CHAPTER 1
INTRODUCTION

NEWS, the acronym for North-East-West-South, needs no introduction to anyone. In India, the age group of 12-14 is considered as the novice stage amongst the newspaper readers. With the ease-of-access which the Internet provides along with an addition to the rise of e-news over the past decade, an educated guess can be made on the populace of young readers. However, NEWS has never been and will never be dependent upon the age band of readers. This serves as a problem for any neophyte as any kind of update over past events or any sort of addition to an existing article may leave a huge knowledge gap in readers marking the information as useless. Adding the problem of fake NEWS and wrong information often not addressed by media sources, a user may remain misinformed forever despite the wrong information not having any background data to verify it. This, in fact, is not only true about NEWS but also for any kind of factual information. The same kind of problem migrates to research scholars. A survey of related work for any kind of work in a field of interest hogs limited resources such as time. Though we have good documentation regarding past affairs and existing knowledge, each and every one of us is limited by the time span to gather related data from multiple sources. This field of accessing knowledge from multiple sources about an event has not seen much work as any kind of work in structuring knowledge or information about any domain is a mere addition of new knowledge to existing facts. This fact is justified as Humanity has always perceived the knowledge in a similar manner. The information has always been practiced as data improved with experience. The phrase, "Learn more to know how little you know" will always remain true to us. Humans have published multiple articles and books, formulated numerous knowledge bases, created probably uncountable blogs and journals for the passage of knowledge from one generation to next but never addressed to the problem how entire knowledge regarding a specific term can be retrieved to be made use of.

The crux of issue is to retrieve information related to any kind of textual source. The primary reason to take these kinds of source(s) into consideration is because it remains as the most popular choice to advertise the recent most knowledge. For e.g. NEWS or journals are communicated via text and media only.

Our work addresses the above-mentioned issue how to fetch existing related knowledge on any kind of extracted information in unstructured textual format, which can be used as immediate or intermediate knowledge for further use. While doing so, we must take care of the following

- Knowledge from unstructured format must be bought down to a structured scheme for better understanding as well as retrieval of information. Since the existing knowledge is already in a structured form somewhere else, it is vital the new one follows a similar path. This also yields another useful asset – Any kind of update of knowledge is much effortless as data retrieval is easy because of choice of exploring distinct entities.
- Updating Information must not distort existing knowledge. A point to be noted here is the existing knowledge is not the knowledge fetched from external sources but knowledge from unstructured forms. The mentioned should be followed strictly since the intent of interest prioritizes the information from unstructured forms more than external sources.
- Information retrieval from this new structure must be capable of both –
  - Addition of new knowledge as well as
  - Retrieval of knowledge

The rest of the paper is organized as follows – Chapter 2 contains related work done in this field, Chapter 3 mentions how we've addressed and dealt with the issue, Chapter 4 highlights what we've achieved and Chapter 5 concludes the paper acknowledging works which can be done in future.

CHAPTER 2
RELATED WORK

To resolve the issue of retrieval of information from unstructured sources, for any kind of further use, has seen work done in fragments. There are multiple works regarding annotation of terms in the unstructured text as well as detection of entities and relationships among the entities both with or without the usage of some existential, general as well as specific, knowledge. Knowledge from semantic web has also been structured after the growth of Worldwide Web. Such kinds of work, though used to retrieve information about any topic, can be used as a potential to gain or rather grow and populate another structure of knowledge.

2.1 Literature Survey

Delia Rusu & *Lorand Dali* et. al. 2018[1] presented a work on extracting subject-predicate-object triplets from unstructured text. They extracted RDF triplicates from parse tress via help of parser dependent techniques. They primarily used four parsers – OpenNLP, Stanford Parser, Link Parser and Minipar. Their work dealt with the extraction of triplicates from treebank parsers as well as linked grammar. *Peter Exner* & *Pierre Nugues* et. al. 2012[2] designed an end-to-end system for extraction of RDF triplicates from unstructured text. The work was focused on entities describing relations & properties amongst them. An ontology mapping scheme was used to generate 189,000 triplicates, in N-Triple format, from 114,000 Wikipedia articles via bootstrap techniques. *Kundan Kumar* & *Siddarth Manocha* et. al. 2015[3] used the approaches of relation extraction via semi-supervised methods to generate a knowledge graph. They used semantic similarities, more precisely cosine similarity, to determine new relations from existing one.

In addition to these kinds of work, there were already some toolkits for creating RDF triplicates from text. **FRED[4]** is a popular tool for Semantic Web. Not being limited to *English*, it provides support for 47 other languages as well. It is available in Python as a REST API – *fredlib*.

Open Calais[5] – a service provided by *Thomson Reuters* provides a structure to unstructured text. It returns relevant and related content such as Events, Topic Codes and Social Tags in addition to entities and relations among them. Most of these tools however are not precise and accurate in determining relations as their primary focus of work is to annotate the entities. While investigating various approaches for relation extraction, many *pattern matching* techniques were also found to being used in real time where the prerequisite to analyze text was not required or already achieved. All these approaches are somewhat supervised or semi-supervised.

2.2 Existing Sources of Knowledge

Whenever a knowledge about a particular domain is formulated into a structural format, the resulting structure is termed as *knowledge base*. These structures can be used for easy retrieval of knowledge or updating other knowledge bases. Project **NELL[6]** was one such research project which aimed to create a system that extract facts from billions of web pages. Some predefined ontology was used to generate relations from a huge *corpus* along with multi-web paradigm. This kind of work is termed as Distant Supervision. Similar projects such as **YAGO, DBpedia, Google Knowledge Vault** which is now **Google Knowledge Graph** are also serving the purpose of knowledge bases where the fetched information from web was being stored. DBpedia[7] is a project for extraction of structured knowledge from Wikipedia articles. Google's Knowledge Graph[8] is currently the back end of *Google Search* and *Google Assistant.* The project was aimed to move from *web of strings* to *web of things*.

All of the knowledge sources are open-source and available on world wide web to use free-of-cost. These knowledge bases are currently being used in different domains such as Artificial Intelligence, QA Systems, chatbots, etc.

We view these knowledge bases as potential sources of knowledge which can be used to fetch related information regarding any content. The information provided by these types of sources is already structured in nature and can be easily converted to other formats, if required.

Problem Definition

To create and maintain a knowledge base from domain knowledge the two main criteria which should be fulfilled are

1. Formulate the unstructured knowledge from sources into a structural format so that it can easily be queried
2. Constant expansion of knowledge from other sources as updating the information is crucial

The first criteria alone is sufficient for the creation of knowledge base. However, this kind of product is fruitless as there is no increment in terms of information as compared to other sources. The expansion of knowledge should also not disfigure existing information.

To achieve the same two tasks concurrently we will use a **Knowledge Graph.** We've chosen Knowledge Graph as we feel it provides more flexibility in storage of information, such as relations, compared to other forms of knowledge base such as *Ontology*. This arrangement of knowledge also provides a better visualization of information. An insight into the information is also uncomplicated. We've tried to create a technique which can mine a knowledge graph where the initial source of domain knowledge is unstructured text in *English* language. Once we've had information in a structural format, we use other sources of knowledge to expand our learning.

CHAPTER 3
PROPOSED METHOD

In our technique, the construction of knowledge graph has been done in three phases

1. Processing Unstructured Text – Entity and Relation Detection
2. Mining Knowledge Graph – Introduction of Nodes and Relations
3. Knowledge Expansion – Extension of knowledge from other sources

Each of the following has been discussed in detail below

Processing Unstructured Text

Since we are dealing with unstructured text, we first need to develop a mechanism which can extract entities from this territory of knowledge. Entities might be defined as any *non-stopword* be it a name, place, living or non-living thing which can have existence or significant meaning. A special mention must be made here to distinguish between entities and noun. A noun is used to identify a class of things, places or people whereas entities are a noun or a group of nouns which together signify a different meaning. Every entity in the unstructured text is a noun but not every noun is an entity. For e.g. consider the words ‘Prime’, ‘Minister’ and ‘India’. All of these are a noun and some alone or in combination may even classify as entities generating different meaning. But the phrase - “Prime Minister of India” acts altogether as a different term with significant meaning denoting the supreme leader of country India.

To detect these entities, a model has to be trained over a corpus – where the linguistic features and entities presented matches with the interested domain. Only after successful training, entities can be correctly detected. A more general approach can also be followed to detect entities with the help of domain-independent linguistic features but results in higher redundancy. This approach, however, allows scope for noise-filtering techniques which can be applied to retrieve more entities as compared to former approach. Since our domain of knowledge is fixed i.e. NEWS, we use an annotation tagger already trained over this domain for successful detection of entities. Furthermore, we restrict our domain only to NEWS over the region of India as that reduces the scope of further training, if required, to a limit.

Once entities are successfully identified, the second task comes into play – *Relations* extraction. A relation signifies a semantic correspondence between two entities. A relationship plays an essential role to entities as it may even sometimes add up to the meaning of entities associated with it. We've used a general approach to extract relations between two entities. After the extraction of entities, the *group of text* occurring between every pair of adjacent occurring entities in a sentence is passed through these phases:

- **Stopword Removal** – Stopwords are domain independent low-information words such as 'he', 'she', 'the' etc. Removal of these words doesn't alter the true intent of information
- **Stemming** or **Lemmatizing –** A stem, or a lemma, is a root keyword occurring in different forms of tense such as "melt", "melting" or "melted". Based on *Porter's Algorithm,* stemming phase doesn't truly distort the original meaning. Lemmatization is an advancement over stemming where the stem keywords are surely from dictionary.

Most of the times, a possibility might occur where there will be no relation between adjacent pair of entities even though there exists a relation in unstructured format. These kinds of issues can neither really be ignored nor handled. To resolve this, we define a custom-made relation - "*related",* which denotes that there might be some sort of relation between a pair of entities which might be not significant enough to name but at the same time not be completely irrelevant as well.

TABLE 1 – Algorithm for extraction of entities and relation from unstructured text

Algorithm: Extraction of Entities and Relations
Input: List of NEWS Headlines, A trained tagger
Output: List of entities linked with each other

```
entities = {}                                          //initially an empty set of entities
temp = {}                                              //a temporary set to store entities
String relation = null                                 //a null string
for each headline in news:
    temp = extractEntities(headline)
    for each entity in temp:
        entities.append(entity)
    end for
    for each pair(a,b) of entities in temp:
        relation = extract_Text_Between(a,b,headline)
        remove_stopwords(relation)
        lemmatize(relation)
        if relation not null                           //if any kind of relation found
        THEN
            add_Relation(a,relation,b)
        else
            add_Relation(a,related,b)                  //adding related relation
        end if
        relation = null                                //relation nullified for next pair
    end for
    temp = {}                                          //temporary set emptied for next headline
end for
return entities
END
```

Knowledge Graph Mining

The previous phase resulted in a list of entities along with some relations defined over them. Now, the task remains to represent the data into a format where the fetched information can easily be queried as well as visualized. We take the aid of Knowledge Graph as our knowledge base. A knowledge graph is a collection of interlinked entities where both the relations and entities boost the meaning of each other and provide room for growth of knowledge. The construction of knowledge graph is termed as mining.

During the mining of Knowledge graph, the following conversion takes place

- Entities to Nodes
- Relationships to Links

Since the Knowledge Graph is essentially a Graph, it must consist of Nodes interlinked with each other. Here, each node signifies an entity which has its own definition, existence and information. The precedent for conversion of entities to nodes require the entity must have a proper denotation of its own. It must hold out its true meaning even if the links relating to it other nodes are severed. In other terms, each node in a knowledge graph can be viewed as an endpoint to other knowledge source where it can hold its metadata.

Links in Knowledge Graph denotes how the nodes are linked with each other and what relation the links holds between them.

TABLE 2 – Construction of Nodes and Links via Conversion

Algorithm: Conversion of Entities and Relations
Input: List of related entities
Output: A Knowledge Graph

```
for each entity in entities
        if validate(entity) = TRUE                //check if Wikipedia article exists
        THEN
                create Node(entity)
        end if
end for
for
for each entity in entities
        if relation present = TRUE
        THEN
                create Link(entitiy#1, entity#2)
        end if
```

Expansion of Knowledge

Once we have a knowledge graph constructed, any kind of update in information can be done in similar manner the way knowledge graph was initially created. However, since the discovery of entities and relations were already done in the initial phase, a new iteration over the same textual content would be futile. The knowledge can only be expanded if we have textual content from other sources which makes the complete process pointless. To expand Knowledge, we now take the help of other knowledge bases which can give us related knowledge already in a structured format so that we can dissemble and reassemble it to our knowledge base. We term this knowledge gained as *external knowledge* or *knowledge from external sources* as the knowledge gained is not implicit to our domain. The entire process of updating knowledge should take care that existing knowledge does not get disturbed. This is necessary because the existing knowledge has been extracted from target source whereas derived knowledge comes from other source of information. If we modify the initial relations, we in turn migrate our knowledge base to other knowledge base and lose out originality. Thus, in order to ensure preservation of authenticity it is necessary to merge the existing knowledge with newly found knowledge such that overlapping is taken care of.

Wikipedia Category Graph

Wikipedia is the most trivial source of knowledge which is constantly updated while taking care of the old information as well. Since our domain deals with highly updated content, as NEWS is usually the recent-most form of information, we must heavily rely on knowledge bases such as Wikipedia as it will not only help in gaining external knowledge but also ensuring no misinformation is relayed into existing knowledge.

Wikipedia articles are usually linked to each other which provides easier access for navigation on the platform. This is possible because Wikipedia categorizes to classify each article into one or more classes. Each article in Wikipedia acts as an entity and the hierarchy of these links between webpages can be used to generate all related contents in terms of list of articles.

When the hierarchy of any entity is drawn from its list of sub-categories, the resulting structure is called Wikipedia Category Graph. The hierarchy is not limited to sub-categories. Each sub-category can then be considered as a distinct category to fetch sub-categories of sub-categories. This continuous process of retrieval of information never stops where each intermediate structure, consisting of entities and its subcategories, is a Wikipedia Category Graph.

We’ve used Wikipedia Category Graph to expand knowledge by continuously exploring sub-categories of each and every node in knowledge graph. While doing so, we’ve taken care about the clash of initial existing relations with new ones. A point to note here is some new relations might get formed between entities which originally never had any links between them. Even though the nodes were already present, the relationship between them is new and considered as a gain in knowledge. All these expansion of knowledge via retrieval of sub-categories must be atomic as if it is not, it will lead the growth to an inconsistent state where partial or incomplete expansion of nodes take place.

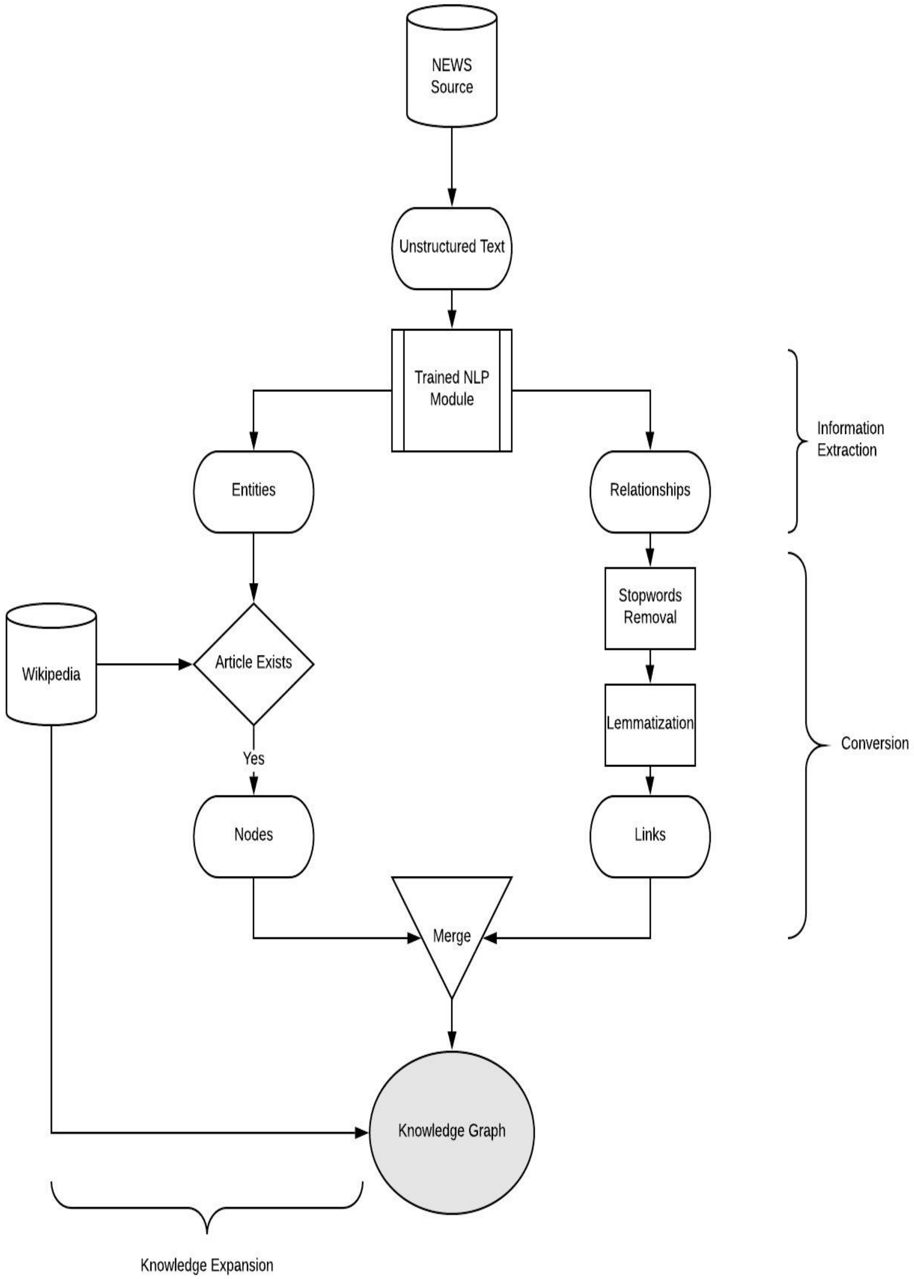

Figure 1 - Overview of Model

Let us consider an example to convey the complete description of our work. Below is a statement from a recent NEWS headline.

"Prime-Minister Modi took a swipe at the Congress at an election rally in Maharashtra citing raids by the Income Tax officials in Madhya-Pradesh"

The first phase involves detection of entities followed by determining of relations among the entities. The entities extracted are underlined whereas the relation derived are highlighted.

"Prime Minister Modi took a swipe at the Congress at an election rally in Maharashtra citing raids by the Income Tax officials in Madhya Pradesh"

After successfully extracting entities and relations, they must be converted into nodes and links respectively as shown. To determine which entity will be converted to a node or not, simply check whether it exists as a Wikipedia article or not. This will also resolve the problem to handle those entities which do not have any sub-categories.

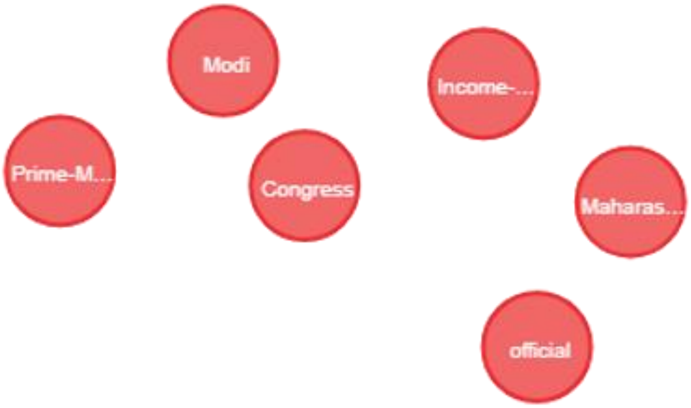

Figure 2 – Entities to Nodes

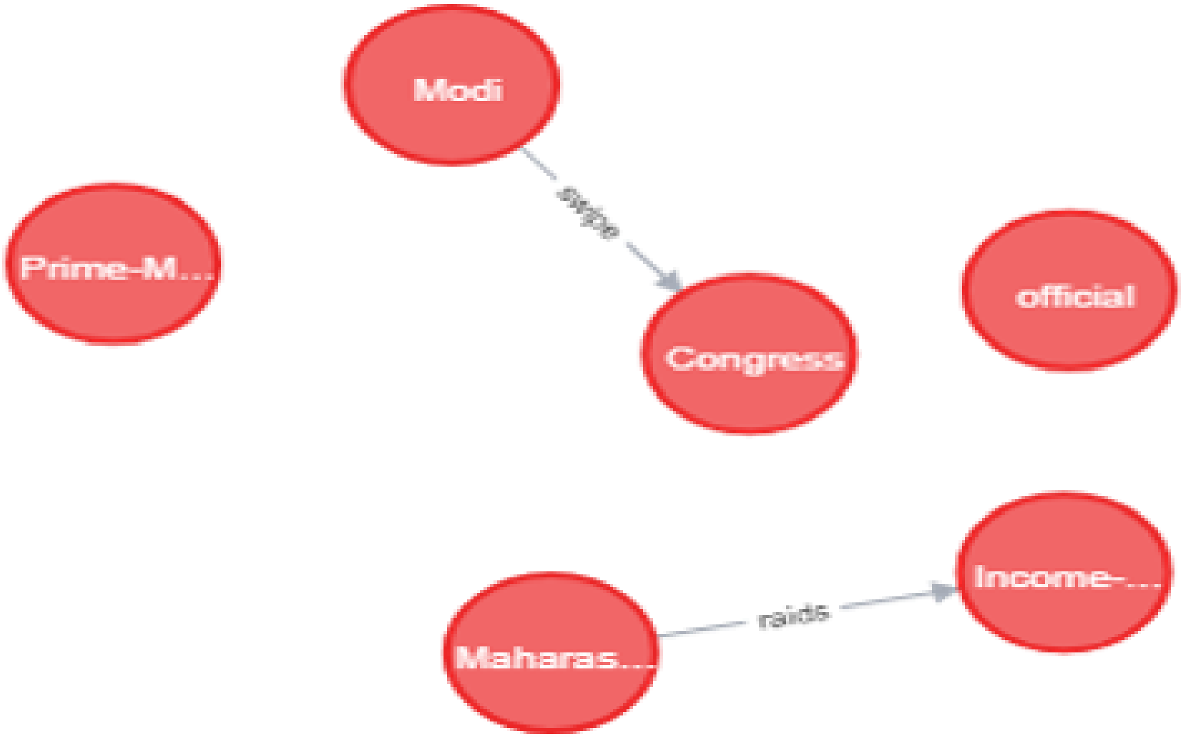

Figure 3 – Relations to Links

Once the relations and entities have been successfully converted to nodes and links, each node can be viewed as source of information in other knowledge base. A WCG of each node is created as shown below. Note that how the original relation remains intact despite addition of new entitles and relations.

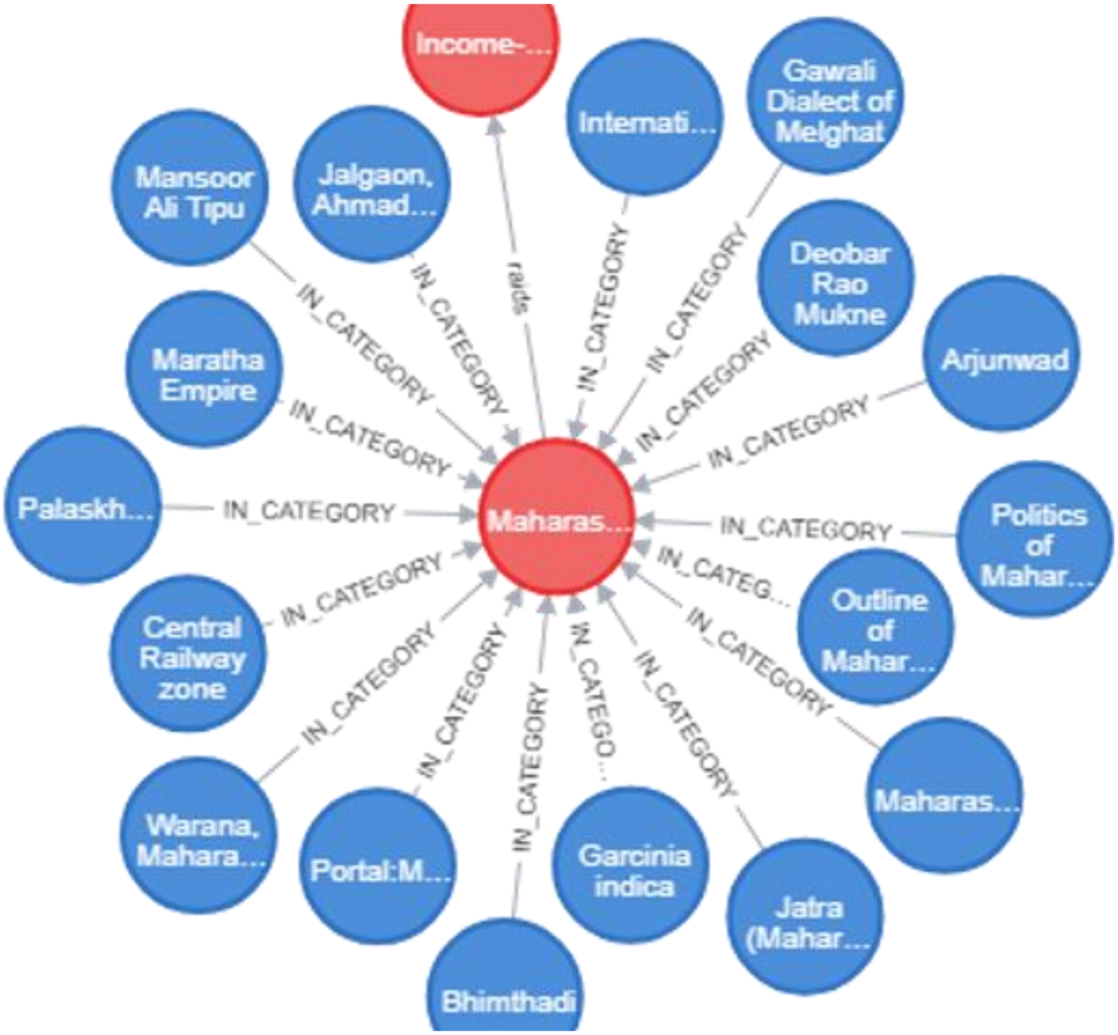

Figure 4 – Expansion of each Node – WCG of an entity

Over the complete exploration of each and every node and sub-category the resulting structure formed is a knowledge graph. The nodes in *red* color are the entities extracted initially from textual source. The nodes in *green* are sub-categories of the initial nodes i.e. red nodes whereas nodes in *pink* are sub-categories of sub-categories.

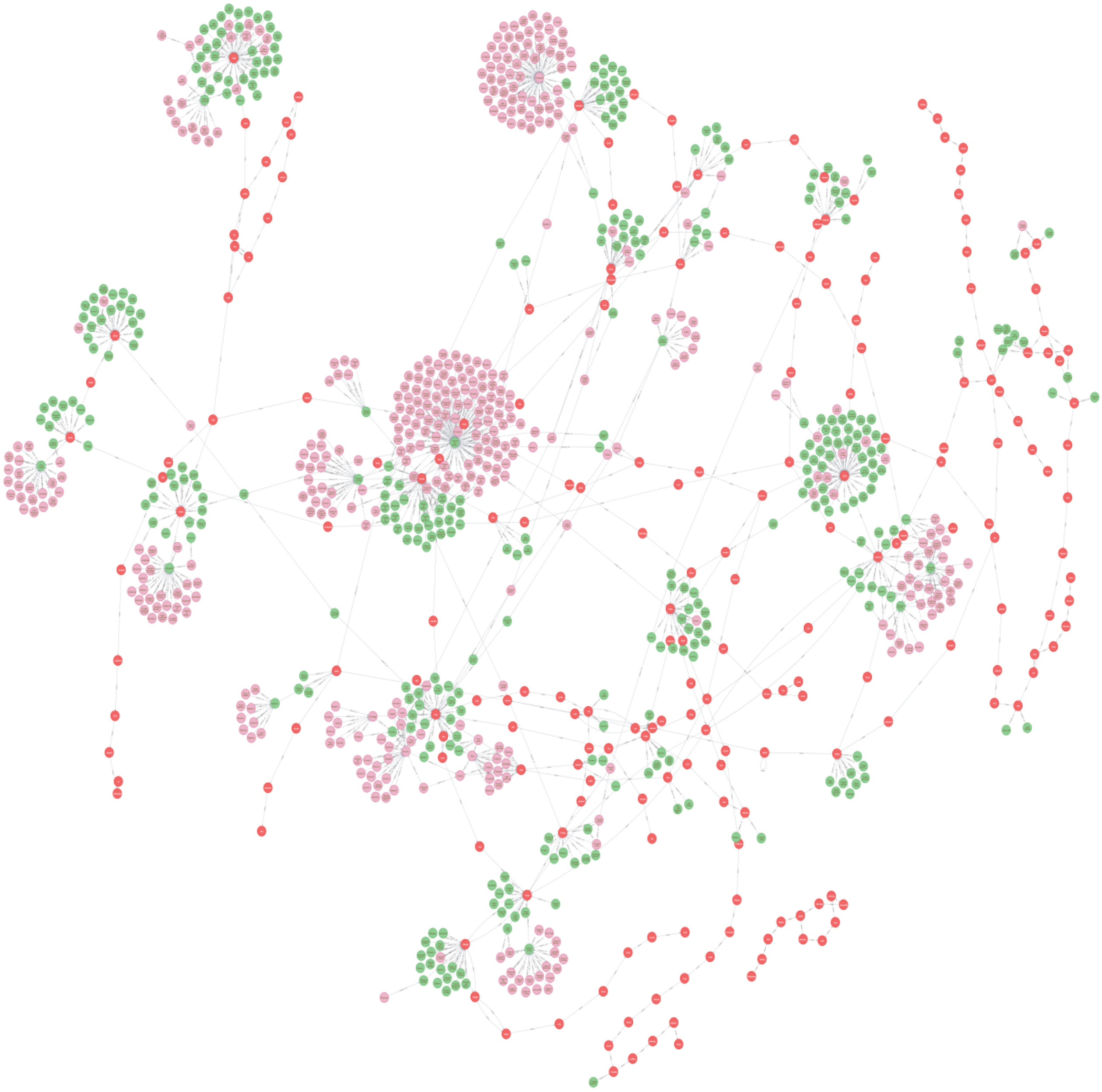

Figure 5 – A Knowledge Graph

CHAPTER 4
RESULTS

There have been multiple works where knowledge from unstructured text were forged to schemas. The addition of knowledge to a knowledge base from other sources of information is also not a new concept. However, to achieve both this task has not been a much-explored area.

Since our work was primarily focused on the detection of entities, we needed a highly accurate trained model to identify entities. A supervised approach was followed to train model. The choice of training dataset had to include sources which came from NEWS sources and blogs. For this purpose, the *corpora* chosen was OntoNotes[7] and multi-task CNN method was followed to correctly predict entities.

TABLE 3 – Named Entity Recognition Model

Metric	**Percentage**
F-score	85.86
Precision	86.33
Recall	85.39

We used 15 NEWS headlines daily to create a new Knowledge Graph every day for a tenure of two weeks. The Knowledge graph constructed from initial sources of information had an average of 38 entities and 27 relations. The expansion of Knowledge graphs was done in two-fold i.e. a Wikipedia Category Graph of level 2 was drawn where level number denotes up to how many levels of sub-categories were the nodes explored. On expanding on first level the increase in number of nodes was 343 and relations were 289. Considering the addition of each node as an increase in knowledge the percentage gain in knowledge was roughly 800-805%. On further expansion the number of nodes rose close to 5438. This kind of increase in knowledge cannot be measured as it may hold multiple redundant information.

For e.g. the expansion of WCG of the node *"India"* after 2 levels will result in formation of node *"Asia"*. If the exploration is further continued it may result in formation of nodes which are not of interest.

TABLE 4 – Knowledge Increase in each phase

Level No	**No. Of Entities**	**No. Of Relations**	**Knowledge Increase (%)**
0	38	27	100
1	343	289	802.5
2	5438	4471	NA

Initial Knowledge increase is considered to be hundred percent because no knowledge in structured format was existing initially.

Observatory analysis revealed the information required was more than necessary at this level and further iteration of WCG will result in useless or unintended information. For e.g. "India" when explored further leads the creation of node, "Asia" which on further expansion shifts the domain altogether to different countries, thus changing the domain of interest.

CHAPTER 5
CONCLUSION & FUTURE WORK

We’ve presented a technique to gather related information for any kind of knowledge presented in textual source. Though we primarily worked with NEWS headlines, our work can easily be generalized or specialized by the selection of parser or re training the model. We believe our work serves as an intermediate approach to multiple applications such as

- Query Resolve Systems – QA Systems are generally famous for using a knowledge base at their backend. We believe a Chatbot or similar sort of application can be developed to query our Graph for information extraction.
- Data Source – Google's own Knowledge Graph is used to improve Google Search Engine capabilities. DBpedia is another example where data is freely available for different use case. Our work can be used to create a knowledge graph for custom-made purposes
- Data Visualization – We've tried to mimic *InfraNodus* which is a paid tool to visualize the text as semantic network. We believe with the expansion of information, Knowledge Graph can be used to generate insight into data.

We believe our work is more of an intermediate approach rather than an immediate approach for any kind of application.

REFERENCES

1. Rusu, Delia, et al. "Triplet extraction from sentences." *Proceedings of the 10th International Multiconference" Information Society-IS*. 2007.
2. Exner, Peter, and Pierre Nugues. "Entity extraction: From unstructured text to dbpedia rdf triples." *The Web of Linked Entities Workshop (WoLE 2012)*. CEUR, 2012.
3. *Kundan Kumar* & *Siddarth Manocha "Construction of Knowledge Graph from Unstructured Text" IIT Kanpur, 2017*
4. Gangemi, Aldo, et al. "Semantic web machine reading with FRED." *Semantic Web* 8.6 (2017): 873-893.
5. Reuters, Thomson. "OpenCalais." *Retrieved June* 16 (2008).
6. Auer, Sören, et al. "Dbpedia: A nucleus for a web of open data." *The semantic web*. Springer, Berlin, Heidelberg, 2007. 722-735.
7. Hovy, Eduard, et al. "OntoNotes: The 90% Solution." *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*. 2006.