendation system. It also contains details of the developers.
The second page of the system's UI displays only the positive reviews of the latest books on the website (www.kirkusreviews.com) that is being scraped in the Web Scraping module(3.5.2).After reading these reviews the user is provided with an option of 'Reading similar content' after each displayed positive review. On clicking this button the next page loads.
The third page after implementing the NER and AE module at the backend displays a list of recommendations of books similar to the content previously read by the user. This is the final output of our system.
To make our system look more attractive to the user we have integrated various CSS methods for styling the web pages.

3.2 Module 2: Web Scraper

The UI after taking the input from the user switches the model's control to the Web Scraper module.
This module is implemented using python's package Beautiful Soup.
To collect the corpus for our project we have scraped the website www.kirkusreviews.com. The imported package panda is used to convert unstructured data from this website to structured data. After which the data is scraped and stored in a table. This table has four main columns of Title, Author, Genre and Review. The output of this module is a .csv format file that contains data about reviews from the site in a tabular form. This file is then sent to the Classification module.
Along with its role in dynamically scraping the above stated website, we have also used this scraping module to collect information and reviews of a large number of books from the same website. These datasets have been used in training our classification module or Sentiment Analysis module

3.3 Module 3: Classification or Sentiment Analysis

The website of book reviews contains all types of reviews some may be positive ones whereas the others might be negative. Another important task of our project is to be able to differentiate between the positive and negative reviews from the scraped data. For this we have created the Classification or Sentiment analysis module. This

module mainly implements the Naïve Bayes Classification algorithm and its upgraded versions. A function called vote classifier takes the individual votes of each of the classifiers used and displays the average of the outputs of each of them. The various classifiers used are : Linear SVC classifier, MNB classifier, Bernoulli NB classifier and Logistic Regression classifier. Each of these classify the scraped content into either positive or negative by tagging them with tags like 'pos' for positive and 'neg' for negative.
The reviews are not only classified plainly as positive or negative, a certain percentage of confidence is also calculated for each review. For example if the review is tagged as "pos, 0.8" (where 0.8 is the confidence), that means the content is 80% positive.

3.3 Module 4: NER and AE

The review selected by the user is sent to this module.The task of NER and AE is then performed on this corpus.NLTK (Natural Language Tool Kit) modules for NER are imported here and run over the extracted corpus.The result of this module is a list of entitites. These entities are then sent to the AE module that simply extacts the attributes of the recognized entities using NLTK libraries. The attributes are then run over the scraped book datasets to find books with similar content.Finally a list of Similar books is displayed to the user.

4. CONCLUSIONS

In this project we have created a Book Recommendation System based on Named Entity Recognition and Attribute Extraction. We In this project we studied the problem of focused named entity recognition.We gave examples to illustrate that focused named entities are useful for many natural language processing applications. The task can be converted into a binary classification problem. We focused on designing linguistic features, and compared the performance of three machine learning algorithms. Our results show that the machine learning approach can achieve near human-level accuracy. Because our system is trainable and features we use are language independent, it is easy for us to build a similar classification model for other languages. Our method can also be generalized to related tasks such as finding important words and noun-phrases in a document In the future, we can integrate focused named entity recognition into real applications, such as information retrieval, automatic summarization, and topic detection and tracking, so that we can further study and evaluate its influences to these systems.

5. ACKNOWLEDGEMENT

We wish to express our sincere gratitude to Dr. U. V. Bhosle, Principal and Dr. S.B. Wankhade, H.O.D of Information Technology Department of RGIT for providing us an opportunity to do our project work on"Named entity recognition and attribute extraction using machine learning".
This project bears an imprint of many people. We sincerely thank our project guide Mr. Ankush Hutke for his guidance and encouragement in carrying out this project work.
Finally, we would like to thank our colleagues and friends who helped us in
completing the Project (Synopsis) work successfully.

6. REFERENCES

[1]. R. Barzilay and M. Elhadad. Using lexical chains for text summarization. In Proceedings of the ACL

[2] F. J. Damerau, T. Zhang, S. M. Weiss, and N. Indurkhya. Text categorization for a comprehensive time-dependent benchmark. Information Processing & Management, 2004.

[3] H. P. Edmundson. New methods in automatic abstracting. Journal of The Association for Computing Machinery, 16(2):264–285, 1969.

[4] J. Y. Ge, X. J. Huang, and L. Wu. Approaches to event-focused summarization based on named entities and query words. In DUC 2003 Workshop on Text Summarization, 2003.

[5]. Artificial Intelligence Applications by Lisa F. Rau (1991)

[6].Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity RecognitionArtificial Intelligence Applications by Erik F. Tjong Kim Sang and Fien De Meulder

[7]. Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition by Jana Strakova´ and Milan Straka and Jan Hajic