A REPORT

ON

CONTRACT ELEMENT EXTRACTION USING DEEP LEARNING

BY

Name of the student
Sharat Patil

ID Number
2016A7PS000075G

Prepared in partial fulfillment of the
Practice School-I Course No.
BITS F221/ BITS F231/ BITS F241
AT
Bank of Maharashtra, Pune

A Practice School-I station of
BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
(June 2018)

A REPORT

ON

CONTRACT ELEMENT EXTRACTION USING DEEP LEARNING

BY

Name of the student	**ID Number**	**Discipline**
Sharat Patil	2016A7PS000075G	B.E.(Hons) Computer Science

Prepared in partial fulfillment of the
Practice School-I Course No.
BITS F221/ BITS F231/ BITS F241
AT
Bank of Maharashtra, Pune

A Practice School-I station of
BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
(June 2018)

ACKNOWLEDGEMENTS

I would like to extend our heartfelt gratitude to all those responsible for providing me with the opportunity of a lifetime to work in one of the most premier Banking Institutions , Bank of Maharastra. I would like to take this opportunity to thank the several entities and personalities responsible for providing us with such a platform.

I would like to thank Mr. Prasant Samantray, faculty in-charge of our of PS station, for helping me throughout my time there

I would like to thank Mr. Sudhanshu Saxena, Chief Manager(Zonal Office),Bank of Maharastra, for taking time out of his busy schedule to guide me.

I would like to thank the PS Division of BITS Pilani University for taking such a brilliant and necessary initiative.

I would like to thank everyone else who was directly or indirectly involved to provide us this opportunity.

ABSTRACT

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI (RAJASTHAN)
Practice School Division

Station: Bank of Maharashtra **Centre:**Pune

Duration: May 22, 2018 to July 14, 2018 **Date of Start**: May 22, 2018

Date of Submission: June 24, 2018

Title of the Project: Contract elements extraction using Deep Learning.

ID No. /Name(s)/Discipline(s) of the student(s):
SHARAT PATIL 2016A7PS0075G Computer Science

Name(s) and designation(s) of the expert(s):

Mr. Sudhanshu Saxena, Chief Manager(Zonal office),Bank of Maharastra.

Name(s) of the PS Faculty: Mr. Prassant Samantray

Key words: Contracts, Natural Language Processing, Deep Learning

Project Areas: Natural Language Processing

Abstract: Contracts are legal texts describing agreements. Law firms, companies, government agencies ,banks etc. need to monitor contracts for a wide range of tasks that are automated using specific information from the contracts.But extracting contract elements is a mechanical job that requires a lot of time. In this project I study how contract element extraction can be automated using deep learning. The main aim of the project is to implement a deep learning architecture to facillitate the automation of contract element extraction using dataset provided by Natural Language Processing Group Department of Informatics - Athens University of Economics and Business . The datasets is in an encoded form to bypass privacy issues.

TABLE OF CONTENTS

INTRODUCTION

Contracts are legal texts describing agreements.Banks need to monitor contracts for various tasks.Many of these tasks can be automated by extracting particular contract elements (e.g., termination dates, legislation references, contractingparties, agreed payments). Contract element extraction, however, is currently performed mostly manually, which is tedious and costly.

Automating contract extractions is a hard task as it is not as straightforward as as being able to recognise the legal provisions - it is about both identification and localisation that is to identify and classify legal terms. For a human these two seem the same; however, for a machine, detecting which of the many different combinations of words best fits a provision type is much more difficult than answering a simple yes or no question.

Another problem is that the programs main function is to perform these extractions on data it has never seen before. The machine is expected to automatically deal with novelties and uncertainties that are associated with legal language. These predictive properties may happen naturally and unconsciously in the brain of an expert lawyer, but are far trickier to attain in a logical computer. In fact the legal language itself is continually evolving which then requires online learning to keep up.

NATURAL LANGUAGE PROCESSING

Natural Language Processing

Def:-The field of study that focuses on the interactions between human language and computers is called Natural Language Processing, or NLP for short. It sits at the intersection of computer science, artificial intelligence, and computational linguistics (Wikipedia).

NLP is a way for computers to analyze, understand, and derive meaning from human language in a smart and useful way. By utilizing NLP, developers can organize and structure knowledge to perform tasks such as automatic summarization, translation, named entity recognition, relationship extraction, sentiment analysis, speech recognition, and topic segmentation.

NLP is characterized as a hard problem in computer science. Human language is rarely precise, or plainly spoken. To understand human language is to understand not only the words, but the concepts and how they’re linked together to create meaning. Despite language being one of the easiest things for humans to learn, the ambiguity of language is what makes natural language processing a difficult problem.

Contract extraction is an example of a problem dealing with Natural Language Processing (generally defined as "understanding spoken and written language"), which is a fast-paced field at the centre of a multidisciplinary crossroad of computation and linguistics, and relies heavily on Machine Learning.

Machine Learning provides very algorithms to deal with these challenges. Their inner workings aren't manually programed, but are automatically infered from the data. Over time, more data does not increase the complexity but generally improves the accuracy, which makes these methods flexible and powerful. These algorithms also make use of the underlying uncertainties that are inherent to predictive extractions, and are an exceptional match for the dynamic and elaborate task at the heart of contract extraction.

For analyzing the complex structures of legal documents, one class of algorithms which has given great results is a subsection of Machine Learning called Deep Learning.

DEEP LEARNING

Deep learning (also known as deep structured learning or hierarchical learning) is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms. Learning can be supervised, semi-supervised or unsupervised. (wikipedia).

Apart from being particularly trendy, Deep Learning is a “disruptive” branch of Machine Learning which harnesses the analytical power of vast networks of algorithms. This technology was born over 40 years ago when computer scientists tried to imitate the brain’s neural structure. Despite being promising, these computations were too slow and costly to be effective. The tremendous advances in computer hardware,especially GPUs,over the last few years have brought neural networks back in the limelight, as deep learning systems are now capable of crunching large amounts of data and handling very sophisticated tasks. This explains both why it is at the forefront of natural language processing, and why it is a perfect candidate for our contractual extractions. The deep architectures allows the systems to really grasp the meaning of sentences rather than recognise sequences of words.One example a highly successful use case is JPMorgan Chase. It has developed a proprietary ML algorithm called Contract Intelligence or COiN. It is now used to analyze the documentation and extract the important information from it.

CONTRACT ELEMENTS

These are the specific parts of the contract that we will extract.

Contract Elements:-

- Contract Title

  The title briefly states the overall purpose of the document.

- Contracting Parties

  Provides information about the parties involved in the agreement.

- Start Date

  Provides information about the date of signing.

- Effective Date

  Provides information about the date from when the contract becomes legally binding or comes into effect.(may be different from the start date.)

- Termination Date

  Provides information about the date till which the contract is effective.

- Contract Period

  Provides information about the time period in which the contract is in effect.

- Contract Value

  Provides information about the price of the agreed transaction.

- Governing Law

  Provides information on which laws applicable to the contract.

- Jurisdiction

Provides information on the court which has jurisdictional power to settle disputes related to the contract.

- Legislation References

  Provides information on the laws that pertain to the contract.

- Clause headings

  Provides information regarding the different clauses present in the contract.

Extraction zones

Extraction zones are area where specific information has a high probablity of being found. For eg the contract title would be found on the cover page. A specific element may multiple possible extraction zones but by restricting it those where it has a higher probablity of being found we increase the accuracy and also lower computation costs.

Extraction Zones (at testing)	Example Clause Heading Words	Contract Elements Typically Included
Cover page and preamble	-	Contract Title, Contracting Parties, Start Date, Effective Date
Term clause	'Term', 'Period', 'Term of Agreement'	Termination Date, Contract Period
Termination clause	'Termination', 'Termination of Agreement'	Termination Date
Governing Law clause	'Governing Law', 'Applicable Law'	Governing Law, Jurisdiction
Jurisdiction clause	'Jurisdiction', 'Jury Trial', 'Venue'	Jurisdiction
Miscellaneous clause	'Miscellaneous', 'Entire Agreement'	Governing Law, Jurisdiction
Contract Value clause	'Lump Sum', 'Salary'	Contract Value
In the text after the recitals, zones starting up to 20 tokens before and ending up to 20 tokens after each line break, not crossing other line breaks		Clause Headings
In the entire contract, zones starting up to 20 tokens before and ending up to 20 tokens after each occurrence of words like 'Act', 'Treaty' etc.		Legislation References

Table 1: Extraction zones where contract elements of different types are searched during testing.

These extraction zones are predetemined.

DATASET

It is said that the most important part of a machine learning projeect is getting the data and getting cleaning it. Due sensitive nature of the type of data required it becomes hard to obtain it. Therefore the only available dataset for such a purpose is a relatively small dataset that has been encoded so that it can be made available for general use.

The datset that is used in this project has been provided by the Natural Language processing group, Department of Informatics, Athens University of Economics and Buisness.

From their paper :-Ilias Chalkidis, Ion Androutsopoulos and Achilleas Michos. 2017. “Extracting Contract Elements.”

> *“The labeled benchmark dataset contains 993 contracts (893 training,*
> *100 test) annotated with gold (correct) clause headings , and 2461 contracts (2,111 training, 350 test) with gold annotations*
> *for the other 10 types of contract elements.*
> *The gold contract element annotations of the labeled dataset were provided by 10 law students. Each contract was annotated by one student.*
>
> *All the contracts of the labeled dataset are in English. We cannot reveal their actual texts, due to privacy and IPR issues, but we provide them in an encoded form, where each vocabulary word*

has been replaced by a unique integer, as already discussed. We also provide hand-crafted features, word embeddings, and POS tag embeddings per token, further discussed below. “

The dataset causes some difficulty as we can only use the word that they tokenised when they obsfucated the datset.This limits the extent to which we can experiment with the dataset. The encoding also prevents us to really see the extraction as we dont know what each token represents. This also prevents us from testing the model on a contract of our choice which renders it to be a purely academic model without much direct practical usage. But they have also provided with embeddings, part-of-speech tag and word length for each token which is extremely usefull.

The Dataset Has:

2000 training contracts

350 test contracts

200 dimensional embeddings

25 dimensional POS tags

Element classes:-

|CONTRACT ELEMENT| |CATEGORY ENCODING| |CLASS|

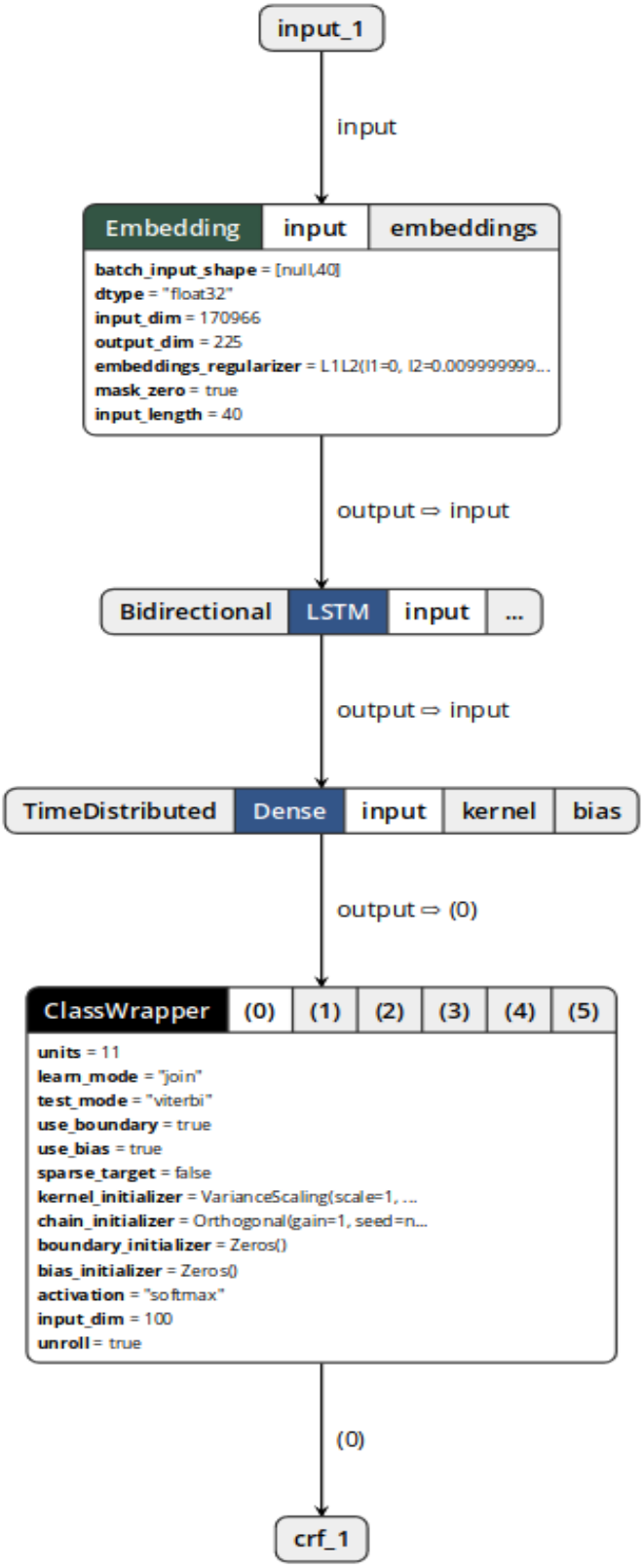
None	β> 0	0
Contract Title	β> TIT	1
Contract Party	β> CNP	2
Start Date	β> STD	3
Effective Date	β> EFD	4
Termination Date	β> TED	5
Contract Period	β> PER	6
Contract Value	β> VAL	7
Governing Law	β> GOV	8
Jurisdiction	β> JUR	9
Legislation Refs.	β> LEG	10

(CLASS is the value predicted by the model)

Sample Text:-

TOKEN_1490[TIT] TOKEN_6[TIT]
TOKEN_15[0] TOKEN_6[0] TOKEN_2384[0] TOKEN_263[0]
TOKEN_28816[STD] TOKEN_28[STD]
TOKEN_25[STD] TOKEN_4376[STD] TOKEN_19[STD]
TOKEN_1530[STD] TOKEN_31[0] TOKEN_78167[CNP]
TOKEN_5565[CNP] TOKEN_1539[CNP] TOKEN_19[0]

MODEL


input_1
input
Embedding
input
embeddings
batch_input_shape = [null,40]
dtype = "float32"
input_dim = 170966
output_dim = 225
embeddings_regularizer = L1L2(l1=0, l2=0.009999999...
mask_zero = true
input_length = 40
output ⇨ input
Bidirectional
LSTM
input
...
output ⇨ input
TimeDistributed
Dense
input
kernel
bias
output ⇨ (0)
ClassWrapper
(0)
(1)
(2)
(3)
(4)
(5)
units = 11
learn_mode = "join"
test_mode = "viterbi"
use_boundary = true
use_bias = true
sparse_target = false
kernel_initializer = VarianceScaling(scale=1, ...
chain_initializer = Orthogonal(gain=1, seed=n...
boundary_initializer = Zeros()
bias_initializer = Zeros()
activation = "softmax"
input_dim = 100
unroll = true
(0)
crf_1

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 69593)             0
_________________________________________________________________
embedding_1 (Embedding)      (None, 69593, 225)        38467350
_________________________________________________________________
bidirectional_1 (Bidirection (None, 69593, 100)        110400
_________________________________________________________________
time_distributed_1 (TimeDist (None, 69593, 50)         5050
_________________________________________________________________
crf_1 (ClassWrapper)         (None, 69593, 12)         780
=================================================================
Total params: 38,583,580
Trainable params: 38,583,580
Non-trainable params: 0
```

The Model is made up of four layers(excluding input):-

1. Embedding Layer
2. Bi-LSTM
3. Dense
4. CRF

As compared to other models that produce state of the art results this is a relatively shallow model.The large number of parameter is due to the size of input and the embedding dimension.

Each of the individual layer will be explained further.

The model was implemented in keras using its fuctional model API.

INPUT

The input is passed in a certain format

Document:
TOKEN_1490[TIT] TOKEN_6[TIT]
TOKEN_15[0] TOKEN_6[0] TOKEN_2384[0] TOKEN_263[0]
TOKEN_28816[STD] TOKEN_28[STD]
TOKEN_25[STD] TOKEN_4376[STD] TOKEN_19[STD]
TOKEN_1530[STD] TOKEN_31[0] TOKEN_78167[CNP]
TOKEN_5565[CNP] TOKEN_1539[CNP] TOKEN_19[0]

X(input):

[1490,6,15,6,2384,263,28816,28,25,4376,19,1530,31,
78167,5565,1539,19]

Y(True output):
[1,1,0,0,0,0,3,3,3,3,3,3,0,2,2,2,0]

(Note the categories were replaced with there class values
0=0
TIT=1
STD=2
CNP=3)

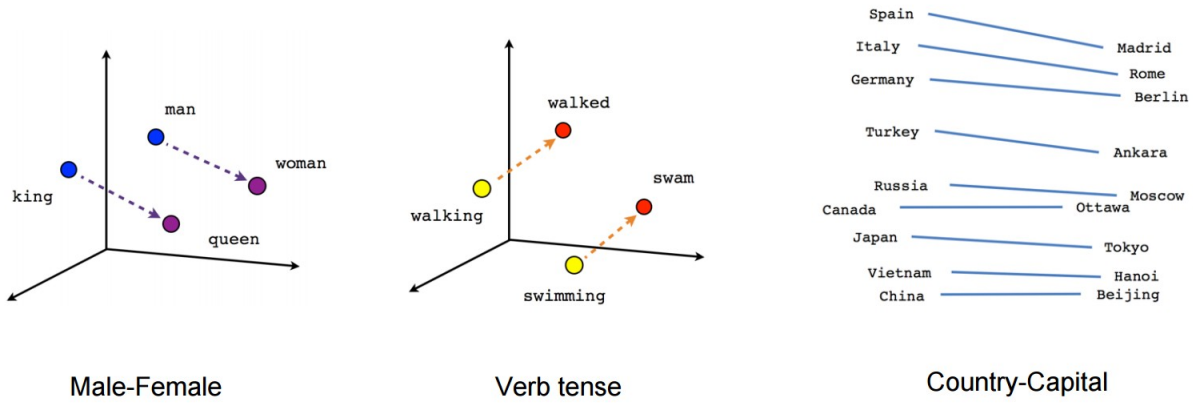
The length of each contract was resticted to 69593 to facilitate

faster training. Post padding with zeros was used.

WORD EMBEDDINGS

Word embedding is the collective name for a set of language modeling and feature learning techniques in natural language processing (NLP) where words or phrases from the vocabulary are mapped to vectors of real numbers. Conceptually it involves a mathematical embedding from a space with one dimension per word to a continuous vector space with a much lower dimension.(Wikipedia)

So word embeddings are learnt vector representations of words.



Male-Female Verb tense Country-Capital

Above is an example of learnt word vectors. Words with simila relation have same vector distance between them. As shown in the example king is to quuen as man is to woman so the distances between the pair wise relation is same in both cases.
Simillarly the verb tenses and the cities and capitals.

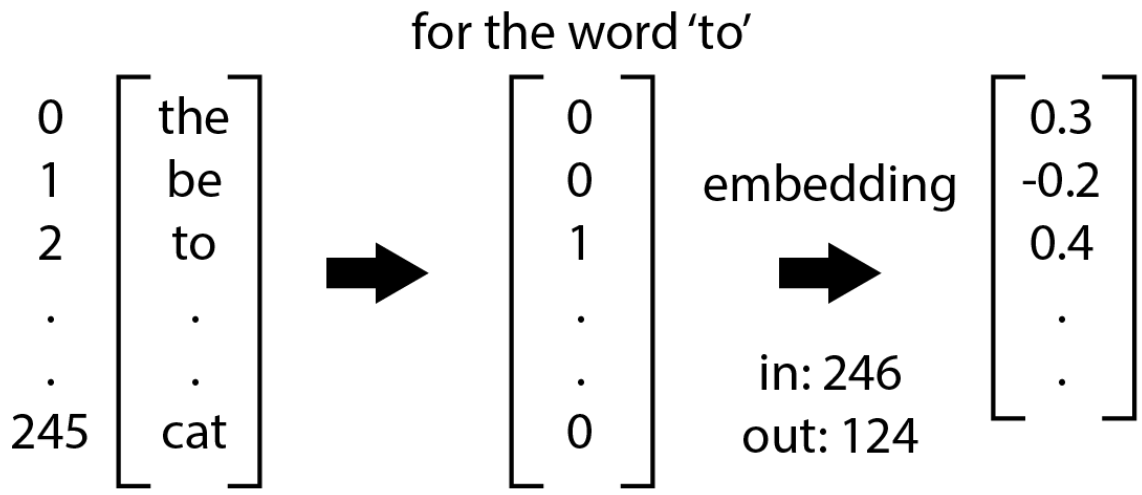
Depending on the task the word embedding for a certain word varies.For us the word ‘act’ has legal meaning but for other use cases it becomes a verb.

EMBEDDING LAYER

An embedding layer works as a mapping

$$f: X \rightarrow y$$

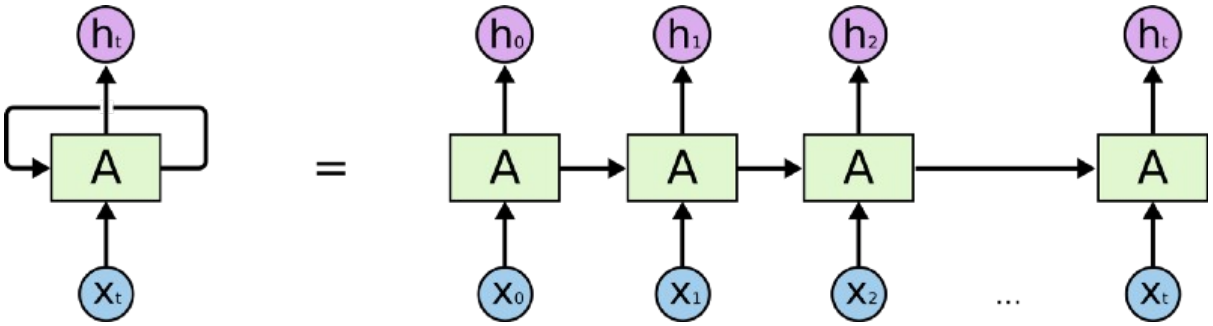
where X is our vocabulary and y is 225 dimensional embeddings. So when the embedding layer recieves a ‘word’(in our case a number representing a word) it maps it to its specific embedding and passes the embedding to the next layer.



In the example above the embedding layer recieves an input for the word to in the form of an one hot encoding of length 246(vocabulary size) and outputs a 124 dimensional vector.

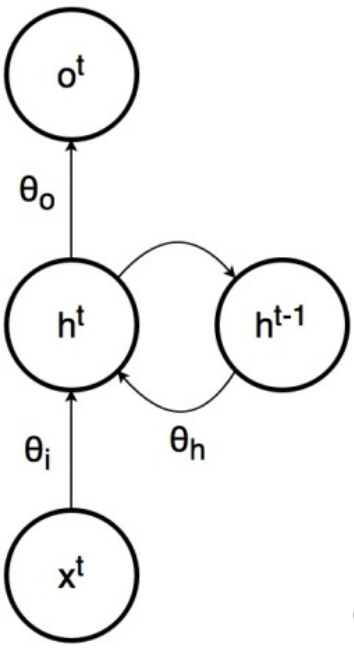
RNN’s

Recurrent Neural Networks or RNNs are a special type of neural nets architectures in which the information cycles through it.



On the left side is the RNN shown with cyclical input. It is unrooled on the right side- (X_0,X_1,X_2...) is asequential input and as you can see the information from processing X_0 is passed on to X_1.

RNNs are extremely usefull in processing sequence data as they are able to link between two inputs.When it makes a decision, it takes into consideration the current input and also what it has learned from the inputs it received previously.



Graphical representatio

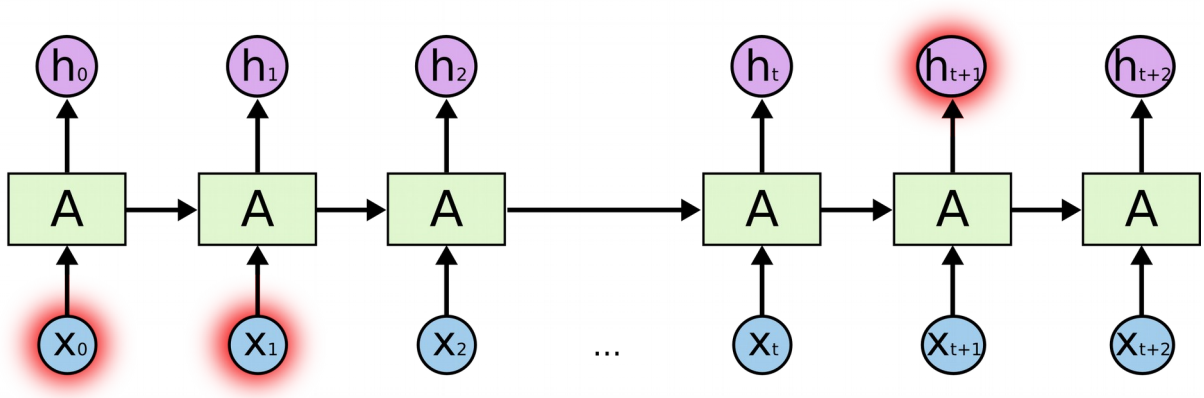
Mathematically :

$$\mathbf{o^t} = f(\mathbf{h^t}, \theta)$$

$$\mathbf{h^t} = g(\mathbf{h^{t-1}}, \mathbf{x^t}, \theta)$$

LONG-TERM DEPENDENCIES:

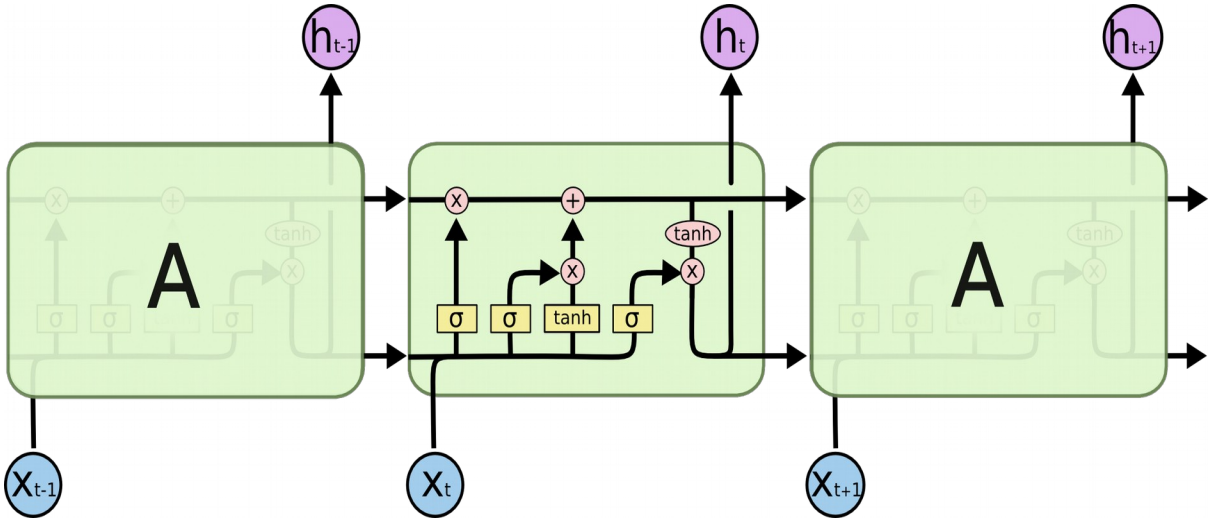
RNNs have difficulty in linking inputs with large gap in them in the sequence.It’s entirely possible for the gap between the relevant information and the point where it is needed to become very large. Unfortunately, as that gap grows, RNNs become unable to learn to connect the information.



The RNN will be able to link X_0 and X_1, but as the distance grows it will fail to do so and therefore h_{t+1} would not be produced taken into consideration some relevant information that the RNN failed to link up.

BI-DIRECTIONAL LONG SHORT TERM MEMORY

BI-DIRECTIONAL LONG SHORT TERM MEMORY or BiLSTMs are a special type of RNNs that weere made to take care of the long-term dependency problem.



The key to LSTMs is the cell state, the horizontal line running through the top of the diagram.The cell state is kind of like a conveyor belt. It runs straight down the entire chain, with only some minor linear interactions. It’s very easy for information to just flow along it unchanged.

The LSTM does have the ability to remove or add information to the cellstate, carefully regulated by structures called gatesGates are a way to optionally let information through. They are composed out of a sigmoid neural net layer and a pointwise multiplication operation.

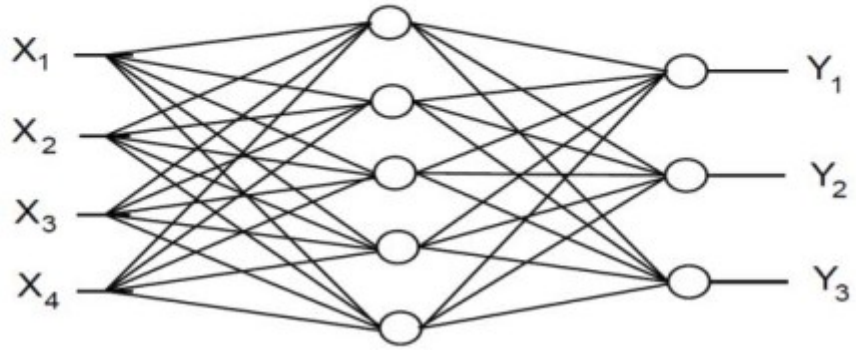
BiDirectional means that the sequence fed to LSTM twice once in th e froward direction and once backwards and the ouputs are conctenated and sent to the next layer.

DENSE

This is a fully connected layer applied in conjunction with the TimeDistributed wrapper.

This wrapper applies a layer to every temporal slice of an input I.e every word in sequence has this layer applied to it.

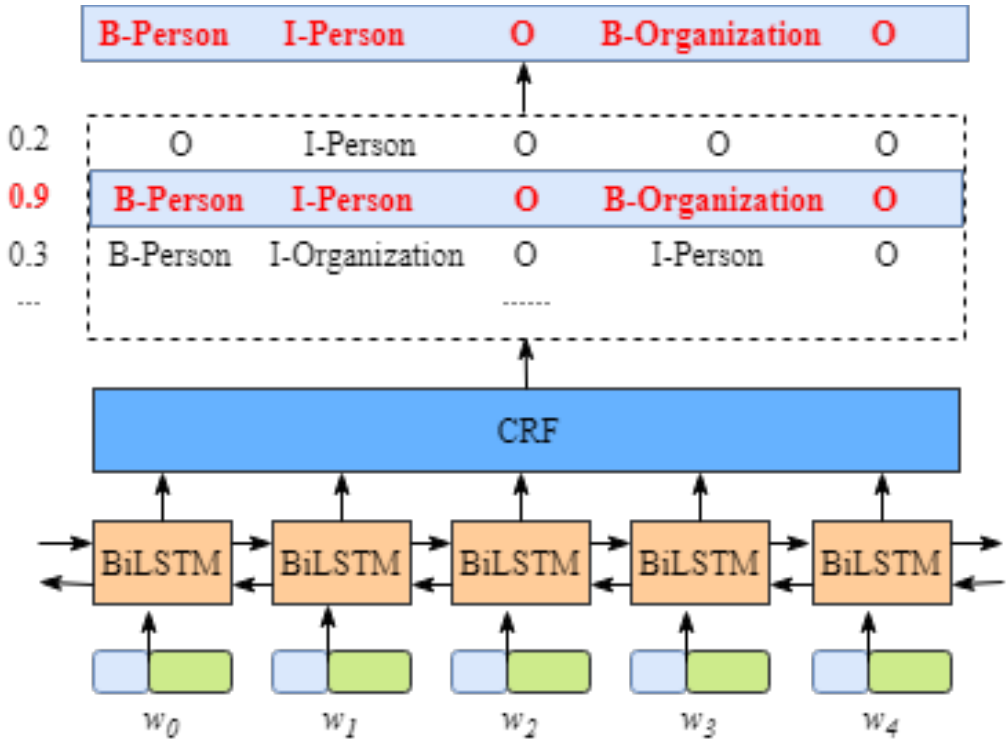
This one of the simplest neural network architecture.



Here X_1,X_2 are values received from the Bilstm for a single word in the sequence.

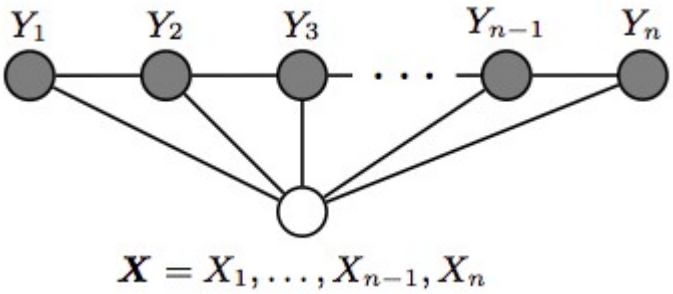
CONDITIONAL RANDOM FIELD

Conditional random fields (CRFs) are a probabilistic framework for labeling and segmenting structured data, such as sequences.The underlying idea is that of defining a conditional probability distribution over label sequences given a particular observation sequence, rather than a joint distribution over both label and observation sequences.



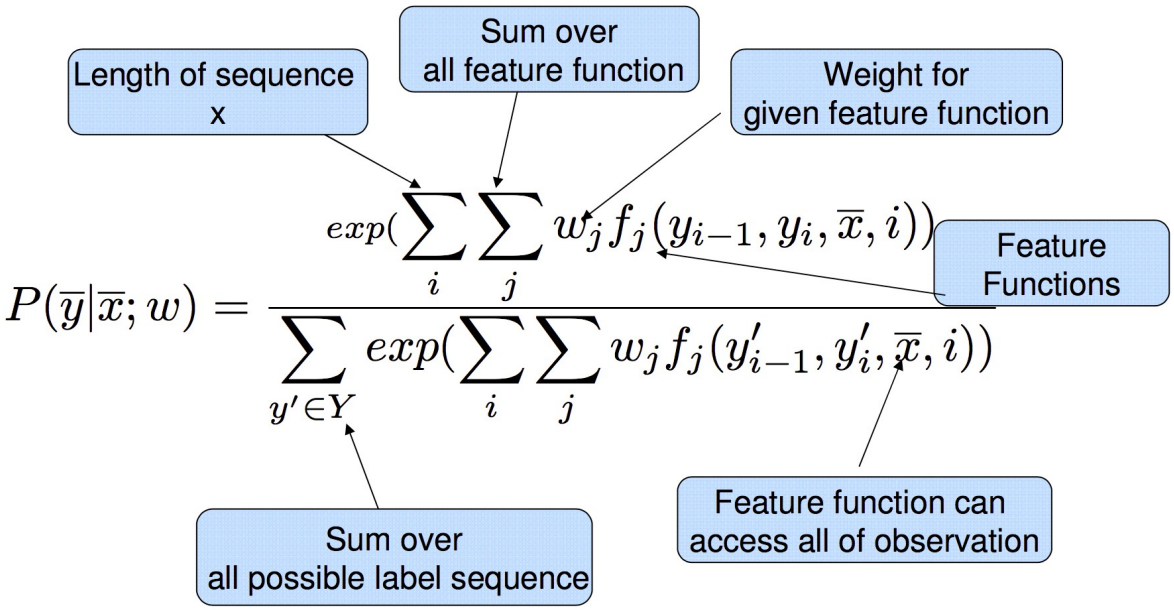
The advantage of CRFs is that they take into consideration the entire sequence I.e they learn constraints between the labels of the words for e.g the crfs are capapble of learning that between two words that are contract title it is not possible to have the start date.

So the CRF takes the enire output from the dense layer for the entire sequence and based on the features it outputs the most likely seuence of tag.



The CRF can be represted as a graph where the lines between the nodes show the dependency between the variables.Here you can see that the output labels are all inter dependent on each other.The CRF is capable of learning these dependencies and uses them to predict the most likely sequence.

Mathematically the CRF maximises the probability:



and the output is:

y*=argmax(y¯) P(y¯|x¯;w

TRAINING

The model was trained for one epoch over the 2000 training contracts.

Training time was 31 hours on a i7 cpu with multithreading on.

The optimizer used was adam optimizer.

METRICS

$$\text{Precision} = \frac{tp}{tp + fp}$$

$$\text{Recall} = \frac{tp}{tp + fn}$$

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

TESTING

The model was evaluated on 350 test contracts.

RESULTS

My-Model

	Id	f1	p	r	s
0	0	0.999855	0.999817	0.999894	24337213
1	TIT	0.901983	0.873234	0.932689	2585
2	CNP	0.831151	0.842347	0.820249	5146
3	STD	0.724796	0.748392	0.702642	1325
4	EFD	0.120983	0.761905	0.065708	487
5	TED	0.234957	0.650794	0.143357	286
6	PER	0.425926	0.932432	0.276000	250
7	VAL	0.122667	0.696970	0.067251	342
8	GOV	0.877108	0.819484	0.943450	2122
9	JUR	0.719714	0.817497	0.642825	2195
10	LEG	0.830986	0.876064	0.790320	5599

Research Paper

BILSTM-CRF		
P	R	F1
0.96	**0.95**	**0.95**
0.98	0.92	**0.95**
0.92	**0.98**	**0.95**
0.95	0.89	0.92
0.65	0.93	0.77
0.55	0.85	0.65
0.72	0.60	0.66
0.99	0.97	**0.98**
0.90	0.88	0.88
0.82	0.94	0.87

The results are as expected even though the research paper outperforms the model but, for one epoch the model has achieved very good accuracy.

(Note: differences between the papers model and mine is that tin the paper they have created a different model for each element, but I have used only one. This actually gave some advantage that the CRF learns much more efficiently.

The size of the model also is smaller)

CONCLUSION

On June 8th 2018 NITI AAYOG released a reprt entitled “Disscussion Paper on National Strategy on AI” which gave reccomendations on research and adopting new AI based technology in various fields. As even the government has started to give focus to the field I decided to work on a project related to it.

This project was undertaken by me to learn more about the field of NLP while working on something that has real world application especially in the field of Banking. In the project I created a small model that showcases the techniques that can be used to tag various parts of a contract. The model though being a very simple implementation in terms of size ,accuracy and scalablity due to computational limits and the size and nature of the dataset. But still it can act as proof of concept for the application of NLP in contract analysis and sequence tagging.

REFRENCES

- Ilias Chalkidis, Ion Androutsopoulos, and Achilleas Michos. 2017.Extracting Contract Elements. In Proceedings of International Conference on Artificial Intelligence and Law, London, UK, June 12–15, 2017 (ICAIL'17), 10 pages. DOI:http://dx.doi.org/10.1145/3086512.3086515
- https://en.wikipedia.org/wiki/Deep_learning
- https://en.wikipedia.org/wiki/Natural_language_processing
- http://semi.org/en/standard-contract-elements
- https://blog.algorithmia.com/introduction-natural-language-processing-nlp/
- https://www.seal-software.com/blog/solving-contract-analytics-using-machine-learning
- http://nlp.cs.aueb.gr/software_and_datasets/CONTRACTS_ICAIL2017/index.html
- http://colah.github.io/
- http://www.davidsbatista.net/blog/2017/11/13/Conditional_Random_Fields/
- https://www.depends-on-the-definition.com/sequence-tagging-lstm-crf/