Text Analytics & Business Application

Text Summarization

Qinglai He

Department of Operations and Information Management

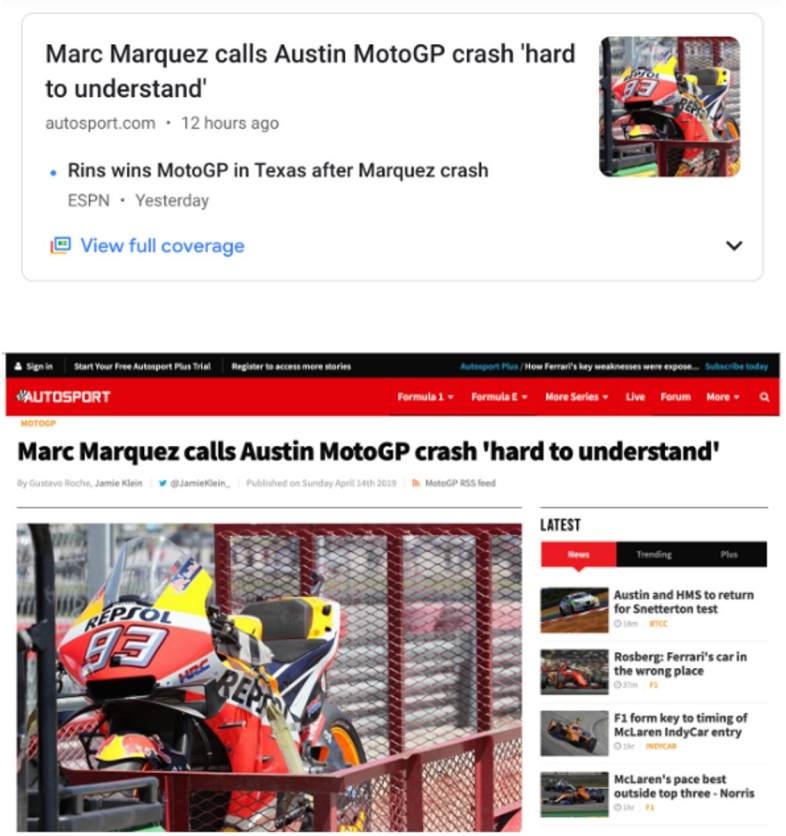
Wisconsin School of Business

WISCONSIN SCHOOL OF BUSINESS

Outline of Today's Class

- Text summarization applications
- Types of text summarization
  - Extraction-based summarization
  - Abstraction-based summarization
- Practical advice

What is Text Summarization?



SUMMARY

Marc Marquez says the crash that brought his Moto GP win streak at Austin to an abrupt end was hard to understand because he was not pushing to the limit.

Having established a gap over second-placed Valentino Rossi of 3.8 seconds, Marquez undid his hard work with a low-side crash at Turn 12 just shy of half-distance and was unable to continue.

Rossi went on to finish second behind Suzuki rider Alex Rins, who scored his first Moto GP win, while Andrea Dovizioso moved to the head of the riders standings by finishing fourth.

Marquez denied a suggestion made by third-place finisher Jack Miller that the Honda rider had pushed too hard in the early laps to break away from the pack and was struggling with an overheating front tyre.

When that was put to him, Marquez responded: Its what I said, on data already we compared and it was very similar to my fastest lap and to other laps.

Text Summarization

- Text summarization refers to the task of creating a summary of a longer piece of text.
- The goal of this task is to create a coherent summary that captures the key ideas in the text.
  - It's useful to do a quick read of large documents, store only relevant information, and facilitate better retrieval of information.

Applications–Newsletters

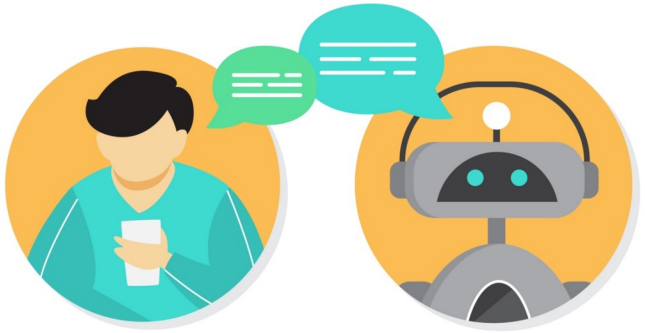
- Many weekly newsletters take the form of an introduction followed by a curated selection of relevant articles.
- Summarization would allow organizations to further enrich newsletters with a stream of summaries (versus a list of links), which can be a particularly convenient format in mobile.

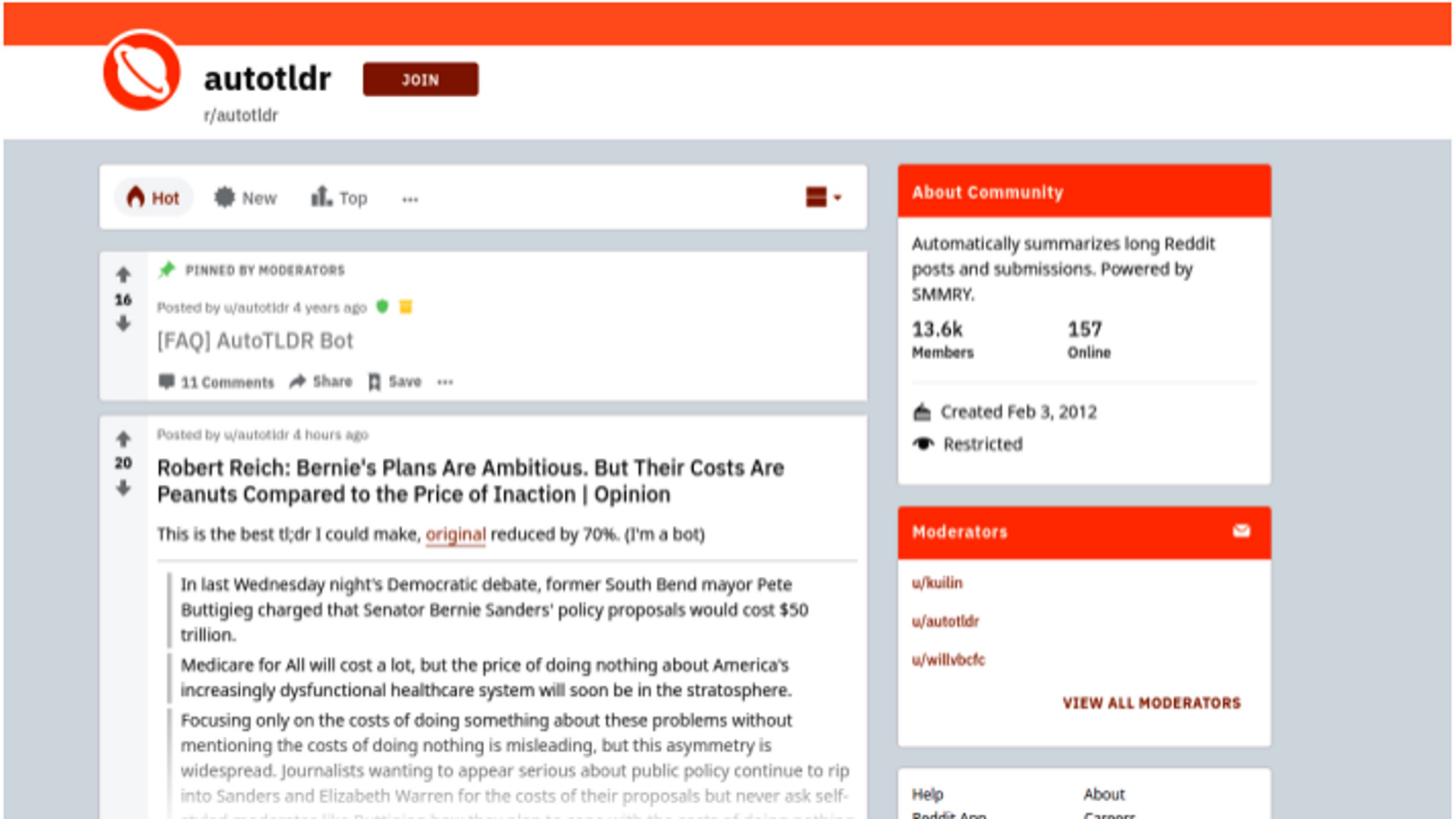
Applications–Media Monitoring

- The problem of information overload and “content shock” has been widely discussed.
- Automatic summarization presents an opportunity to condense the continuous torrent of information into smaller pieces of information.

Applications–Question Answering & Bots

- Personal assistants are taking over the workplace and the smart home. However, most assistants are fairly limited to very specific tasks.
- By collecting the most relevant documents for a particular question, a summarizer could assemble a cohesive answer in the form of a multi-document summary.





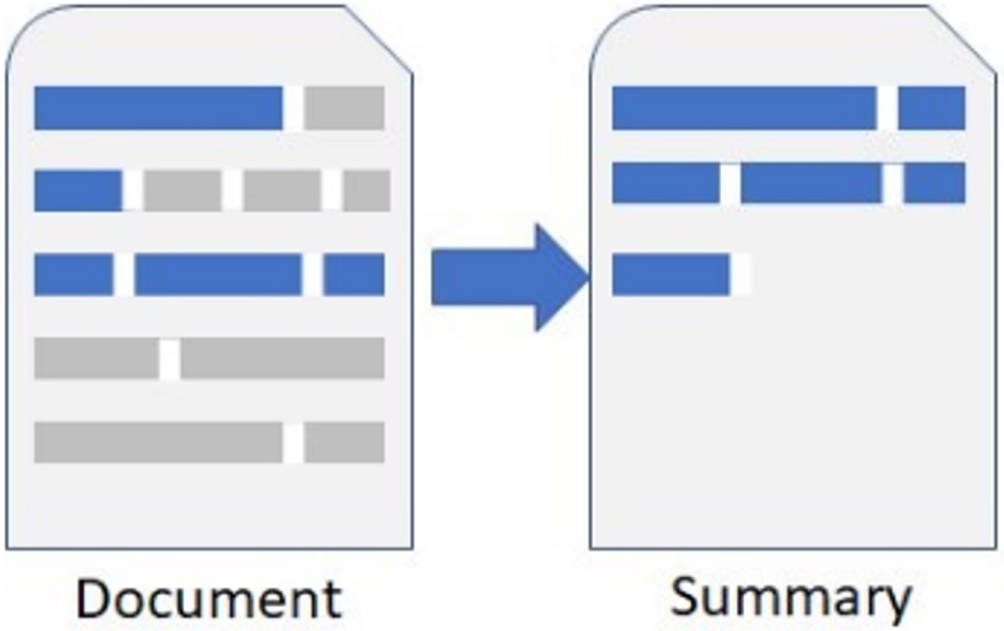
Autotldr bot on Reddit summarizes long Reddit posts by selecting and ranking the most important sentences in the post.

Types of Summarization

- Extractive summarization
  - select important sentences from a piece of text and showing them together as a summary
- Abstractive summarization
  - the task of generating an abstract of the text
- Query-focused summarization
  - create the summary of the text depending on the user query
- Query-independent summarization
  - create a general summary.
- Single-document summarization
  - the task of summarizing a standalone document.
- Multi-document summarization
  - the task of assembling a collection of documents

Extraction-based Summarization

- The extractive text summarization technique involves pulling key phrases from the source document and combining them to make a summary.
- The extraction is made according to the defined metric without making any changes to the texts



Example:

Source Text: Peter and Elizabeth took a taxi to attend the night party in the city.

While in the party, Elizabeth collapsed and was rushed to the hospital.

Summary: Peter and Elizabeth attend party city. Elizabeth rushed hospital.

Some Ideas to Perform Extractive summarization

Let's use a short paragraph to illustrate how **extractive text summarization** can be performed:

"Peter and Elizabeth took a taxi to attend the night party in the city. While in the party, Elizabeth collapsed and was rushed to the hospital. Since she was diagnosed with a brain injury, the doctor told Peter to stay besides her until she gets well. Therefore, Peter stayed with her at the hospital for 3 days without leaving."

Step 1: Convert the paragraph into sentences

Split the paragraph into its corresponding sentences:

1. *Peter and Elizabeth took a taxi to attend the night party in the city*
2. *While in the party, Elizabeth collapsed and was rushed to the hospital*
3. *Since she was diagnosed with a brain injury, the doctor told Peter to stay besides her until she gets well*
4. *Therefore, Peter stayed with her at the hospital for 3 days without leaving*

Step 2: Text processing

Removing the stop words, numbers, punctuation, and other special characters from the sentences:

1. *Peter Elizabeth took taxi attend night party city*
2. *Party Elizabeth collapse rush hospital*
3. *Diagnose brain injury doctor told Peter stay besides get well*
4. *Peter stay hospital days without leaving*

Step 3: Tokenization

Tokenizing the sentences is done to get all the words present in the sentences:

```
['peter','elizabeth','took','taxi','attend','night','party','city','party',
'elizabeth','collapse','rush','hospital', 'diagnose','brain', 'injury',
'doctor','told','peter','stay','besides','get','well','peter',
'stayed','hospital','days','without','leaving']
```

Step 4: Evaluate the weighted occurrence frequency of the words

- Calculate the weighted occurrence frequency of all the words.
- Divide the occurrence frequency of each of the words by the frequency of the most recurrent word in the paragraph.

Word	Frequency	Weighted Frequency
peter	3	1
elizabeth	2	0.67
took	1	0.33
taxi	1	0.33
attend	1	0.33
night	1	0.33
party	2	0.67
city	1	0.33
collapse	1	0.33
rush	1	0.33
hospital	2	0.67
diagnose	1	0.33
brain	1	0.33
injury	1	0.33
doctor	1	0.33
told	1	0.33
stay	2	0.67
besides	1	0.33
get	1	0.33
well	1	0.33
days	1	0.33
without	1	0.33
leaving	1	0.33

Step 5: Substitute words with their weighted frequencies

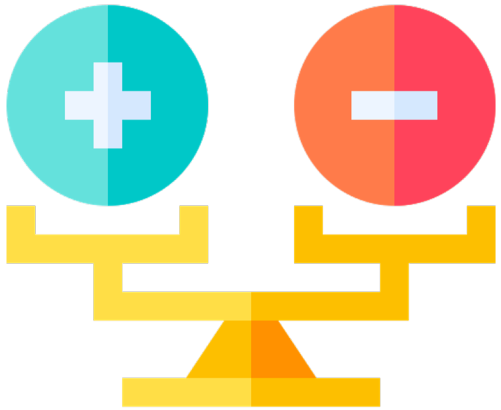
Sentence	Add weighted frequencies	Sum	
1	Peter and Elizabeth took a taxi to attend the night party in the city	1 + 0.67 + 0.33 + 0.33 + 0.33 + 0.33 + 0.67 + 0.33	3.99
2	While in the party, Elizabeth collapsed and was rushed to the hospital	0.67 + 0.67 + 0.33 + 0.33 + 0.67	2.67
3	Since she was diagnosed with a brain injury, the doctor told Peter to stay besides her until she gets well.	0.33 + 0.33 + 0.33 + 0.33 + 1 + 0.33 + 0.33 + 0.33 + 0.33 +0.33	3.97
4	Therefore, Peter stayed with her at the hospital for 3 days without leaving	1 + 0.67 + 0.67 + 0.33 + 0.33 + 0.33	3.33

From the sum of the weighted frequencies of the words, we can deduce that the first sentence carries the most weight in the paragraph

Extraction-based Summarization

Pros:

- Unlikely to change the meaning of text
- In built explainability. We can visualize sentence scores; explore gradient based approaches to compute contribution of each input token to score prediction

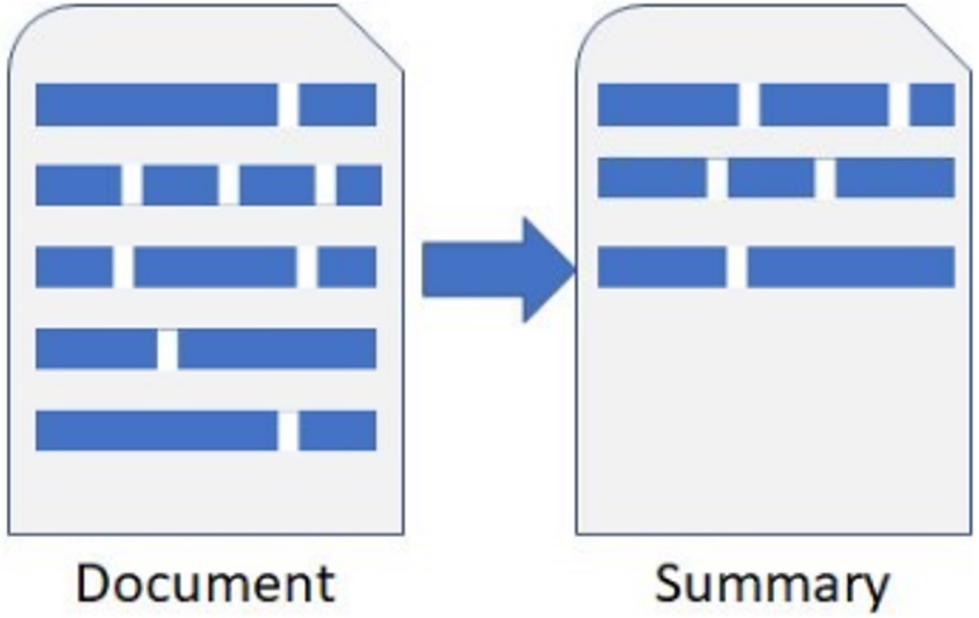


Cons:

- Extracted sentences can be awkward and grammatically strange when assembled.
- Perhaps be more compute intensive than the abstractive approach since we are making predictions for each sentence.

Abstraction-based Summarization

The abstraction technique entails paraphrasing and shortening parts of the source document, and it can overcome the grammar inconsistencies of the extractive method.



Example:

Source Text: Peter and Elizabeth took a taxi to attend the night party in the city.

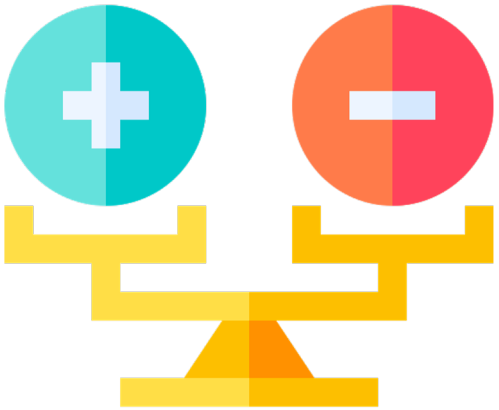
While in the party, Elizabeth collapsed and was rushed to the hospital.

Summary: Elizabeth was hospitalized after attending a party with Peter.

Abstraction-based Summarization

Pros:

- Large datasets exist.
- End to end training can allow a model generate grammatically correct summaries.
- Models can paraphrase, similar to what humans do.

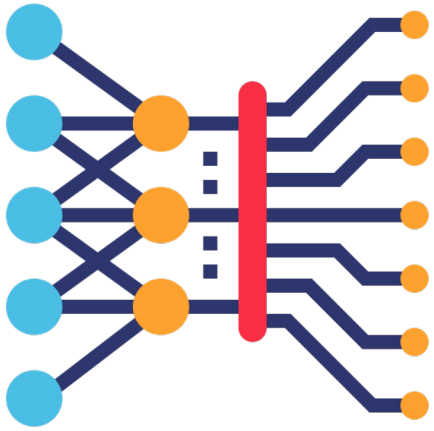


Cons:

- Model can “hallucinate” information that is not contained in the original document or factually incorrect. This can result in summaries that are different in meaning compared to the original document

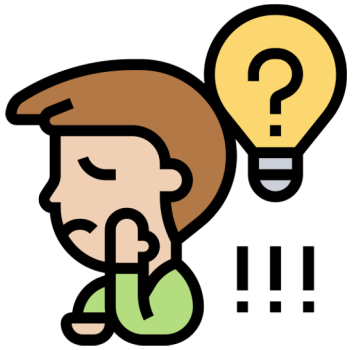
Abstractive Summarization as a Research Topic

- Abstractive summarization is not widely used in practical applications.
- News headline generation, news summary generation, and question answering are three common use cases for abstractive summarization.
- Recent research in deep learning and reinforcement learning has shown promising results for abstractive summarization.



Practical Issues in Deploying a Summarizer

- Deploying a summarizer requires considering practical issues like **pre-processing** and **text size**.
- Pre-processing steps like sentence splitting play an important role in output summary quality.
  - Custom solutions may be necessary for different data formats.
- Most summarization algorithms are sensitive to text size.
  - To work around this, the summarizer may be run on text partitions or top/bottom sections of the text.
  - Alternatively, it may make sense to run a summarizer on other selected parts of the text.



Practical Advice

- In most cases, off-the-shelf summarizers will be used rather than developing a summarizer from scratch.
  - Considering factors like speed.
- If existing algorithms do not suit the project scenario or perform poorly, it may be necessary to develop a custom summarizer.

Practical Advice–Evaluation

- Metrics like accuracy and coherence may not fully capture the quality of the summary.
- In research, summarization approaches are evaluated using a common dataset of reference summaries created by humans.
- Recall-oriented understudy for gisting evaluate (ROUGE) is a common set of metrics based on n-gram overlaps used for evaluating automatic summarization systems.
- ROUGE may not suit your exact use case. You can create your own evaluation set or ask human annotators to rate the summaries produced by different algorithms in terms of coherence, accuracy of the summary, etc.

Example

python + colab

Take 10 minutes break…