GALGOTIAS UNIVERSITY

School of Computing Science and Engineering
Greater Noida, Uttar Pradesh
2021-2022

PROJECT GROUP NUMBER – BT3341

PROJECT GUIDE – RUNUMI DEVI

(Research Paper)
(2021-2022)

SUMMARIZING CONTRACT DOCUMENT USING GATE SOFTWARE

Varsha Vashisht
B.Tech CSE
Galgotias University
Jharkhand, India
Varsha_vashisht.scsebtech@ galgotiasuniversity.edu.in

Supriya
B.Tech CSE
Galgotias University
Bihar, India
supriya.scsebtech@galgotiasuniversity.edu.in

Abstract

Information extraction is the process of extracting information from both formal and informal documents in order to enable organizational acquisition , classification and archive on a website. Summarising is one of the methods of information extraction where shortening of data takes place computationally. A Contract Document is a written formal document that describes the basis of a contract covering the roles of both the offerer who offers the contract and acceptor who accepts the contract . This document not only includes obligations, and detailed description of the work or service such as drawings, details, procedures, or any other circumstances but also commercial information including prices,agreement, payment conditions, etc. In this paper a framework is proposed which will summarise a contract for the user. The summarisation will be done by using Natural Language Processing (NLP) operations such as Named entity recognition, where all the numerical values and terms along with the conditions related to contract are identified . Co-occurrence of the named entities is used as a criteria to identify the summary of the contract using GATE tool. The evaluation is done using the Model evaluator abstract.

Keywords

- Contract document
- Text summarisation
- Text enginnering
- Text annotation
- Text extraction

Introduction

The computer has remained notable for speed, information processing, exchange and storage but still unable to comprehend and interpret the information it is made to store, manipulate or exchange. With the high information overload across several domains, the task of processing and extracting meaningful facts from "these sea" of information is increasingly laborious, inefficient and ineffective. Individuals and organizations are finding an increasing gap between the acquisition of information and their meaningful use, despite the increasing influx and access to the information due to the inability of the computer – the core information processing tool – to comprehend and interpret the information. This may account for the poor decision bedeviling every aspect of the world in recent times; as humans have to study, understand and extract useful facts for decision making from the sea of information; a task that would have been more efficient and reliable if computers could comprehend the information and work in cooperation with humans in extracting and interpreting required facts from available information. As regards decision making, poor decision in law will not only be a disaster to the legal profession but also to the society it controls. The legal profession, world over, keeps track of their legal information in form of statutes, legislation and case law. Of these legal recordings, the most active is case law as legal decisions are inherently case based – "stare decisis". For efficient and quality legal decision therefore, the computer must be made to comprehend case law and assist legal practitioners in the task of extracting relevant facts from available information. Making the computer comprehend and summarise information (i.e. automatic summarisation), is the essence of semantic annotation and extraction.

Literature Survey

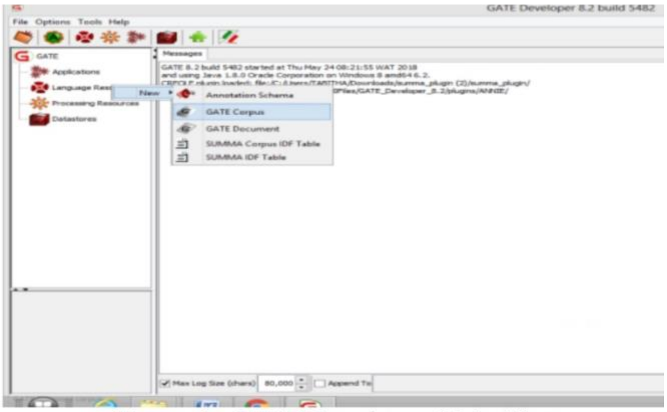
A summary is a text produced from one or more texts, which conveys important information in the original text, and is of a shorter form. Radev, et al defined summary as "a text that is produced from one or more texts, that conveys important information in the original text(s), and that is not longer than half of the original text(s) and usually, significantly less than that". Automatic text summarization is the task of using computers to produce a concise and fluent summary while preserving key information content and overall meaning.

Popov, et at 2017 described an approach towards semantic web information extraction and presented a model for semantic content enrichment. The model was implemented on a system called the Knowledge and Information Management (KIM) platform. KIM performs information extraction based on ontology and a massive knowledge base. The Information Extraction (IE) process in KIM was based on the General Architecture for Text Engineering (GATE) platform and it directly reused some of GATE's generic Natural Language Processing (NLP) components. The system's information extraction approach was based on the recognition of Named Entities (NEs) with respect to formal upper-level ontology. However, Popov, et al were only concerned with named entity extraction and not text summarisation. The work of Schilder and McCulloh was centred on temporal information extraction from legal documents. The work analysed the kinds of temporal information that could be found in the diverse types of legal documents; by providing a comparison of the different legal document types (case law, statute or transactional documents).

Although, the work focused on temporal information in legal text, how the information could be automatically extracted and how one could do reasoning with the extracted temporal information in order to add more value to the document; the work carried out extraction without annotation and thus not amenable to machine comprehension.

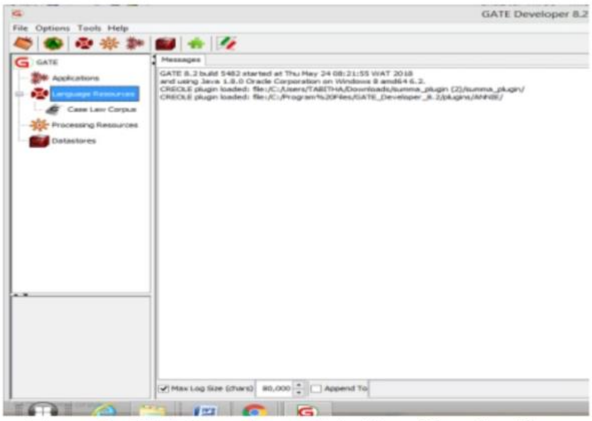
Wiebe, et in 2019 described the manual annotation of corpus based on opinions, emotions, and sentiments amongst other private states in language. The research stemmed from the desire to aid analysts in government and political domains to automatically track attitudes and feelings of people about happening events from news and online forums. The work presented multiple answers to non-factual multiperspective questions based on opinions from different sources. Annotation gold standard was realized manually and they made use of GATE which used the gold standard as basis to annotate other document sentences. However, the work's IE was abstractive and not extractive.

FLOWCHART



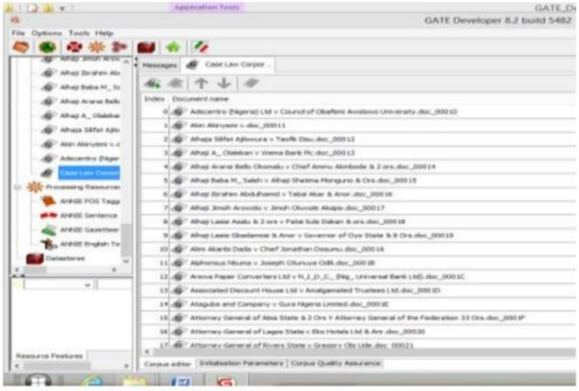
```
GATE 8.2 build 5482 started at Thu May 24 08:21:55 WAT 2018
and using Java 1.8.0 Oracle Corporation on Windows 8 amd64 6.2.
CREOLE plugin loaded: file:/C:/Users/TABITHA/Downloads/summa_plugin (2)/summa_plugin/
                      0Files/GATE_Developer_8.2/plugins/ANNIE/
```

Figure 1: GATE Developer Main View



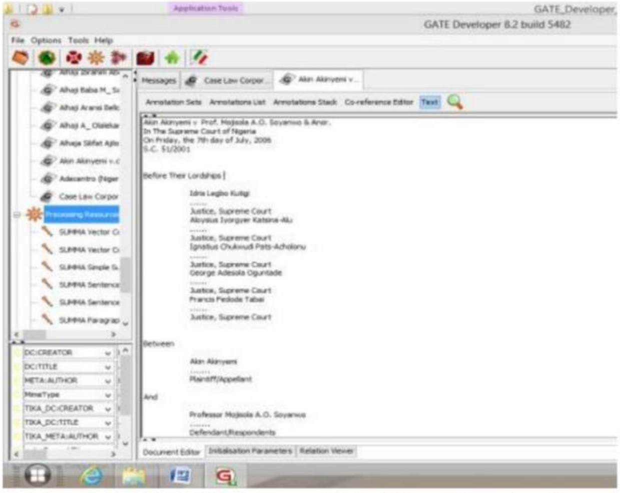
```
GATE 8.2 build 5482 started at Thu May 24 08:21:55 WAT 2018
and using Java 1.8.0 Oracle Corporation on Windows 8 amd64 6.2.
CREOLE plugin loaded: file:/C:/Users/TABITHA/Downloads/summa_plugin (2)/summa_plugin/
CREOLE plugin loaded: file:/C:/Program%20Files/GATE_Developer_8.2/plugins/ANNIE/
```

Figure 2: GATE Developer Main View Showing the Created Case Law Corpus



Index	Document name
0	Adecentro (Nigeria) Ltd v Council of Obafemi Awolowo University.doc_00010
1	Akin Akinyemi v.doc_00011
2	Alhaja Sitfat Ajibowura v Taofik Disu.doc_00012
3	Alhaji A._Olalekan v Wema Bank Plc.doc_00013
4	Alhaji Arana Bello Okomalu v Chief Aminu Akinbode & 2 ors.doc_00014
5	Alhaji Baba M._Saleh v Alhaji Shetima Monguno & Ors.doc_00015
6	Alhaji Ibrahim Abdulhamid v Tabal Akar & Anor.doc_00016
7	Alhaji Jimoh Arowolo v Jimoh Oluwole Akapo.doc_00017
8	Alhaji Lasisi Asalu & 2 ors v Fatai Sule Dakan & ors.doc_00018
9	Alhaji Lasisi Gbadamosi & Anor v Governor of Oyo State & 8 Ors.doc_00019
10	Alimi Akanbi Dada v Chief Jonathan Dosunmu.doc_0001A
11	Alphonsus Nkuma v Joseph Otunuya Odili.doc_0001B
12	Arewa Paper Converters Ltd v N.I.D.C._(Nig._Universal Bank Ltd).doc_0001C
13	Associated Discount House Ltd v Amalgamated Trustees Ltd.doc_0001D
14	Atagubo and Company v Gura Nigeria Limited.doc_0001E
15	Attorney General of Abia State & 2 Ors V Attorney General of the Federation 33 Ors.doc_0001F
16	Attorney-General of Lagos State v Eko Hotels Ltd & Anr.doc_00020
17	Attorney-General of Rivers State v Gregory Obi Ude.doc_00021

Figure 3: GATE Developer Main View Showing the Loaded Case Law and ANNIE Resources



Akin Akinyemi v Prof. Mojisola A.O. Soyanwo & Anor.
In The Supreme Court of Nigeria
On Friday, the 7th day of July, 2006
S.C. 51/2001

Before Their Lordships

Idris Legbo Kutigi
....
Justice, Supreme Court
Aloysius Iyorgyer Katsina-Alu
....
Justice, Supreme Court
Ignatius Chukwudi Pats-Acholonu
....
Justice, Supreme Court
George Adesola Oguntade
....
Justice, Supreme Court
Francis Fedode Tabai
....
Justice, Supreme Court

Between

Akin Akinyemi
....
Plaintiff/Appellant

And

Professor Mojisola A.O. Soyanwo
....
Defendant/Respondents

Figure 4: GATE Developer Main View Showing the Selected SUMMA Processing Resources

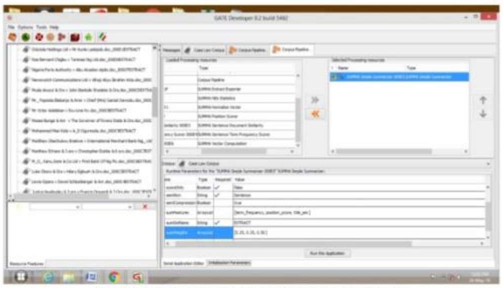


Figure 6: GATE Developer Main View Showing Summarised Case Law

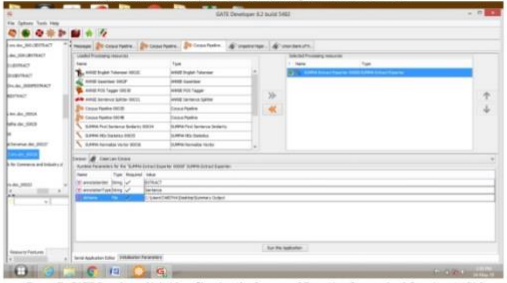


Figure 7: GATE Developer Main View Showing the Process of Exporting Summarised Case Law to Disk

Methods used in project :

This paper adopted the General IE System Architecture approach – the defacto approach to text annotation and extraction [26]. The annotation and summarization of case law were based on case elements. The case elements considered include: case title, name of court, date of judgment, judge(s), suit number, parties in court, lead judge – where there exists more than one case decider, fact of the case, cause of action etc. In particular, annotation of the selected Nigeria Supreme Court case law was performed using GATE with A Nearly New Information Extraction System (ANNIE) components while extractive summarisation of the annotated case law was carried out using GATE with SUMMA plug-ins. The study made use of 72 Nigeria Supreme Court electronic case law. To annotate case elements of the selected case law, the case law corpus was created and loaded on the GATE platform. To create case law corpus, Language Resources was right clicked on the Resource Tree in the GATE Developer Main View which displays on launching GATE; as shown in Figure 1. Then, New+GATE corpus was selected and thereafter, parameters for the corpus in the Parameter Dialog Box were set. The “ok” button was then clicked. On successful creation of the corpus, the corpus name displayed on the Language Resources menu of the To load the case law into the created corpus, the created corpus (Case Law Corpus) on the Language Resources menu of the Resource Tree, was right clicked. “Populate” button was then selected and the directory where the case law were stored on the Dialog Box that appeared was supplied. The “ok” button was clicked to complete the task. On successfully loading the case law, the main view then displayed the loaded case law in the Language Resource menu of the Resource Tree; as shown in Figure 3. Subsequently, the required annotation “Processing Resources” beneath the created case law corpus in the Resource Tree were loaded; by repeatedly right clicking and selecting New+Additional Required Resource until all required processing resources were loaded. The loaded processing resources immediately displayed on the Processing Resources menu in the Resource Tree; as shown in Figure 3. The loaded processing resources were: ANNIE Sentence Splitter (for sentence segmentation), ANNIE English Tokenizer (for tokenization), ANNIE POS Tagger (for POS tagging) and ANNIE Gazetteer (for entity and relation detections). The required SUMMA Processing Resources were also loaded following the same iterative process as that of the ANNIE Processing Resources. When successfully loaded, the SUMMA Processing Resources were displayed in the GATE Developer main view as shown in Figure 4.The selected SUMMA Processing Resources were: SUMMA NEs Statistics, SUMMA Position Scorer, SUMMA Sentence Document Similarity, SUMMA Normalize Vector, SUMMA Term Frequency Filtering, SUMMA Vector Computation, SUMMA First Sentence Similarity, SUMMA Sentence Term Frequency Scorer, SUMMA Simple Summarizer and SUMMA Extract Exporter. The tasks of the selected SUMMA Processing Resources are detailed in [27]. The next task was to “Run” the resources on the loaded cases. This was done by right clicking “Applications” button on the Resource Tree of the GATE Developer Main View. Then, Create New Application+Corpus Pipeline was selected . Thereafter a dialog box with the parameters for the new corpus pipeline was displayed and the “ok” button on it was clicked to complete the task. This immediately created a corpus pipeline below the Application message in the Resource tree as shown in Figure 5. The created Corpus Pipeline was then populated with the loaded Processing Resources. The Corpus Pipeline’s parameters were set as required to achieve the desired annotation and summarisation. To populate the Corpus Pipeline, the Corpus Pipeline was right clicked and the “shows” button that displayed thereafter was selected. The loaded Processing Resources were then displayed for selection in the order they will be Run; as shown in Figure 5

Evaluation:-

A very critical part of IE is the evaluation of the annotated text on which extraction was done. The importance of evaluation in text engineering stems from the fact that what cannot be measured and expressed in either quality or quantity is inconsequential to man and oftentimes cannot be relied upon. Commonly, processes and operations in IE and IR are measured for purposes of dependency and trust using metrics such as Precision, Recall and F-measure. Consequently, this research paper measured the Precision, Recall and F-measure of the case law’s annotation for extractive summarisation using GATE; since GATE is the platform of annotation, extraction and summarization. GATE is a complete text engineering tool not only because it supports most text engineering processes but also because it enables the processes, artifacts and systems built on it, to be evaluated for performance quality [28]. A veritable tool in GATE for evaluating annotation including those for IE is the Annotation Diff Tool (ADT). ADT is able to calculate the Precision, Recall and F-measure of the annotated text under evaluation according to three different criteria of strict, lenient and average [28]. The

Strict measure considers all partially correct responses as incorrect (spurious),the Lenient measure considers all partially correct responses as correct while the Average measure allocates half weight to partially correct responses (i.e. it takes the average of strict and lenient).These metrics for evaluating IE systems are defined as follows.

(i) Strict Criteria

$Precision = Correct\ Correct + Spurious + Partial$

(1) $Recall = Correct\ Correct + Missing + Partial$

(2) $F - Measure = (\beta\ 2 + 1) recision \times Recall\ (\beta 2 \times Precision) + Recall$ (3)

(ii) Lenient Criteria

$Precision = Correct + Partial\ Correct + Spurious + Partial$ (4)

$Recall = Correct + Partial\ Correct + Missing + Partial$ (5)

$F - Measure = (\beta\ 2 + 1) recision \times Recall\ (\beta 2 \times Precision) + Recall$ (6)

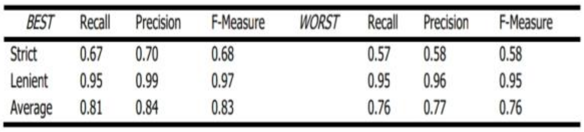
(iii) Average Criteria

$Precision = Correct + 1\ 2\ Partial\ Correct + Spurious + Partial$ (7)

$Recall = Correct + 1\ 2\ Partial\ Correct + Missing + Partial$ (8)

$F - Measure = (\beta\ 2 + 1) recision \times Recall\ (\beta 2 \times Precision) + Recall$ (9)

In all, β reflects the weighting of precision vs. recall. However in GATE's ADT
was carried out using GATE with SUMMA plug-ins. The study made use of 72 Nigeria Supreme Court electronic case law.



BEST	Recall	Precision	F-Measure	*WORST*	Recall	Precision	F-Measure
Strict	0.67	0.70	0.68		0.57	0.58	0.58
Lenient	0.95	0.99	0.97		0.95	0.96	0.95
Average	0.81	0.84	0.83		0.76	0.77	0.76

Result Interpretation and Discussion:-

Table 1 captures about the best and about the worst performance of the automatic summarisation annotation of the selected Nigerian case law using GATE with ANNIE and SUMMA plug-ins. A strict recall of 0.67 means that 67% of the sentences, words and phrases in the case law were correctly and completely annotated as it should be annotated; a lenient recall of 0.95 means 95% of the sentences, words and phrases in the case law were annotated while an average recall of 0.81 means roughly 81% of the sentences, words and phrases in the case law were correctly annotated as it can be annotated. A strict precision of 0.58 means that 58% of the annotated sentences, words and phrases in the case law were correctly and completely annotated as it should be annotated; a lenient precision of 0.96 means 96% of the annotated sentences, words and phrases in the case law were annotated correctly while an average precision of 0.77 means roughly 77% of the annotated sentences, words and phrases in the case law were correctly annotated as it can be annotated. A strict F-measure of 0.68 means that 68% of the annotated sentences, words and phrases in the case law were of excellent annotation quality; a lenient Fmeasure of 0.97 means that 97% of the annotated sentences, words and phrases in the case law were of fair annotation quality while an average F-measure of 0.83 means that 83% of the annotated sentences, words and phrases in the case law were of good annotation quality. The average F-measure best captures the system's performance as it mitigates outliers. The implication of this result is that the developed summariser is capable of 83%, but guarantees 76%, retention of the original case law's meaning. The summarised case law is not 100% in meaning compared to their original version as expected. The reason for this is probably due to the poor support of the tools used with respect to the indigenous names and culture of the Nigerian people as it affects her legal system.

Conclusion:-

Legal reasoning and judicial verdicts is highly dependent on case law. This ever increasing number of case law make the task of comprehending case law cumbersome even for experienced legal practitioners and this stifles their efficiency. This paper adopted the General IE Systems Architecture approach and deployed GATE platform with ANNIE and SUMMA plugins for automatic extractive text summarisation of some Nigeria Supreme Court case law. The automatic summarised case law which abridged the original case law by about 80% was established to be 83% reliable in the semantic preservation of its original version in the context of case elements. The result calls for creation of indigenous plug-ins to existing text engineering tools.

References:-

1) www.wikepedia.com
2) https://www.makeuseof.com/tag/websites-aid-daily-routine/
3) https://in.pcmag.com/web-sites/72977/10-sites-you-have-to-check-every-day
4) https://buffer.com/resources/daily-success-routine/
5) https://www.calendar.com/blog/what-are-the-advantages-of-scheduling/