Supervised extraction of catchphrases from legal documents

ABSTRACT

The long and complicated textual structure of legal court case documents necessitates the need for a concise representation of the involved legal issues, which are captured through legal 'catchphrases'. Catchphrases are usually identified manually by legal experts. With the rapidly increasing number of digitized legal documents, automated extraction of such catchphrases is an essential phase of legal document indexing and legal information retrieval. In this paper, we address the task of automated catchphrase identification from legal case documents, using supervised techniques. We model the catchphrases as named entities, and train a supervised named entity recognizer to identify them through sequence labeling. We compare our proposed entity recognizers against a pre-existing set of baselines built for catchphrase extraction. We use a dataset comprising of legal court case documents from the Supreme Court of India. We observe that our proposed techniques exhibit human-competitive performance compared to the existing baselines.

KEYWORDS

Legal Information Retrieval, Legal catchphrases, Supervised methods, Sequence labeling

1 INTRODUCTION

The two primary sources of law in countries adhering to the *Common Law System* are - the laws promulgated by the legislature, and the precedents. Precedents are prior cases that enable lawyers to prepare their legal reasoning by providing an insight into the proceedings of the Court on similar preceding/prior cases that are not directly referenced in the statutes.

Legal documents can be lengthy and convoluted [3], making the task of understanding them strenuous and time-consuming, even by legal practitioners. Thus, there arises the need for a concise representation of the legal statement/proceeding, which is captured through 'catchphrases'.

'Catchphrases' are short phrases or words that occur within the text of legal documents. As noted in [7], catchphrases serve an 'indicative as well as an informative function'. They represent all the legal points considered in the case instead of simply summarizing the key points of a decision. This extraction of catchphrases is mainly done manually by legal experts as is seen in the case of *Manupatra Legal Search System* (http://www.manupatra.in/). [1] The voluminous amount of digitally available legal documents necessitates the need for automated extraction of catchphrases, as opposed to their manual identification by legal experts which is an onerous and costly task.

Almost all prior works of catchphrase extraction [11–13, 19, 22] have divided the task into two parts - firstly to generate the candidate phrases for a given document and then ranking them using a scoring function. The focus was built mainly upon how well a scoring function was built. These methods although were shown to perform well were restricted to the domain they were tested upon.

We remodel the catchphrase extraction task as a sequence labelling task as is done in case of most supervised *Named Entity Recognition (NER)* tasks. To this end, catchphrases are considered to be entities of 'legal' type. We employ a few well-performing named entity recognizers to identify 'legal' entities out of a document text. It is found that these NER taggers provide not just quantitatively superior catchphrases than the existing methods, but are qualitatively human-competitive as well, in the sense that we find new catchphrases that were previously undiscovered by the human annotators and were recognized as relevant by legal experts.

We evaluate our methods both quantitatively and qualitatively. The quantitative evaluation is done by using a dataset of 400 court case statements from the Supreme Court of India, annotated manually by the legal experts at *Manupatra*. And for the quantitative evaluation we hired some legal experts to manually evaluate a set of catchphrases while they were uninformed of the methods these were extracted from.

2 RELATED WORK

The application of information retrieval and natural language processing in the legal domain has gained recent research interest [10, 17, 21] . As more material is added to legal literature, automated techniques are required to facilitate query and search[10] summarizing[4, 16], translating legal documents [6] and the like.

The long and convoluted nature of legal documents necessitates some concise representation of the legal proceedings, which are captured through the catchphrases. Thus automated extraction of catchphrases is also an essential component in Legal IR.

There have been some prior works involving legal catchphrases. **Galgani et. al** [7] identified sentences with catchphrases embedded in them to summarize legal documents. The PS-legal method of [12] proposed an unsupervised methodology to extract the catchphrases in an automated fashion. They identified an initial set of viable candidate phrases (noun phrases) from a legal document and subsequently ranked them using a scoring function. The work of **Tran et al.** [20] employs a deep neural model to automate the extraction task.

We present a different approach to this problem wherein we remodel the catchphrase extraction task as a sequence labelling task as is done in case of most supervised *Named Entity Recognition (NER)* tasks. These techniques have been employed for other domain specific purposes. **Alvarado et al** [2] applied named entity recognition (NER) on financial documents to identify phrases pertaining to credit risk.

3 EXISTING SUPERVISED MODELS FOR CATCHPHRASE EXTRACTION

3.1 KEA - Keyphrase Extraction Algorithm

KEA [22] is a machine learning based tool which automatically extracts catchphrases from the document.[2] KEA is a standalone system which doesn't require any hand-crafted or corpus-specific

[1] *Manupatra* is a widely used online system that provides manually annotated catchphrases for Indian court case documents apart from other legal resources.

[2] http://www.nzdl.org/Kea/.

Case Id	Statements containing catchphrases
2002.INSC.493	They, therefore, did not apply for any **licence** nor paid **excise duty** ... on 19th November, 1993, the Collector (Appeals) confirmed the order of the **Assistant Collector** ... is not due to any fraud, **collusion** or willful mis-statement or suppression of fact
2015.INSC.547	that the **employees** who continued in the M.D. University on allocation**/absorption** with change of employer ... pension is not only **compensation** for loyal service rendered ... there should be no **discrimination** between one person ... even taking the **retirement age** as 58, should be excluded.
2003.INSC.304	beneficial to reproduce Rule 53 of the Rajasthan **civil Services** (Pension) ... service or has attained the age of 50 years, whichever is earlier, the **appointing authority**, upon having been satisfied ... difficulties faced by the Judicial Officers in **discharge** of their duties ... continuation of such proceeding despite permitting the **employee** concerned to retire

Table 1: A set of legal documents and few selected sentences from the documents. Gold standard catchphrases are indicated in bold. The documents for any case title YEAR.INSC.CODE can be found at http://liiofindia.org/in/cases/cen/INSC/YEAR/CODE.html, for instance, the first case is available at http://liiofindia.org/in/cases/cen/INSC/2002/493.html

features as input. Given some training examples of documents with their annotated keyphrases it learns a model by itself, which can then be applied on new documents, for extraction of catchphrases. The thorough functioning of KEA is presented in [22].

3.2 MAUI

MAUI[13] is a topic indexing model built upon four open-source components: the KEA[22] for n-gram keyphrase extraction, Weka for topic indexing, Jena that integrates extrinsic vocabularies and Wikipedia Miner[3]. To wield supervision it goes through the training examples of catchphrases along with their documents and creates the model which can later be used for extracting new catchphrases. This method is limited to extracting catchphrases that has been encountered in the training data. MAUI has parameters that can limit the size of the trained model as well as boost the speed of training.

4 SEQUENCE LABELLING MODELS FOR CATCHPHRASE EXTRACTION

In this section we propose several supervised methodologies of extracting catchphrases by formulating the task as a sequence labelling problem. We adopted the BIO scheme in accordance with standard practices in sequence labelling problems and named entity extraction tasks. According to the BIO scheme, the catchphrases were marked as entities of type 'LEG' while the other words were marked 'O'. The catchphrases *'amount of compensation'* and *'state'* would be labelled as shown.

amount	B-LEG
of	I-LEG
compensation	I-LEG
that	O
could	O
be	O
awarded	O
to	O
the	O
state	B-LEG

Consequently, we have formulated the catchphrase extraction problem as a classification problem of predicting the correct tag in the text sequence.

[3] github.com/dnmilne/wikipediaminer

The models that we explore in this section are (1) Spacy, (2) Conditional Random Fields (CRF), (3) Bi-directional LSTM with CRF and (4) Bi-directional GRU with CRF.

4.1 Spacy

Spacy[4] is an open-source toolbox meant for commercial NLP applications such as POS-Tagging, dependency parsing and the like. For entity recognition, it uses brown cluster features as well as case normalization data that makes it domain-independent. It trains a neural model over integrated word vectors and is quite fast in the training process.[1]

4.2 Conditional Random Field Models (CRF)

We employ CRF of [8] to learn the conditional probabilities of the 'LEG' tags. We experiment with different features of the word w_i to infer its corresponding tag t_i.

- Case-folded word to model a bag of words approach. (F_1)
- POS (Part of Speech) tag of w_1 since previous work [12] has observed that catchphrases are primarily noun phrases (F_2).
- Legal importance score of w_1 as defined in [12] (F_3)
- GloVe embedding[14] of w_1 obtained by training on the entire legal corpus of 400 documents. (F_4)

We also include simple and conventional orthographic features such as the suffix of the word to infer the particular stem, the case of the word, whether the word is alphanumeric or not and the like. Finally, we consider each of the aforementioned features for the words which are situated within a window size of 3 from w_1 to encode the dependencies of the sequence labels.

Comb 1	Comb 2	Comb 3	Comb 4
F_1	F_1+F_2	F_1+F_2+F_3	F_1+F_2+F_3+F_4

Table 3: Combination of features

We experiment with different combinations of these features as shown in Table 3. For each of these combinations, we run the CRF model using the L-BFGS algorithm for 150 iterations. The coefficients for the L1 and L2 penalty were set to 0.1 and 0.01 respectively.

4.3 Bi-directional LSTM with CRF

The current state of the art methods [15] learn contextual word representations using bidirectional language models where the character and word representations are learnt from the internal states of the

[4] https://spacy.io

Case	Gold standard	CRF	Bi-LSTM	Bi-GRU
2002.INSC.493	[license] [excise duty] [collusion] [assistant collector]	[license] [excise duty] [collector]	[license] [excisable to duty] [assistant collector]	[license] [excise duty] [assistant collector]
2015.INSC.547	[compensation] [absorption] [discrimination] [employee] [retirement age]	[discriminatory] [compensation] [employment] [employee] [surrender]	[compensation] [absorption] [discriminatory] [employee] [interest]	[compensation] [refund] [commercial] [age of retirement]
2003.INSC.304	[civil service] [appointing authority] [employee] [discharge]	[government servant] [appointing authority] [employee] [detention] [superannuation age]	[civil service] [judicial officer] [appointing authority] [employee] [superannuation]	[civil service] [judicial service] [appointing authority] [employee] [detention] [superannuation]

Table 2: Sample of phrases extracted from the test set

deep network. We have deployed a modified version of LSTM-CRF network as described in [9]. We use a time distributed LSTM layer to learn the character embedding of dimension 25. These character embeddings are concatenated with word-embedding vector of dimension 100 and finally fed to the Bi-LSTM layer. A fully connected layer of dropout 0.5 is learnt on the BiLSTM layer to obtain 100 features which are trained through a CRF layer to give the final prediction of tag.

4.4 Bi-directional GRU with CRF

The BiGrU-CRF model is similar to the BiLSTM-CRF architecture as described in the previous section. The only difference is that the Bi-directional LSTM cell is replaced with a standard GRU - Gated Recurrent Unit cell [5]. Both the BiLSTM-CRF and BiGRU-CRF model are trained for 25 epochs with a batch size of 32 using Adam optimizer having learning rate of 0.001, $\beta_1 = 0.9$ and $\beta_2 = 0.99$.

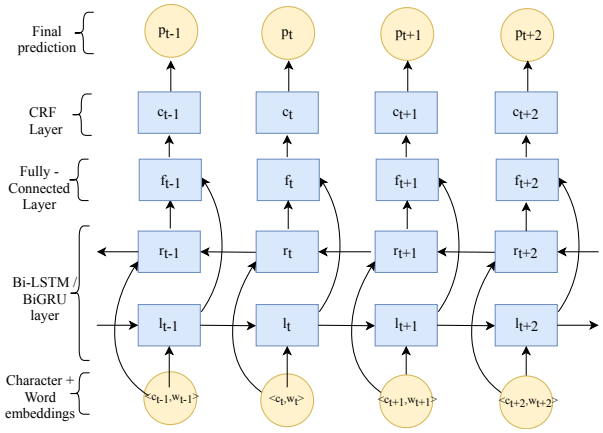

Figure 1: Architecture for BiLSTM-CRF and BiGRU-CRF

5 DATASET

To evaluate the efficacy of the aforementioned techniques, we use a dataset of 400 court case statements from the Supreme Court of India.[5] The gold standard catchphrases for each of the 400 documents have been manually identified by the legal experts at Manupatra legal search system.[6]

[5] http://liiofindia.org/in/cases/cen/INSC/
[6] http://www.manupatra.in/

6 EVALUATION OF THE METHODOLOGIES

6.1 Quantitative Evaluation

We employ two types of strategies for evaluating the catchphrase extraction algorithms - (i) a set based approach (ii) a summary based approach. The set based approach computes the **Precision, Recall** and **F-Score** of the extracted set of catchphrases and the gold-standard. The summary based approach computes the Rouge-L scores of the catchphrases from each method against the gold standard (as shown in [20]), treating both of them as summaries. The results for these methods are shown in **Table 4** . We perform 10 fold cross validation to evaluate the performance of the supervised models.

	Exact Match			Rouge-L		
Method	Prec	Rec	F1	Prec	Rec	F1
KEA	6.274	6.117	6.172	11.611	13.876	11.761
MAUI	11.760	11.605	11.659	15.051	19.239	15.645
Spacy	34.499	5.967	8.832	8.598	41.128	8.998
CRF	33.158	30.017	26.580	30.732	33.243	24.654
BiLSTM-CRF	32.991	21.449	22.179	21.597	31.383	18.732
BiGRU-CRF	31.119	24.358	23.310	24.805	29.453	20.235

Table 4: Results for the quantitative evaluation of supervised methods

An interesting observation was that the CRF models did not show any marked improvement in performance while using different feature combinations. In fact, the inclusion of PS-Legal scores and Glove vectors decreased the score slightly since they were unable to provide any additional information beyond the bag of words. Another interesting observation was that the neural architectures performed significantly poorer than all the CRF models. However after manually inspecting the tags returned by the models, we discovered that they seemed quite relevant. For example, in Table 2 the ground truth and CRF method extracted the phrase *retirement age* as a catchphrase, while the BiGRU extracted *age of retirement* as the catchphrase for the same case. This prompted us to undertake a qualitative evaluation described in the next section.

6.2 Qualitative Evaluation

Much like summarization, catchphrase extraction is a subjective task. Given the crowdsourced nature Manupatra, from where we obtained the gold standard tags, it is difficult to construe exactly what constitutes a catch phrase, since the standards may differ from one expert to another. Thus, by strictly comparing the tags extracted by our dataset to the gold standard, we suspected that we were losing

out on some catchphrases which may be subjectively useful to a legal expert. We, therefore decided to do a qualitative analysis of the tags extracted by each of our methods along with the gold standard tags, to assess if there is any merit in the tags that our methods generated.

We took 15 random documents and for each legal document, we compiled a superset of catchphrases comprising the ones extracted by BiLSTM, BiGRU, CRF as well as the gold standard catchphrases. We provided this extensive set of catchphrases to 3 legal practitioners with the task of identifying the relevant ones. The annotators did not have any prior knowledge about which catchphrase originated from which method, to ensure a true unbiased evaluation. We then computed the consistency scores [18] of the relevant catch-phrases for the different methods. The consistency score for two sets, X and Y is denoted by

$$CS = \frac{2|X \cap Y|}{|X| + |Y|} \quad (1)$$

Method	Annotator 1	Annotator 2	Annotator 3
CRF	0.502	0.426	0.428
BiLSTM-CRF	0.521	0.502	0.499
BiGRU-CRF	0.528	0.503	0.508
Gold-standard	0.514	0.443	0.427

Table 5: Qualitative evaluation done on 15 random test documents. The consistency scores have been reported in this case

We observe that the supervised methods were given a higher score than the gold standard by the legal experts. All three experts independently agreed that the tags produced by the BiGRU-CRF method were semantically more relevant to the case, according to them, than all the other methods (including the gold standard). This confirms our suspicion about the challenges of learning from crowdsourced annotations, especially for such a subjective task.

A look at some examples of catch phrases that were marked relevant by annotators but were missing in the original ground truth reinforces this.

Case id	Gold standard	Annotator (additional)
2016.INSC.306	Exhortation, Rigorous Imprisonment	Compensation, Complainant
2016.INSC.725	Coastal Zone, Policy Document,	Sale deed, documentary evidence

Table 6: Manupatra and additional annotator catchphrases

We would like to safely conclude that our neural catchphrase extraction methods are at least human comparable, and may even be subjectively better than the tags provided by Manupatra.

7 CONCLUSION AND FUTURE WORK

In this work, the task of legal catchphrase extraction is modelled as a sequence labelling task, in a supervised setup. It is observed that, existing named entity recognizers (NERs) adapt seamlessly to legal catchphrase extraction task to the extent they outperform some existing catchphrase extractors. So, we infer that these NERs can be used for future tasks involving legal catchphrase extraction.

In our qualitative analysis of these named entity recognizers, we also find that they provide human-comparable results for the task of catchphrase extraction. In other words, these NERs were able to identify catchphrases which were *missed* by the Manupatra annotators. This indicates that the NERs used in this paper can effectively identify legal catchphrases for cases where catchphrases are absent and so is legal expertise. Therefore, we look forward to using these methods for future cases for automatic catchphrase esp. in situations where legal expertise is not available.

In the future, we also intend to expand the ground-truth set of catchphrases for all the documents and release them to the research community.

REFERENCES

[1] F. N. A. Al Omran and C. Treude. 2017. Choosing an NLP Library for Analyzing Software Documentation: A Systematic Literature Review and a Series of Experiments. In *Proc. 2017 IEEE/ACM MSR*.

[2] Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*. 84–90.

[3] Stefanie Brüninghaus and Kevin D. Ashley. [n. d.]. Improving the Representation of Legal Case Texts with Information Extraction Methods. In *Proc ICAIL 2001*. 42–51.

[4] Jianpeng Cheng and Mirella Lapata. [n. d.]. Neural Summarization by Extracting Sentences and Words. In *Proc. ACL 2016*. 484–494.

[5] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *CoRR* abs/1406.1078 (2014). arXiv:1406.1078 http://arxiv.org/abs/1406.1078

[6] Ahmed Elnaggar, Christoph Gebendorfer, Ingo Glaser, and Florian Matthes. 2018. Multi-Task Deep Learning for Legal Document Translation, Summarization and Multi-Label Classification. *CoRR* abs/1810.07513 (2018). arXiv:1810.07513 http://arxiv.org/abs/1810.07513

[7] Filippo Galgani et al. 2012. Towards Automatic Generation of Catchphrases for Legal Case Reports. Springer-Verlag, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-28601-8_35

[8] John D. Lafferty et al. [n. d.]. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proc. ICML 2001, pages = 282–289,*.

[9] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 260–270. https://doi.org/10.18653/v1/N16-1030

[10] Daniel Locke and Guido Zuccon. [n. d.]. A Test Collection for Evaluating Legal Case Law Search. In *Proc. SIGIR 2018*. 1261–1264.

[11] Debanjan Mahata et al. 2018. Key2Vec: Automatic Ranked Keyphrase Extraction from Scientific Articles using Phrase Embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the ACL: Human Language Technologies, Volume 2 (Short Papers)*. Association for Computational Linguistics, 634–639. https://doi.org/10.18653/v1/N18-2100

[12] Arpan Mandal et al. [n. d.]. Automatic Catchphrase Identification from Legal Court Case Documents. In *Proc. CIKM 2017*. 2187–2190.

[13] Olena Medelyan. 2009. Human-competitive automatic topic indexing. http://cds.cern.ch/record/1198029 Presented on July 2009.

[14] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation.

[15] Matthew Peters et al. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the ACL: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, 2227–2237. https://doi.org/10.18653/v1/N18-1202

[16] Seth Polsley, Pooja Jhunjhunwala, and Ruihong Huang. 2016. CaseSummarizer: A System for Automated Summarization of Legal Texts. In *Proc. of COLING 2016*. 258–262.

[17] Adam Roegiest, Alexander K. Hudek, and Anne McNulty. 2018. A Dataset and an Examination of Identifying Passages for Due Diligence. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*. 465–474. https://doi.org/10.1145/3209978.3210015

[18] L. Rolling. 1981. Indexing consistency, quality and efficiency. , 7 pages. https://doi.org/10.1016/0306-4573(81)90028-5

[19] Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. *Automatic Keyword Extraction from Individual Documents*. John Wiley & Sons, Ltd, Chapter 1, 1–20. https://doi.org/10.1002/9780470689646.ch1

[20] Vu D. Tran, Minh Le Nguyen, and Ken Satoh. 2018. Automatic Catchphrase Extraction from Legal Case Documents via Scoring using Deep Neural Networks. *CoRR* abs/1809.05219 (2018). arXiv:1809.05219 http://arxiv.org/abs/1809.05219

[21] Pengfei Wang, Ze Yang, Shuzi Niu, Yongfeng Zhang, Lei Zhang, and ShaoZhang Niu. [n. d.]. Modeling Dynamic Pairwise Attention for Crime Classification over Legal Articles. In *Proc. SIGIR 2018*. 485–494.

[22] Ian H. Witten et al. 1999. KEA: Practical Automatic Keyphrase Extraction. In *Proc. ACM Conference on Digital Libraries*.