A Semi Supervised Approach for Catchphrase Classification in Legal Text Documents

Imran Sarwar Bajwa[1*], Fatim Karim[1], M. Asif Naeem[2], Riaz Ul Amin[3]

[1] Department of Computer Science & IT, The Islamia University of Bahawalpur
[2] School of Computer and Mathematical Sciences, Auckland University of Technology, New Zealand
[3] School of Computing Science, University of Glasgow, UK

* Corresponding author. Email: imran.sarwar@iub.edu.pk

Manuscript submitted March 2, 2016; accepted May 20, 2016.
doi: 10.17706/jcp.12.5.451-461

Abstract: An agreement between a user and also the owner of a software program known as software license that allows a user to try to certain things that will somewhat be an infringement of copyright law. Typically, a software license agreement is based on set of rules that a user has to comply with while using the software. Sometimes, the price of the software and licensing fees is usually described elsewhere, but also discussed in the licensing agreement, however, typically used catchphrases in software License Agreement make the text of the agreement complex for a common user. Most of the users are very keen to understand the agreement before purchasing software, and it's very necessary for users and business managers to understand these agreements due to further use this software, because these software's are high cost and risky. In this paper, a semi supervised approach is presented to extract catchphrases from software license agreement and provide meanings to minimize the understanding complexity for a reader. The results of experiments show that the proposed approach for automatic extraction of catchphrases from software license agreements make its interpretation easy and straightforward.

Keywords: Catchphrases, legal text, natural language processing, software license.

1. Introduction

Every text document has some legal points of great importance with respect of identifying precedents defined as catchphrases. Catchphrases are presented all the legal point deliberated instead that just summarizing the key point(s) of a decision. Hence, catchphrases [1] have an indicative function rather than informative. A legal text is somewhat very different from common speech. Typically skilled by those persons who have best knowledge of legal filed and requires speed and accuracy in which documents are classified. These legal specialists look for information relevant to specific cases in these summaries and often summarize decisions [2]. In this section, we discuss related work presented regarding the legal text summarization work which is done in previous years. Legal text summarization is done by many techniques in the past [3]. We have added some previous related research about legal text summarization in natural language processing. The software license agreement is one of the most important but highly flouted part of software installation process, one of the primary reasons could be the complexity, length and the inability of a common user to understand it in totality, which bolsters the user to skip to the end and just accepts the terms, without a mere understanding of the implications of actions that he/she is about to undertake. However, people think it is necessary to understand the Software License Agreement before they sign it.

Such examples are shown in Table 1.

Table 1. Examples of Catchphrases List from Typical License Agreements

Intellectual Property Rights	Means any and all rights existing from time to time under patent law, copyright law, semiconductor chip protection law, moral rights law, trade secret law, trademark law, unfair competition law, publicity rights law, privacy rights law.
Limited Title	Your Limited Title shall be further subject to Your return of such Hardware pursuant to this Agreement.

Table 1 shows an example of legal catchphrases taken from license agreements. Here, we found that some of catchphrases are very case specific, usually there are some high level like important catchphrases (e.g. intellectual property rights or limited title), this catchphrases cannot understandable for a common user [4]-[8].

Extraction of catchphrases is important from software license agreements for technical user's and business managers because these software's are purchased according high fees, so these license agreements are not understandable for common user and business managers. To understanding this license agreement extraction of catchphrases are very important rather to hire technical persons to understand these license agreements [9]. However, software licenses are difficult to understand for a common user because every license agreement has contained a lot of catchphrases (important legal points in a document) [10], these catchphrases are difficult to understand for a user. Software license agreement is essential to understand for business mangers when they purchasing software, therefore, in this thesis we purpose an automated approach to make these catchphrases understandable for common user and business managers [11], [12]. In order to encourage the user to understand prior to accepting the software license agreement he/she should have an interpreted and straightforward version of it. This can be done through extract catchphrases from software license agreement and provide meanings to minimize the complexity. In this research, we aim to address this issue be providing some assistance to the reader by providing meanings of the extracted catchphrases given in a software license agreement.

To our best of knowledge, a few approaches have been presented to extract catchphrases from legal text and summarize the legal text documents [13]-[20]. However, no specific work has been presented to extract catchphrases from software license agreements. Since the nature of software license agreements is different from legal documents such as document structure, catchphrases, etc. A dedicated approach is required to process the natural language based software license agreements and extract the catchphrases.

2. Using EM for Catchphrase Classification

In this research we have used algorithm to solve our problem that is defined in Section 1, these sequential steps are as follows:

1) We taking input from software license agreement.
2) Get this input in POS tagger which gives tagged information for each word.
3) Identify the boundary of sentence and separate it.
4) Find the nature of sentence.
5) Tokenize these sentences.
6) Find subject, check the tagged name entity.
7) Find main verb, which shows an action in the sentence.
8) Find object, same as done for finding subject.
9) Generating parse tree generation and dependencies.
10) Now add phrase to the phrase list.
11) Apply these rules repetitively sentence by sentence to the whole paragraph.
12) Create thematic segments for source data using rhetorical roles.

13) Eliminate the irrelevant data at filtering stage.
14) Check the terms and extract catchphrases.
15) Show output in table style format.

Here, Semi-supervised classification algorithms have been performing better for natural language text classification better as compared to the supervised classification algorithms. Semi-supervised algorithms such as the Expectation-Maximization (EM) [21] successfully used for classification of NL text documents. EM is one of the iterative algorithms that are capable of completing missing data. EM typically computes the expected value for each missing value in data based on maximum likelihood estimation of existing values in the data and such expected values fill the missing values in the data. Such ability of EM algorithms makes it suitable for classification of NL data that is inherently incomplete.

Input of EM approach is a collections R_l of labeled legal text and R_u of unlabeled text. The EM algorithm is applied in two stages: Expectation stage, and Maximization stage. In Expectation stage, the missing data is filled in and in the Maximization stage, the parameters are estimated on the basis of filled data. To perform these two stages, the training of naive Bayesian classifier performed as shown in equation (1). For training, the labeled examples used only that becomes initial phase of EM and then this process is iteratively repeated.

$$\mathrm{P}(c_j \mid v_i) = \frac{\mathrm{P}(c_j)\prod_{k=1}^{|v_i|}\mathrm{P}(v_i \mid c_j)}{\sum_{r=1}^{|C|}\mathrm{P}(c_r)\prod_{k=1}^{|v_i|}\mathrm{P}(v_i \mid c_j)} \tag{1}$$

Here, R is a set of legal text that are representation of software requirements and an ordered list of vocabulary items. Each vocabulary item in a legal text r_i is represented by v_i, where text is list of vocabulary as V = < v_1, v_2, … , $v_{|n|}$ >. Here, the vocabulary items are classified into target UML class model elements. In this paper, the possible types of UML class model items are denoted as classes, C = {c_1, c_2, … , $c_{|n|}$}. Here, classification s performed by computing the posterior probability, P(c_j | R_i), where *cj* is a class and *vi* is a legal text vocabulary item.

For the maximization phase, equations (2) and (3) are used, where equation (2) represents the Bayesian probability and the multinomial model.

$$\mathrm{P}(c_j) = \frac{\sum_{i=1}^{|R|}\mathrm{P}(c_j \mid v_i)}{|R|} \tag{2}$$

In equation (3), the Laplacian smoothing is used, where $N(v_t, r_i)$ is the total number of times the vocabulary item v_t appears in a rule r_i and where P(c_j | r_i)   {0, 1, 2, … . n} depends on the class label of a vocabulary item. Here, the naive Bayesian classifier helps in identifying the class with the top P(*cj*|*di*) value and that class is assigned to that vocabulary item.

$$\mathrm{P}(v_i \mid c_j) = \frac{1+\sum_{i=1}^{|R|}N(v_t, r_i)\,\mathrm{P}(c_j \mid r_i)}{|V|+\sum_{s=1}^{|V|}\sum_{i=1}^{|R|}N(v_s, r_i)\,\mathrm{P}(c_j|r_i)} \tag{3}$$

Here, an important issue is raised in [22] where it is investigated that the basic version of EM works for only where there exists one–to–one relationship in classes and mixture components. Since, in NL, there can be a few cases where a class can be associated with multiple mixture components and such situation leads to a one-to-many relationship in classes and mixture components. It is discussed in (Zhang, W., 2015) that multiple mixture components M-EM performs better than basic EM with text classification. In our approach, we experimented with both variations of EM approach.

3. Used Approach

In this section we discuss the architecture of the solution. In which these modules has been discussed: lexical analysis, syntax analysis, Thematic Segmentation, filtering and selection of catchphrases. The architecture of the used approach is shown in Fig. 1:

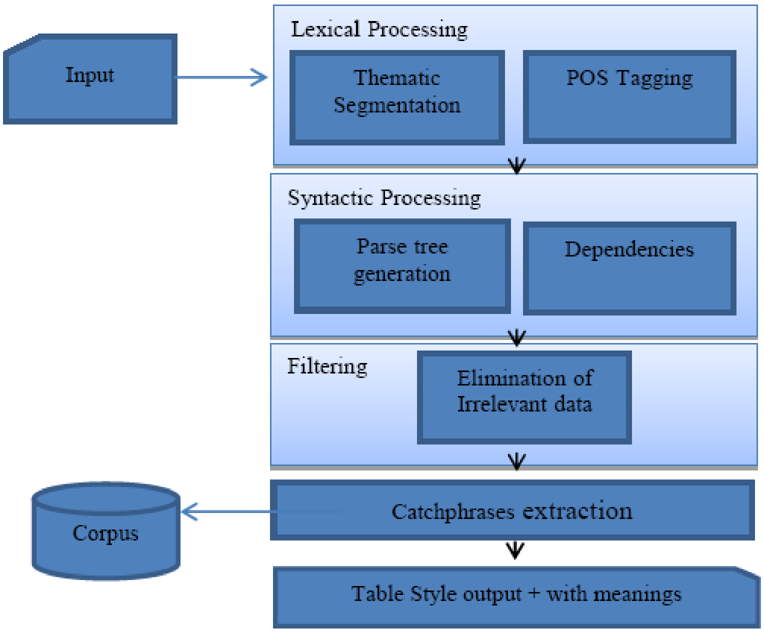

Fig. 1. The practical steps for producing table style output.

The output is built in four phases (Fig. 1): lexical processing, syntactic processing, filter of less important units such as citations of articles of the law, and choose the relevant text units. Following is description of each phase.

3.1. Input Acquisition

For recognizing catchphrases we are taking input of legal text from software license agreement, which is a case study of legal text. A legal document is taken as an input. Each sentence from that document will be considered and catchphrases will be detected according rules. Organized the first phase of treatment in the NL analysis involves a number of processing units for data processing complex English and these units as shown in Fig. 1 presents the framework of NL-parsing.

3.2. Lexical Processing

NL analysis begins with the lexical processing of a plain text file containing the text English language programs legal. This is the process of making streams of characters and also called scanning.

It is known that the process of converting a string of characters in a series of symbols and lexical analysis. The function is called to perform the analysis of lexical analyzer, scanner, or laxer. To build a diagram that embodies icons structure of the source language is a simple way to build a lexical analyzer, the result codes and then planned to hand translates to the program. Can be produce lexical analysis efficiency in this way. Lexical analysis is the first phase of the natural language processing in which include these stages: tokenization, wholesale division, marking some of the words (POS) and morphological analysis.

Role of a Lexical Analyzer: the lexical analyzer is the first phase of the complex. Its main task is to read the input letters and output produces a series of symbols used by the analyzer to analyze syntax. To determine how to work with in the processing of lexical and we have an example of the license agreement.

Sentence Splitting: Sentence splitting is to divide the sentence is determined wholesale margin and then each sentence is stored separately in an array list. Split in Java method is available which allows you to split the regular expression that is used to divide the series into a string array. After sentence splitting of example 3.1, following is the output:

Tokenization: After sentence splitting the tokenization of the English text input that is taken from software license agreement. In the analysis of lexical entries read from left to right and in the group of icons. Symbols are a sequence of characters with a collective meaning. In the first step the software requirements and read and use for tokenize (recognized symbols) Lexical analyst recognizes some cases of icons such as: system "software", etc. These specific cases are lexemes. In fact symbols formed by a group of characters that we called sequential lexemes.

Parts-of- Speech (POS) Tagging. According to the nature of every word Part Of Speech Tagger tag the words. Grammatical structure of sentences is presented by POS tagging. Each word is assigned its part of speech i.e. noun, verb, adverb etc.

Thematic Segmentation: according to rules divide the text into structural blocks is done by thematic segmentation. We develop these structural blocks with the specific knowledge of software license agreement, and match the catchphrases from our corpus for every thematic block. We develop three thematic segments is as follows:

License Grant: Subject to the conditions of this Contract, and in consideration of Your payment of all appropriate taxation and charges as set forth in the Shopping Trolley Area of the Search engines Store web site ("Fees"), Search engines allows to You (and You accept to conform with) a non-sublicensable, non-transferable, non-exclusive.

Ownership: For reasons of this Contract, "Intellectual Property Rights" means any and all privileges current every now and then under certain law, trademark law, semiconductor processor protection law, ethical privileges law, trade secret law, signature law, unjust competitors law, advertising privileges law, comfort privileges law, and any and all other exclusive privileges, and any and all programs, renewal, additions and corrections thereof, now or hereafter in power and effect globally.

Terms and Conditions: Topic to Your transaction of all Charges, the phrase of the certificate provided herein for any Item shall start upon the time frame of shipping by Search engines or its specific broker and may be ended as set forth herein.

Extraction of catchphrases from thematic segments is as follows in Table 2 using example 1.

Table 2. Thematic Blocks and Extracted Catchphrases from License Agreement

Blocks	Thematic Segments	Catchphrases with meanings
1	License Grant	**Fees:** Your payment of all applicable fees and taxes as set forth in the Shopping Cart Section. **Software:** installed in computer hardware.
2	Ownership	**Intellectual property:** any and all rights existing under patent law, copyright law etc. **Limited Use:** Your Restricted Headline shall be further topic to Your come back of such Components pursuant to this Contract.
4	Terms & Conditions	**Warranties:** should get that a system will function in all content aspects as described in the appropriate system certification for one year after distribution. **Conditions:** license agreement is under all applicable fees and taxes.

3.3. Syntactic Parsing

Grammatical analysis used to determine the structure of the text input. This structure consists of a hierarchy of terms, and the smallest is called the underlying code largest wholesale. It is based parser that we use in our approach to the rules of the English language. At this stage, after the grammatical analysis of the text tree analysis (see Fig. 1) is generated for further analysis. In syntactic analysis Stanford parser generate parse tree and its dependencies.

Parse Tree: Parse tree generation done by Stanford parser, it consists of a hierarchy of terms; the smaller is the basic code is called the largest wholesale. For example after syntactic processing of example 3.1, following is the output:

```
(ROOT
 (S
  (NP
   (NP (DT The) (NN software))
   (CC and)
   (NP (NNS services)))
   ... ... ...
```

Dependencies: We use Stanford parser to generate its dependencies. For example after syntactic processing of example 3.1, following is the output:

```
det(software-2, The-1)
nsubjpass(provided-6, software-2)
cc(software-2, and-3)
conj(software-2, services-4)
... ... ...
```

3.4. Filtering

To determine which parts of the text, which can be removed without loss of relevant information for output and this is done through the filter. Content is less important to the summary, such as units of martyrdom is the occupation of a large amount of text, and more than 30% of the judgment. So we remove such content within blocks of thematic sectors. Therefore, we liquidation of two segments: the arguments and defences associated with citations from the previous things or signals to the appropriate legislation and that report and the views of the parties in the litigation. In the case of the elimination of quotation of legislation (such as the License Agreement), we provide a reference citation in the validity of data in the field of power. Direct and indirect: two types of indicators on which it is based identify Citation in: first, a direct indicator is one of the linguistic signs that we are classified into three categories: acts and concepts (noun, Chuck, recipe) and supplementary data. Examples of acts of the quote are: Closure of the provision, read, identifies, and indicate, and references to, say, the state, see, and summarizing. Examples of concepts are: the following page, paragraph, section, subsection, pursuant to. Supplementary indicators include certain preposition, numbers and signs reading (colon, the quotation marks) and relative terms. Second, neighbouring units quoted phrase is indirect citations.

3.5. Corpus of Agreement Catchphrases

Documents of license agreement that record terms are usually stored electronically in different databases. Software license agreement of any software is one example of such database which is use as source data for our system. Software license agreement is one of the sources of legal materials on the net and that is easily available on net, with many searchable documents.

We accessed the case reports from software license agreements and extract these terms that may be used as and more similar to catchphrases. The corpus of these terms is in Annexure A. Our corpus of software license agreement is used as knowledge base which we extract catchphrases using our tool.

Looking at the collected list of catchphrases that is in annexure A, we presented in our corpus 360 words and almost 250 terms that are taken from software license agreements, and then we extracted some statistics from given text and calculate the precision (P), recall (R) and F-measure. We found that some of catchphrases are very case specific, usually there are some high level like important catchphrases (e.g. intellectual property rights, license grant and application provider) and some more common catchphrases (e.g. copyright) that can be found in a range of license agreement. Example from license agreement is given in Table 3.

Table 3. Examples of Catchphrases List from Typical License Agreements

Application Provider	**Reserves all rights not expressly granted to You. The Product that is subject to this license is referred to in this license as the "Licensed Application."**
Usage Rules	This license does not allow You to use the Licensed Application on any iPod touch or iPhone that You do not own or control.

In our corpus check these terms word by word if first word is matched then check second from corpus if second word also matched then check third word. After that return the output the term is matched or not. This poses some limit to what we can achieve from extraction based system, we consider that automatic extraction of catchphrases could bring as additional benefit, an increased consistency for catchphrases choice among software license agreement.

3.6. Selection of Catchphrases

For each structural level of the output selection builds a list of the foremost effective candidate catchphrases. We've an inclination to see for each word related to the next information: position of the paragraphs at intervals the thematic section, position of the paragraphs at intervals the document, and position of the sentences at intervals the paragraph, distribution of the words in document. we\'ve known some cue words and linguistic markers that square measure counting on the given info in every bedded phase. For example selection of catchphrases from software license agreements we first determine the thematic segmentation then choose catchphrases from thematic segmentation.

3.7. Relevance Identification

Various techniques can be used to extract fragments of the text is important. Approaches such as regular expressions are used to identify patterns in text, based on the terms or conditions braid / certain structures. However, when you consider the legal texts manually, we realized that the recognition of important content, several aspects of the text need to be considered. Looking in a single sentence in itself is clearly not enough to decide its importance: we must also consider information document the scope to find out what is the case today about, at the same time we need to look at the information on the scope of habeas corpus to decide what is peculiar to this issue. For this reason, we have developed several methods to identify potential catch phrases in the legal text, based on the different types of attributes, which form the basic building blocks of the judging system. We have got this idea from (Galgani, F., *et al.*, 2012) where the authors used 16 numerical attributes, in our work we used 13 attributes and one in addition F-measure. These attributes are given below:

Tf (Term frequency): Term t occurs how many times in document d.

AvgOcc: AvgOcc how many times occurs in the corpus.

Df (Document frequency): Number of documents in a collection that contains a term t at least once divided by the total number of documents is computed.

TFIDF: we combine the term frequency with inverse document frequency to produce a composite weight for computing weight for each term in each document. It is computed as the vigorous of the term in a document, for example TFIDF is 12 it's mean that the term 12 highest TFIDF in this document. The TF-IDF defines as:

$$tf - id = tf * idf$$

CpOcc: The set of all known catchphrases present in the corpus which how many times the term occur.

FcFound: This uses the known catchphrases to compute the ratio between how many times (that is in how many documents) the term appears both in the catchphrases and in the text of the case, and how many times in the text.

$$FcFound = (t)\,\frac{NDocstext\&catchp.(t)}{NDocstext\ (t)}$$

CitSen: All the sentences that cite the target case which how many times the term occur.

CitCp: All the catchphrases that are cite or cited by the target case which how many times the term occurs.

4. Experiments and Results

A case study of license agreement taken from Mini Google license agreement has been discussed in this section. We solved this case study in this research. Following is a part of the problem statement for the case study, solved in this research.

Case Study 1: “For purposes of Mini Google license Agreement, "Intellectual Property Rights" means any and all rights existing from time to time under patent law, copyright law, semiconductor chip protection law, moral rights law, trade secret law, trademark law, unfair competition law, publicity rights law. Intellectual property rights means any and all other proprietary rights, and any and all applications, renewals, extensions and restorations thereof, now or hereafter in force and effect worldwide. All ownership rights, title, and Intellectual Property Rights in and to the Products shall remain in Google and/or its licensors, except that title to the Hardware shall pass to You upon receipt of all Fees by Google ("Limited Title"). Your Limited Title shall be further subject to Your return of such Hardware pursuant to this Agreement. All ownership rights, title, and Intellectual Property Rights in and to the content accessed through the Product are the property of the applicable content owner and may be protected by copyright and/or other applicable laws.”

After lexical processing, thematic segmentation was performed according to rules divide the text into structural blocks is done by thematic segmentation. We develop these structural blocks with the specific knowledge of software license agreement, and match the catchphrases from our corpus for every thematic block. We develop six thematic segments is as follows:

After syntax analysis and parse Tree Generation by the Stanford parser, filtering was done to determine which parts of the text, which can be removed without loss of relevant information for output and this, is done through the filter. Content is less important to the summary, such as units of martyrdom is the occupation of a large amount of text, and more than 30% of the judgment. So we remove such content within blocks of thematic sectors. Therefore, we liquidation of two segments: the arguments and defences associated with citations from the previous things or signals to the appropriate legislation and that report and the views of the parties in the litigation. In the case of the elimination of quotation of legislation (such as the License Agreement), we provide a reference citation in the validity of data in the field of power. Finally, the catchphrases were extracted. As shown in Table 4.

Table 4. Thematic Blocks and Extracted Catchphrases from License Agreement

Blocks	Thematic Segments	Catchphrases with meanings
1	Definitions	**Academic institutions:** Indicates a degree-granting academic organization. **Agreement:** Means this Nationwide Equipment Application Certificate Contract, together with any and all appropriate Particular Product Addenda. **Authorized application:** those programs that you make with growth editions of the SOFTWARE that you have validly certified. **Software:** Means the software applications and other rule provided with this Contract that you are approved to set up and use in compliance.
2	License Grant	**Fees:** Your payment of all applicable fees and taxes as set forth in the Shopping Cart Section. **Software:** installed in computer hardware.

3	Third party components	**Third party contractor:** those persons who engaged and use the software. **Product:** responsible for each product that is used.
4	Terms & Conditions	**Warranties:** should get that a system will function in all content aspects as described in the appropriate system certification for one year after distribution. **Conditions:** license agreement is under all **applicable fees** and **taxes**.
5	Ownership	**Intellectual property:** any and all rights existing under patent law, copyright law etc. **Patent law:** clear law **copyright law:** official document **semiconductor chip:** An integrated circuit **protection law:** security laws **Moral rights law:** *Ethical privileges are privileges of creators of branded works.* **Trade secret law:** Any useful professional details that provides a company with a benefits over opponents who do not have that details. **Trademark law:** The signature proprietor can be an personal, business company, or any lawful enterprise. **Unfair competition law:** The law of unjust competitors is mainly consists of torts that cause an financial damage. **Publicity rights law:** the right of advertising is mostly secured by condition typical. **Limited Use:** Your Restricted Headline shall be further topic to Your come back of such Components pursuant to this Contract.
6	Limitations	**Shipment date:** Topic to Your transaction of all Charges, the phrase of the certificate provided herein for any Item shall start upon the time frame of shipping. **Termination:** Upon cancellations of this Contract, all permits, and any other privileges and solutions offered by Search engines to You as set forth in this Contract, shall stop instantly.

In this section we calculate the results using above solved case study, in this research and two more case studies solved elsewhere, which we extract total catchphrases, correct and incorrect catchphrases and missing catchphrases. Using these values of catchphrases we can calculate the value of Precision, Recall and F-Measure. Here, P indicate the precision value and R indicate the Recall value. Table 5 shows the results of P, R and F-measure.

Table 5. Calculated Values of P, R and F-Measure

Case studies	Total Catchphrases	Correct Catchphrases	Incorrect Catchphrases	Missing Catchphrases	P	R	F-Measure
1	34	25	4	5	86.2	73.5	79.3
2	29	24	3	2	88.9	82.8	85.7
3	55	40	10	5	80.0	72.7	86.7
4	62	45	10	7	81.8	72.6	76.9

Charts show the values of precision, recall and F-measure using Table 4. Following are the charts of case studies 1, 2, 3 and 4.

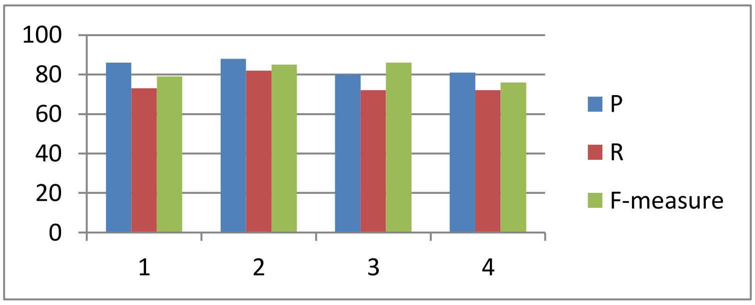

Fig. 2. Graph shows the overall precision, recall and f-measure values.

In this section, we solved case studies related to license agreements of Mini Google license agreement and End User License Agreement. These case studies solved using all phases that are defined in Fig.1 in chapter 3 one by one, hence for doing this we can calculate the values of Precision, Recall and F-measure. At the end these results shows in charts.

5. Conclusion

As we know from the best our knowledge catchphrases is extracted from legal case report, legal documents and the work that was done earlier is the legal text summarization, and extraction techniques was used to extract legal text.Most of the users are very keen to understand the agreement before purchasing software, and it's very necessary for users and business managers to understand these agreements due to further use this software, because these software's are high cost and risky. Catchphrases meant to know, every document has some legal points of great importance with respect of identifying precedents. A dedicated approach is required to process the natural language based software license agreements and extract the catchphrases and their meanings. This poses some limit to what we can achieve from extraction based system, we consider that automatic extraction of catchphrases could bring as additional benefit, an increased consistency for catchphrases choice among software license agreement.

References

[1] Alam, Y. S. (2007). Analyzer to identify phrases and the functional roles in sentences: Its architectural aspects. *Proceedings of PACLIC: Vol. 21* (pp. 67-75).

[2] Al-Rababah, K. S., Shatnawi, S. M., & Al-Rababah, A. (2011). Identifying significant single phrases in submitted free-order Arabic natural language questions. *Proceedings of International Conference on Information Society (i-Society)* (pp. 446-449).

[3] Farzindar, A., & Lapalme, G. (2004). Legal text summarization by exploration of the thematic structures and argumentative roles. *Proceedings of Text Summarization Branches out Workshop* (pp. 27-34).

[4] Bajwa, I. S., & Choudhary, M. A. (2006a). A rule based system for speech language context understanding. *Journal of Donghua University, 23(6)*.

[5] Bajwa, I. S., Naeem, M. A., Chaudhri, A. A., & Ali, S. (2011). A controlled natural language interface to class models. *Proceedings of ICEIS: Vol. 2* (pp. 102-110).

[6] Bajwa, I. S., Naeem, M. A., & Riaz-Ul-Amin, M. A. C. (2006b). Speech language processing interface for object-oriented application design using a rule-based framework. *Proceedings of 4th International Conference on Computer Applications*.

[7] Bajwa, I. S., Lee, M., & Bordbar, B. (2012). Resolving syntactic ambiguities in natural language specification of constraints. *Proceedings of Computational Linguistics and Intelligent Text Processing* (pp. 178-187). Springer Berlin Heidelberg.

[8] Bajwa, I. S., & Choudhary, M. A. (2006c). Natural language processing based automated system for UML diagrams generation. *Proceedings of The 18th Saudi National Computer Conf. on Computer Science* (NCC18). Riyadh, Saudi Arabia: The Saudi Computer Society (SCS).

[9] Farzindar, A., & Lapalme, G. (2004). Letsum, an automatic legal text summarizing system. *Legal Knowledge and Information Systems*, 11-18.

[10] Galgani, F., Compton, P., & Hoffmann, A. (2012). Citation based summarisation of legal texts. *Proceedings of PRICAI 2012 on Trends in Artificial Intelligence* (pp. 40-52). Springer Berlin Heidelberg.

[11] Galgani, F., Compton, P., & Hoffmann, A. (2012). Combining different summarization techniques for legal text. *Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data* (pp. 115-123). Association for Computational Linguistics.

[12] Grover, C., Hachey, B., & Korycinski, C. (2003). Summarising legal texts: Sentential tense and argumentative roles. *Proceedings of the HLT-NAACL 03 on Text Summarization Workshop: Vol. 5* (pp. 33-40). Association for Computational Linguistics.

[13] Hachey, B., & Grover, C. (2006). Extractive summarisation of legal texts. *Artificial Intelligence and Law, 14(4),* 305-345.

[14] Jones, S., Lundy, S., & Paynter, G. W. (2002). Interactive document summarisation using automatically extracted keyphrases. *Proceedings of the 35th Annual Hawaii International Conference on System Sciences* (pp. 1160-1169). IEEE.

[15] Yousfi-Monod, M., Farzindar, A., & Lapalme, G. (2010). Supervised machine learning for summarizing legal documents. *Advances in Artificial Intelligence*, 51-62, Springer Berlin Heidelberg.

[16] Farzindar, A., & Lapalme, G. (2009). Machine translation of legal information and its evaluation. *Advances in Artificial Intelligence*, 64-73, Springer Berlin Heidelberg.

[17] Francesconi, E. (2008). Towards semantic interpretation of legal modifications through deep syntactic analysis. *Proceedings of The Twentieth First Annual Conference on Legal Knowledge and Information Systems: Vol. 21* (p. 202).

[18] Galgani, F., & Hoffmann, A. (2011). Lexa: Towards automatic legal citation classification. *Advances in Artificial Intelligence*, 445-454, Springer Berlin Heidelberg.

[19] Lesmo, L., Mazzei, A., & Radicioni, D. P. (2009). Extracting semantic annotations from legal texts. *Proceedings of the 20th ACM conference on Hypertext and Hypermedia* (pp. 167-172). ACM.

[20] McCarty, L. T. (2007). Deep semantic interpretations of legal texts. *Proceedings of the 11th International Conference on Artificial Intelligence and Law* (pp. 217-224). ACM.

[21] Zhang, W., Tang, X., & Yoshida, T. (2015). Tesc: An approach to text classification using semi-supervised clustering. *Knowledge-Based Systems, 75*, 152-160.

[22] Zhang, W., Yang, Y., & Wang, Q. (2015). Using Bayesian regression and EM algorithm with missing handling for software effort prediction. *Information and Software Technology, 58*, 58-70.

Imran S. Bajwa has a PhD in computer science from School of Computer Science, University of Birmingham, UK and is an active research in the domain of natural language processing and artificial intelligence. He has also have been in University of Coimbra on a FCT research Project. He has more than 100 conference and journal publications in his field of research. He is also a General co-chair of IEEE ICDIM 2013 and IEEE INTECH 2016. Currently, Dr. Bajwa is an assistant professor of computer science at the Islamia University of Bahawalpur, Pakistan.

Muhammad A. Naeem has a PhD in computer science from University of Auckland, New Zealand. Currently, the author is a senior lecture in School of Engineering, Computer and Mathematical Sciences, Auckland University of Technology, New Zealand. Dr. Asif Naeem is an active research in the field of data and text analysis. He is a member of Data Analysis Research Group at Auckland University of Technology, New Zealand.

Riaz Ul Amin has PhD in Computer science from Univerity of Glasgow. Dr. Riaz is currently Assistant Professor of computer science in the Faculty of Emerging Sciences in Balouchistan University of Information Technology, Engineering, and Management Sciences, Quetta.