Named Entity Extraction From Disaster Reports

By

Alice Irankunda (alice.irankunda@aims.ac.rw)
Supervised by Dr.Yabebal FANTAYE

June 2017

AN ESSAY PRESENTED TO AIMS RWANDA IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF MASTER OF SCIENCE IN MATHEMATICAL SCIENCES

Abstract

Reports are key source of information for all activities within organizations. Electronic reports are generated day to day in [Jan: an] unstructured way. It is still a big challenge to know automatically what the reports are talking about. For big organizations like International Federation of Red Cross(IFRC) [Jan: Space before parenthesis.] where they work [Jan: s/where they work/which work] in humanitarian domain, some information from their reports are quietly [Jan: Quietly? This must be a typo and I'm not sure what you mean here.] necessary and very important. Within million reports, automation of extracting information saves time and increase [Jan: s/increase/increases] quality. Needed information from the reports are called Named entities. [Jan: Rephrase: "In this thesis we are concerned with extracting specific pieces of information called named entities, such as <example(s)>" (or similar). Then mention packages you used as below. Do not capitalize "named entities".] This research, We [Jan: Do not capitalize mid-sentence.] used machine learning algorithms such as Stanford NER, polyglot and Natural language toolkit [Jan: Capitalize packages names as in their documentation.] to extract named entities from IFRC reports. We are looking for the answer of " Who did what when How" [Jan: No space after opening quote.] [Jan: Rephrase quote to makes sense, e.g., "Who did what, when and how?"] from the documents.

Contents

1. Introduction of the Research

[Jan: Change to "Introduction".]

In every organization there is a way to communicate. One of the most popular way to transmit the information is to produce a written report which explains how different activities of the organization are going. For large organizations there are huge number of reports and it is so [Jan: Delete "so".] challenging to go through each and every report manually. This research has an aim of providing an easy way of visualizing and extracting the important information locked in reports from NGO [Jan: s/NGO/NGOs] and large organisations.

In 1919, the International Federation of Red Cross and Red Crescent societies (IFRC) has been founded, it has some millions of reports related to humanitarian support, [Jan: Change comma to full stop.] How to know automatically the number of people who suffered from a disease? How to know the fraction of fund spent on shelter?

In this research, we tried to use a combination of statistics formulae and techniques of Natural Language Processing (NLP) to find the solution for the extracting entities, Big data and Machine learning for analysing the huge data by using statistical and computing algorithms. [Jan: This sentence is too long, split it into two thoughts.] Entity can be defined as an instance of existence of something, for example what is the activity done on what place when and how ? [Jan: No space before question mark.]

Document modelling by extracting entities is one of the way to deal with natural big data linguistic problems where entity can be considered as a single unit of data like location, people, organization and so on. Entities can be classified based on their relationship.

These are key procedures to be performed for extracting entities:

- The sentences which compose a report must be parsed.
- Entities must be identified in the report and classified.
- Relationship between entities must be modelled.

A report is composed by [Jan: s/by/of] paragraphs, each paragraph is made by sentences. Natural Language Processing techniques deal with sentences and content based analysis by splitting the sentences into tokens then remove the common words and work with corpus to get entities. [Jan: s/remove/removing, s/and work/in order to work. Consider splitting this sentence into two.] The meaning of a word can depend on its surroundings as well as it can be independent. [Jan: s/as well as it can be independent/but it can also be independent of it] For extracting significant entities, the context of a word is one of the points to be considered carefully.

2. Literature review

[Jan: Capitalize "review" in the chapter title.]

In today's life, many organizations are generating unstructured data while they are communicating. There are plenty of entities to be extracted. In this research, all reports we considered are written in English.

To label the boundaries of sentences is one of the important prerequisite steps in Natural Language Processing. The punctuation marks cause some ambiguity (Baluja et al., 2000) for example it is challenging to differentiate the point in abbreviations and a full stop. To handle this ambiguity some systems use the special purpose-regular expression grammar, exception rule method etc.

David Palmer and Marti A. Hearst worked on the problem of punctuation marks. (Palmer and Hearst, 1994). They developed an efficient system with high accuracy in automatic labelling the boundaries of the sentence by using the feed forwarding neural-networks where the input was the POS probabilities of all tokens which are surrounding the punctuation and output was found as the label to be assigned to the token.This work was able to correct up to 98.5% for punctuation of sentence-boundaries. A proposed new approach was how to represent the context of punctuation marks without ambiguities.

This research will also look at how neural networks can be used to label different tokens.

Capitalization can be used in different ways such as the beginning of the proper noun, the abbreviation, the post of high level profile people etc. Considering the English language text, if we are given a particular token it is not by chance to determine whether it is a name or not. Some of the approaches to indicate a name are to use capitalization, detection of sentence boundaries and dictionaries (Baluja et al., 2000).

[Jan: The text above should be general introduction to this chapter, but you write about very specific problems (punctuation, capitalization) which are not general. You should put it into an appropriate section.]

2.1 Parse Tree

One of the sentences that compose our sample report says: "Assessment reports indicated 117 deaths, 544 people injured, 12,794 homes damaged and 7,384 houses destroyed", Suppose that this sentence is called "S" [Jan: Full stop after last sentence.]

There are two mains steps which can [Jan: s/can/are] be performed to get the entities from this sentence:

- **Tokenizing**: This is a procedure of taking a sentence and extract [Jan: s/extract/extracting] the composing atomic linguistic elements e.i. [Jan: s/e.i./i.e.] words, verbs, punctuations, adjectives etc . S has the following tokens: ['Assessment', 'reports', 'indicated', '117', 'deaths', ',', '544', 'people', 'injured', ',', '12,794', 'homes', 'damaged', 'and', '7,384', 'houses', 'destroyed']

- **POS**: part-of-speech is a process of attaching to every linguistic element of the sentence a corresponding tagg [Jan: s/tagg/tag] based on grammar rules. The POS of S are: [('Assessment', 'JJ'), ('reports', 'NNS'), ('indicated', 'VBD'), ('117', 'CD'), ('deaths', 'NNS'), (',', ','), ('544', 'CD'), ('people', 'NNS'), ('injured', 'VBN'), (',', ','), ('12,794', 'CD'), ('homes', 'NNS'), ('damaged', 'VBN'), ('and', 'CC'), ('7,384', 'CD'), ('houses', 'NNS'), ('destroyed', 'VBD')]

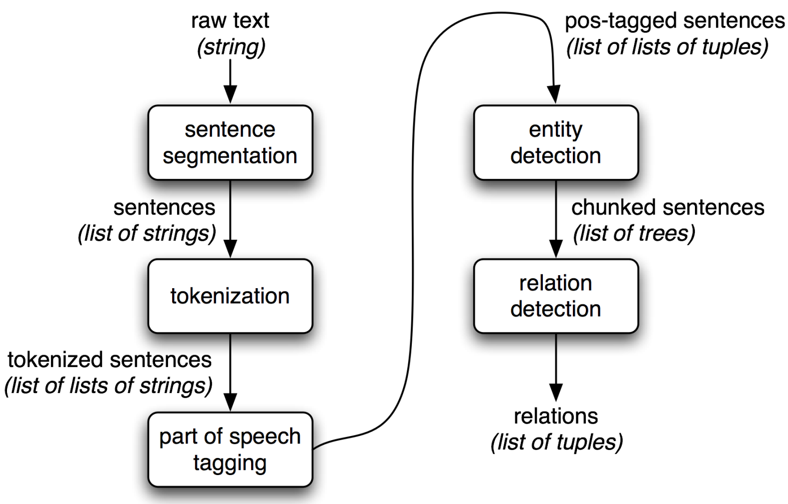
  The meanings of the used tags for S:

[Jan: Last sentence should be after itemize (and in a new paragraph).]

- JJ: **Adjective**: 'Assessment'
- NNS: **Noun, plural**: 'reports', 'deaths', 'people','houses'
- VBD: **Verbs, past tense**: 'indicated', 'injured', 'damaged', 'destroyed'
- CD: **Cardinal Number**: '117', '544' '12,794','7,384',
- CC: **Coordinate Conjugation**: 'and'

The parse tree is formed based on the POS, the classification of word and the way words are arranged in a sentence show a kind of relationship between words. [Jan: Split this sentence into two. Pay attention to grammar.] [Jan: Where is the figure taken from? Please cite properly.] [Jan:

Figure 2.1: General Extraction Process



Please provide citation for this section.]

2.2 Named Entity Recognation and Classification NERC

[Jan: Put “NERC” in parentheses in section name.]

The term "Named entity" has been coined in 1996 in "sixth Message understanding Conference" (MUC-6 R. Grishman and Sundheim 1996). [Jan: No manual citations, use \cite in LaTeX. Also it is not usual to write the conference name, just “has been coined in 1996 by \cite”.] Entity can be referred as a task, the entity is "named" when it is restricted to one or many rigid designators (Sharnagat, 2014), example: persons, location, product are the named entities.

Based on the classification of Standard Generalizes Markup Language (SGML) a task can be divided into three subtasks:

1. ENAMEX: location,product,country ,organization
2. NUMEX : percentage,quantity
3. TIMEX : time, date

[Jan: Spaces after commas above.]

The entities from different reports. [Jan: This is not a sentence (no verb).] For extracting entities in a report there are different models which can be used like Hidden Markov model, Supporting Vector Machine (SVM) , [Jan: No space before comma.] etc.

2.3 Hidden Markov Model

[Jan: Please say in the beginning what is this model used for (in the entity extraction context).] This model is based on Bayesian probability inference which has been initiated in 18th century. HMM is the earliest applied model for Natural Entities Recognition for English language.The way to perform these tasks is to find the most likely sequence of tagged names(TN) given a sequence of words(SW). [Jan: In prose there is always space before parenthesis. So, above it should be “words (SW)”. Only for mathematical functions, like f(x) it is the opposite.]

$$P(TN|SW) = \frac{P(SW|TN)P(TN)}{P(SW)} \tag{2.3.1}$$

The equation (2.3.1) is conditional probability, $P(TN|SW)$ can be called posterior and it is the probability of an event Sequence of word occurring given Tagged names has observed. [Jan: Formula and explanation above are inconsistent.] [Jan: Do not capitalize “sequence” or “tagged”.] $P(SW|TN)$ is also called likelihood e.i. it is the probability of observing the sequence of words(SW) when the given hypothesis tagged name (TN) is true. On another hand [Jan: s/on another hand/on the other hand] P(TN) doesn’t depend on the evidences, P(TN) is called prior e.i. that it is true even if there is no given evidence at all(masters thesis). [Jan: What do you mean by “masters thesis”?] We can be ignored P(SW) [Jan: We can be ignored/We can ignore] and the remaining objective is to maximise the probability of getting the sequence of tagged names when sequence of words is given.

$$Max\,[P(TN|SW)] \tag{2.3.2}$$

From the equation (2.3.2) of the maximization, the following estimation can be made. [Jan: Please say that this estimation is due to assumption that the probabilities of tags are independent from each other.]

$$P(TN){\approx}\prod_{i=1}^{n} P(TN_i|TN_{i-1}) \tag{2.3.3}$$

Where TN_i is a tag in the sequence of names (TN), for the likelihood probability can be estimated as :

$$P(SW|TN){\approx}\prod_{i=1}^{n} P(SW_i|TN_i) \tag{2.3.4}$$

The above estimations was for a small sequence where TN_i is a tag in the sequence of names (TN) and SW_i is a tag at index i in a sequence words (SW). For the large training corpus, the needed step is estimate based on the number of times the tag occurs and the position of the tag in a given corpus.

$$P(T_i|T_{i-1}) = \frac{K(T_{i-1},T_i)}{K(T_{i-1})} \tag{2.3.5}$$

Based on the training corpus, $K(T_{i-1},T_i)$ is referred as a how many times the tag T_i occurs after the tag T_{i-1}. In the corpus, $K(T_{i-1})$ is considered as the number of occurrences for the tag T_{i-1}.

Therefore the estimation can be performed as follow:

$$P(C_i|T_i) = \frac{K(T_i,C_i)}{K(T_i)} \tag{2.3.6}$$

From the equation (2.3.6), the term $K(T_i,C_i)$ is referred as the sum of the times that a word "C_i" has a tag $T_i$in the training corpus. The process of computing the posterior using the above steps is called Markov model.

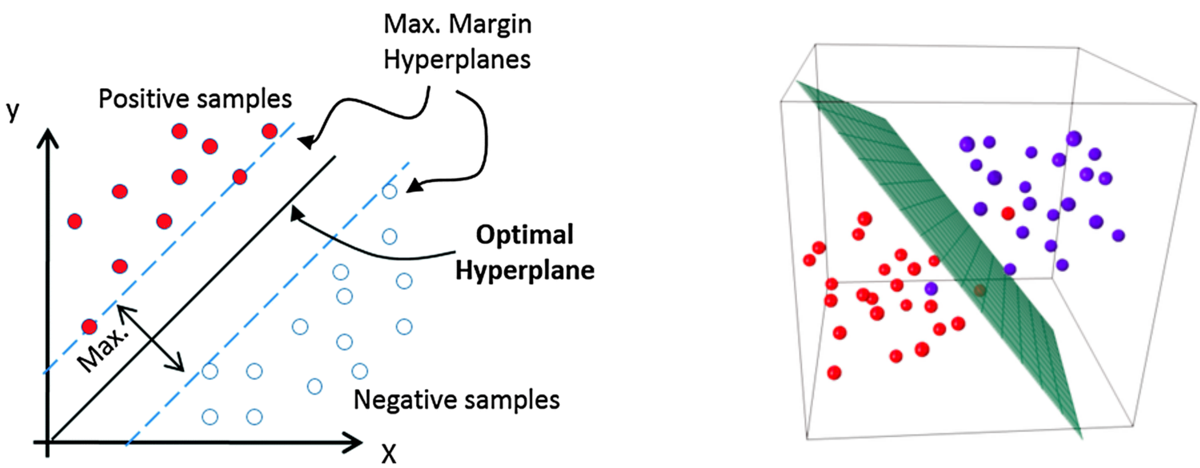
It is one of the most powerful statistical and machine learning (ML) techniques in modelling and high qualified in entities extraction. When the researcher is willing to train new data, HMM is very robust and efficient in computations. One of the limitations of HMM is that the researcher must have the notion of model topology and statistical techniques on how to deal with large amount of training data.

2.4 Supporting Vector Machine (SVM) based model

This model has an aim of classifying the named entities by separating the documents into two categories. The document must belong to one category, either positive or negative. SVM can classify linear data as well as non linear with a purpose of maximizing the margin between negative and positive documents.The plane which separate those two categories is called "hyperplane".

The main idea behind SVM modelling is to work with features and find the hyperplane. The hyperplane must separate all given samples regardless the dimensions.

2.4.1 Linear Supporting Vector Machine . [Jan: Why do you enforce new page here? Please don't do that.] !



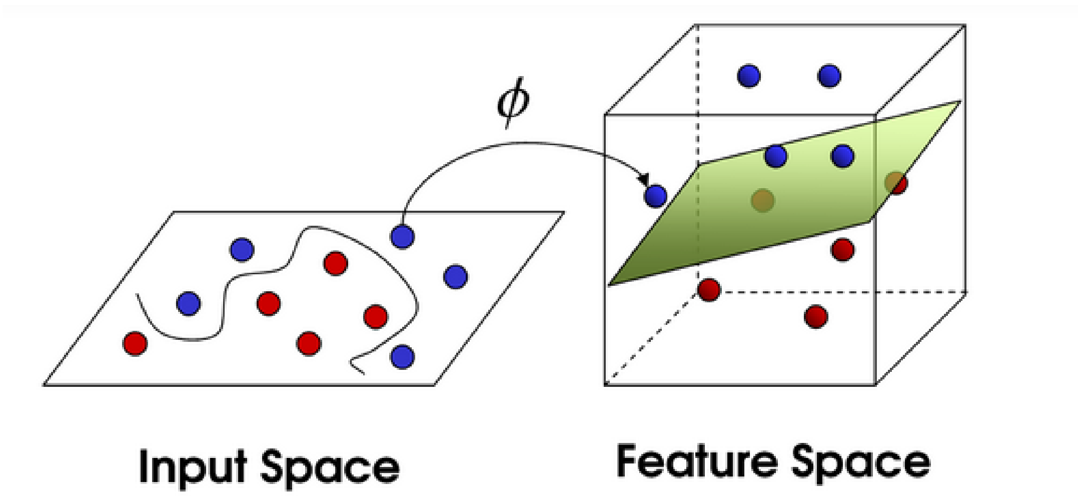
(a) Two dimensional SVM

(b) Multi dime

[Jan: Fix this figure. You can put two parts on top of each other.] For linear sample data, it is simple to plot the hyperplane to handle the separation. Data are spread separately between positive documents and negative documents. The way data are represented SVM decided whether to use linear modelling or not. IFRC reports are considered as multi dimensional documents.

2.4.2 Non Linear data. Sometimes the representation of data is quite mixed way so that you cant plot hyperplane easily. When the hyperplane can not be plotted as a straight line, SVM has a function to linearise non linear data. When the data are not linear, SVM has a way to linearise them by using a function. ϕ maps data to the higher dimensional space. Straightforwardly, the classification became linear. Figure 2.3 shows the way a function ϕ linearised the data.

Figure 2.3: Nonlinear SVM classification



2.5 Disadvantages of SVM

[Jan: Merge this section with the previous one.] The classification of particular documents is not easy to be performed by SVM without destroying the constructed weights but with hand-written rule model. Machine learning uses decision tree procedure than SVM. [Jan: s/than/rather than] In addition the decision tree has a detailed boolean-like model which is more popular to user. [Jan: Replace "popular" with another word (I'm not sure what you mean. "Accessible"?)]

2.6 Some Terminologies

[Jan: This is not a good title. "Some Definitions" would be better. Even better if it said what the definitions are related to (for example, "NLP Definitions"].

[Jan: I did not understand much from this section. What is IE? Definition of hand-written rule does not mention what it is. Also other definitions are not clear. What is the purpose of this section?]

Hand-written rule

It is one of the standard approaches of NER and IE, it has been used for extracting the patterns from automated pages such as amazon, NLP is so useful for unstructured humman-written text by delivering part-of-speech (POS), syntactic parsing and categories of semantic words.

Rule /pattern based extraction

Many IE systems uses rule/pattern to extract words and also phrases by looking to the context of those words or based on the their surroundings.(Califf and Mooney, 2003). Some system decided if the procedure of extracting the words should rely on the meaning of each word independently or on the context of their surroundings in a phrase. The limitation of this method is that some words do not have a closer mining to their surroundings that is why Patwardhan Siddharth with help of Ellen Rilo in workshop called "ACL 2006" presented another approach which was generating an automated IE system to learn patterns from a large fixed data set within a specific domain (Patwardhan and Riloff, 2007)

Bag of Words

It is referred to the multi-set of words represented by Natural Language Processing and Information Retrieval(IR). They are used for classifying the documents.

Our research deals with reports generated through a template, compared to the work of (Patwardhan and Riloff, 2007) templates usages is a limitation.

2.7 Text classification and Naive Bayes

It is one of the most important algorithm in text classification by using base rule and bag of words to classify the entities (Manning, 2012).The user instead of going through the report and start posing many queries, text classification algorithm transient the need information. Its aims is to build a function θ which takes the bag of words and returns the class of sentiment C either positive or negative. [Jan: Please don't use newpage, here or elsewhere.] !

$$\theta$$

$$\Updownarrow$$

ARCS initiated its response immediately after the earthquake struck to address the immediate needs. The National Society (NS) regional branches were at the forefront of the response and worked with Disaster Response Units (DRU). ARCS staff and volunteers were deployed promptly to support rescue efforts, provide first aid to the injured and distribute immediate relief supplies to affected people alongside undertaking initial assessments. A total of 900 volunteers were mobilised to support this response operation. ARCS also supported to transport critically injured people to hospital and mobilized community members for voluntary non-remunerated blood donations.

$$\Updownarrow$$

$$C$$

The procedure is to look for all words and retrieve those which form the subsets. Bag of words are formed after throwing away all words except the subsets.The use of the function θ is for attributing to each item of the bag of words a sentiment.

[Jan: I did not understand what this algorithm is for and how it works. I don't see how the picture is relevant to the algorithm. Please provide the source for the picture.] !

2.8 Machine learning for Named Entities

The natural language processing is not enough to handle the sophistication and ubiquity of textual data. Deep learning using machine learning techniques has been introduced to solve this problems. The advantages of machine learning for Named Entities:

- Manual extraction of entities is too expensive.
- Fast processes of extraction.
- extraction done by learning algorithms and Natural language tools. [Jan: Correct capitalization above.] !
- No limitation of languages.

3. Research Methodology

3.1 Data and tools

World wide non governmental organizations publish some of their reports on their official websites. Web scrapping is one of the ways to extract data from website to the local machines. We downloaded the reports about appeals in pdf format from IFRC website. We used R-scripts for web scraping form our co-supervisor professor Xavier. [Jan: Is Xavier formally co-supervisor? If yes, you should put him on the front page. Is he a professor? Please clarify this. Also use his full name.]

We downloaded 1262 reports which have been submitted between 1^{st} January 2015 and 31^{st} December 2016. To differentiate the reports, each report has a report Id but different reports can refer to the same appeal Id. [Jan: What is appeal id? Please explain.] As an international organization which insights [Jan: "Which insights"? I don't understand what you mean here.] on the largest humanitarian activities in the world, IFRC reports we have talk about disasters and cash transfer program. Cash Transfer Program (CTP) describes the money used by IFRC to buy food, shelter, etc. [Jan: Try describing better what CTP is.]

3.2 Data Analysis and Filtering

[Jan: The section title is not relevant to the contents.]

Portable document format (PDF) has content which can not be extracted and manipulated easily. The data we have has to be changed into another format in order to pull the information we need. We managed to transform 1260 reports. Our folder has the 1260 txt [Jan: Don't use math mode for this. You can use verbatim: \verb+txt+] files which is considered as dataset. For analysing the data, we used python programming language. [Jan: Capitalize Python.]

Our data reported on different areas of the World such as continents, countries and cities. For example "Europe IB23102015 23Oct2015.txt" covers European continent , "Afghanistan MDRAF003 05Nov2015.txt" and "Japan 0 16Apr2016.txt" reported on specific countries and "Port au Prince country cluster 0 04Oct2016.txt" reported on the most popular [Jan: s/popular/populous] city of Haiti.

3.3 Supervised vs Unsupervised Machine Learning

- **Supervised** is a machine learning part which deals with "labelled data", data are categorized and classified. We have a csv document which summarize [Jan: s/summarize/summarizes] the appeals what we have. The shape of this document is 25 columns and 3997 rows. The "CTP" feature indicate if the appeal is classified as a Cash Transfer Program document or not. Among 3997 appeals, 404 are CTP. [Jan: Please explain how the CSV file relates to your data.] [Jan: Make two paragrahps: one with definition of SL and one introducing your example.]

- **Unsupervised** can be defined as a way machine learning processes "unlabelled data". the data are unstructured, uncategorised and unclassified. The reports we have are good example of unlabelled data.

  – clustering is a technique for analysing data by identifying hidden groups in a data set. The hidden groups helps the machine to classify them data into small groups called "cluster" based on similarities or relationship found in data (Dy and Brodley, 2004). [Jan: Some typos here, please correct (also capital letter in the beginning).]

Unsupervised Machine Learning is very important, the analysis of data require the machine to use its brain Supervised learning. [Jan: I don't understand this sentence.]

3.4 NLP Corpus

Corpus is a set of large data which are semi-structured. To extract entities from corpus is simple [Jan: s/simple/simpler] than to deal with unstructured data. To get the corpus we filtered the data by using unicode of utf-8. [Jan: This is not accurate. You deleted non-printable characters from the converted files.]

To get compatible data, we have to filter using the Unicode provides canonical and compatible equivalence.

Regular Expressions: [Jan: Start with saying what REs are for (to search for patterns in strings)]. in Python, regular expression has operations and modules like "re.py" aND so on. they are used to manipulate characters in strings. regular expressions use a backslash ("\") to indicate a special form without invoking the meaning of the special form. There are many regular expressions functions but some of what we used the most are :

– re.split(): this function split par pattern and return the list of string.

– re.search(): it returns match objects. [Jan: Some typos above.]

– the match object ".end()": in a search string, it returns the end position of the match.

– the match object: in a search string, it returns the start position of the match.

ASCII stands for American Standard Code for Information Interchange. it is uses numbers to represent text by using 128 characters. Computer uses ASCII exist within unicode for storing texts easily. All ASCII uses unicode. [Jan: ASCII and Unicode (note capitalization) are different things and you are confusing them here.] ASCII characters are used to send and receive the e-mails, for text files and data conversions.

– ASCII-encoder: transform text to numbers.

– ASCII-decoder: transform numbers to text.

[Jan: What are those points for? What is their relation to the rest of the paragraph?]

Figure 3.1: ASCII TABLE

Dec	Hx	Oct	Char		Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL	(null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH	(start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX	(start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX	(end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT	(end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ	(enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK	(acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL	(bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS	(backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB	(horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF	(NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT	(vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF	(NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR	(carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO	(shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI	(shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE	(data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1	(device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2	(device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3	(device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4	(device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK	(negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN	(synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB	(end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN	(cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM	(end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB	(substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC	(escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS	(file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS	(group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS	(record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US	(unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Figure 3.1 demonstrates Unicode standard. [Jan: Acknowledge the source of the figure.] It provides a unique number to each character composing a text regardless the language, program or platform. UTF stands for Unicode Transformation Format. Unicode characters are set into binary values 0 and 1. UTF-8 for encoding 8 bytes, UTF-16 for encoding 16 bytes and UTF-32 which is a standard for encoding 32-bytes are three current standards. For our corpus we used UTF-8. !

After getting semi-structured documents, we removed the StopWords [Jan: s/StopWords/stop-words] which are defined as unnecessary words for extraction of entities from corpus. !

Normally the StopWords return vast amount of unwanted information. Some example of English Stop Words: almost, are, or, details, during, upon and so on.

Now we can check how for all of 1260 documents and count the Stop Words to be removed from vocabularies of corpus. We trained corpus by nltk package called FreqDist which uses frequency distribution of each word occurs in corpus, then the module of nltk technique called "nltk.corpus.PlaintextCorpusReader" helped us to get total 58104 stop words over the whole 7796263 vocabularies. [Jan: Please fix grammar above.] !

3.5 Modules and packages

The procedure of extracting entities requires many linguistic packages. Machine learning algorithms check grammar rules, punctuation marks and syntaxes. For some research eras , [Jan: Delete this beginning and just write "Some documents can have..."] documents can have special characters such as emoticons for emotions. To install these package, you must understand clearly !

how they work and the content type of your documents. [Jan: Do not use "you", write in passive voice. Also, what do you mean "to install you must understand how they work and the content type"? This is just not true.] This is a list of packages we used for extracting IFRC entities.

- **os** : This module is known as miscellaneous operating system interfaces. os represents the functionality of operating system with independent functions such as os.path.isfile(), os.path.exist, os.path.isdir, etc.

  Its functions are important for building andependant platform. [Jan: s/andependant platform/platform-indpendent programs] The programmes [Jan: s/programmes/programs] written using os module can execute in Windows and Linux regardless the machine operating system.

- **nltk**: Natural Language Toolkit is one of core packages for linguistic modelling . With various important built-in functions nltk is able to manipulate documents. The main idea behind nltk is to use $nltk\_corpus$ to collect all documents as one dataset, then split the documents into sentences using ne_chunk and remove the stopwords by importing $stopword$ from $ntk.corpus$, lastly apply machine learning algorithms to extract entities.

- **PyPDF2** is able to extract specific information from a pdf document based on the section they belong to. This package locates top section, title, author, etc. It has many functions such as splitting documents pages, merge document pages, encrypt and decrypt documents and so on. It can be compared to pdftk

- **pandas** is an open source with high performance structure within various build in functions. Dataframe design for presenting many data in organized way. Pandas is powerfull in data analysis, flexible, fast and manipulation took for any language. In our research, We used pandas for making the frames of our data. [Jan: I could not understand this description.]

- **codecs** module offers unicode string for encoding and decoding. codecs is used for handling errors and gives freedom to access internal registry. Codecs are not limited to text but mostly are for text encodings which is for encoding text to bytes. Additionally, there exits codecs for encoding text to text, some codecs can encode and decode at the same time.

- **defaultdictionary** has basic content of difference between verbs, nouns, adjectives, adverbs etc. It uses tagger to assign each word composes the sentence a corresponding POS as explained in Chapter 2.1 classes of words are inverted by NLP, it refers to categories of words in dictionary. [Jan: What do you mean? Defaultdict in Python is just a dictionary with default value for missing key instead of key error. It has nothing to do with parsing.]

- **Python String** module which returns a string with trailing text removed. [Jan: The module does not return anything, its methods do. Their names are "rstrip" and "lstrip".] It has two methods to strip text on both sides, right strip() and left strip(). Trailling text can be unwanted space, extension, punctuation marks, etc.

  To indicate the position of the character to be stripped we use left(l.strip()) which removes the character at the beginning of a string or right(r.strip()) to remove the character at the end of the string.

- **regular expressions** regex is a module which finds out the patterns between strings by setting rules for text. bytecodes compile those pattern rules and execute using matching rules. example methods for re are explained into Chapter 3.4

- **polyglot** is used to extract entities from many languages. It is multilanguage application supporter built as natular language pepeline.

- **Stanford** is one of most brilliant algorithms to extract entities from documents corpus. it has classifier models, jar files which are free downloads. Stanford has many packages to handle linguistic problems.

[Jan: In my opinion the whole section above needs rewriting.] !

3.6 Extraction of Entities

To extract entities we used default dictionary built in collection package of nltk. Our dataset now is a folder containing 1260 corpus files, we used nltk chruncker to get sets of sentences of corpus. let have a look for our sample document the way sentences are split. [Jan: From the figure it looks these are not sentences, just lines.] !

Figure 3.2: Set of sentences

```
['DREF operation n MDRAF003 Glide n EQ-2015-000147-AFG\n', 'Date of Issue: 26 May 2016 Date of disaster: 26 October
2015\n', 'Operation start date: 3 November 2015 Operation end date: 2 March 2016\n', 'Operation budget: CHF 465,684
Current expenditure: CHF 379,353\n', 'Number of people affected: 65,653\n', '1\n', 'Number of people assisted: 14,0
00 people (2,000 families)\n', 'Host National Society(ies) present (n of volunteers, staff, branches):\n', 'The Afg
han Red Crescent Society (ARCS) has at least 1,800 staff, 25,000 volunteers and 34 provincial branches and\n', 'sev
en regional offices nationwide. A total of 13 branches of ARCS are involved in the earthquake response, with some
\n', '700 volunteers mobilized to support activities\n', 'to the benefit of affected people.\n', 'N of National Soc
ieties involved in the operation:\n', 'The International Federation of Red Cross and Red Crescent Societies (IFRC)
 with the Movement partner actively\n', 'involved in supporting the ARCS response. IFRC and ARCS also maintained go
od coordination with other movement\n', 'partners, the International Committee of the Red Cross (ICRC), partners wi
th present in Afghanistan that include the\n', 'Canadian Red Cross Society, Danish Red Cross, Norwegian Red Cross,
 and Qatar Red Crescent Society. However,\n', 'Red Crescent Society of the Islamic Republic of Iran, Red Cross Soci
ety of China and Turkish Red Crescent Society\n', 'do not have offices in Afghanistan but have supported the earthq
uake response through bilateral arrangements with\n', 'ARCS.\n', 'N of other partner organizations involved in the
 operation:\n', 'Afghanistan National and provincial Disaster Management Authorities, Ministry of Rural Rehabilitat
ion and\n', 'Development (MRRD), UN agencies (WFP, UNICEF, WHO), International Organization for Migration (IO
M),\n', 'International Rescue Committee (IRC), People in Need (PIN), Care International and Oxfam.\n', 'Partners wh
o have contributed to the replenishment of this DREF include Canadian Red Cross Society/\n', 'Canadian Government
 (DFATD), DG ECHO, and Netherland Red Cross/ Netherlands Government (SEF). The\n', 'unspent balance of CHF 86,331 w
ill be returned to the DREF pot.\n', 'A. Situation analysis\n', 'Description of the disaster\n', 'Around 13:40 loca
l time (UTC +4:30) on 26 October 2015, a magnitude 7.5 earthquake struck Badakhshan province\n', 'in the north-east
region of Afghanistan. Badakhshan, Nangarhar, Baghlan and Kunar provinces were ranked the most\n', 'affected provin
ces. The Afghanistan National Disaster Management Authority (ANDMA) coordinated the initial\n', 'assessments in par
tnership with in-country humanitarian partners.\n', 'Assessment reports indicated 117 deaths, 544 people injured, 1
2,794 homes damaged and 7,384 houses destroyed.\n', 'In Badakhshan province alone, more than 51,000 people were aff
ected. The province also reported to have the most\n', 'extensive damages to properties. Kunar and Nangarhar provin
ces were recorded to have the highest number of\n', 'deaths and casualties as a result of the earthquake. Food and
 non-food items (NFIs), emergency shelter, and\n', 'psychosocial support services were identified to be among the i
mmediate needs. As the country moved into winter\n', 'season, winterization materials were being prioritized in the
response plan. Access to the affected population\n', '1\n', 'Afghanistan Earthquake, OCHA Situation Report No. 3 (a
s of 12 November 2015)\n', 'DREF Final Report\n', 'Afghanistan: Earthquake\n', '\x0cremained the most significant c
hallenge in delivering humanitarian assistance in a timely and effective manner. With\n', 'the support of the gover
nment, roads were cleared to pave way for humanitarian actors to reach the earthquake\n']
```

Figure 3.2 shows the 45 first lines of the sample document. each each line is ended by $'\backslash n'$.

3.7 Top Section Dataset

From the analysis of IFRC pdf reports, most of them have a small table on the top. this table gives the image of what the report is talking about. This table summarizes what the document is talking about. For example the total amount of money spent in recovering a disease, the number of people who participated in a given activity, the location and so on.

While we were transforming the pdf data into txt format, this table occupied almost 25 first lines. Due to the limited time of the research, We decided to split those twenty five first lines of each document. the collection of those first twenty five documents has been considered as our new corpus.

Now we can use one of the algorithms to extract entities and for classification. [Jan: ...from this table.]

3.8 Stanford Named Entities Recognition

The data to be trained is unlabelled. Named Entities Recognizer labels the data to be extracted easily. it recognises sequence of words and its classification is mainly to name of persons, localization and organization.

Stanford Named Entities Recognition is an extractor implemented in java. It takes the sequence of words and label them Stanford named entities recognition is a able to identify correctly the named recogniser which labels sequences of words in a text. The next step is to split the sentences into set of words called tokens. By using the Stanford NER tokenizer where token can be tagged.

- **Stanford NER Tagger** is a package which has modules for classifying tokens with the taggs. A tagg can be defined as one of classes of significant words like nouns, adjectives etc. we used the package Stanford POS Tagger to classify the words.
- **Stanford NER Models** are many Stanford has different models such as "stanford-corenlp-full-2016-10-31", "stanford-ner-2014-01-04" which is the version we used.
- **Stanford Classifier** is a package which classify the entities into defined categories. It has four specific classes such as "Locations", "Persons", "Organizations" and "Others".

We specified the named entities that we wanted to extract. We classified them into the four categories by Stanford classifier. The last category called "others" combined all numerical entities such as time, amount of money, number of people, percentage, etc.

The reports from our corps are order by appeal numbers, the entities are in classified by nltk algorithm.

Figure 3.3: IFRC entities from Stanford NER

	- Global - MAA00001 22Jul2015.txt	**- Global - MAA00006 24Apr2015.txt**	**- Global - MAA00010 10Nov2015.txt**	**- Global - MAA00021 02Jun2015.txt**	**- Global - MAA00028 01May2015.txt**	**- Global - MAA00029 21Jun2016.txt**	**- Global - MAA00040 02Jun2015.txt**	**- Global - MAA00040 10Nov2015.txt**
locations	[Neonatal]	[Geneva, Geneva]	[Bolivia]	[Sendai, Japan, Geneva, Cali, Colombia]	[Geneva, Panama, Kuala Lumpur, Nairobi, Dubai,...	[Syria, Iraq, Afghanistan, Libya, Ukraine, Yem...	NaN	NaN
organizations	[Global Health Report Health Department 2014 T...	[National Societies, NSKD, International Feder...	[DREF 2013 Number Amount, Red Cross Red Cresce...	[Preparatory Committee of WCDRR, DRR, Fourth G...	[IFRC Global Logistics Service, IFRC, National...	[Red Cross, Red Crescent, IFRC, IFRC, Middle E...	[Federation of Red Cross, Rules for Disaster R...	[IFRC, Crisis Management Department (DCM) Gl...
other	NaN	[2014, 2014, 2014 The Difference Overview The ...	[31 %, 69 %, 2014, 2 per cent, 2013, April 201...	[2015, March 2015, July, November 2014, June, ...	[2015, Strategy 2020, 2015, 2014, 2015, 2014, ...	[2014, 2014, 1990, 7 %, 2014]	NaN	[January June 2015, January 2015 12 months 72]
persons	NaN	NaN	NaN	NaN	[Sierra Leone]	[Jaime Sepulveda, Christopher Murray]	NaN	[Pankaj Mishra, Hakan Karay]

From Figure 3.3, Consider the for the report "-Global MAA00029 21Jun2016.txt", locations row shows that the report covered Syria, Irak, Afganistan, Libia, Ukraine, Yemen, etc. The extraction of entities separates clearly the categories.

3.9 Natural Language ToolKit (NLTK)

Natural language toolkit is one of the algorithm to extract named entities. It has different modules which are used to process the data alongside the extraction. [Jan: Make capitalization consistent across the thesis: NLTK, Natural Language ToolKit.] NLTK chunkparser is a one of nltk module which uses Regular expressions. NLTK tokenize which splits the sentences into small units called tokens. This module helps the NLTK tagger to identify words independently.

Generally NLTK classify the entities into four categories which are known as Location, Organization, Persons and Others. [Jan: Singular or plural? Be consistent.]

Figure 3.4: IFRC entities from NLTK

	- Global - MAA00001 22Jul2015.txt	**- Global - MAA00006 24Apr2015.txt**	**- Global - MAA00010 10Nov2015.txt**	**- Global - MAA00021 02Jun2015.txt**	**- Global - MAA00028 01May2015.txt**	**- Global - MAA00029 21Jun2016.txt**	**- Global - MAA00040 02Jun2015.txt**	**- Global - MAA00040 10Nov2015.txt**
locations	NaN	NaN	NaN	NaN	NaN	[West Africa, West Africa, Caribbean]	NaN	NaN
organizations	[Global Health Report Health, Contents, CBHFA,...	[Global, Difference, National Society, NSKD, N...	[Overview Statistics, DREF, CHF Total, DREF, D...	[DRR, HFA2, DRR, WCDRR, WCDRR, HFA2, UNISDR, W...	[IFRC Global Logistics Service, GLS, IFRC, Nat...	[oPt, Ebola Virus Disease, EVD, Sahel, Horn, R...	[Red Cross, Red Crescent, MAA00040, DCMs, DCM,...	[DEVELOPMENT, UPDATE, INTERVENTION, DCM, CHF, ...
other	[Disease, Maternal, Neonatal, Child, Sanitation]	[Geneva, Long, Geneva]	[Bolivia, Bolivian]	[Sendai, Japan, Geneva, Cali, Colombia]	[Geneva, Panama, Dubai, Las Palmas, Ebola, Ira...	[Iraq, Afghanistan, Libya, Palestinian, Yemen,...	NaN	NaN
persons	[Annual, Annexes Annex, Health, First Aid]	[Knowledge Development Division, Term Planning...	[Start, Red Cross Red Crescent, Emergency Fund...	[Billion Coalition, Climate, Climate]	[Overview, Kuala Lumpur, Nairobi, Guinea, Arab]	[Latin America, Global Health, Jaime Sepulveda...	[Crisis Management, Rules, Disaster Relief, Gl...	[Disaster, Crisis Management Department, Globa...

[Jan: Are you sure this figure is correct? It seems there are lots of obvious locations classified as "other".]

From Figure 3.4, Consider organizations extracted from the report "-Global-MAA00021 02 Jun 2015.txt", NLTK entities classifier was able to extract DRR, HFAR, WCDRR, HFAR2, UNISDR, etc. The classifier uses nltk tagger and default dictionary which help it to identify the names, verbs and adjectives.

3.10 Polyglot Named classifier

Compared to previous entities extractor, Polyglot has only three categories which are "Persons", "Locations" and "Organizations". For nltk, any entity which is classified into those three categories is not considered as named. [Jan: I don't understand last sentence.]

Figure 3.5: IFRC entities from Polyglot

	- Global - MAA00001 22Jul2015.txt	- Global - MAA00006 24Apr2015.txt	- Global - MAA00010 10Nov2015.txt	- Global - MAA00021 02Jun2015.txt	- Global - MAA00028 01May2015.txt	- Global - MAA00029 21Jun2016.txt	- Global - MAA00040 02Jun2015.txt	- Global - MAA00040 10Nov2015.txt
locations	NaN	[Geneva, Geneva]	[Bolivia, Bolivia]	[Sendai, Japan, Geneva, Cali , Colombia, Cali]	[Geneva, Panama, Kuala Lumpur, Nairobi, Dubai,...	[Syria, Iraq, Afghanistan, Libya, Ukraine, Yem...	NaN	NaN
organizations	[Health, First, Adolescent, Sanitation, Cross,...	[Global, National Society and Knowledge Develo...	[Crescent, Cross, Red Crescent Societies, Disa...	[World Conference, UNISDR, WCDRR, Community Re...	[Logistics, Global Logistics Service, National...	[Global Health]	[nternational, of Red, Red Crescent, Managemen...	[Crisis Management Department, Crisis Management]
persons	NaN	NaN	NaN	[DRR]	[GLS, GLS, GLS]	[Jaime Sepulveda, Christopher Murray]	NaN	[Simon Eccleshall, Pankaj Mishra, Hakan Karay,...

Let us take an example report "-Global-MAA000029 21 Jun 2016.txt" from Figure 3.5, the entities which are classified as "Persons" Jaime, Sepulveda and Christoper Murray.

3.11 Sample Files

The type of data we have can be considered into two different ways. There are some reports which are classified as CTP documents. This documents cover the overview of how IFRC money was invested in humanitarian activities.

Non CTP reports are focused on other activities which didnt require IFRC to invest money.

The next step is to take example report document for CTP and Non-CTP to have a comparison on the extracted entities.

[Jan: Please work more on this chapter. There are many language mistakes (I did not point out all of them). Also please stop using newpage, Latex will put the tables reasonably on its own.] !

4. Results Discussion and Testing

4.1 General Overview

To extract and classify entities, We used Stanford, NLTK and Polyglot. These entities extractor have common categories which are "Persons", "Location" and "Organization", Additionally NLTK and Stanford NER has another category which is called "others". This last category is not very clear. It combines numbers, percentage and unclassified entities. This can cause the confusion for to the organization. The core categories are those three first groups.

Among these three entities extractor, Stanford requires time to run compared to others.

The named entities must be setted by the organization based on its interest. [Jan: s/setted/set] Some reports are composed by many pages but some few point must be highlighted. Templates in reporting are important, they made life easy.

Before extracting the entities, You must know what the document is talking about. What the organization is struggling to know from the report.

Named entities from NLTK, Polyglot and Stanford are useful. They tried to summarise the primary information such as locations, persons and organizations.

Sometimes, extracted named entities are not sufficient. You can used Regular expressions to respond perfected the will of the organization. [Jan: Please do not address the reader directly ("you"). Also you are repeating yourself in some of the points.]

4.2 Case Study Results

After analysing 1260 documents, Let us take one sample file and work on top section composed by 25 lines.

Consider a document which is specific to African region. "Africa regional office MDR60002 03 Nov2015.txt". We are requested to extract name of Persons who participated in IFRC activities.

We had a function to extract four categories of entities by Stanford NER. It is only to specify the category we are interested in. To identify persons names manually is also possible.

4.3 Testing

For the security purpose testing gives a guarantee of correctness. It is a major chapter for assertion of the research quality. Typically our research is a part of big data and Machine learning. We used analysis and statistical testing to make sure that the results are true. [Jan: I did not understand this paragraph at all.]

The process of extracting entities can be done in different ways. Either manually or by the use of machine learning algorithms. The manual way has many disadvantages as explained in Chapter 2.8.

Computer algorithms have impact for solving human problems. However we have to do a comparison for a small dataset between algorithm results and human results. The correctness of a tested dataset gives a confidence for remaining datasets.

IFRC uses the templates formats to produce their report. It is way of structuring a content of the document. The use of templates made most IFRC reports to have almost the same size of top section. Top section contains important summary as explained in Chapter 3.7 .

Due to the time limitations, We tested some sample documents and We concluded for all top sections of the reports.

JSON file

By taking the sample file, We extracted names of entities in JSON format. JSON stands for Java Script Object Notation. It is built based on two universal data structures such as a pair composed by a name and a value, and ordered list of values which is considered as an array, sequence, vector or list.

Figure 4.1: JSON File Structure

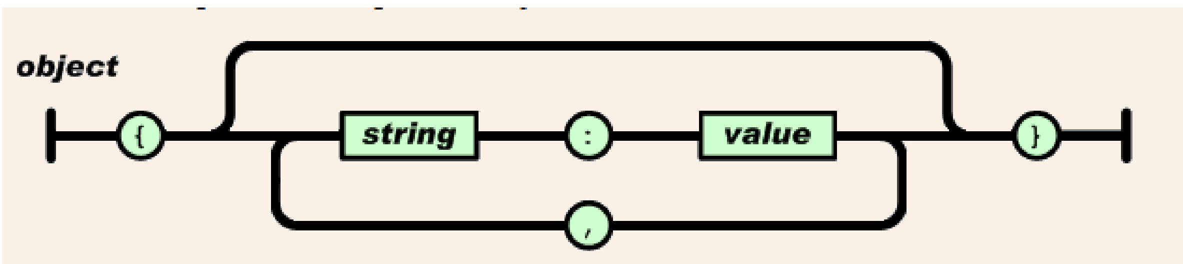


Figure 4.1 [Jan: Please acknowledge the source of the figure.] refers to the structure of our json file. It contains a small dictionary which has one feature of proper names. We are able to identify three people who participate in IFRC sample report.

```
{'BothNames': {'0': 'Mamadou Basilah',
  '1': 'Tommy Trenchard',
  '2': 'Norbert Allale'}}
```

Figure 4.2: Persons Names Extracted by Hands

After extracting three proper names as the Figure 4.2 shows, We are now going to do a comparison. We can compare the output of the algorithms.

Figure 4.3: Comparasion

	Hand-labeled True BothNames	Stanford NERC Authors	Polyglot NERC Authors
0	Tommy Trenchard	Mamadou Basilah	Mamadou Basilah
1	Mamadou Basilah	Norbert Allale	Tommy Trenchard
2	Norbert Allale	Norbert Allale	Norbert Allale

In Machine Learning, there are three ways of testing the quality of algorithms. As We extracted entities from IFRC reports, to be sure on the work of algorithms, We calculated recall, precision and accuracy.

Precision has been calculated as a fraction of relevant instances over retrieved instances.

Recall has been gotten as a fraction of retrieved relevant instances over sum of relevant instances.

Prediction is made by algorithms to predict the name of persons in sample document. The correctness can be calculated based on comparison between what predicted and what extracted by hands. The figure 4.4 is the results of comparison between Polyglot names of

Figure 4.4: Precision

	Predicted Negative	Predicted Positive
Negative Cases	0	0.0
Positive Cases	0	3.0

[Jan: I think this example is pretty interesting. Consider adding more. For example, it would be interesting to see an instance where the algorithms are not 100% correct and understand why. That way you wouldn't have to worry about inflating number of pages with \newpage...]

5. Conclusion and Future work

Entities extraction has been performed using natural language toolkit, polyglot and Stanford named entity recognition. Evaluation of entity extraction is normally done by the metrics of precision, accuracy and recall between algorithms and named extracted by human hands. This research argues that top section of report has meaningful metrics. The results demonstrate that a process of extracting names of persons in top section of reports was well done.

As future work, the next step for entity extraction is to work on other sections of a document. To combine all used approaches into a software which can automatically visualised entity named by organization such as budget, number of people suffered from a disaster etc.

References

Shumeet Baluja, Vibhu O Mittal, and Rahul Sukthankar. Applying machine learning for high-performance named-entity extraction. *Computational Intelligence*, 16(4):586–595, 2000.

Mary Elaine Califf and Raymond J Mooney. Bottom-up relational learning of pattern matching rules for information extraction. *Journal of Machine Learning Research*, 4(Jun):177–210, 2003.

Jennifer G Dy and Carla E Brodley. Feature selection for unsupervised learning. *Journal of machine learning research*, 5(Aug):845–889, 2004.

Christopher Manning. Information extraction and named entity recognition, 2012.

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. Text classification and naive bayes. *Introduction to information retrieval*, 1:6, 2008.

David D Palmer and Marti A Hearst. Adaptive sentence boundary disambiguation. In *Proceedings of the fourth conference on Applied natural language processing*, pages 78–83. Association for Computational Linguistics, 1994.

Siddharth Patwardhan and Ellen Riloff. Effective information extraction with semantic affinity patterns and relevant regions. In *EMNLP-CoNLL*, volume 7, pages 717–727, 2007.

Rahul Sharnagat. Named entity recognition: A literature survey. *Center For Indian Language Technology*, 2014.