International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT21743

Drug Safety Report Generator

Sudhir Dubey*, Pruthviraj Bhamre, Akshay Patil, Rahul Kumar
SOCSE, Sandip University, Nashik, Maharashtra, India

Article Info
Volume 7, Issue 4
Page Number: 38-49

Publication Issue :
July-August-2021

Article History
Accepted : 01 July 2021
Published : 07 July 2021

ABSTRACT

This document provides an overview on identifying ICSR (Individual case safety reports) & Drug Safety Classification of Adverse Drug Events from free Text Electronic Patient Records and Information. As a remarkable rise is observed in the usage of digital health records the potential for extensive clinical data extraction has drawn much attention. We intend to separate the causes and effects of unfriendly drugs from the records. We have therefore promoted a machine learning-based framework for the planned signature test of hostile drugs or safe phrases in the event of a report. In addition, the framework also uses named substance recognition based on word references to identify drugs and diseases that are present at the same time. The framework evaluation of physical comments in the corpus and a context-related analysis of consumption, which was carried out on preselected drugs, showed convincing results.

Keywords : Adverse Drug Events, Drugs, Extraction Text, Dependency Parser, Patient identification, Medical History, Natural Language Processing, Text Mining, ICSR (Individual Case Study Report)

I. INTRODUCTION

The side effects of drugs are a serious problem faced by patients, healthcare providers, administrative departments, and drug manufacturers. Although the strict measures to determine the potential of drug use are preliminary clinical, the application in a wide range of fields can reveal other risks that are not detected in the clinic. Temporarily included according to the number of designated patients. Once the advertisement is approved, professionals will use the spontaneous adverse reaction reporting system to report the adverse drug exposure, and then conveniently analyze it to ensure safe use of the drug. Regardless, underreporting is a major problem in pharmacovigilance, especially since the number of reports received from specialists is tiny. Common case reports in biomedical screenings are significant advantages in terms of consistency with SAERS because of their abundance, accelerated maturation, and key information barrier. Because of their unstructured nature, legitimate composition tests are exploratory, monotonous, and work-based. Recently, advances in a general programming language (NLP) and information extraction (IE) have become universal. Inbuilt recognizable substance tests with biomedical names, component compounds, or related

occasions Efforts to dispense with potentially harmful drug use cases have resulted in different types of free data. The organization consists of different frameworks, e.g.

1. Individual Case Study Report (ICSR) is an adverse event report for an individual patient.

Important elements that needs to be determined from ICSR are:

(1) An Identifiable Patient who has experienced an adverse reaction after taking drugs.

(2) A suspect drug is a drug that caused the adverse reaction.

(3) An Adverse Event is a side effect occurring with a drug

2. Natural Language Processing (NLP) is a branch of computer science focused on developing a system through which computers can communicate with humans in natural languages.
3. NLP and Text Mining are used to extract Adverse Drug Events from safety reports and Bio- medical text without need for a large training set.

Hence, this work aims on transforming a machine learning-based compound extraction framework for recognizable evidence and the extraction of drug-related unfriendly hits from case reports, based on a cosmology-driven philosophy, the subjective evaluation of the framework shows vigorous results. The methodology provides an adaptable self-help level for drug health professionals to gather potential hostile medication occasions to be submitted as free content information and display an organized layout that is naturally saved in any content, PDF, or Word document layout.

II. LITERATURE SURVEY

We have reviewed information extraction and related software engineering practices that have focused on the wellness area of drugs to distinguish the signs of unfavorable drug responses from various sources of information such as detailed unrestricted records, electronic medical and clinical records. has become more important to general well-being, particularly with the development of information stores that contain reports of adverse drug reactions that require rapid preparation to find signs of antagonistic reactions, or sources of information that may contain such signals but require information or text extraction methods in order to highlight the importance of the previous commitment of PC researchers in this area, we classify and survey current methods and, above all, recognize the regions in which further exploration should be undertaken.

The prerequisite of programmed retrieval of valuable information with the vast amount of literary information that human inquiry is used to work with is entirely distinguishable. Market development is subject to online news data, responses, and procedures.

1. Existing system

Previously there was no such software to reduce the work of a man overworked, there was only a complete work done by men and handwritten. Tables are made up of different columns with different meanings such as serial number, Suspect drug, side effects etc., this cost about $ 256 million a year because so many people were doing that work.

Each written report took approximately 90 to 120 minutes to compile and submitted to the FDA (Food And Drug Administration). As a result of this human activity many mistakes were made which resulted in many errors in the number of reports produced.

It was a very busy task for both human and FDA staff and these records were very difficult to manage because all the work was done with hard copy, these records were difficult to maintain and update. It became a major problem for organizations.

- **Disadvantages**

1. A lot of money was spent because of the old methods.
2. Many people had to work for this tedious job.
3. Many human errors have been made, resulting in incorrect reporting.
4. It took about 90 to 120 minutes, but one more report was needed, and thousands of reports were added.
5. The FDA had a similar problem keeping these multiple records as the growing population, side effects of drugs and diseases made it even more difficult.

III. PROPOSED SYSTEM

The idea is to develop an app based on the WHO Web site to automatically store records of drug side effects in a systematic way, so the user can easily access the content and understand it easily and use it continuously.

The application retrieves document files such as word document, text document and pdf document inputs for applications and converts data in a structured format and saves data using different visual modules.
The program uses NLP, Data Mining, machine learning algorithms and dictionaries such as the WHO dictionary and the ADR dictionary to execute the program.

The main purpose of the improved system is that large workloads by employees will be reduced and storage and storage of metadata will be much easier and less stressful.

This program is able to extract data from free text sources, text documents.

TABLE I
PROPOSED SYSTEM

Parameters	Traditional System	Proposed System
Implement ation	Its implemented by creating tables for ICSR manually	Its implemented solely by web based application using NLP, Text mining and Machine Learning algorithms
Workload	Man labor is put to work to handle the work and hence a tedious job	All the work is handled by the application
Time	The time required for generating ICSR is 90-120 minutes per case	The time required for generating ICSR is 2-15 minutes
Errors	There may be a chance missing a component or mention a wrong field in ICSR	Generates perfect ICSR with no errors
Funding	The cost of appointing man labor is 250 crore INR per annum	The cost of overall application comes around 30K-50K

IV. CONCEPT AND WORKING

The concept is to foster the digital software for WHO to clearly hold up the facts of the negative influences of medicinal drugs in a prepared arrangement, so the

client thinks that it's less complicated to get to the substance and realize it effortlessly and use them further.

The web application obtains report records such as text files, text reports, and PDF file information from the user input, converts the information into organized customization, and stores the information in different perception modules.

The program uses NLP, data mining, machine learning calculations and algorithms, and word references such as the WHO word reference and ADR word reference to run the application.

Here is the short algorithm for developing the web based application:

A. Start
B. Check authentication
C. Enter Login and Password
D. If the user wants to update the info, the user may pass the records in word , text or pdf format and the data will be processed automatically
E. If new data is entered, auto refresh the application and implement the updates automatically
F. If the user wishes to check the adverse effects of drugs, he can get access to the structured data for his use
G. The application saves the updates
H. Stop.

V. DESIGN CONCEPT

1. Corpus Preparation:

The data set used for training and validation of the relation extraction system is the ADE corpus. The ADE corpus contains 2972 MEDLINE case reports that are manually annotated and harmonized by three annotators. The corpus contains annotations of 5063 drugs, 5776 conditions (e.g. diseases, signs, symptoms), and 6821 relations between drugs and conditions representing clear adverse effect implications. The ADE corpus contains annotations of relations between drugs and conditions that represent True relations. This represents a sparsely annotated dataset.

In the ADE-EXT corpus, 120 manually annotated True relations were not suitable for the NLP task. Examples include overlapping inter-related entities such as acute lithium toxicity where lithium is related to acute toxicity. After removal of nested annotations, the ADE-EXT corpus was compass into a training set (ADEEX-TRAIN) and a test set (ADE-EXT-TEST).

2. Relation Extraction Workflow:

For the identification and extraction of drug-condition entity pairs that fit into adverse effect relations, the Java Simple Relation Extraction (JSRE) system. JSRE provides a re-trainable and scalable supervised classification platform that uses Support Vector Machines (SVMs) with different kernels specially designed for the NLP and relation extraction. All sentences in ADE-EXT-TRAIN and ADE-EXT-TEST contain drug-condition pairs labeled as either True or False.

The ADE-EXTTRAIN was used as data for training and cross-evaluation of JSRE whereas the ADE-EXT-TEST was used as an independent test set.

3. Mapping annotation ontology against Ontology of Adverse Events:

The use of ontologies has proven of great value in biomedicine, also since it enables machine reasoning, abstraction and automatic hypothesis generation. We therefore had interest in investigating if the knowledge encoded in the annotations of the ADE corpus could be semantically connected to the AEO.

For doing this, we manually compared the definitions of the entities of AEO and of ADE annotation ontology. The basic design patterns of AEO, ADE and CLEF as from the original papers, emphasizing shared entities using green and red color

VI. REQUIREMENTS FOR DEVELOPMENT

1. Hardware

1) Processor: Intel Core 2 Quad CPU Q6600 @ 2.40GHz (4 CPUs) / AMD Phenom 9850 Quad-Core Processor (4 CPUs) @ 2.5GHz. or above
2) Memory: 4GB or more
3) HDD Space: 60GB or more

2. Software

1) Operating System: Windows XP or above
2) Web Browser: IE4, Google Chrome or Mozilla Firefox
3) Development Tools: JSP, CSS3, HTML 5 and JavaScript
4) File System: Drug and Adverse Reaction Dictionaries

VII. DESIGN AND CODE

1. System Design and Description

The system status diagrams show the system, complete and input and its effects from / to external objects. Content drawings can be created using two types of building blocks:

1. Businesses (Actors): labeled boxes; one in the center representing the program, and around several boxes for each external character
2. Relationships: labeled lines between business and system

The diagram below shows the levels at which the system undergoes the ICSR process. Initially the input is provided in the form of a word, pdf or text document and then transferred. Thereafter NLP, Text mining and machine learning algorithms are used to produce final case reviews.

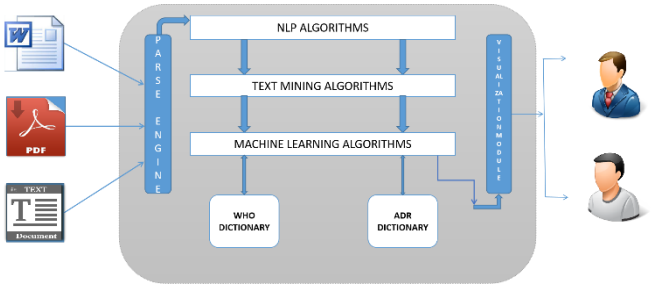


Figure 1. Architecture Diagram

2. Database Design

In our project, we have used a lot of information to verify and secure the user and the ICSR for security, that does not allow unauthorized access to our system and we make the ICSR completely flawless.

To achieve our goal, we have created three different details of the different roles of user login, side effects and other medical related terms.

1. User login details:
- The first database contains two separate tables that define the username and user password.
- The second database assigns roles to the user.

2. ADR and WHO Dictionary database:
- This dictionary contains all the information about the side effects that a patient may experience while using the medication.
- The data used from the algorithms is compared to the database and according to the comparison, a report is generated.
- These dictionaries are downloaded from free sources available online.

3. Patient's database:
- This dictionary contains all the information about the patient temporarily.
- Includes patient age, height, gender and country of origin.

- With the exception of the parameters specified by the patient above, it depends on the case.

3. UML Diagrams

1. LogIn use case diagram

- The diagram below shows that at first, the user needs to prove his identity by entering the right username and password.
- If login is successful then the user will be directed to the upload page otherwise, the user will be prompted with an unsuccessful login prompt.

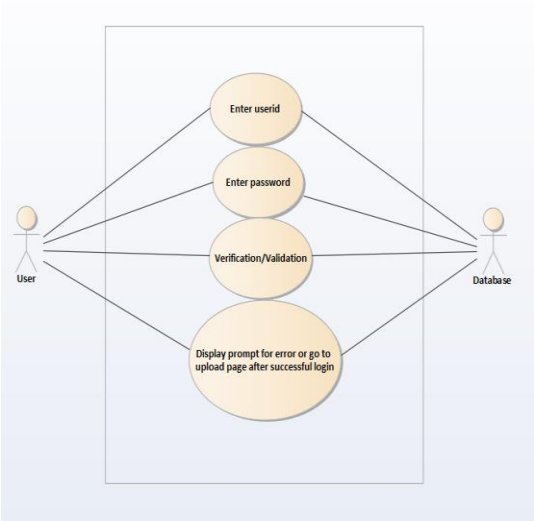


Figure 2. LogIn UML Diagram

1. Upload use case diagram

- After the login procedure, the user will be directed to the upload page.
- Now, the user has the choice either to upload a file in three different forms – text, word or pdf document or can simply enter the case in TextArea for processing it.
- This diagram shows the document is uploaded and for which report will be generated.

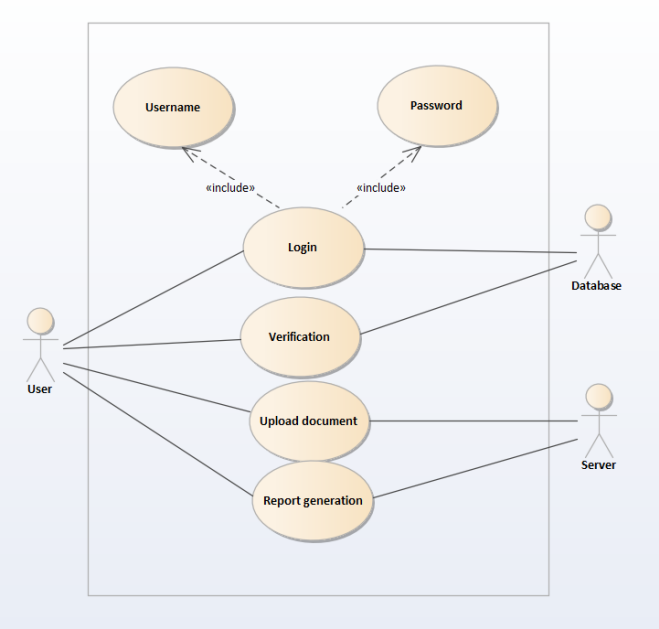


Figure 3. Upload UML Diagram

2. Processing use case diagram

- After the login procedure, the user will be directed to the upload page.
- Now, the user can upload a file in three different forms – text, word or pdf document or can simply enter the case in TextArea for processing it.
- This diagram shows that text cased typed in the text area will be processed and a report will be generated for the same.

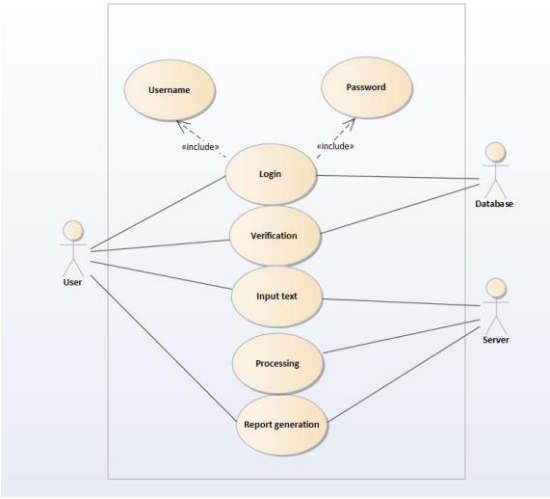


Figure 4. Processing UML Diagram

3. Parsing use case diagram

- After a document is uploaded in the form of text document or typed text, the data is forwarded to the parse engine.
- The parse engine will parse the data, that is, it will generate raw data and will be given as input to NLP.
- The parser used is Stanford parser.

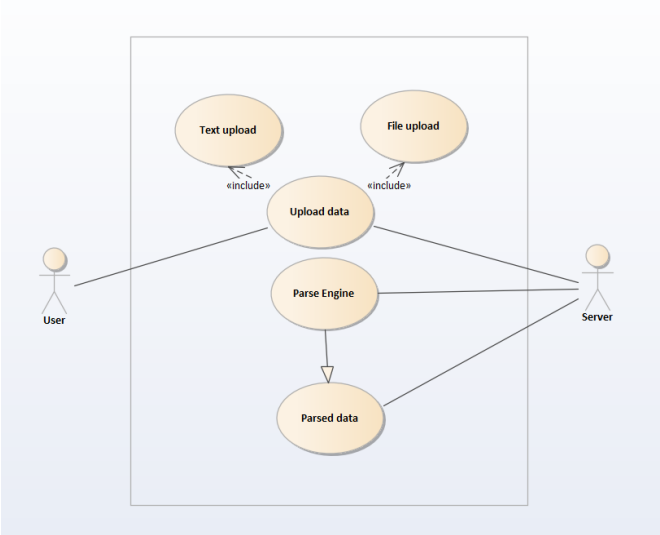


Figure 5. Parsing UML Diagram

4. NLP use case diagram

- The input to NLP will be parsed data generated by the parser.
- NLP consists of three main blocks-
  - Tokenization
  - POS Tagger
  - Chunker
- The output generated will be more simplified.

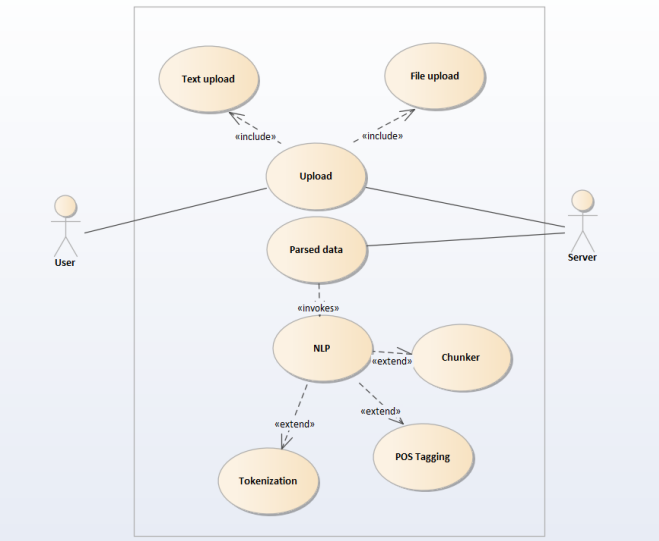


Figure 6. NLP process UML Diagram

5. Report Generation use case diagram

- The input from NLP will now go through text mining algorithms.
- This mined data will be compared and verified with WHO and ADR dictionaries and will generate the test case review in structured format.
- The user can either refer to the report generated on the browser or can download an XML file onto his system for further reference.

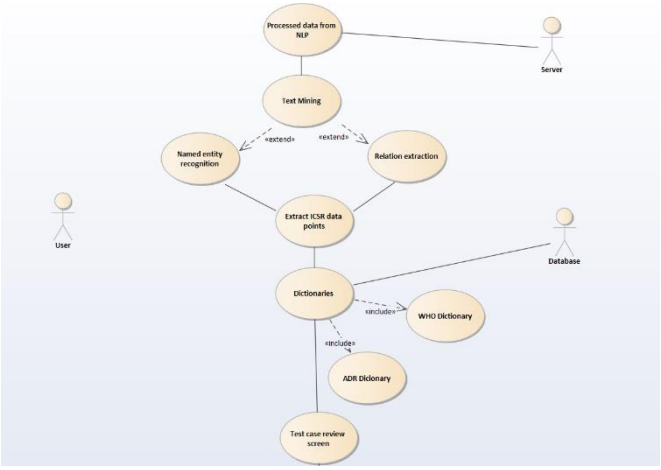


Figure 7. Report Generation UML Diagram

4. Modules and Implementation

1. Parsing:

- Purpose:

For breaking a data block into smaller chunks by following a set of rules, so that it can be more easily

interpreted, managed, or transmitted by a computer, we are using parses. The parser that is used in the project is Stanford parser.

- Input:

A 10 year old boy takes metformin which induces abdominal pain. (PDF, Text, Word document or typed ADR cases)

- Output:

10 years old boy takes metformin induced abdominal pain

(The data contents of the file are checked and according to it raw data is extracted)

- Files used by module:

Its code files and input files (text, word and pdf document)

- Algorithm:
  - Start
  - Get input in the form of .txt, .doc/.docx and .pdf format
  - Parse that data using Stanford parser
  - Parsed output is generated
  - End

2. NLP:

- Purpose:

NLP can be used to interpret free text and make it analyzable. There is a tremendous amount of information stored in free text files, like patients' medical records, for example. Prior to deep learning-based NLP models, this information was inaccessible to computer-assisted analysis and could not be analyzed in any kind of systematic way. But NLP allows analysts to sift through massive troves of free text to find relevant information in the files. NLP consists of 3 main processes.

1. Tokenization-

Process of splitting up text into smaller parts called tokens .

2. POS Tagger-

This is used to remove stop words in the case statements.

3. Chunker-

It provides a partial syntactic structure of a sentence.

- Input:

Output of above parsing algorithm.

(10 years old boy takes metformin induced abdominal pain)

- Output:

10 years, boy, metformin, abdominal pain

- Files used by module:

Its code files and input files (text, word and pdf document)

- Algorithm:
  - Start
  - Get input in the parsed form
  - Process the parsed data using NLP algorithm
  - More furnished output is generated
  - End

3. Text Mining:

- Purpose:

Text mining, also referred to as text data mining, roughly equivalent to text analytics, is the process of deriving high-quality information from text. High-quality information is typically derived through the devising of patterns and trends through means such as statistical pattern learning. To achieve this, various text mining algorithms are used such as

- Input:

Output of above NLP algorithm.

(10 years, boy, metformin, abdominal pain)

- Output:

10 Boy Metformin Abdominal pain

- Files used by module:

Its code files and input files (text, word and pdf document)

- Algorithm:
  - Start
  - Get input in the simplified form
  - Mine the terms from the input and compare them to the dictionaries in the database

- Text is extracted as output
- End

4. Machine Learning:

- Purpose:

It is the science of getting computers to act without being explicitly programmed. Machine learning helps our project to automatically accommodate different cases without any interaction with the system. It basically handles all of the processing done to generate ICSR.

- Input:

A 10 year old boy takes metformin which induces abdominal pain.

- Output:

Final Test case review (Shown in the output itself)

- Files used by module:

Its code files and input files (text, word and pdf document)

- Algorithm:
  - Start
  - Parse the input file using Stanford parser
  - Process the parsed input using NLP to get more simplified resultant
  - Perform text mining on output data from NLP
  - Compare mined terms from output with the dictionaries in the database
  - Display output in structured format
  - End

5. Dictionaries:

- Purpose:

These dictionaries are used to check whether the case is valid or invalid. The final result is verified by comparing with the data of these two dictionaries:

1. ADR Dictionary(Contains adverse drug reactions)
2. WHO Dictionary(Contains drugs and other drug related terms)

- Input:

Final processed result.

- Output:

Indicates whether the final test case review is valid or invalid (Shown in the output)

- Files used by module:

WHO Dictionary and ADR Dictionary

- Algorithm:
  - Start
  - Check the result with dictionaries
  - If terms do not match, case is invalid otherwise case is valid
  - Output is indicated and shown on browser
  - End

6. Visualization modules:

- Purpose:

Through this, the unstructured form of report is shown in structured form. The output will be shown on the browser in the report form.

- Input:

Raw unstructured data.

- Output:

Final test case review is displayed
(Also shows whether case is valid or invalid)

VIII. Output/Results

1. Screens

Login Screen

This screen appears when the user enters the location of the application either on a local network or on cloud. This provides a secure way of using the system. Only the users registered to the system will be able to gain access to the system. A User must have a valid username and password to login. If the user does give wrong input username and password, he will be prompted for an unsuccessful login.

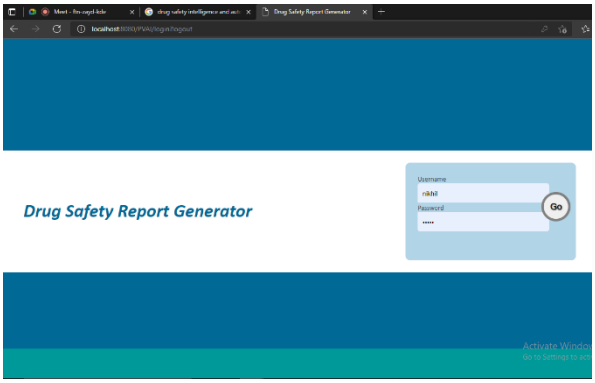


Figure 8. LogIn Screen

Upload Screen

This screen is a "Case Report Upload Screen" of the application which provides users with options to choose/upload the case reports for extracting the ICSR data and predicting the validity of the case. The screen provides the user with ability to upload the case in the pdf, word or text document and can also type the case in the following given text area and start the process by clicking on process or upload.

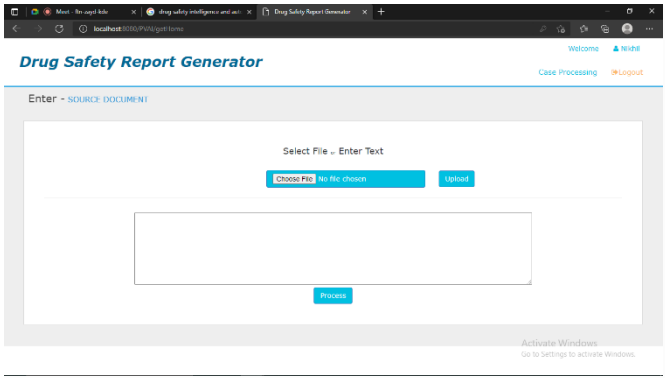


Figure 9. Upload Screen

Case Review Screen

The result screen returns the ICSR data extracted from the process of NLP, Text Mining and Machine Learning on the chosen case reports. The screen shows the detailed results including the entire extracted data one at a time. The screen also shows the validity of the case classified using Machine Learning algorithms. It also provides the facility to view the analysis of the extraction as well as validity of the case. The result is displayed in editable form which can be copied for further external user required processing.

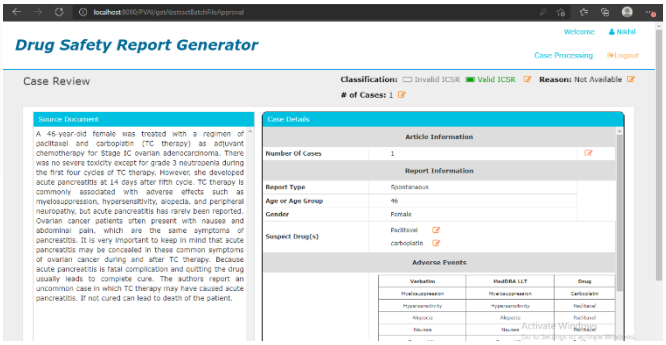
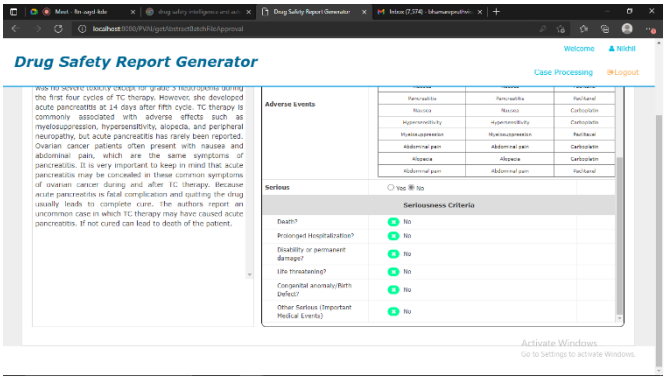


Figure 10. Case Review 1



IX. Application

- Medical field (Drug sellers, Experienced Doctors).
- The web based application is developed to be only used by experienced doctors, drug sellers, practicing doctors and other authorized users only.

X. Conclusion

This work reports on the adaptation of the JSRE machine-based learning system for diagnosing and eliminating drug side effects in the event of reports. The method of enriching the corporation was described in small detail and its subsequent use in construction

separation model. System performance testing showed promising results. System performance can be improved in a number of ways. In the current test, only the default features adopted by JSRE were used. The use of feature representation to include additional features for example from sentence search trees can further enhance results. Developing additional strategies such as post processing to differentiate relationships with non-contextual meanings can help to find more relationships. Reported test results indicate the nature of the study in identifying the adverse drug effect from the text. There are many strategies to follow immediately. The authors plan to measure the effectiveness of many branded business tags against the ADE corpus for drug identification and conditional in the text. Current experiments were performed on the ADE corpus, as they were the only ones available at the time of the project, but at the time of writing this report had been issued a new corporation, namely the EU-ADR corpus (van Mulligen et al., 2012). It will be interesting to see if the performance of JSRE in the ADE corpus will be different compared to the EU-ADR corpus. Similarly, the results of the benchmarking of programs to remove commercial and public relations programs such as SemRep, Luxid MER Skill CartridgeR, RelEx, MedScan will be implemented. The effect of the disclosure of the relationship in the text will be investigated to support the discovery of the signal and to identify the possible novel or negative results reported. The use of extruded data ontologies has been reported (Wimalasuriya and Dou, 2010; Pandit and Honavar, 2010), we plan to evaluate the use of various available tools (eg ODIE, semantics) using AEO ontology and compare the effectiveness of conflicting ontology methods and the manner indicated here. The result of the current work has shown promising results and has the potential to reduce manual learning time, speed up the process of tracking signals, thereby ensuring safe drug use in the market.

XI. Future Work

1. Application can be made more efficient using various optimization techniques.
2. There may be some common bugs that should be fixed in future development.
3. In terms of security various authorization techniques can be used to make application more secure.

XII.REFERENCES

[1]. Gurulingappa, H., Fluck, J., Hofmann-Apitius, M., and Toldo, L. (2011). Identification of adverse drug event assertive sentences in medical case reports. First International Workshop on Knowledge Discovery and Health Care Management (KD-HCM), European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases

[2]. Data Mining and Knowledge Discovery, Delamarre, D., Lillo-Le Louet, A., Jamte, A., Sadou, E., Ouazine, T., Burgun, A.,and Jaulent, M. (2010). Documentation in pharmacovigilance: using an ontology to extend and normalize Pubmed queries. In Studies Health Technology Informatics, volume 160, pp 518–522.

[3]. Burges, C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition.

[4]. rumaki, E., Miura, Y., Tonoike, M., Ohkuma, T., Masuichi, H., Waki, K., and Ohe, K.(2010). Extraction of adverse drug effects from clinical records. In Studies Health Technology Informatics, volume 160, pp 739–743.

[5]. Giuliano, C., Lavelli, A., Pighin, D., and Romano, L. (2007). FBK-IRST: Kernel Methods for Semantic Relation Extraction. In Proceedings of the Fourth International Workshop on Semantic Evaluations.

[6]. Benton, A., Ungar, L., Hill, S., Hennessy, S., Mao, J., Chung, A., Leonard, C., and Holmes, J. (2011). Identifying potential adverse effects using the web: A new approach to medical hypothesis generation. Journal of Biomedical Informatics, 44, pp 989–996.

[7]. (ECML PKDD) Gurulingappa, H., Mateen-Rajput, A., Roberts, A., Fluck, J., Hofmann-Apitius, M., and Toldo, L. (2012). Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. Journal of Biomedical Informatics.

Cite this article as :

Sudhir Dubey, Pruthviraj Bhamre, Akshay Patil, Rahul Kumar, "Drug Safety Report Generator", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 4, pp. 38-49, July-August 2021. Available at doi : https://doi.org/10.32628/CSEIT21743
Journal URL : https://ijsrcseit.com/CSEIT21743