Ontology Based Concept Similarity

# Breaking the Given Data into Individual Sentences:

The given data is stored into a text file. The text in the data is then send through PunktSentenceTokenizer in the NLTK Library. PunktSentenceTokenizer is a trained tool that separates the sentences from the text and indexes them.

# Coreference the Sentences and its objects:

# The sentences are then connected using coref (coreference tool of the Stanford CoreNLP). Coref is a trained model of the CoreNLP Database that is able to connect the objects in the sentence to its referred pronouns/reference other sentences.

E.G :-

The constitution of India postulates a society in which social, economic and legal justice is available to all on the basis that they are equal. To enforce the constitutional mandate of equality before the laws, the State has to ensure access to justice, that is to say, that opportunities for securing justice are not denial to any citizen by reason of economic or other disability. According to new concept, when a poor litigant is compelled to contest the case against a rich person, he should not only be given financial legal aid but the poor downtrodden person should also be furnished aid with advice and assistance in settlement of disputes by negotiation, conciliation, compromise, arbitration or any other means. This could be possible only when we change our outlook and re-orient our thinking process and then the theme of social justice be able to change the fate of the poor litigants.

Is Processed into:

The constitution of India postulates a society in which social , economic and legal justice is available to all on the basis that social , economic and legal justice are equal .

To enforce the constitutional mandate of equality before the laws , the State has to ensure access to justice , that is to say , that opporunities for securing justice are not denial to any citizen by reason of economic or other disability .

According to new concept , when a poor litigant is compelled to contest the case against a rich person , a poor litigant should not only be given financial legal aid but the poor downtrodden person should also be furnished aid with advice and assistance in settlement of disputes by negotiation , concilliation , compromise , arbitration or any other means .

The constitution of India could be possible only when we change we outlook and re-orient we thinking process and then the theme of social justice be able to change the fate of the poor litigants .

# Tagging each and every word in a sentence:

# Each word of every sentence is then tagged (Part of speech tagging) using the pos\_tag feature in nltk library. The pos\_tag allocated each and every word its token and its role in the sentence (e.g. Proper noun, Adjective, Object, Adverb etc).

# Selection of Keywords and Entities:

A rule is then made to select the keyword and entities from the text. The rule is Adjective followed by a noun/series of nouns to select the keywords from each sentence. Given the rule a RegexpParser was made to parse the entire Sentence and chunk the words that were following the rules together using the nltk library.

**RegexpParser:**

A grammar-based chunk parser. chunk. RegexpParser uses a set of regular expression patterns to specify the behaviour of the parser. The chunking of the text is encoded using a Chunk String, and each rule acts by modifying the chunking in the Chunk String. The rules are all implemented using regular expression matching and substitution.

A grammar contains one or more clauses in the following form:

{<DT|JJ>} *# chunk determiners and adjectives*

}<[\.VI].\*>+{ *# chink any tag beginning with V, I, or .*

<.\*>}{<DT> *# split a chunk at a determiner*

<DT|JJ>{}<NN.\*> *# merge chunk ending with det/adj*

## After getting the chunked data the chunks were extracted using regex and basic string algorithms.

# Creation of A Knowledge Graph:

A graph is generated using chunks and its dependencies on other chunks. After this , the nodes are fed into a pandas data frame and networkx library is used to create a graphs. After the graph is created matplotlib is used to show the created graph.

# Selection Of Core Learning Points Using Knowledge Graph and a trained Machine Learning Module:

|  |  |
| --- | --- |
|  | 1.What a neural net uses to optimize its cost function??? |
|  |  |
|  | Given the factors on which the output of the neural network or hence the cost function depends are the dimensions. |
|  | Now given the different values of dimension at each point of time cost function is calculated , now to find the local minima in the graph of the values plotted by varying these dimensions, we use gradient descent. |
|  |  |
|  | ## 2. What did we do??? |
|  | Now we know that the output of our program is dependent on three initial conditions ( no\_of\_connected\_nodes,length of the given topic, its frequency in the graph. |
|  |  |
|  | Now what we did was to multiply 3 factors : k1,k2,k3 to each of the value and using the value we calculate our ans. |
|  |  |
|  | Now on varying each of the values we will get different answers of the cost function --> that basically calculates the difference between the output coming from the students and the expected output. |
|  |  |
|  | Now consider a 3-d graph that plots the k1,k2,k3 constants and cost function. |
|  |  |
|  | Now in order to get the most optimum result , what we have to do is find the global minima of this graph ==> but since our sample space of the graph is very small , we can just go through the entire graph |
|  | and the find the min value , which is what we did. |
|  |  |
|  | Clearly a 3-d graphs can be iterated in O(n^3) time. |
|  |  |
|  | Putting these values in our program gets the closest results to the expected output. |
|  |  |
|  | We plan to run this algorithm on the teachers notes and obtain the value and use the obtained value to extract output from the students graph as well. |

# Selecting Base Nodes:

Utilizing the Knowledge Graph , articulation points ( articulation point : A vertice is an articulation point if removint it and edges through it disconnects the graph. ) are extracted via a funtion in articulation\_points in NetworkX library. These points are then used as base nodes over which the concept Graph is created.

# ID And Generic Name Extraction:

A Concept ( Base Node ) is searched on Wikiped. From the Wikipedia Page its WikiData ID is extracted. Utilizing the WikiData ID its WikiData Name ( Genereic Name ) is extracted . The operations are performed bu utilizing Wikipedia , Requests and BeautifulSoup Libraries.

# WikiData Relation Linking:

WikiData’s query Language SPARQL is used to connect the concept to its Ancestors ( Concepts of which given conept is a part of/ Subclass of) and Decendents ( Concepts which are part of/ Subclass of given concept ).

# Creating Concept Graph:

WikiData Relation Linking for each base nodes provdies a graph with all possible relations of Base Nodes/Concepts as per wikidata ( Relation Graph ). Shortest Relations between all Base Nodes are filtered out utilizing shortedt\_path function in NetworkX library.

From these filtered out function a new Graph is constructed containg only base nodes and nodes neccesary for relation establishment between any two base nodes. ( Creation of the Conept Graph from the Relation Graph was due to abundance of nodes in Relation Graph which made it difficult to be read by Users. )

Matplotlib is then used for plotting the obtained Graph.