

Autonomous Presentation Creator (for educational purposes only)

Project Phase II (ECD402)
Evaluation 3 (May 9th, 2021 , Monday)



PROJECT TEAM MEMBERS

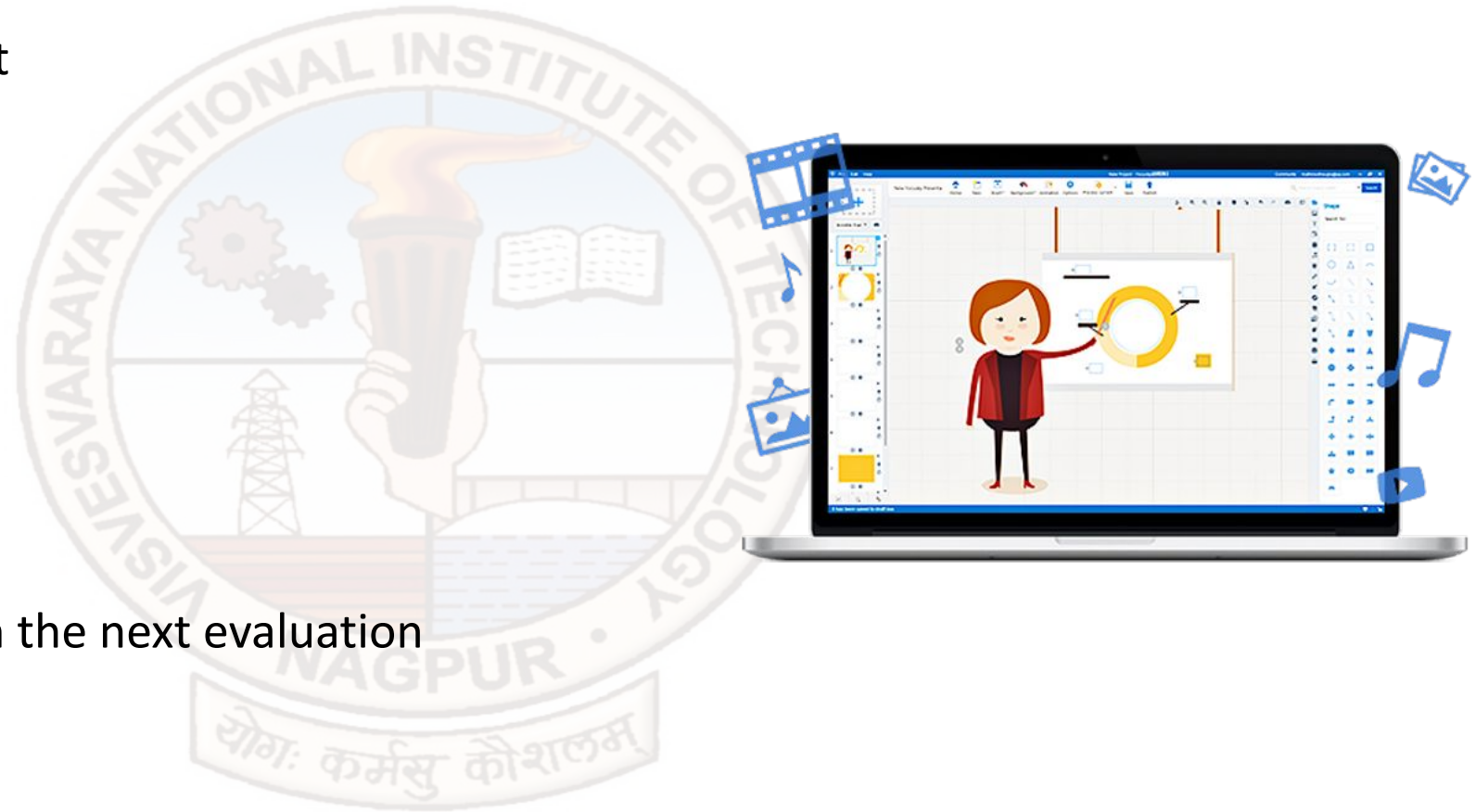
1. Pushpesh Raj (BT18ECE072)
2. Mayank Bumb (BT18ECE111)
3. Ramisetti Sai Mounish (BT18ECE131)
4. Abdul Majid Khan (BT18ECE135)

NAME OF SUPERVISOR

Dr. Anamika Singh

Contents

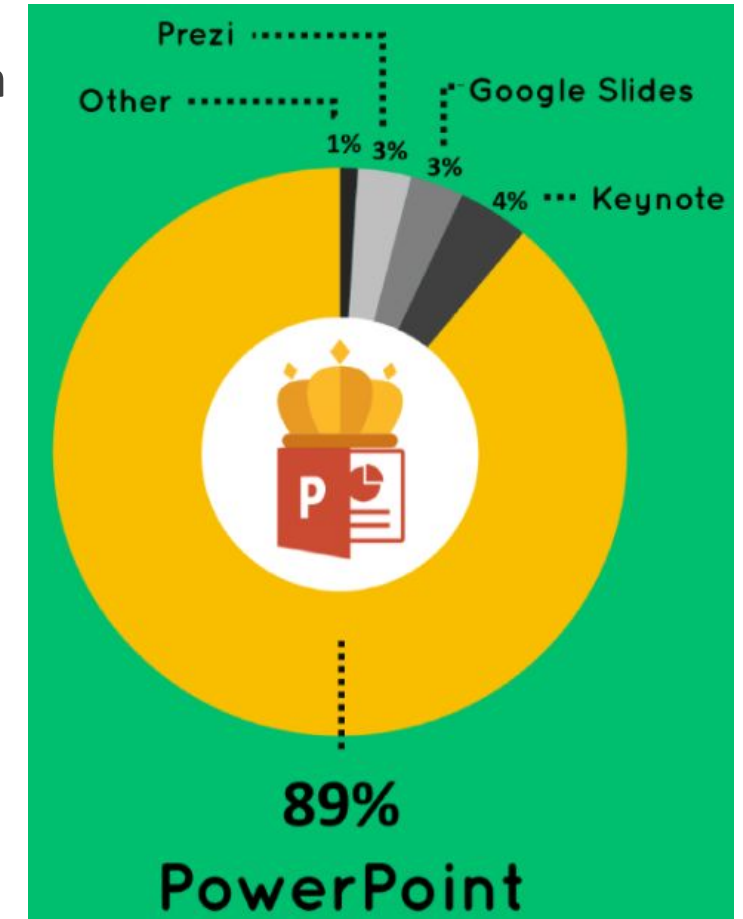
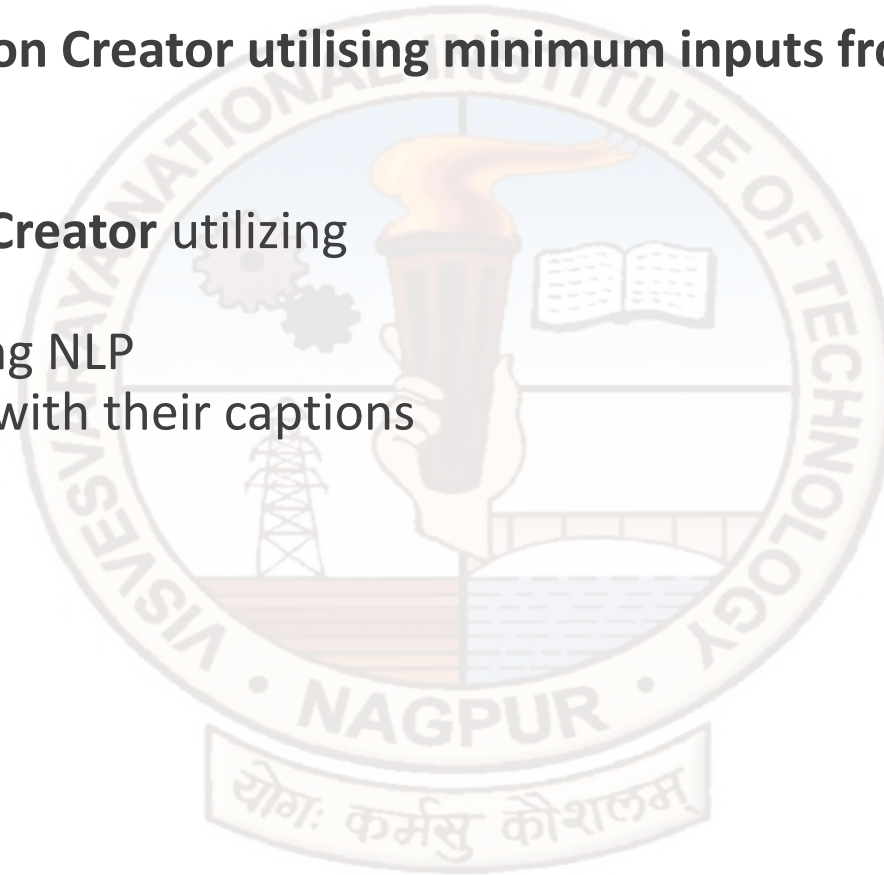
- Problem Statement
- Block diagram
- Timeline
- Literature Survey
- Web Scraping
- XLNet Model
- BART Model
- Python-pptx
- Work to be done in the next evaluation
- Conclusion



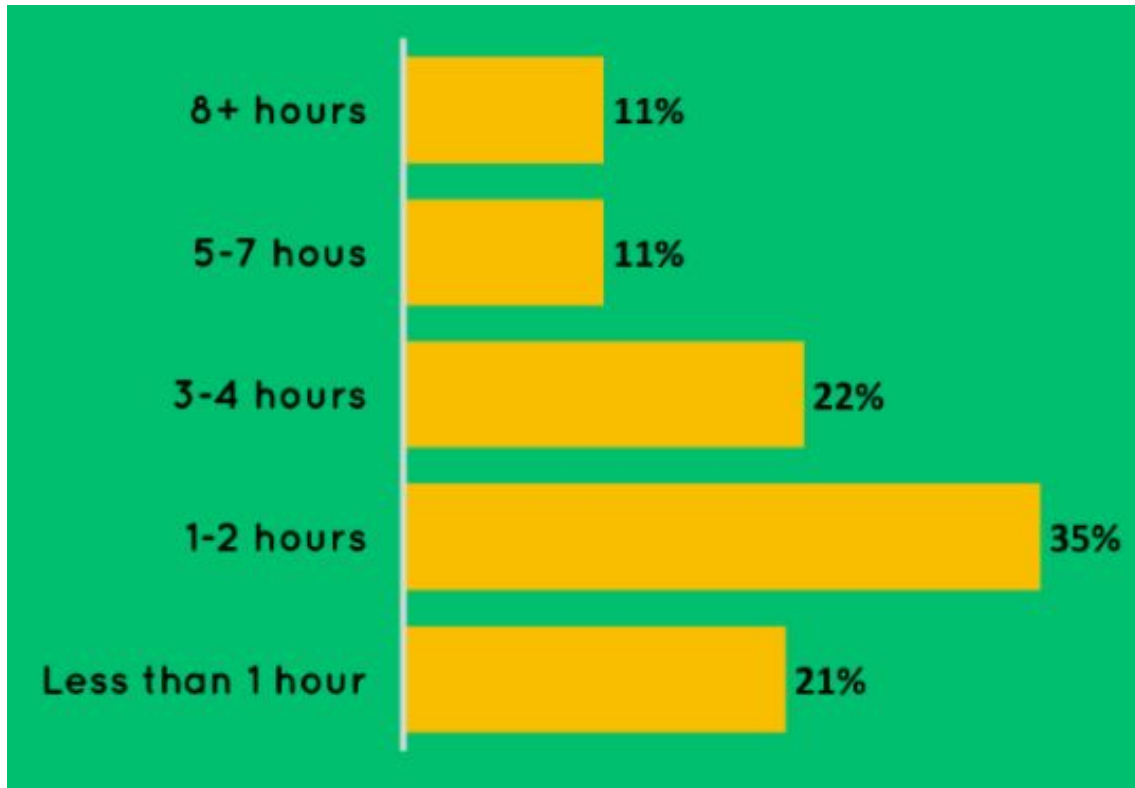
Problem Statement

An Autonomous Presentation Creator utilising minimum inputs from the user.

- **Autonomous Presentation Creator** utilizing
 - Web Scraping
 - Text summarisation using NLP
 - Image extraction along with their captions
 - Presentation Design



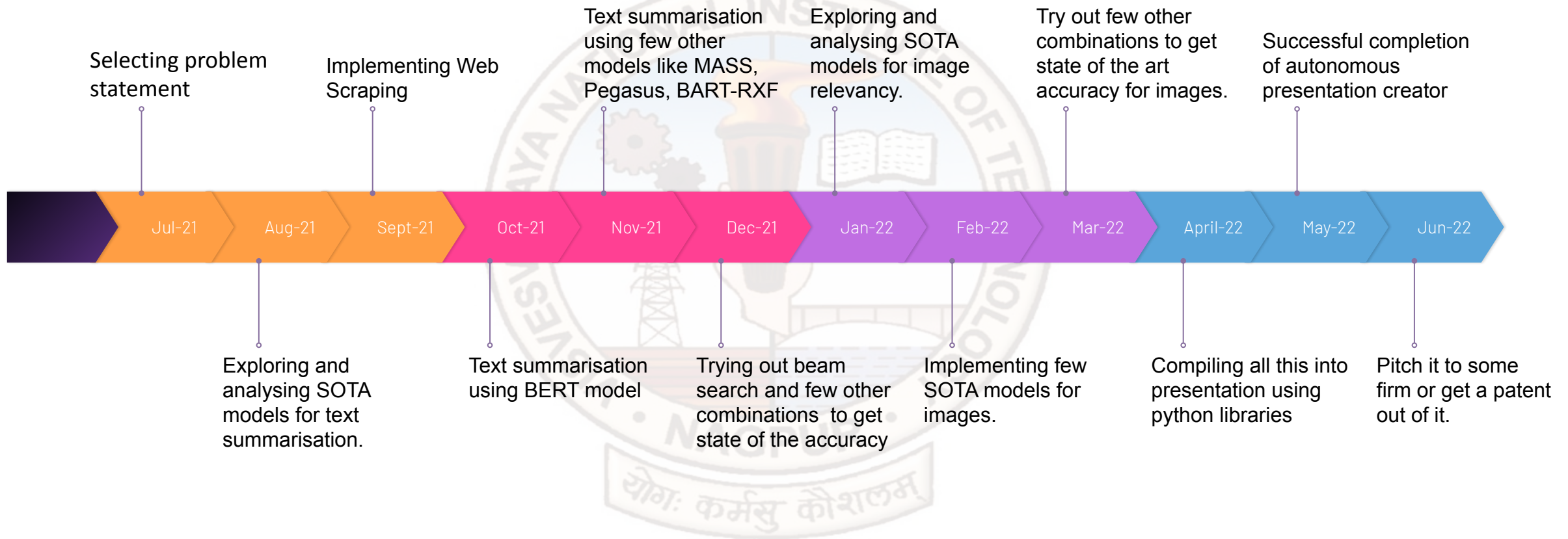
Time consumed



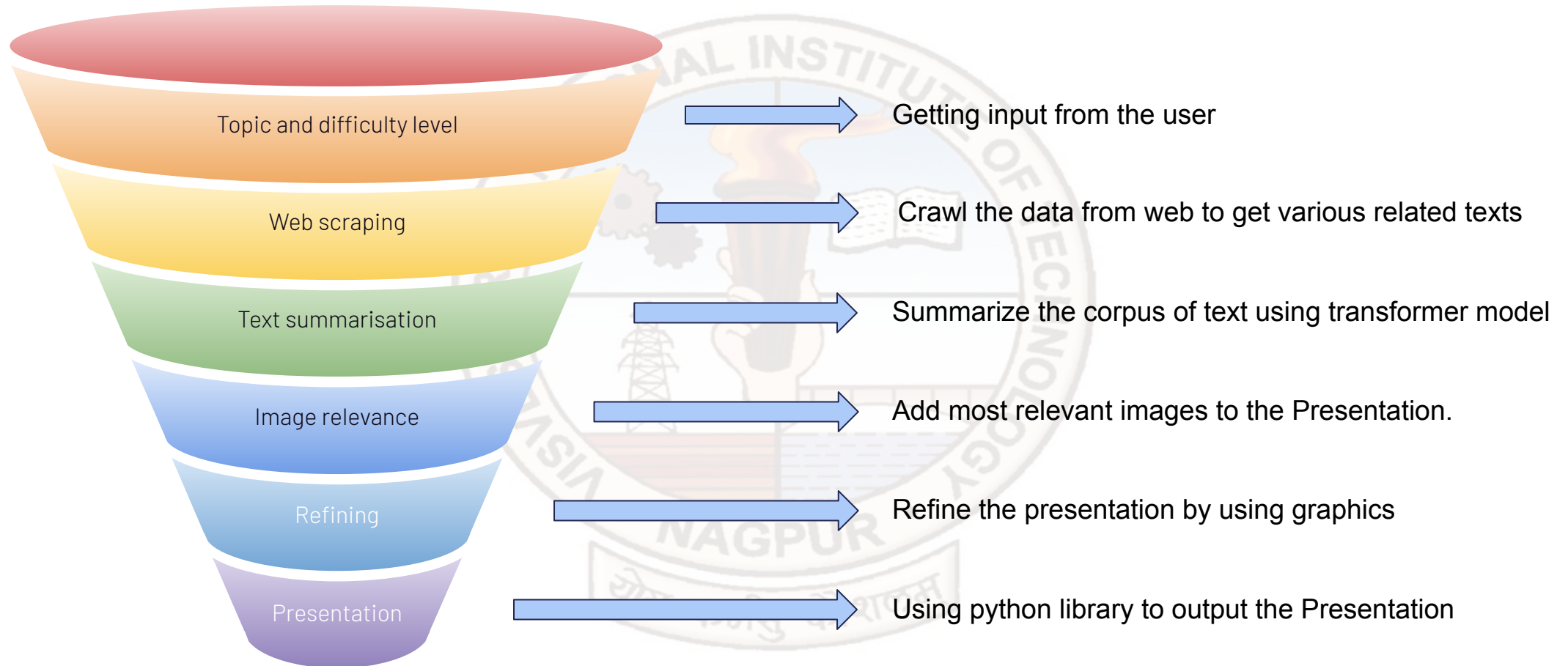
Potential Market

1	Number of users	• 500 million worldwide
2	Number of presentations created daily	• 30 million worldwide
3	Use of presentation for teachers	• 6 million teachers use it
4	Use of presentation for business and education	• 120 million worldwide
5	Estimated sale of presentations	• 100 million dollars

Timeline for the Project Completion



Block diagram



Web Scrapping

- The Data required for making the presentation is scraped from the wikipedia sources and **urlopen** package is used to open URL(HTTP).
- **Beautiful Soup framework** in Python library is used for web scraping to pull the data from the HTML file of our corresponding web page provided by the user.



CLASSIFICATION OF DATA INTO SEGMENTS

- Topic as input is stored along with Wikipedia domain as a string.
- Using BeautifulSoup for HTML and selenium for scraping different URLs.
- Ordered tree traversal for the division of classes separately on each subtopic.
- Extracting headlines and corresponding images from that section.
- Iteration to extract paragraph from <p> tag.
- Cleaning of the string.

Final Steps involved in Scraping of Data and Images in Classified Groups

- First the input is taken as a string from the user and it is appended with Wikipedia domain name so that we can get the finalise url of the corresponding topic.
- Once we got the url we use urlopen package to open the url.
- After this beautiful soup library is used to get the html file for the url.
- Then we use soup.find_all function present in beautiful soup for ordered traversal in the HTML file.
- While traversing in html file we extract different types of heading (i.e from heading 1 to heading 3) using H1 to H3 tags.

Continued...

- The heading tags are taken individually and are divided as subtopic segments.
- Basically we are classifying the whole ppt into segments using these subtopics.
- Then for each of these segment we are extracting the paragraph as p tag and store it as a string.
- The next stage is cleaning these paragraph string as it contains several irregularities like numeric keyword for reference and black slash n as it is used in html to end the line.
- These paragraphs are main content which we process in our NLP part.

Continued...

- Along with the paragraph we extract all the URLs of images using image attributes and anchor tags.
- We use these images url and append the image along with paragraph in the final ppt whose process will be explained later in this presentation.
- Along with the images, their description is also extracted using caption tag.
- All these information of each subsegment is passed further in the form of dictionary which we use later in this project.

Image of Input for Text extraction

Nuclear fission

From Wikipedia, the free encyclopedia

Nuclear fission is a [reaction](#) in which the [nucleus](#) of an [atom](#) splits into two or more smaller [nuclei](#). The fission process often produces [gamma photons](#), and releases a very large amount of [energy](#) even by the energetic standards of [radioactive decay](#).

Nuclear fission of heavy elements was discovered on 17 December 1938, by German chemist [Otto Hahn](#) and his assistant [Fritz Strassmann](#) in cooperation with Austrian-Swedish physicist [Lise Meitner](#). Hahn understood that a "burst" of the atomic nuclei had occurred.^{[1][2]} Meitner explained it theoretically in January 1939 along with her nephew [Otto Robert Frisch](#). Frisch named the process by analogy with [biological fission](#) of living cells. For heavy [nuclides](#), it is an [exothermic reaction](#) which can release large amounts of [energy](#) both as [electromagnetic radiation](#) and as [kinetic energy](#) of the fragments ([heating](#) the bulk material where fission takes place). Like [nuclear fusion](#), for fission to produce energy, the total [binding energy](#) of the resulting elements must be greater than that of the starting element.

Fission is a form of [nuclear transmutation](#) because the resulting fragments (or daughter atoms) are not the same [element](#) as the original parent atom. The two (or more) nuclei produced are most often of comparable but slightly different sizes, typically with a mass ratio of products of about 3 to 2, for common [fissile isotopes](#).^{[3][4]} Most fissions are binary fissions (producing two charged fragments), but occasionally (2 to 4 times per 1000 events), *three* positively charged fragments are produced, in a [ternary fission](#). The smallest of these fragments in ternary processes ranges in size from a proton to an [argon](#) nucleus.

Output after classification of extracted text

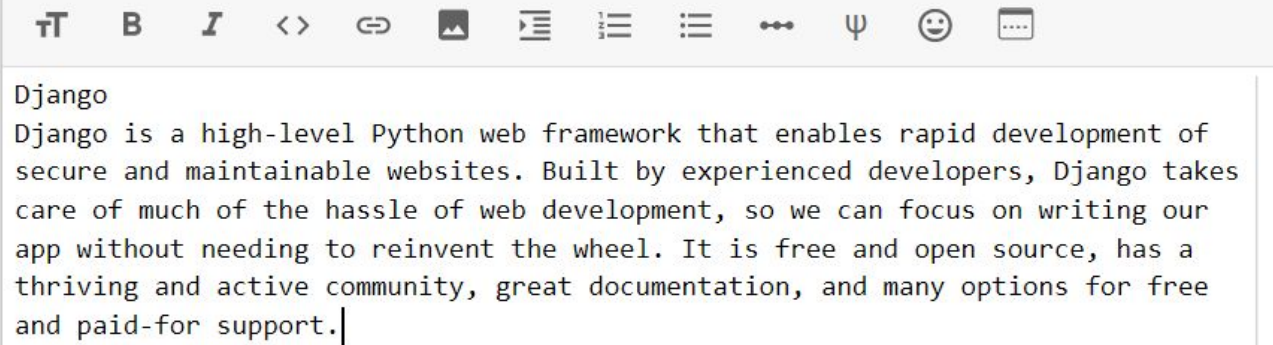
Introduction--> Nuclear fission is a reaction in which the nucleus of an atom splits into two or more smaller nuclei. The fission process often produces gamma photons, and releases a very large amount of energy even by the energetic standards of radioactive decay. None Nuclear fission of heavy elements was discovered on 17 December 1938, by German chemist Otto Hahn and his assistant Fritz Strassmann in cooperation with Austrian-Swedish physicist Lise Meitner. Hahn understood that a "burst" of the atomic nuclei had occurred. Meitner explained it theoretically in January 1939 along with her nephew Otto Robert Frisch. Frisch named the process by analogy with biological fission of living cells. For heavy nuclides, it is an exothermic reaction which can release large amounts of energy both as electromagnetic radiation and as kinetic energy of the fragments (heating the bulk material where fission takes place).

Physical overview--> Nuclear fission can occur without neutron bombardment as a type of radioactive decay. This type of fission (called spontaneous fission) is rare except in a few heavy isotopes. None In engineered nuclear devices, essentially all nuclear fission occurs as a "nuclear reaction" – a bombardment-driven process that results from the collision of two subatomic particles. In nuclear reactions, a subatomic particle collides with an atomic nucleus and causes changes to it. Nuclear reactions are thus driven by the mechanics of bombardment, not by the relatively constant exponential decay and half-life characteristic of spontaneous radioactive processes.

Extraction from Pdf (given by user)

Django

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so we can focus on writing our app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support.

A screenshot of a text editor window. The title bar is light gray and contains icons for font size, bold, italic, code, link, image, list, table, indent, undo, redo, smiley, and a menu. The text area is white and contains the following text: "Django\nDjango is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so we can focus on writing our app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support." The text is in a monospaced font.

Django
Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so we can focus on writing our app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support.

Screenshot of output of text extracted from pdf

Image Scraping

- The images in the Wikipedia page can be extracted using BeautifulSoup.
- A function is built using `urlopen()`, `BeautifulSoup()`, and `soup.find_all()` to get urls of images in a list.
- The 'soup' in `soup.find_all` is obtained using BeautifulSoup.
- `Soup.find_all` has many parameters(search terms) of which the most important ones are Tag name and `Class_`.
- As far as images are concerned the search terms are "a" and `class_="image"`.
- From the urls obtained from `find_all` method, only those that contain 'src' attribute are to be appended in the URL list.

Image Scraping (Along with Caption)

- From the above process we can get the image or its URL but we need the caption(image description) too!
- From the adjacent figure, “3+2 = 5 with apples, a popular choice in textbooks” is the image description and we need to extract this along with image.
- So firstly we need to collect urls of all the image descriptions in a list along with the image URL list.
- Just like extraction of image URLs, we use ‘soup.find_all’ with “div” as head and “thumbinner” as class to extract the image descriptions of all the images i.e. “div” and class_=”thumbinner” are the parameters in soup.find_all.
- Now we have a list with all the image urls and another list with all the captions.

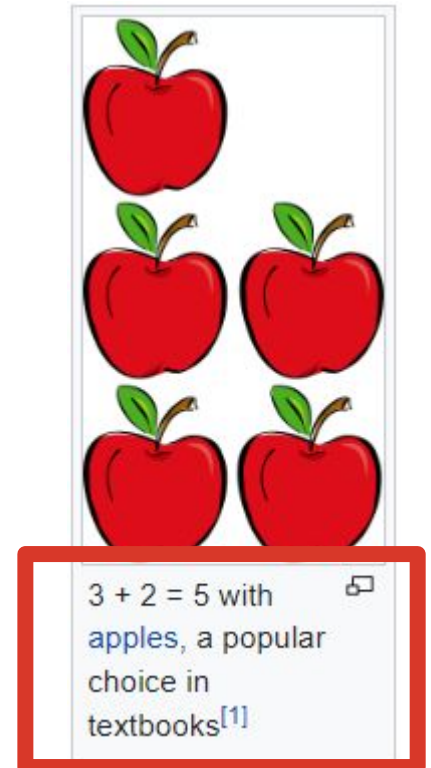


Image Scraping

(Joining image and caption)

- After getting lists of image urls and image descriptions, our job is to combine both i.e, join the corresponding image description at the bottom of the image.
- This can be done using OpenCV or PIL and NumPy.
- The image array(img) and description box array(temp) should be created and concatenated together. The height of img and temp will be same and the width of temp will be decided based on the number of characters in the caption.
- Now the text can be drawn on the concatenated image using cv2.putText. One can manually decide the font, font colour, line thickness etc. of the text.
- The images achieved after this process look like the adjacent figure.



Copyright clarification for usage of images from Wikipedia

- Most of the images on Wikipedia are non copyrighted. Very few images which cannot be replaced by free images (like company logos etc.,) are copyrighted.
- But these can be used for **non-commercial or educational purposes** by referencing the source.
- Also there is a sister project of Wikipedia called Wikimedia Commons in which all the images related to a topic are non copyrighted!!
- All the images from Wikipedia Commons are free to use for any purposes.



Category

Discussion

Category:Addition

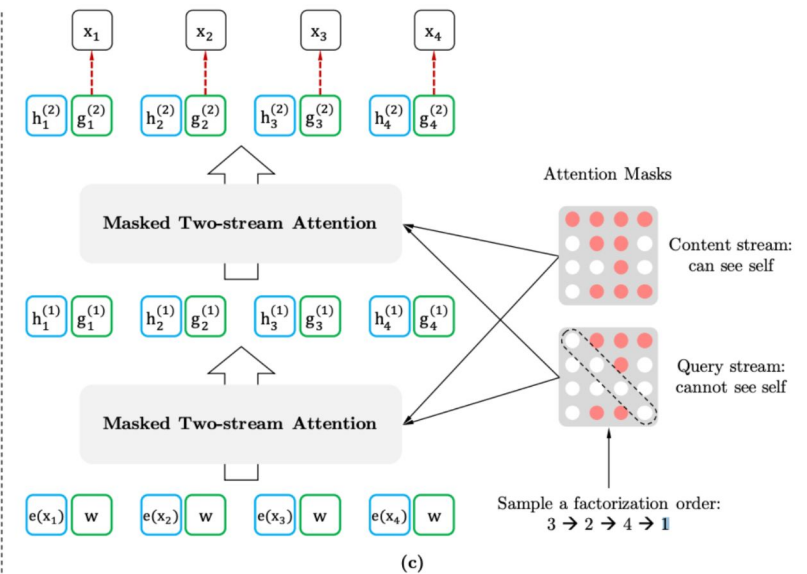
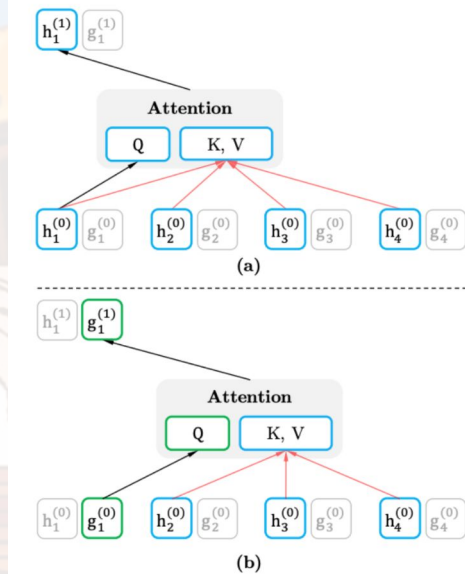
From Wikimedia Commons, the free media repository

XLNet MODEL

● 2019 Datasets :

1. Custom

- XLNet is a **generalized autoregressive model where next token** is dependent on all previous tokens.
- It integrates the idea of auto-regressive models and bi-directional context modeling, yet overcoming the disadvantages of BERT.
- BERT masks the data and tries to predict the masked data using a bi-directional context whereas XLNet uses permutation objective.



Comparison of best models

Text Classification on AG News

Leaderboard

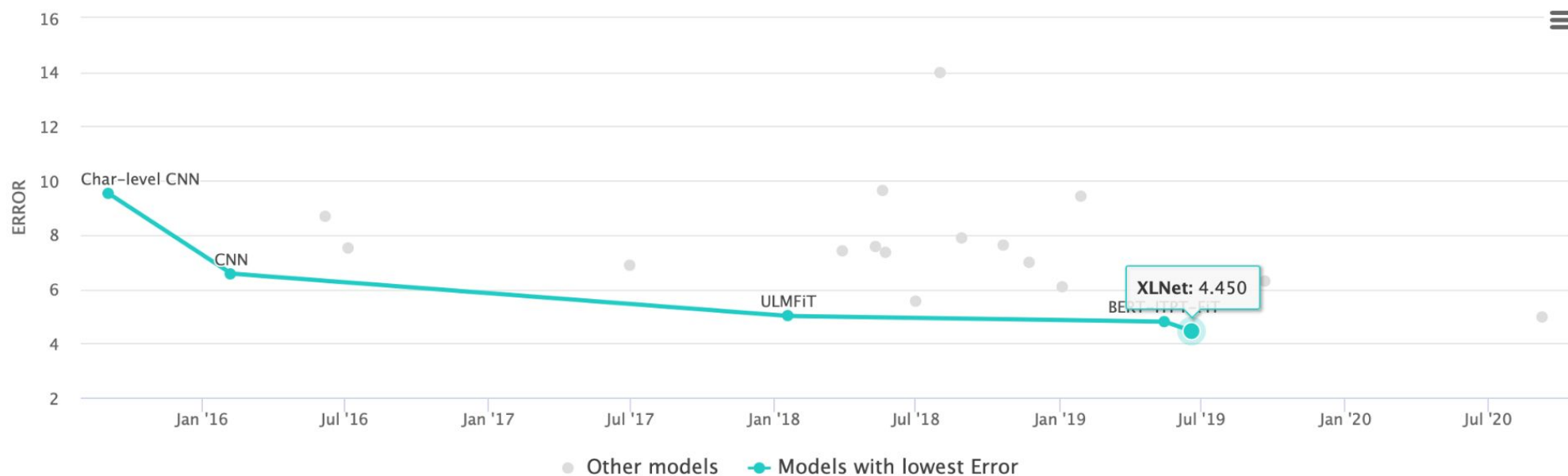
Dataset

View

Error

by

Date



Comparison and Literature Survey Conclusions



- ROUGE is a classification based dataset that assigns a score corresponding to the accuracy of a bot generated summary to that of a human written one.

Encoder ⇔ Decoder	Dataset	ROUGE Score
RNN ⇔ LSTM	DUC 2004, Gigaword Corpus	28.97
Bi-RNN ⇔ RNN	LCSTS	35.00
Transformer	Gigaword, Newsroom	38.72
Bi-LSTM ⇔ LSTM	CNN/Daily mail	39.53
Transformer	CNN/Daily mail, LCSTS	44.79

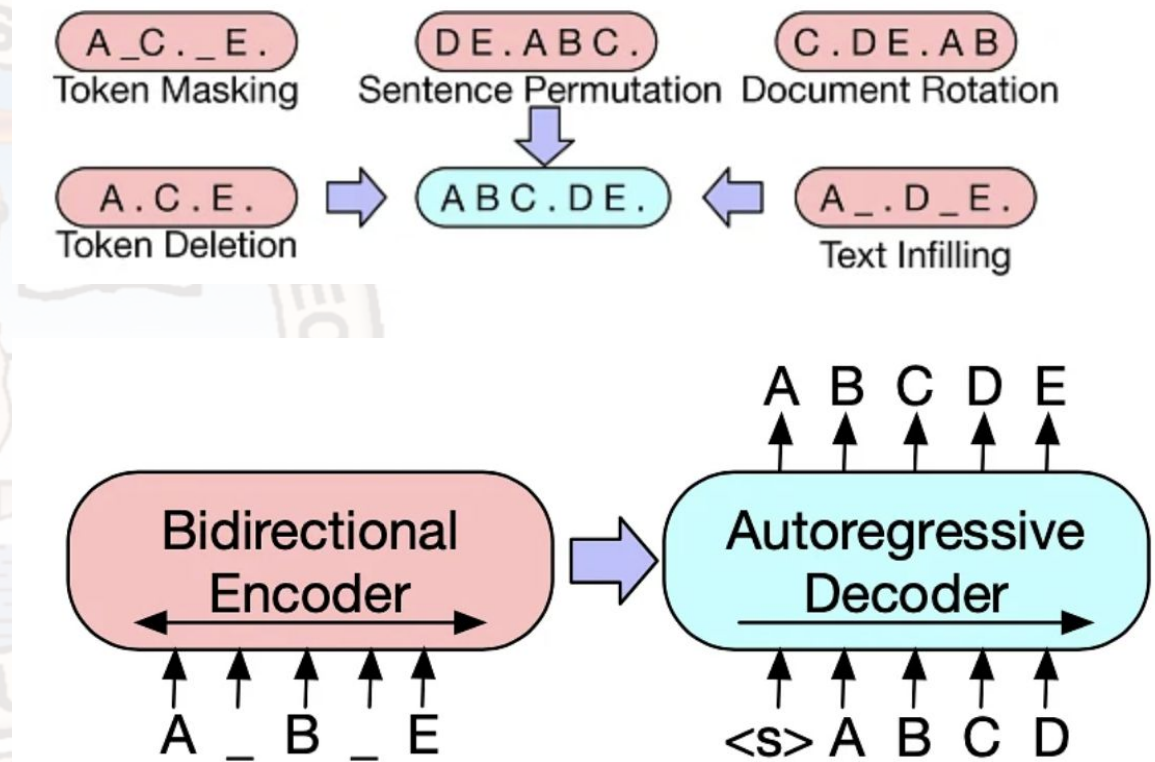
BART MODEL

● 2019 Datasets :

1. CNN news

Bidirectional and Auto-Regressive Transformer
[1][2]

- BART is a sequence-to-sequence model trained as a denoising autoencoder.
- The BART model can be fine-tuned to domain-specific datasets to develop applications
- Bidirectional encoder(BERT): It uses both the information from left and right in the text to find the best representation of its input sequence.
- Autoregressive decoder(GPT): It means that it produces output from the hidden state which is the output of the encoder. This subsequently goes into a recurrent structure that uses prediction from previous state to generate next step.



Comparison of best models

Abstractive Text Summarization on CNN / Daily Mail

Leaderboard

Dataset

View

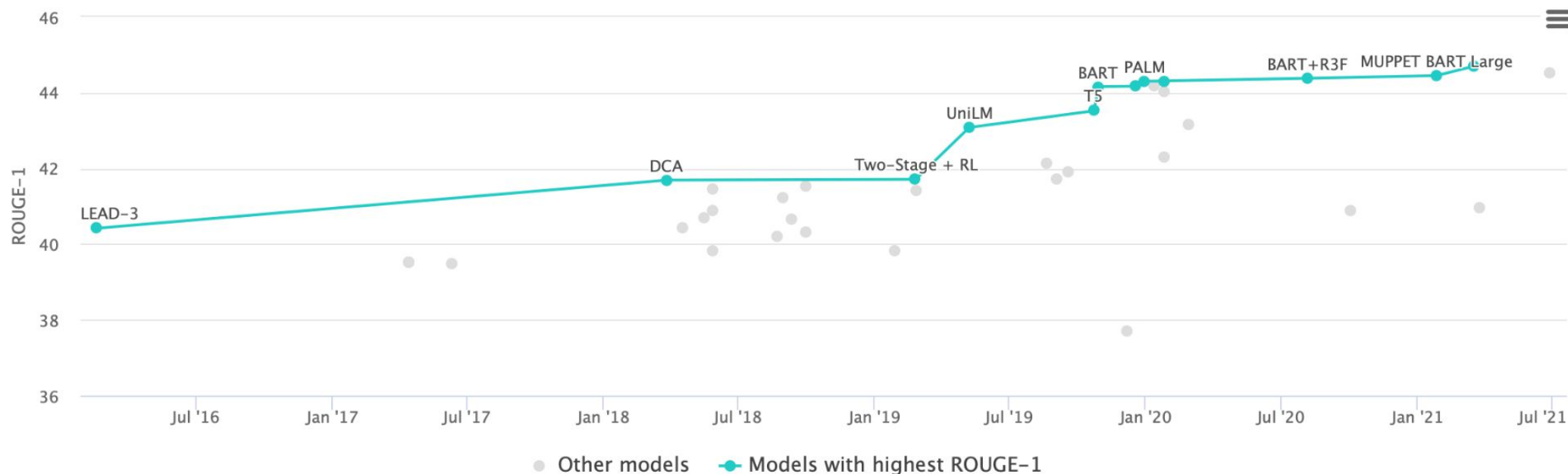
ROUGE-1

by

Date

for

All models



IMPLEMENTATION OF TEXT SUMMARIZATION MODELS

Summarization of addition from wikipedia using BART model

- The addition of two whole numbers results in the total amount or sum of those values combined.
- Addition belongs to arithmetic, a branch of mathematics.
- Performing addition is one of the simplest numerical tasks.

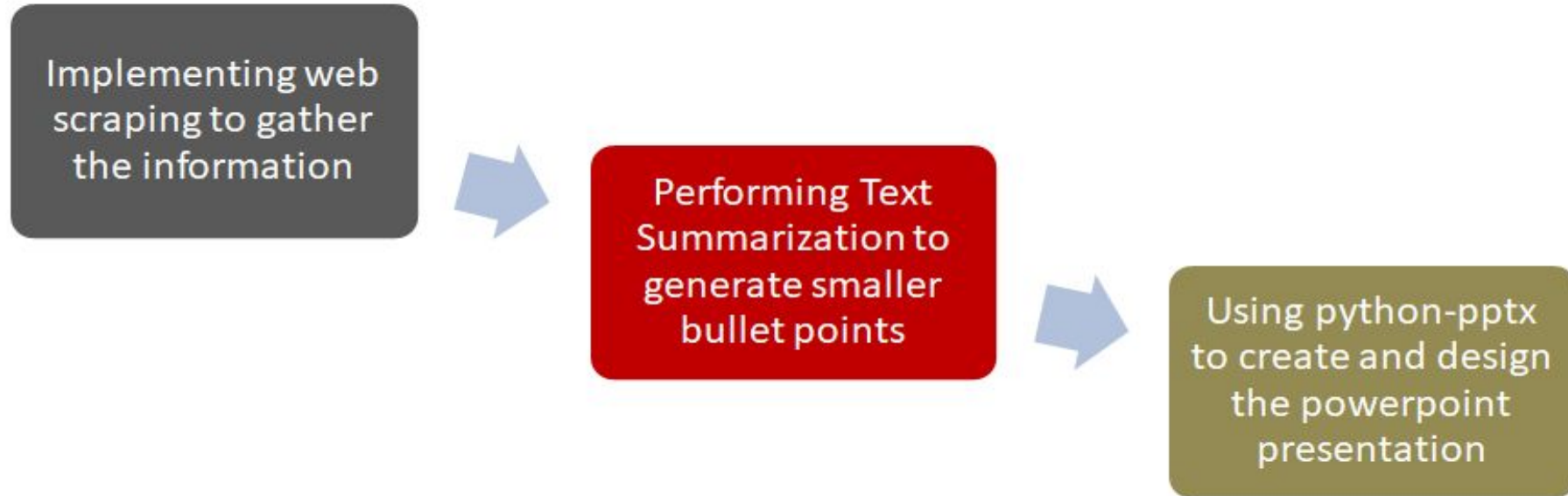
Summarization of addition from wikipedia using BERT model

- Addition (usually signified by the plus symbol $+$) is one of the four basic operations of arithmetic, the other three being subtraction, multiplication and division.
- Addition belongs to arithmetic, a branch of mathematics.
- In algebra, another area of mathematics, addition can also be performed on abstract objects such as vectors, matrices, subspaces and subgroups.

Summarization of addition from wikipedia using NLTK library

- Repeated addition of 1 is the same as counting; addition of 0 does not change a number.
- Performing addition is one of the simplest numerical tasks.
- Addition has several important properties.

Creating and Design of a PowerPoint Presentation



Introduction to Python-pptx

Early Open Source initiative to make ppt making easy using python commands

Utilizes Python Image Library and LXML packages to draw shapes and layouts

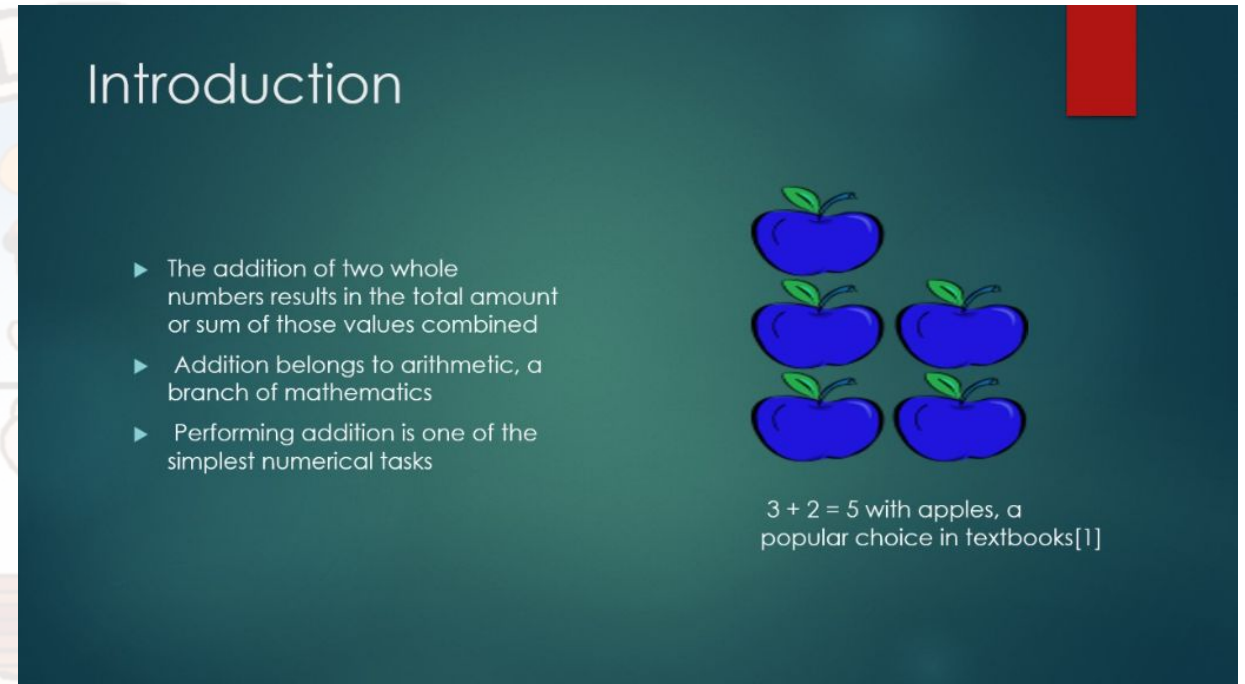
Python-pptx

We are working on a source code with several parameters and conditions taken from the user to generate a PPT.

If implemented the user will only input the title of his presentation and the entire ppt will be made and saved in his folder.

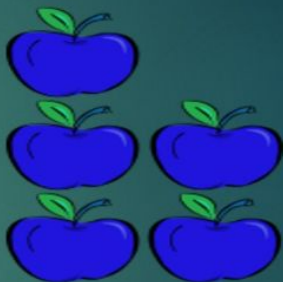
Themes of PPT

- ❖ Objects in VBscript is created.
- ❖ These objects are basically presented as layouts.
- ❖ Different layouts for slides of PPT are given to the user depending on his/her needs.



Introduction

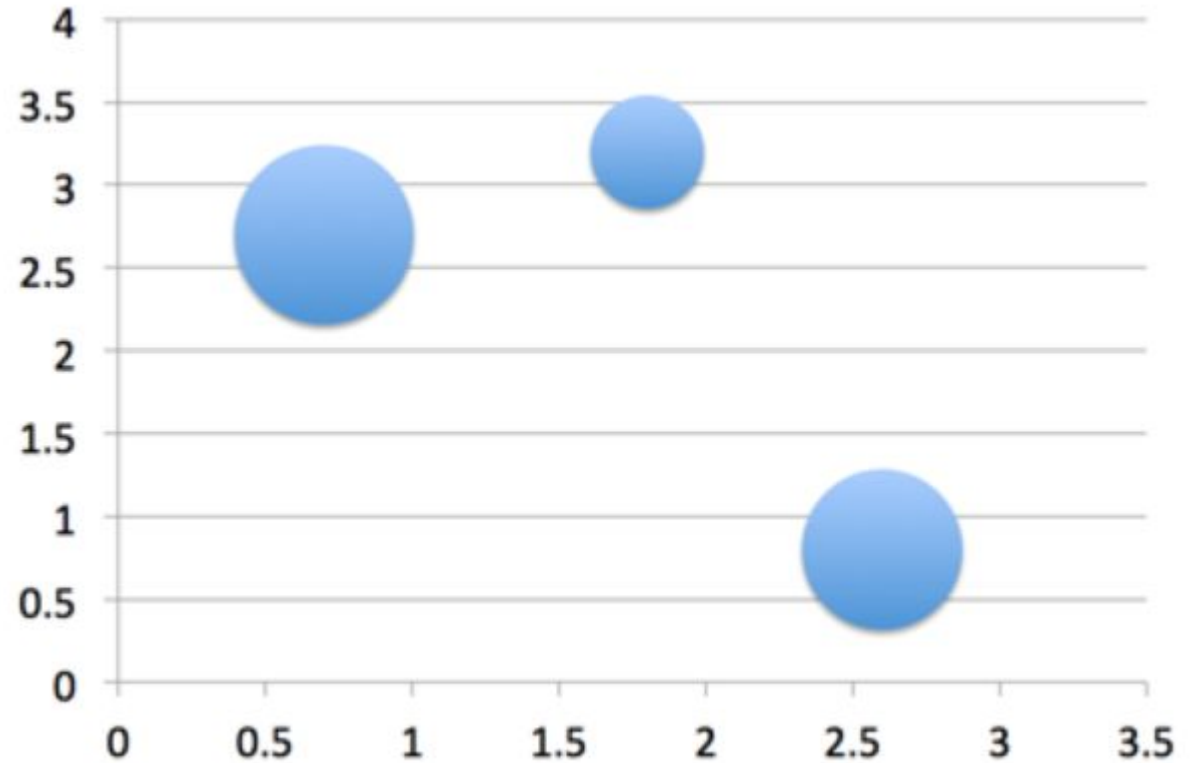
- ▶ The addition of two whole numbers results in the total amount or sum of those values combined
- ▶ Addition belongs to arithmetic, a branch of mathematics
- ▶ Performing addition is one of the simplest numerical tasks



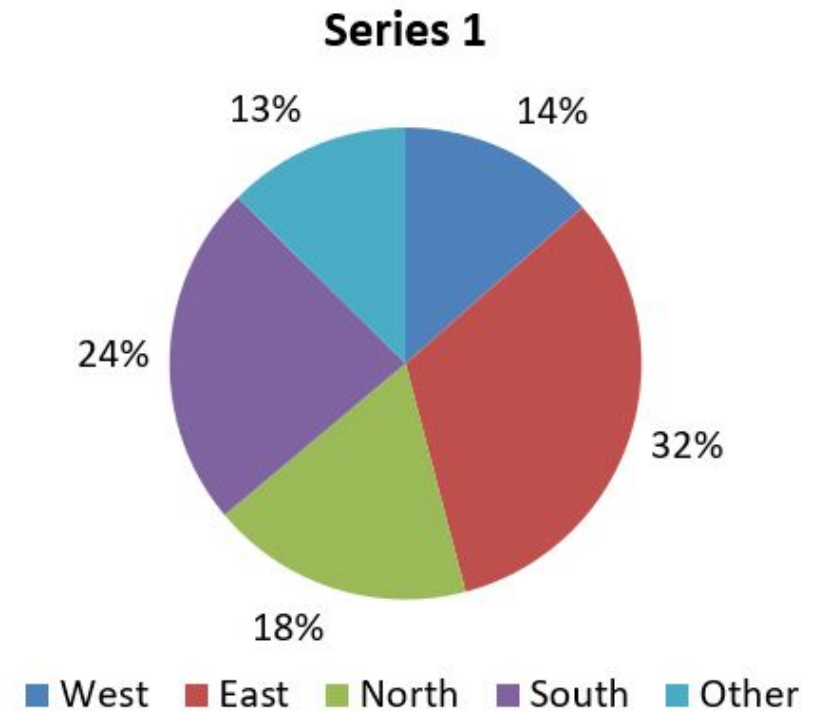
3 + 2 = 5 with apples, a popular choice in textbooks[1]

DATA VISUALISATION ASPECT

- Statistical Data -> Graphs
- More Visuals
- Requires dedicated statistical data scraping algorithm



Further Implementations of python-pptx



RESEARCH PAPER

Now coming to the code, firstly the modules required must be imported. requests, os, tqdm, BeautifulSoup, urljoin and urlparse are imported. A function is built using urlparse() to check if the input URL is valid i.e. if it contains domain name and protocol. Next, a function that collects URLs of all the images in the web page is built using requests and BeautifulSoup. BeautifulSoup is used to access the HTML form. Next, all the img tags which contains the 'src' attribute are extracted and which doesn't contain the 'src' attribute are skipped. Some URLs have unnecessary characters at the end starting with a question mark(?), those must be removed if any are present. All the URLs are checked if they are valid using the function built in the beginning to check if the input webpage URL was valid.

Next, a function is built to download all the images using the URLs collected. This function is built using os library and file handling methods. The main function is built to download each image by passing the URL and path(obtained in the download function). Finally, the main function is called by passing the URL of the Wikipedia webpage and the folder name as input.

XLNet[10]: Generalized Auto regressive Pretraining for Language Understanding a generalized auto regressive pre-training method that enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order and overcomes the limitations of BERT[11] by integrating ideas from Transformer-XL, the state-of-the-art auto regressive model, into pretraining.

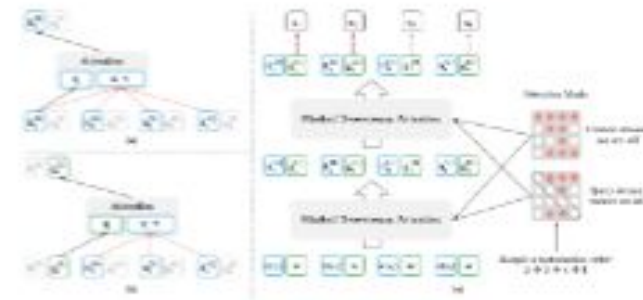


Fig. 2. (a): Content stream attention, which is the same as the standard self-attention. (b): Query stream attention, which does not have access information about the content x_{zt} . (c): Overview of the permutation language modeling training with two-stream attention.

References

1. Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
2. Canete, José, et al. "Spanish pre-trained bert model and evaluation data." Pml4dc at iclr 2020 (2020).
3. Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems*. 2017.
4. Suleiman, Dima, and Arafat Awajan. "Deep learning based abstractive text summarization: Approaches, datasets, evaluation measures, and challenges." *Mathematical Problems in Engineering* 2020 (2020).
5. Syed, Ayesha Ayub, Ford Lumban Gaol, and Tokuro Matsuo. "A Survey of the State-of-the-Art Models in Neural Abstractive Text Summarization." *IEEE Access* 9 (2021): 13248-13265.
6. Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global vectors for word representation." *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014.
7. Elman, Jeffrey L. "Finding structure in time." *Cognitive science* 14.2 (1990): 179-211.
8. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780.
9. Chung, Junyoung, et al. "Empirical evaluation of gated recurrent neural networks on sequence modeling." *arXiv preprint arXiv:1412.3555* (2014).
10. python-pptx — python-pptx 0.6.21 documentation