

# Automatic Comprehension and Summarisation of Legal Contracts

Sibusiso Kubeka\*, Abejide Ade-Ibijola

*Formal Structures, Algorithms and Industrial Applications Research Cluster, Department of Applied Information Systems, University of Johannesburg, Bunting Road Campus, Johannesburg, 2006, South Africa*

---

## ARTICLE INFO

### Article history:

Received: 31 August, 2020

Accepted: 09 September, 2020

Online: 10 March, 2021

---

### Keywords:

contract

clauses

automatically comprehend and

summarise

context-free grammars

---

## ABSTRACT

*Contracts may range from a simple agreement between a tenant and a landlord or a gym contract, or it could be as important as an employment or marital contract. No matter the level of importance, individuals are legally obligated to obey and carry out all clauses in the contract. In this paper, we have identified that the majority of people seldom read through the entire contracts for several reasons such as the size of the contracts i.e. bulky contracts or the inability to fully comprehend a contract. As a solution to the identified problem, this paper presented a software tool that automatically comprehends and summarises legal contracts. We designed context-free grammar (CFG) rules for the recognition of critical clauses found in contracts. These CFG rules were implemented in the software tool. An evaluation of this tool showed that it was able to identify critical clauses in contracts to an accuracy of 79.2%.*

---

## 1 Introduction

In this section, we present an overview and contributions of the paper. The history of contract laws dates back to prehistoric civilizations which was profoundly influenced by ancient Roman and Greek thoughts [1]–[2]. At the time contract laws were established around a form of action identified as the action of “assumpsit”, which was put into effect in the early 16<sup>th</sup> century as a solution for the breaches in informal contracts made by word of mouth [3, 4]. As the years have gone by contract laws have evolved and there have been several developments with regards to the type of contracts [5, 6].

With all these new developments of contract laws comes several challenges and unwanted problems, such as the recent discovery that majority of contract receivers fail to thoroughly comprehend bulky contracts which has led to a large number of law cases related to breaches in modern day contracts by contract receivers [7, 8]. Contract receivers are easily exploited by organisations given that they do not have a full understanding of the clauses identified in the contract [9, 10].

In this paper we discussed in great detail a feasible solution to the problems at hand. We mainly focused on developing a web application that is capable of comprehending any legal contract, extracting semantic information as well as generating accurate summaries that includes all the important clauses stated in a contract.

Furthermore, we incorporated an Artificial intelligence concept of natural language processing (NLP) into our proposed solution in order to automatically comprehend and extract critical sections as well as to find patterns from within a contract [11]–[12].

Natural language processing is a phenomenon that has been in existence for close to 50 years and focuses on techniques such as automatic comprehension, summarisation and information extraction. NLP is by far more efficient and will help save time as compared to processing the bulky contracts manually [13].

In order to develop the web application we needed to start with the cleaning out of unnecessary information and white spaces by performing lexical analysis on the contracts, we then outlined a context-free grammar to identify syntactic patterns in the contracts, and finally we had semantic rules that were used to make conclusions on the basis of the identified patterns [14]–[15].

Figure 1 is a graphical representation of the logical flow, that we used to automatically comprehend and summarise a legal contract document. The diagram outlines the different phases a contract will undergo, these phase include pre-processing, document slicing and finally compiling.

The contributions of the paper are as follows. We have:

1. conducted extensive literature reviews on contracts,

---

\*Corresponding Author: Sibusiso Kubeka, University of Johannesburg, Bunting Road Campus, Johannesburg, South Africa, skubeka32@gmail.com

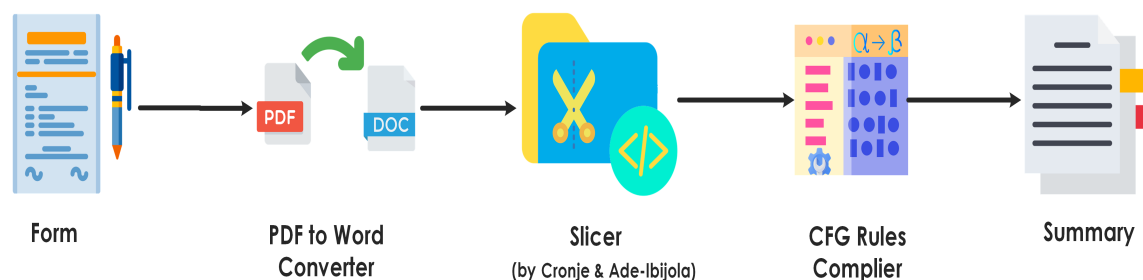


Figure 1: Flow diagram of the comprehension and summarisation process.

2. performed lexical and document slicing of contract documents,
3. identified any hidden implications as well as critical clauses that one should look out for in a contract,
4. formulated context-free grammar rules to help with the identification of patterns within contracts, and
5. implemented the CFG rules into the web application to automatically comprehend and summarise contract documents.

## 2 Background

In this section, we provided a detailed literature review on contracts, the structure of contracts, different types of contracts that exist, various critical clause found in contracts, information extraction, automatic text comprehension and summarisation, and related work.

### 2.1 Contracts

A contract can be defined as a promise enforceable by law, it is a legal written agreement that requires the mutual approbation of two or more parties [16, 17]. There are only two roles identified in contracts namely: the person making an offer and the person accepting the proposed offer. If for any reason one of the involved parties fails to adhere to the condition stated in the contract, the other party is entitled to legal reparation [18].

### 2.2 Structure of Contracts

In this section, we outlined the typical structure of a contract. Contract structures will differ depending on the type of contract.

There are many different types of contracts available and each contract serves a specific purpose [19, 20]. Types of contracts include: General business contracts (Partnership agreements, indemnity agreements, loan agreement and property and equipment lease), Sales-related contracts (Bill of sales, purchase order and security agreement), and lastly Employment contracts (General employment contract, Non-compete agreement and independent contractor agreement).

The structure of a contract typically includes the following categories. Parties to the contract: the details of all involved parties. Recitals: factual information as well as the purpose of the contract. Definitions: explanations of keywords. Conditions precedent: the

state of affairs required. Operative provisions: the rights and obligations of the parties involved. Boilerplate (or Miscellaneous) clauses: important clauses found in a contract. Schedules: the appendix to a formal document. Signatures: distinctive pattern used for identification of parties. The main objective of identifying the structure of a contract is to gain a better understanding in order to be able to recognize the syntax or patterns and create semantic rules to make inferences on identified patterns [21].

### 2.3 Critical Clauses in Contracts

Within a contract, a clause is defined as a specific section or prerequisite that focuses on an aspect of an agreement. Generally, clauses are used to provide details on aspects such as party duties and roles, rights, regulations, as well as benefits [11, 22, 23]. In this section, we provide a list of critical clauses and important information typically found in legal contract documents. The objective is to identify which information is relevant and should be extracted from an original contract, to create an informative and accurate summary.

#### Recitals clause

This is a contract clause commonly referred to as the “whereas” clause, it is usually associated with the purpose of a contract. This clause provides the contract receiver with an idea of what the contract is about and why they should sign the contract.

#### Contract Parties

this section in a contract usually contains information about all parties involved in the agreement. It typically includes the company details, contract receiver information, and any other third party.

#### Dates

We must be able to extract dates such as “11 July 2020” or “11-07-2020”, as they could be representing important information such as the Commencement or end date of a contract, or it could be a representation of important milestones or conditions that need to be accomplished i.e. “lease period”, “termination dates”, etc.

#### Period/Duration

This could indicate for instance the duration of the contract or number of hours an individual is required to work in a week. The extraction of keywords such as “10 Hours”, “15 days”, or “three years” is critical and should, therefore, be

included in the summary. Duration and period in contracts are a representation of “working hours”, “inspection period”, and “eviction notice”.

### Money/ currency

It is important to extract money and different currency symbols such as “R10”, or ZAR, USD, JYP, etc as these units could represent information such as fines or penalties, salaries, or rent amount. Within a contract money variables are a representation of “Cancellation fees”, “Compensation and benefits”,

### Definitions

The summary must include all definition statements found in a contract, these statements normally begin with words such as “is defined as”, “refers to”, “shall mean”, and “means that”.

### Conditions

Conditional sentences consist of condition clauses such as “if/ if not”, “when/ when not”, “where”, “as soon as”, “upon the occurrence”, and many other clauses or conditions found in a contract [11].

### Amounts

It is important to include non-financial amounts when extracting information for summaries. This could include quantities such as “Three hundred thousand”.

## 2.4 Information Extraction

Information Extraction (IE) can be defined as the automatic extraction of structured information from unstructured source documents. It takes natural language inputs and can convert them into structured texts using a set of indicated criteria. Information extraction is normally associated with the following sub-tasks: Name Entity Recognition and Linking, Conference Resolution, and Relation Extraction which form part of the segments of NLP tasks like Text Summarisation, Machine Translation, etc. IE technologies are mainly tasked with discovering valuable and relevant information, to construct a significant representation of the original document [24, 25].

There are two types of approaches for automated information extraction using NLP, namely the rule-based approach, and the machine learning-based approach [24, 25]. The machine learning-based approach makes use of machine learning algorithms for text processing based on text features of a certain text [26]. On the other hand, the Rule-based approach employs manually coded rules for text processing. The rule-based approach creates reiterative and refined rules to expand the precision of text processing. The Rule-based approach generally requires more human effort and makes use of several NLP techniques such as tokenization, parts of speech (POS) tagging, phrase structure analysis, and sentence slicing [24]. In this paper, we will make use of context-free grammar (CFG) for syntactic analysis to reduce the number of patterns required in information extraction.

## 2.5 Automatic Text Comprehension and Summarisation

Automatic text comprehension is a phenomenon that has been in existence since the early 70s and has since become popular in today’s society [27, 28]. Text comprehension is a far more complex task when compared to other computational tasks such as reading or writing (inputting). It is the ability to gain a thorough understanding of a given statement. Understanding a text automatically deduces that one has the ability to form an interconnected and cohesive interpretation of the information found in that particular text. There are several Natural language Processing (NLP) applications that require comprehension of source documents and it is important to note that the majority of those applications comprise of semantic processing of a natural language [28, 29].

Automatic text summarisation is a process related to the construction of a brief and precise version of an original text with the use of a computational software program. The aim is to create summaries that are considerably shorter and contain relevant and important information found within the subsequent document i.e. a contract [30, 31]. Automatic text summarisation can be achieved using two approaches:

### Extraction

The extraction-based approach to automatic summarisation focuses on extracting important information based on keywords or key-phrases [32]. The information extracted is not modified nor represented in a new way or form.

### Abstraction

The abstraction-based approach, on the other hand, is aimed at using techniques such as paraphrasing in order to provide summaries that are more structured and represent information in a new way, this usually includes information that was not present in the original document. In short, abstractive summaries are semantic representations of the source document.

In this paper, we looked into creating summaries for contract documents using the extraction approach. There is an all-encompassing range of practices that have been recommended for constructing summaries in automatic text summarisation. And it is said that most efficacious practices are based on either the positioning of words/ sentences in a text or on text retrieval. Text retrieval techniques are statistically more fruitful when implemented on source documents for text summarisation.

## 3 Related Work

From the research that we have conducted it is safe to conclude that very few attempts have been made to automate the comprehension and summarisation of legal documents. There are numerous articles and literature reviews have been published on automatic comprehension and summarisation of source documents using CFGs. Several papers have also been written on the use of natural language processing and machine learning on contracts and legal texts.

### FINCHAN

In [29], the author presented an innovative technology that

used a Context-Free Grammar to create a subset of every statement in a given Instant message. This article focused on creating a software application for automatic comprehension and summarisation of financial chats.

### LexNLP

In [11], the author proposed a new technology that uses natural language processing as well as machine learning to extract information in legal and regulatory documents. LexNLP is a software designed to enable end-users to turn an unstructured contract into a structured data object. The main aim of LexNLP is to make tasks such as regulation analysis and migration of legal documents a lot easier.

### Source Code

Automatic Summarisation of Source Codes in [15], the author proposed a solution that enables software developers to read shorter and accurate source codes to save development time. The solution involves using techniques from automatic text summarisation. The paper states that a combination of summarisation techniques is more effective when summarising source codes.

### Semantics in Law

Automatic Extraction of Semantics in Law Documents in [33] present a tool called SALEM, that uses Natural language Processing practices to execute two main tasks, namely classification of law paragraphs in relation to their regulatory substance, and extraction of relevant information in the form of text fragments. The article outlines the architecture of the system as well as a report based on a case study on Italian laws.

### Rule Extraction

NLP Approaches for Rule Extraction from Legal Documents by Dragoni presented a software tool that intermixes several Natural Language Processing techniques in order to extract rules found in contracts or legal documents. The paper proposed the combination of linguistic information together with syntax-based extraction of rules from a contract. Furthermore, these techniques will be combined with the logic-based extraction of dependencies.

### Summarisation of Legal Texts

In [34], the author proposed a method that automatically summarises legal texts using graph-based summarisation algorithms. Each legal document has its own set of connected graphs, the connected graphs show a cluster which shared the same clause or topic. This method does not require any self-developed linguistic features.

### LetSum

An automatic Legal Text Summarising system in [35], the author presented a system that can establish the thematic structure of a legal document and group it into four sections, namely Introduction, context, legal text analysis and lastly the conclusion. From there, the system is able to recognize the relevant text for each section.

## 3.1 The Gap

People tend to accept or sign contracts without reading through them thoroughly which could result in; Exploitation of contract receivers, the legal cost acquired by contract receivers, etc. Most text summarisation tools are not designated for contract comprehension and summarisation. There has not been any technology developed to provide contract receivers with accurate textual summarisation of critical clauses found in contracts.

In this paper, we were able to identify a gap in that there are no automatic tools that have been developed to help with comprehending and summarising legal contracts. This tool will help the people that struggle with reading bulky contracts by providing them with relevant summaries of contracts.

## 3.2 Reason for using Context-free Grammars

In this section, we outline the two main reasons for using CFG rules to automatically comprehend contract documents.

**Usage of a Document Slicer** When developing the tool, we identified that using a document slicer to slice through a contract to recognize relevant sections would increase the accuracy of the findings of the applied CFG rules. For our web application, we made use of an automatic slicer API [36, 37].

**Distribution of Grammar Rules** The use of CFG rules ensures effortless distribution of grammar rules amongst other researchers and scientists that aim to comprehend or synthesize similar documents. Within the science field, it is critical to have work that can effortlessly be distributed, reusable, or modified by individuals occupied with similar content.

**Use of FLAT** Formal Languages and Automata Theory (FLAT) refers to a popular feature of Theoretical Computer Science. FLAT is known for having a lot of mathematical theories with very little real-life implementation [38]. The application of FLAT helps with identifying new applications within areas such as computer science education.

## 3.3 Definition of Terms

**Definition 1 (Context-Free Grammar)** *Context-free grammar (CFG) is a set of recursive rules utilized to create patterns of a given text. A CFG is formally defined as a 4 tuple,  $G = (V, \Sigma, P, S)$  where:*

1.  $V$  represents the variables or nonterminal characters (the nonterminal characters indicates the types of clauses found in a sentence)
2.  $\Sigma$  represents a finite set of terminal characters (these are the alphabets of a given language).
3.  $P$  represents the production rules.
4.  $S$  is the start symbol. The relation  $S \in N$  must always hold. [39].

**Definition 2 (Lexical analysis)** *Lexical analysis also referred to as tokenization or lexing is a computer science process that involves converting a sequence of characters into tokens. A token is usually divided into two sections namely the token name and an attribute value [40, 41]. A lexer is a program that performs lexical analysis and it is often used to find specific patterns such as white spaces, new lines, and digits or letters in any text or document.*

## 4 Context-free Grammar for Comprehension of Contracts

### 4.1 Basis

Here we specify the building block for the rules that is presented later in this section. We have three major building blocks namely  $n$ ,  $spc$ , and lastly  $rand\_text$ .  $n$  represents any numeric value,  $spc$  represents tabs and spaces found within a text,  $rand\_text$  represents any random combination of letters that make up words.

### 4.2 CFG rules

The following are CFG rules for generating the subset of all critical clauses and important information found in contracts. Figure 2 is the syntax tree diagram outlining the structure of contracts. Contracts are grouped into three categories as discussed earlier, namely General business contracts  $\langle gb \rangle$ , Sales-related contract  $\langle sr \rangle$ , and Employment contracts  $\langle emp \rangle$ . A contract  $\langle contr \rangle$  includes components such parties to the contract  $\langle pc \rangle$  and critical clauses  $\langle cc \rangle$ . The production of these components is as follows:

$$\langle contract \rangle \rightarrow \langle gb \rangle | \langle sr \rangle | \langle emp \rangle \quad (1)$$

$$\begin{aligned} \langle gb \rangle &\rightarrow \langle pc \rangle \langle recital \rangle \langle CC\_comp \rangle \\ &(\langle CC\_gb\_opt \rangle | \lambda) \end{aligned} \quad (2)$$

$$\begin{aligned} \langle sr \rangle &\rightarrow \langle pc \rangle \langle recital \rangle \langle CC\_comp \rangle \\ &(\langle CC\_sr\_opt \rangle | \lambda) \end{aligned} \quad (3)$$

$$\begin{aligned} \langle emp \rangle &\rightarrow \langle pc \rangle \langle recital \rangle \langle CC\_comp \rangle \\ &(\langle CC\_emp\_opt \rangle | \lambda) \end{aligned} \quad (4)$$

$$\langle CC\_comp \rangle \rightarrow \langle term\_date \rangle \langle comm\_date \rangle | \quad (5)$$

$$\rightarrow \langle comm\_date \rangle \langle term\_date \rangle \quad (6)$$

All types of contracts contain a section that represents the parties to the contract, hence production Rules 2, 3, and 4 consist of  $\langle pc \rangle$ . A parties to the contract section is made up of details about

the company  $\langle com \rangle$  and one or more contract receiver  $\langle cr \rangle$ .

$$\langle pc \rangle \rightarrow \langle com \rangle (\langle cr \rangle)^+ \quad (7)$$

$$\begin{aligned} \langle cr \rangle &\rightarrow \langle first\_name \rangle \langle spc \rangle \langle last\_name \rangle \\ &\langle cell\_no. \rangle (\langle email \rangle | \lambda) \end{aligned} \quad (8)$$

$$\langle com \rangle \rightarrow \langle com\_name \rangle \langle com\_info \rangle \quad (9)$$

$$\langle com\_info \rangle \rightarrow x \in (\langle address \rangle, \langle tel\_no \rangle, \langle fax \rangle)^3 \ni: {}^3P_3 \quad (10)$$

$$\langle address \rangle \rightarrow \langle area\_no. \rangle \langle spc \rangle \langle street\_name \rangle \quad (11)$$

$$\rightarrow \langle city \rangle \langle spc \rangle \langle postal\_code \rangle \quad (12)$$

$$\langle tel\_no \rangle \rightarrow \langle tel\_no. \rangle \langle n \rangle \quad (13)$$

Here  ${}^3P_3$  is a k-permutation of  $x$  (i.e. address, tel\_no, and fax notices without repetitions).

Recital clauses are found in all contract types. We observed that the word “WHEREAS” is often associated with the recital in a contract, and if a line in the contract start with “WHEREAS” (usually written in capital letters), it often means that the purpose of the contract  $\langle poc \rangle$  is to follow.

$$\begin{aligned} \langle recital \rangle &\rightarrow (\langle rand\_text \rangle | \lambda) \langle WHEREAS \rangle \\ &\langle spc \rangle \langle poc \rangle \end{aligned} \quad (14)$$

$$\langle poc \rangle \rightarrow (\langle rand\_text \rangle | \lambda) \quad (15)$$

The symbols  $\langle CC\_gb\_opt \rangle$  from production Rule 2 is composed of four optional critical clauses: deposit  $\langle deposit \rangle$ , lease period  $\langle lease\_period \rangle$ , rent  $\langle rent \rangle$ , and eviction notices  $\langle evic\_not \rangle$ .

$$\begin{aligned} \langle CC\_gb\_opt \rangle &\rightarrow x \in (\langle deposit \rangle, \langle lease\_period \rangle, \\ &\langle rent \rangle, \langle evic\_not \rangle)^4 \ni: {}^4P_4 \end{aligned} \quad (16)$$

Here  ${}^4P_4$  is a k-permutation of  $x$  (i.e. deposit, lease period, rent, and eviction notices without repetitions).

The symbols  $\langle CC\_sr\_opt \rangle$  from production Rule 3 is composed of: description of goals  $\langle goal\_desc \rangle$ , tax/rate  $\langle tax \rangle$ , delivery instructions  $\langle del\_inst \rangle$ , and inspection period  $\langle insp\_period \rangle$ .

$$\begin{aligned} \langle CC\_sr\_opt \rangle &\rightarrow x \in (\langle goal\_desc \rangle, \langle tax \rangle, \\ &\langle del\_inst \rangle, \langle insp\_period \rangle)^4 \ni: {}^4P_4 \end{aligned} \quad (17)$$

Here  ${}^4P_4$  is a k-permutation of  $x$  (i.e. description of goals, tax, delivery instructions, and inspection periods without repetitions).

The symbols  $\langle CC\_emp\_opt \rangle$  from production Rule 4 is composed of: compensation  $\langle compen \rangle$ , benefits  $\langle ben \rangle$ , working hours  $\langle hours \rangle$ , roles and responsibilities  $\langle roles \rangle$ .

$$\begin{aligned} \langle CC\_emp\_opt \rangle &\rightarrow x \in (\langle compen \rangle, \langle ben \rangle, \\ &\langle hours \rangle, \langle roles \rangle)^4 \ni: {}^4P_4 \end{aligned} \quad (18)$$

$$\begin{aligned} \langle compen \rangle &\rightarrow \langle salary \rangle | \langle wages \rangle | \langle pay \rangle | \\ &\langle remuneration \rangle | \langle earnings \rangle \end{aligned} \quad (19)$$

$$\langle roles \rangle \rightarrow \langle responsibility \rangle | \langle duties \rangle \quad (20)$$

$$\begin{aligned} \langle leave \rangle &\rightarrow \langle sick\_leave \rangle | \langle annual\_leave \rangle | \\ &\langle maternity\_leave \rangle \end{aligned} \quad (21)$$

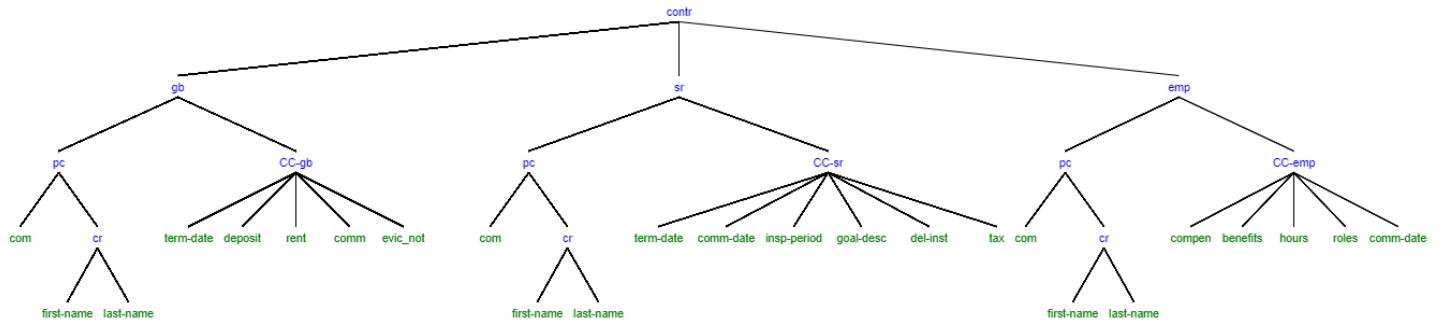


Figure 2: Syntax tree outlining the structure of contracts.

Here  ${}^4P_4$  is a k-permutation of x (i.e. compensation, benefits, working hours, and roles and responsibilities without repetitions).

We also observed that the words “shall mean”, “is defined as”, “means that” is often associated with definitions found in a contract, and if a line in the contract contains these words it often means a definition is to follow.

$$\begin{aligned} \langle \text{def} \rangle &\rightarrow (\langle \text{rand\_text} \rangle | \lambda) (\text{shall mean} | \\ &\text{is defined as} | \text{means that}) \\ &\langle \text{spc} \rangle \langle \text{definition} \rangle \end{aligned} \quad (22)$$

$$\langle \text{definition} \rangle \rightarrow (\langle \text{rand\_text} \rangle | \lambda) \quad (23)$$

**Money or Currency:**

$$\langle \text{cur} \rangle \rightarrow \text{ZAR} | \text{INR} | \text{ZMW} | \text{JPY} | \quad (24)$$

$$\rightarrow \text{MUR} | \text{KES} | \text{UGX} | \text{USD} | \text{EUR} | \quad (25)$$

$$\rightarrow \text{CNY/RMB} | \text{GBP} \quad (26)$$

**Date and Duration:**

$$\langle \text{date} \rangle \rightarrow \langle \text{day} \rangle \langle \text{spc} \rangle \langle \text{mon} \rangle \langle \text{spc} \rangle \langle \text{year} \rangle \quad (27)$$

$$\langle \text{day} \rangle \rightarrow \langle \text{Mon} \rangle | \langle \text{Tue} \rangle | \dots | \langle \text{Sun} \rangle \quad (28)$$

$$\langle \text{mon} \rangle \rightarrow \langle \text{Jan} \rangle | \langle \text{Feb} \rangle | \dots | \langle \text{Dec} \rangle \quad (29)$$

$$\langle \text{year} \rangle \rightarrow \langle \text{n} \rangle \langle \text{n} \rangle \langle \text{n} \rangle \langle \text{n} \rangle \quad (30)$$

$$\langle \text{dur} \rangle \rightarrow \langle \text{year} \rangle | \langle \text{mon} \rangle | \langle \text{days} \rangle \quad (31)$$

**Acts and laws:** This section includes the important acts and laws that are found in a contract document.

$$\langle \text{Act} \rangle \rightarrow \langle \text{Act} \rangle \langle \text{n} \rangle^+ \langle \text{of} \rangle \quad (32)$$

$$\rightarrow \langle \text{year} \rangle \langle \text{rand\_text} \rangle | \lambda \quad (33)$$

$$\langle \text{Act} \rangle \rightarrow (\langle \text{rand\_text} \rangle | \lambda) \quad (34)$$

## 5 Implementation and Results

In this section, we outline the implementation details as well as the results of the proposed tool for automatic comprehension and summarisation of legal contract documents.

### 5.1 Implementation Details

The Context-Free Grammar rules defined in Section 4 of this paper were executed in a web application using Microsoft’s Visual Studio 2019 as the development environment. We used the .Net framework with Visual Basic as our programming language. Furthermore, we made use of a document slicer API by Cronje and Ade-Ibijola [36].

### 5.2 Web Application

In section, we present a brief explanation on the sections found in our web application, we will also provide screenshots of the relevant parts of the web application. The web application is made up of three major sections, namely, the home page section , contract summarisation section and the summary section. Figure 3-5 below are the respective screenshots of each of mentioned sections. Table 1 is an example of a summary produced by the tool.

## 6 Evaluation of Tool

In this section, we presented an in-depth evaluation of our tool based on a performance analysis we had conducted on the tool. We also present the user perception of the tool based on the results gather from a survey we had conducted.

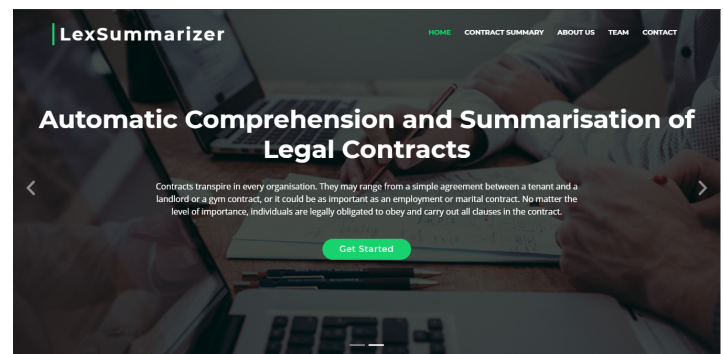


Figure 3: Homepage of the web Application.



Table 1: Example of a summary produced by the tool.

Contract section	Clauses from the tested contract	Information extracted from the contract	Production rule(s)
Acts and Laws	1	The terms and conditions set out herein will constitute the employee’s contract with the company. Where a basic condition of employment is not specifically mentioned, the relevant legislation will be applicable (e.g. the Basic Conditions of Employment Act, Act 75 of 1997, the Labour Relations Act, Act 66 of 195, etc.).	32
	11	The payment of maternity leave will be determined in terms of the provisions of the Unemployment Insurance Act, Act 30 of 1966 read together with the provisions of the Basic Conditions of Employment Act, Act 75 of 1997.	34
	12	in the event of death of the employee’s spouse or life partner, parent or adoptive parent, grandparent, child, adopted child, grandchild or sibling.	32-34
	17	The company subscribes to the principle of freedom of association as stipulated in the Constitution of the Republic of South Africa, Act 108 of 1996, and thus the employee may join any organisation/trade union of his/her own choice as regulated in the Labour Relations Act, Act 66 of 1995.	32-34
Leave	8	Leave must be applied for in writing in the form and manner prescribed by the company from time to time, and may only be taken after approval by the company or its delegated authority.	21
	9	An application for sick leave must be supported by a certificate from a registered medical practitioner.	21
Remuneration	4	The employee’s total monthly remuneration will be R , payable in arrears on.	19
	5	Benefits	18
	6	The remuneration method in 7.2.1 above will be the normal method of remuneration. Should the employer need to use one of the other options due to circumstances, he/she will inform the employee accordingly, preferably in writing, before the commencement of overtime.	19
	13	The employer may not deduct any monies from the employee’s salary unless the employee has agreed thereto in this contract or in writing on each occasion.	19
Working Hours	7	The employer may also by agreement grant two (2) paid working days off in lieu of payment.	17
Termination	3	The employer will be entitled to terminate this contract in terms of the Disciplinary Procedures referred to in paragraph 20 hereof.	5-6
Dates and Duration	16	The appointment under this contract is a full time appointment and the employee shall devote his/her full commitment, energy and attention to the employer’s business.	27-31
	10	on a date from which a medical practitioner or a midwife certifies that it is necessary for.	27-31
False positive error	2	During the period of employment within the company the employee will report to (the) and obey instructions given by him/her and any other person duly.	N/A
	15	This limitation of trade is restricted to the nature of the employer’s business, products and services.	N/A
	14	The employee will not be liable to the employer for information divulged in terms of legislation or a court order compelling him/her to do so.	N/A

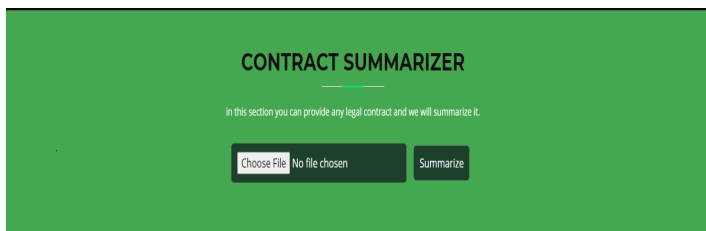


Figure 4: Contract summarisation section of the web Application.

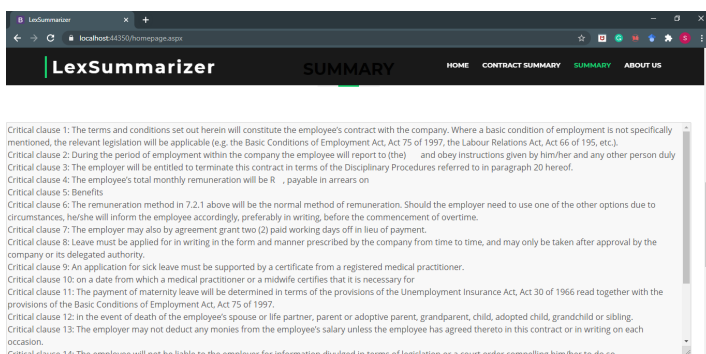


Figure 5: Summary produced by our CFG rules.

### 6.1 Accuracy of Tool

The analysis was founded by one perspective, which was, the accuracy in identifying critical clauses in a contract. To measure the accuracy of our tool, we tested it on five different contract documents in which all critical clauses were manually identified before the commencement of the test.

From the first contract were able to identify a total of 22 clauses and our tool was able to identify 17 clauses in the same contract document giving us an accuracy of 77.3%. In the second contract, we manually recognised a total of 14 clauses, and the tool identified 11 critical clauses, which gave us an accuracy percentage of 78.6%. Contract three consisted of five critical clauses and the tool recognized four clauses giving us an accuracy of 80%. The fourth contract we tested consisted of 24 manually recognised clauses and the tool was able to recognise 21 clauses from the same contract. The last contract consisted of seven critical clauses and the tool identified four of those clauses which gave an accuracy of 57.1%.

In total, we were able to manually identify 72 critical clauses from the five tested contracts and our tool was able to identify 57 of these clauses. From these results, we were able to calculate an aggregate accuracy percentage of 79.2%. Given that there are different types of contracts that exist, we can observe the vast difference in the tool’s accuracy with every test we conducted.

When conducting our tests on the five different contract documents, we recognised that in some cases the tool presented a few

statements that did not fall under the critical clause category in the summary, which indicated that our tool produced false-positive errors. To calculate the rate of the false-positive errors we use the formula  $\frac{FP}{FP+TN}$ , where FP represented the number of false positives and TN was the number of true negatives. The total number of false-positive errors in each of the tested contracts is as follows, contract one had three, contract two had no false-positive errors, contract three consisted of two, contract four had a total of four false-positive errors and lastly contract five consists of two false-positive errors. After calculations, we identified that the tool produced an aggregate false positive rate of 0.13 (13%).

4 and yes
a few time. yes I did
Mostly everything
Just once. Most of the time, but only relevant stuff.
I have signed many software licenses, I have also signed bursary agreements and contracts regarding my studies, contracts for opening bank accounts as well... For the normal contracts I read and understood every clause, but often with software licenses I sign without reading every clause
2, No not every clause
Twice. Yes.
3 times

Figure 6: Number for time participants have received a contract.

## 6.2 User Perception of Tool

To make our data collection and analysis process less complicated we divided our online survey into two sections, the first section contains questions based on general information and the user’s general knowledge and experience on contract documents. The second section included questions that were based on the user’s experience of our automatic comprehension and summarisation tool.

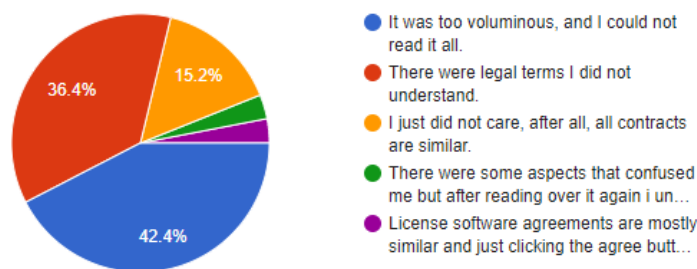


Figure 7: The challenges individual have when reading a contract.

From the raw data, we were able to recognize that of the 39 individuals that have read a contact, at least 26 respondents (66.7%) have been exposed to at least 2 contracts in their lives. We were also able to determine that 29 individuals out of the 39 respondents did not understand every clause in the contract documents they have received and read. This gave us a total percentage of 74.3 respondents that struggled to fully comprehend the clauses in a contract document. We were then able to conclude that the introduction of a tool that can automatically comprehend and summarize legal contract documents could provide great assistance to the majority of

individuals who receive contracts but struggle to fully comprehend the terms and clauses found in these contract documents.

Figure 7 outlines that voluminous contracts and the inability to understand legal terms as the main problems individuals generally face when reading a contract. Most of the survey participants agree that tool is useful and relevant.

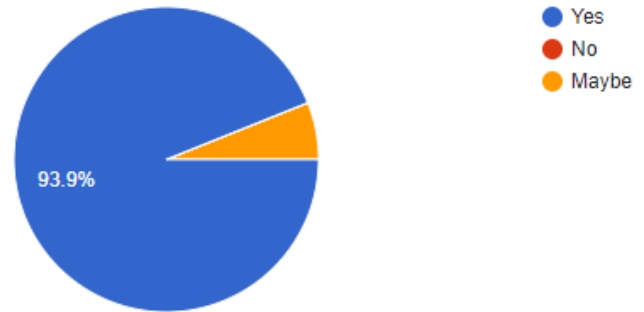


Figure 8: Usefulness of the tool.

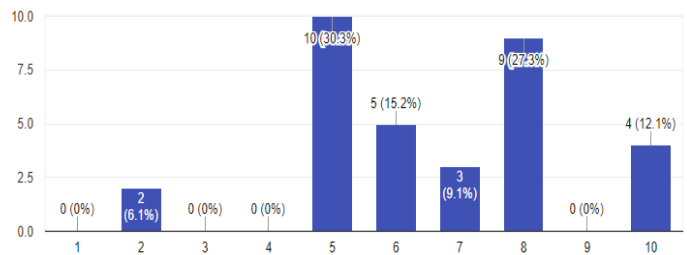


Figure 9: Accuracy of to from participants that tested the tool.

## 7 Conclusion and Future Work

In this paper we presented a newly formulated approach to automatically comprehend and summarise legal contract documents using formal grammar rules. We started by defining the structure of contracts to gain an in-depth understanding of the different sections found in a contract and their level of importance. We were then able to identify critical clauses that one would normally find in a contract document. From these critical clauses we were able to design CFG rules to outline the syntax of a legal contract document.

Our concept was implemented in a web applications using the .Net framework. When implemented, the web application was able to take any contract document, pre-process the document using the following techniques: PDF-to-Word conversion and lexical analysis. From there the contract is sliced into important sections using a document slicer API. Lastly, the contract is comprehended using our grammar rules and then an accurate summary can be provided.

We were able to perform an in-depth evaluation of our tool based on a performance analysis to measure the accuracy of the tool. The tool was tested on five different contracts and from these tests, we calculated an aggregate accuracy percentage of 79.2%. Furthermore, we identified that the tool produced an aggregate false positive rate



of 0.13 (13%). There was a respectable difference in each of the tests we conducted and we believe these differences were because of the various types of contracts that exist and the way they are structure.

In the near future, we will improve on the comprehension aspect of the presented CFG by designing random forbidden and permitting context grammars.

## References

- [1] F. Kessler, "Contracts of Adhesion—Some Thoughts about Freedom of Contract," *Columbia Law Review*, 7(2), 1–18, 1943, doi:10.2307/1117230.
- [2] J. W. Chisholm, "The 1990 contract: Its history and its content," *British Medical Journal*, 1990, doi:10.1136/bmj.300.6728.853.
- [3] A. W. B. Simpson, "The Horwitz Thesis and the History of Contracts," *The University of Chicago Law Review*, 35, 115, 1979, doi:10.2307/1599448.
- [4] M. J. Horwitz, "The Historical Foundations of Modern Contract Law," *Harvard Law Review*, 1974, doi:10.2307/1340045.
- [5] R. M. Parizi, Amritraj, A. Dehghantanha, "Smart contract programming languages on blockchains: An empirical evaluation of usability and security," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, doi:10.1007/978-3-319-94478-4\_6.
- [6] P. Steyaert, C. Lucas, K. Mens, T. D'Hondt, "Reuse contracts: Managing the evolution of reusable assets," in *Proceedings of the 11th ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, OOPSLA 1996*, 1996, doi:10.1145/236337.236363.
- [7] L. Budworth, A. Prestwich, B. Sykes-Muskett, K. Khatun, J. Ireland, F. Clancy, M. Conner, "A feasibility study to assess the individual and combined effects of financial incentives and monetary contingency contracts on physical activity," *Psychology of Sport and Exercise*, 2019, doi:10.1016/j.psychsport.2019.04.021.
- [8] W. Zou, D. Lo, P. S. Kochhar, X.-B. D. Le, X. Xia, Y. Feng, Z. Chen, B. Xu, "Smart Contract Development: Challenges and Opportunities," *IEEE Transactions on Software Engineering*, 2019, doi:10.1109/tse.2019.2942301.
- [9] T. Wilhelmsson, C. Willett, "Unfair terms and standard form contracts," in *Handbook of Research on International Consumer Law*, Second Edition, 2018, doi:10.4337/9781785368219.00012.
- [10] F. Gómez Pomar, J. Ganuza, M. Artigot, "Standard contract terms and legal controls: a view from the law and economics theory of contract (forthcoming)," 2018.
- [11] M. J. Bommarito, D. M. Katz, E. M. Detterman, "LexNLP: Natural language processing and information extraction for legal and regulatory texts," 2018, doi:10.2139/ssrn.3192101.
- [12] N. E. Okwunma, *Automatic Comprehension of Customer Queries for Feedback Generation*, Ph.D. thesis, 2019.
- [13] Y. Kim, J. Lee, E. B. Lee, J. H. Lee, "Application of Natural Language Processing (NLP) and Text-Mining of Big-Data to Engineering-Procurement-Construction (EPC) Bid and Contract Documents," in *Proceedings - 2020 6th Conference on Data Science and Machine Learning Applications, CDMA 2020*, 2020, doi:10.1109/CDMA47397.2020.00027.
- [14] K. Merchant, Y. Pande, "NLP Based Latent Semantic Analysis for Legal Text Summarization," in *2018 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2018*, 2018, doi:10.1109/ICACCI.2018.8554831.
- [15] S. Haiduc, J. Aponte, L. Moreno, A. Marcus, "On the use of automated text summarization techniques for summarizing source code," in *2010 17th Working Conference on Reverse Engineering*, 35–44, IEEE, 2010.
- [16] S. R. M. Rahim, Z. Z. Mohamad, J. A. Bakar, F. H. Mohsin, N. Isa, "Artificial intelligence, smart contract and islamic finance," *Asian Soc Sci*, 14(2), 145, 2018.
- [17] C. Dirican, "The impacts of robotics, artificial intelligence on business and economics," *Procedia-Social and Behavioral Sciences*, 195, 564–573, 2015.
- [18] J. Levin, "Relational incentive contracts," *American Economic Review*, 93(3), 835–857, 2003.
- [19] M. Suprpto, H. L. Bakker, H. G. Mooi, M. J. Hertogh, "How do contract types and incentives matter to project performance?" *International Journal of Project Management*, 2016, doi:10.1016/j.ijproman.2015.08.003.
- [20] H. Dagan, "Types of Contracts and Law's Autonomy-Enhancing Role," *Tourism Tribune*, 104–113, 2018.
- [21] S. Lyu, et al., "The legal structure of the right of unilateral change in travel contracts," *Tourism Tribune*, 33(5), 104–113, 2018, doi:10.1109/20183167907.2018.27.
- [22] U. Mattei, "The Comparative Law and Economics of Penalty Clauses in Contracts," *The American Journal of Comparative Law*, 1995, doi:10.2307/840646.
- [23] T. Eisenberg, G. P. Miller, E. L. Sherwin, "Arbitration's Summer Soldiers: An Empirical Study of Arbitration Clauses in Consumer and Nonconsumer Contracts," *SSRN Electronic Journal*, 2011, doi:10.2139/ssrn.1076968.
- [24] J. Zhang, N. M. El-Gohary, "Semantic NLP-Based Information Extraction from Construction Regulatory Documents for Automated Compliance Checking," *Journal of Computing in Civil Engineering*, 2016, doi:10.1061/(asce)cp.1943-5487.0000346.
- [25] J. Zhang, N. El-Gohary, "Extraction of construction regulatory requirements from textual documents using natural language processing techniques," in *Congress on Computing in Civil Engineering, Proceedings*, 2012, doi:10.1061/9780784412343.0057.
- [26] P. J. Tierney, "A qualitative analysis framework using natural language processing and graph theory," *International Review of Research in Open and Distributed Learning*, 13(5), 173–189, 2012, doi:10.1016/j.irrodl.2012.08.065.
- [27] W. Bakari, P. Bellot, M. Neji, "Towards an automatic text comprehension for the arabic questionanswering: Semantic and logical representation of texts," 2018.
- [28] F. Bravo-Marquez, G. L'Huillier, P. Moya, S. A. Ríos, J. D. Velásquez, "An automatic text comprehension classifier based on mental models and latent semantic features," in *ACM International Conference Proceeding Series*, 2011, doi:10.1145/2024288.2024317.
- [29] A. Ade-Ibijola, "FINCHAN: A grammar-based tool for automatic comprehension of financial instant messages," in *ACM International Conference Proceeding Series*, 2016, doi:10.1145/2987491.2987518.
- [30] P. W. McBurney, C. McMillan, "Automatic source code summarization of context for java methods," *IEEE Transactions on Software Engineering*, 42(2), 103–119, 2015.
- [31] A. T. Ying, M. P. Robillard, "Code fragment summarization," in *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*, 655–658, 2013.
- [32] J. Fowkes, P. Chanthirasegaran, R. Ranca, M. Allamanis, M. Lapata, C. Sutton, "Autofolding for source code summarization," *IEEE Transactions on Software Engineering*, 43(12), 1095–1109, 2017.
- [33] C. Soria, R. Bartolini, A. Lenci, S. Montemagni, V. Pirrelli, "Automatic extraction of semantics in law documents," 253–266, *European Press Academic Publishing*, 2007.
- [34] M. Y. Kim, Y. Xu, R. Goebel, "Summarization of legal texts with high cohesion and automatic compression rate," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2013, doi:10.1007/978-3-642-39931-2\_14.
- [35] A. Farzindar, G. Lapalme, "Letsum, an automatic legal text summarizing system," *Legal knowledge and information systems, JURIX*, 11–18, 2004.

- [36] M. Cronje, A. Ade-Ibijola, "Automatic slicer API," personal communication.
- [37] M. Cronje, A. Ade-Ibijola, "Automatic slicing and comprehension of CVs," in 2018 5th International Conference on Soft Computing & Machine Intelligence (ISCM), 99–103, IEEE, 2018.
- [38] J. Shallit, *A second course in formal languages and automata theory*, volume 179, Cambridge University Press Cambridge, 2009, doi:10.1017/CBO9780511808876.
- [39] A. Ade-Ibijola, "Synthesis of integration problems and solutions," in 2019 6th International Conference on Soft Computing and Machine Intelligence, ISCM 2019, 185–190, IEEE, 2019, doi:10.1109/ISCM147871.2019.9004291.
- [40] D. Watson, "Lexical Analysis," in *A Practical Approach to Compiler Construction*, 37–73, Springer, 2017.
- [41] J. Vakhrusheva, S. Khan, R. Chang, M. Hansen, L. Ayanruoh, J. J. Gross, D. Kimhy, "Lexical analysis of emotional responses to "real-world" experiences in individuals with schizophrenia," *Schizophrenia Research*, 2020, doi:10.1016/j.schres.2019.11.045.