Q① Summarization

Ⓐ Extractive summarization - involves pulling keyphrases from the source & combining them to make a summary. Extraction is made accd. to defined metric without making changes to the text. The summary can be grammatically incorrect.

- eg. Joseph & Mary rode on a donkey to attend the annual event in Jerusalem. In the city, Mary gave birth to a child named Jesus.
- Summary: Joseph & Mary attend event. Jerusalem. Mary birth Jesus.

Ⓑ Abstractive summarization - paraphrasing & shortening parts of the source document. When abstraction is applied for text summarization in DL problems, it can overcome the grammar inconsistencies. The algorithms relay the most useful information from the text. While abstractive summarization is better than extractive, developing its algorithms are more difficult.

- Abstractive example of same sentence - Joseph & Mary came to Jersulaem where Jesus was born.

Ⓐ - Extractive Summarization Algorithm.

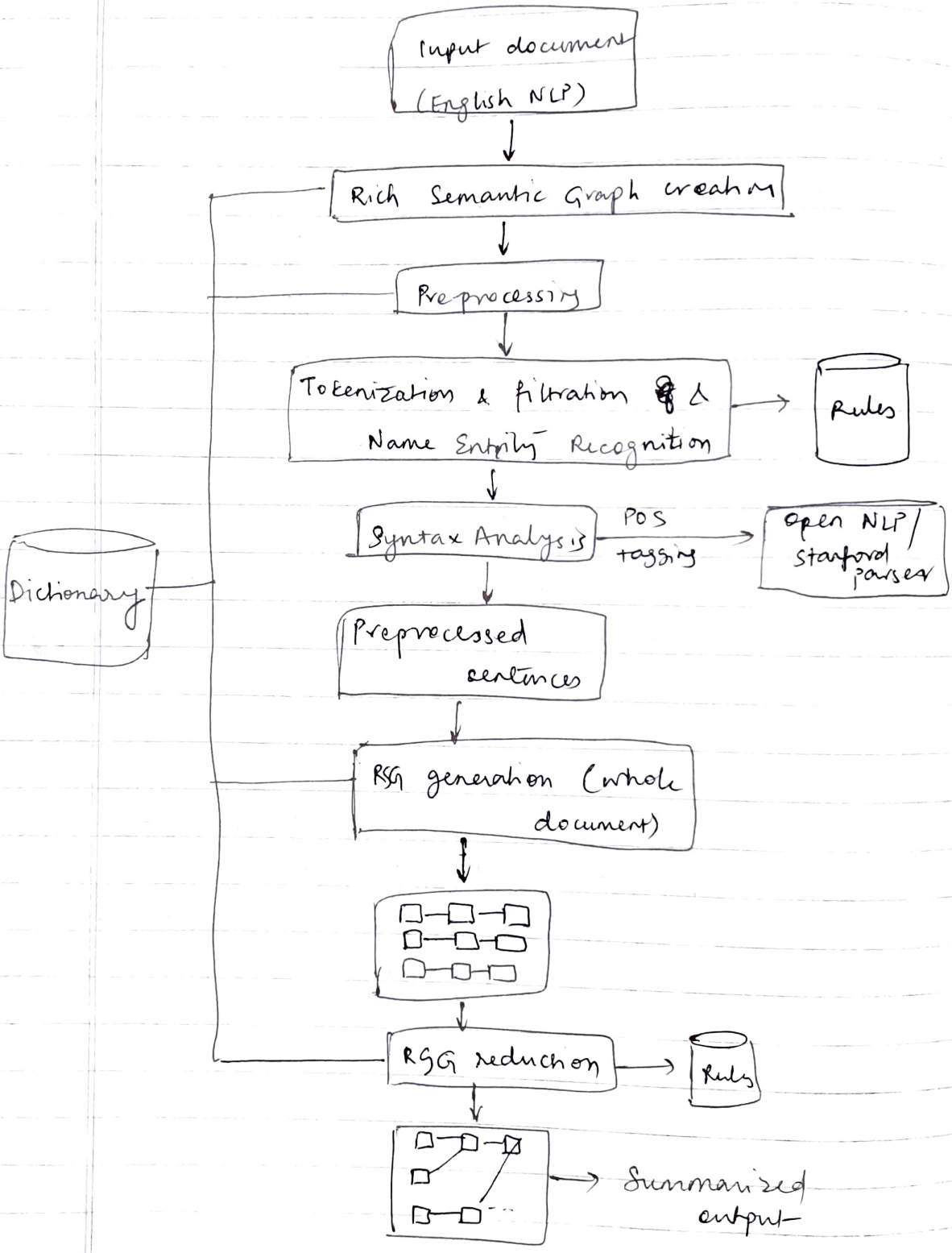
1. Introduce a method to extract the merited keyphrases from the source document. For example, POS tagging, word sequences, or other linguistic patterns to identify keyphrases.
2. Gathered text documents with positively labeled keyphrases should be compatible with stipulated extraction technique. To increase accuracy, negatively labeled keyphrases can also be created.
3. Train a binary machine learning classifier to make the text summarization. Some of the features can be: ⓐ length of keyphrase, ⓑ freq. of keyphrase ⓒ most recurring word in the keyphrase, ⓓ no. of characters in keyphrase.
4. Finally, in the test phrase, create all the keyphrase words & sentences & carry out classification.

(5) Example of Abstractive Summarization for MARATHI documents

- The idea is to summarize an input Marathi document by creating semantic graph called rich semantic graph (RSG) for the original document, reducing the generated semantic graph, & further generating the final abstract summary.
- The approach consists of the following phases:

1. Marathi text document as an input
2. RSG creation phase: In RSG creation, analysis of text is done, finds the sentences & produces the tokens for the complete doc. For every word it creates POS tags & detects the words into predefined categories such as names, locations, etc. After this it generates the graph for every sentence & concatenates RISE semantic subgraphs. These are mixed to show whole doc correctly.
3. RSG reduction phase: Target is to reduce the RSG of the text to more reduced graph. Rules such as merging, consolidation or deletion are applied to reduce the graph.
4. Summary generation from reduced RSG: Targets to obtain the abstractive summary from reduced RSG. To do this, domain ontology is accessed; it has the data required in the same domain of RSG & to obtain the final output. Also, the WordNet ontology is used to obtain multiple texts according to the word synonyms. These are ranked & the highest ranked text is selected.

- eg. single text document in Marathi



Q.2 Conditional Random Fields

- CRFs are probabalistic graphical models for sequential or structured data. They are used to perform classification taking into account context delivered by the sequence. We make structured predictions, where segments are assumed to be related with each other.
By doing so, important contextual information which would be lost in individual classifications, can be given to the model. For eg, words in a sentence are grammatically connected. After an adj. it is more likely to find a noun than a verb. This can be used to label the noun in sentences.
- Construction of a CRF for Pos tagging : Let x be the input sentence of length N : $x = (x_1, x_2, \ldots, x_n)$. Let x_i be the word at position i, let y be the label vector of the sentence & y_i the label of word x_i.
- First part of the problem deals with feature extraction. For this, a set of m feature functions f_i are defined. Each is applied to every word x_i in the sentence. We write $f_i(y, x, i)$ to denote the dependency of the function on vector x & y & the application at position i. Next, the model is trained in order to learn the weights w_j on them. Next, weighted features are added up across all words & functions. Outputs are scores s for each vector y.

$$s(y|x) = \sum_{j=1}^{m} \sum_{i=1}^{N} w_j f_i(y, x, i)$$

Finally scores are transformed into probabalities

$$P(y|x) = \frac{e^{s(y|x)}}{\sum_{y'} e^{s(y'|x)}}$$

Since the feature values can depend on not just the input x but also on output y, we can make coherent predictions.

If each label was predicted independently, predictions might often not make sense as a whole. However, because of the same reason, this makes the training & prediction process much more computationally intensive.

Q(3) Applications of WSD

- Word Sense Disambiguation is applied in almost every application of language technology

(A) Machine Translation or MT is the most obvious application of WSD. In MT, lexical choice for the words that have distinct translations for different senses is done by WSD. The senses in MT are represented as words in the target language. Most of the MT systems do not explicitly use WSD module.

(B) Information Retrieval (IR) may be defined as a software program that deals with the organization, storage, retrieval, & evaluation of information from document repositories. The system basically assists users in finding the information they require but it does not explicitly return the answers of the queries provided to the IR system. Like MT, current IR systems do not explicitly use WSD modules and they rely on the concept that the user would type enough context in the query to only retrieve relevant documents.

(C) Text mining - in most applications, WSD is necessary to do accurate indexing of text. For example, WSD helps intelligent gathering systems to do flagging of the correct words. For example, medical intelligent system might need flagging of "illegal drugs" rather than "medical drugs".

(D) Lexicography - WSD & lexicography can work together in loop because modern lexicography is corpus-based. With lexicography, WSD provides rough empirical sense groupings as well as statistically significant contextual indications of sens

Q.(4) Dictionary-based WSD Approach

- As the name suggests for disambiguation, these methods primarily rely on dictionaries, treasures & lexical knowledge base. They do not use corpora evidences for disambiguation. A major drawback with all other approaches of WSD in scalability. All require a considerable amount of work done to create a classifier for each ambiguous entry in the lexicon. Instead, attempts to perform large-scale disambiguation have focused on the use of machine readable dictionaries.
- In this style of approach, the dictionary provides both the means for constructing a sense tagger and target senses to be used.
- The first implementation of this is from Lesk.

In this approach, all the sense definitions of the word to be disambiguated are retrieved from the dictionary. These senses are then compared to the dictionary definitions of all the remaining words in the context. The sense with the highest overlap with these context words is chosen as the correct sense.

- The problem is that dictionary entries for the various senses of target words are relatively short, & may not provide sufficient material to create adequate classifiers. More specifically the words used in the context & their definitions must have direct overlap with the words contained in the appropriate sense definition in order to be useful.

One way to remedy this problem is to expand the list of words used in the classifier to include words related to, but not contained in their individual sense definitions.

Q.(5) What is anaphora resolution? What is anaphora? Examples.

- Anaphora Resolution which mostly appears as pronoun resolution is the problem of resolving references to earlier or later items in the discourse. These items are usually noun phrases representing objects in the real world called referents but can also be verb phrases, whole sentences or paragraphs
- Reference to an entity that has been previously introduced into the discourse is called anaphora, and the referring expression is said to be anaphoric.
- There are three types of anaphora

(A) Pronominal : This is the most common type where referent is referred by a pronoun. Example - "John found the love of his life.", where 'his' refers to 'John'.

(B) Definite Noun Phrase : The antecedent is referred by a phrase of the form " the <noun phrase> ".
Continued example - " The relationship did not last long ". where "the relationship" refers to "the love" in the sentence.

(C) Quantifier / Ordinal : The anaphor is a quantifier such as "one" or an ordinal such as "first". Continued Example - " He started a new one ", where "one" refers to the relationship.