CS2034B / DH2144B

Data Analytics: Principles and Tools

Western
UNIVERSITY · CANADA

Week 9

Analytics

Textual Analytics

Textual Analytics

- Is the process of computationally deriving **meaningful** information from the textual data
- Also known as text mining
- It involves
  - Structuring and parsing text (including removing **stop words**, punctuation, etc.)
  - Deriving patterns from the now structured data
  - Interpreting and evaluating the output

Textual Analytics

- Is the process of computationally deriving **meaningful** information from the textual data
- Also known as text mining
- It involves
  - Structuring and parsing text (including removing **stop words**, punctuation, etc.)

Stop Words

Common words in the English Language that provide no or little meaning.

Examples: the, is, at, which, on, that, this, want, who, are, a, i

Textual Analytics Applications

- **Plagiarism detection**
- **Security:** Monitoring social media for terrorists
- **Bio Surveillance:** Google Flu Trends
- **Literature:** Searching databases, creating, indexing for retrieval
- **Automation of content analysis:** Document summarization, concept extraction, categorization
- **Search**: Search engines
- **Relevance**: Ad placement
- **Sentiment Analysis**: Public option on topic, business, product, person, stock, commodity

Document Summarization

- Tries to automatically create a representative summary of the entire document, by finding the most informative sentences
- Process of reducing a textual document computationally to create a summary that retains the most important points of the original document
- Main idea is to find a representative subset of the data, which contains the information of the entire dataset

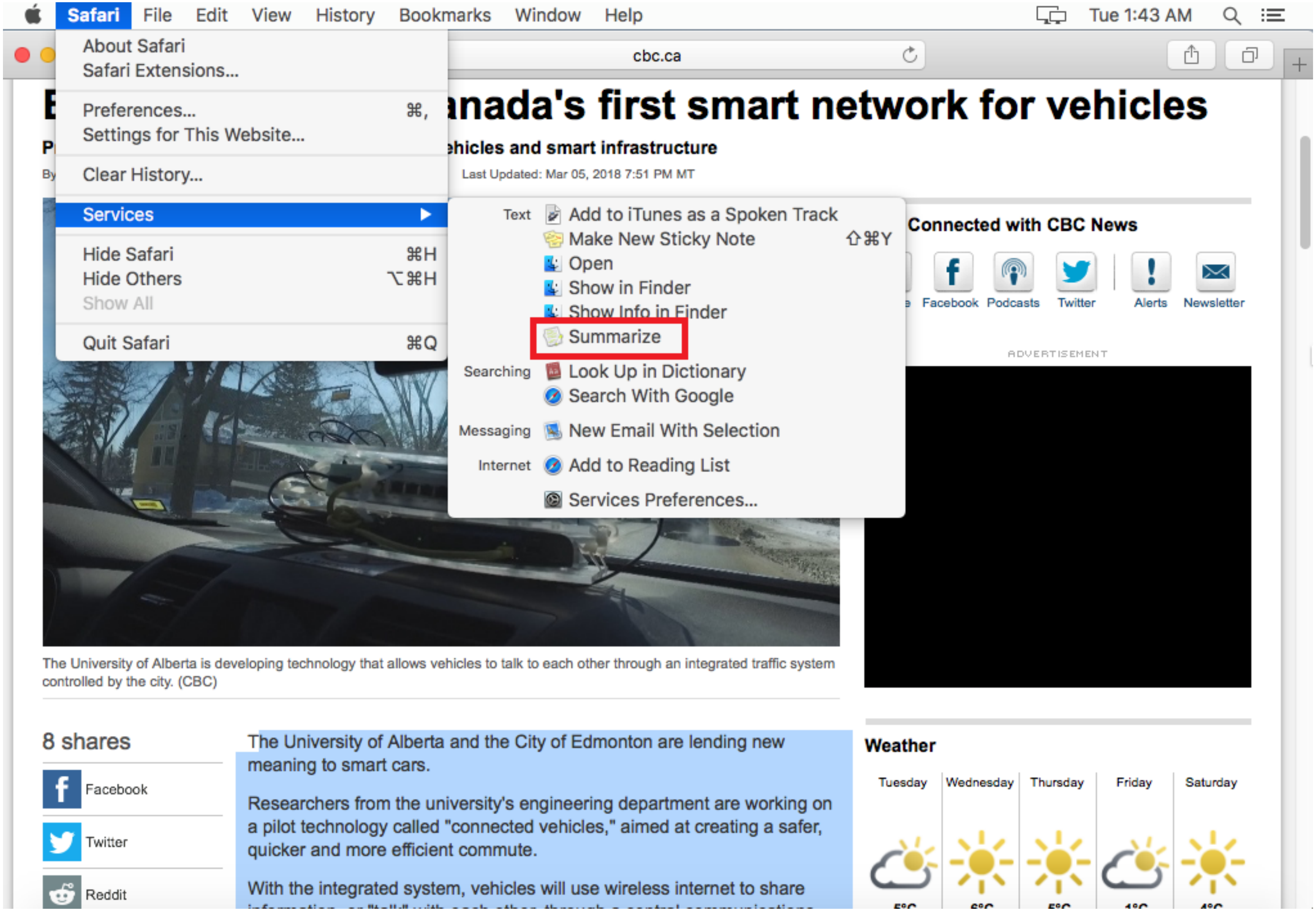
Document Summarization

- Tries to automatically create a representative summary of the entire document, by finding the most informative sentences
- Process of reducing a textual document computationally to create a summary that retains the most important points of the original entire dataset

Examples:

- Word 2010 AutoSummarize
- MacOS X Summarize Service

MacOS X Summarize Service



anada's first smart network for vehicles

ehicles and smart infrastructure

Last Updated: Mar 05, 2018 7:51 PM MT

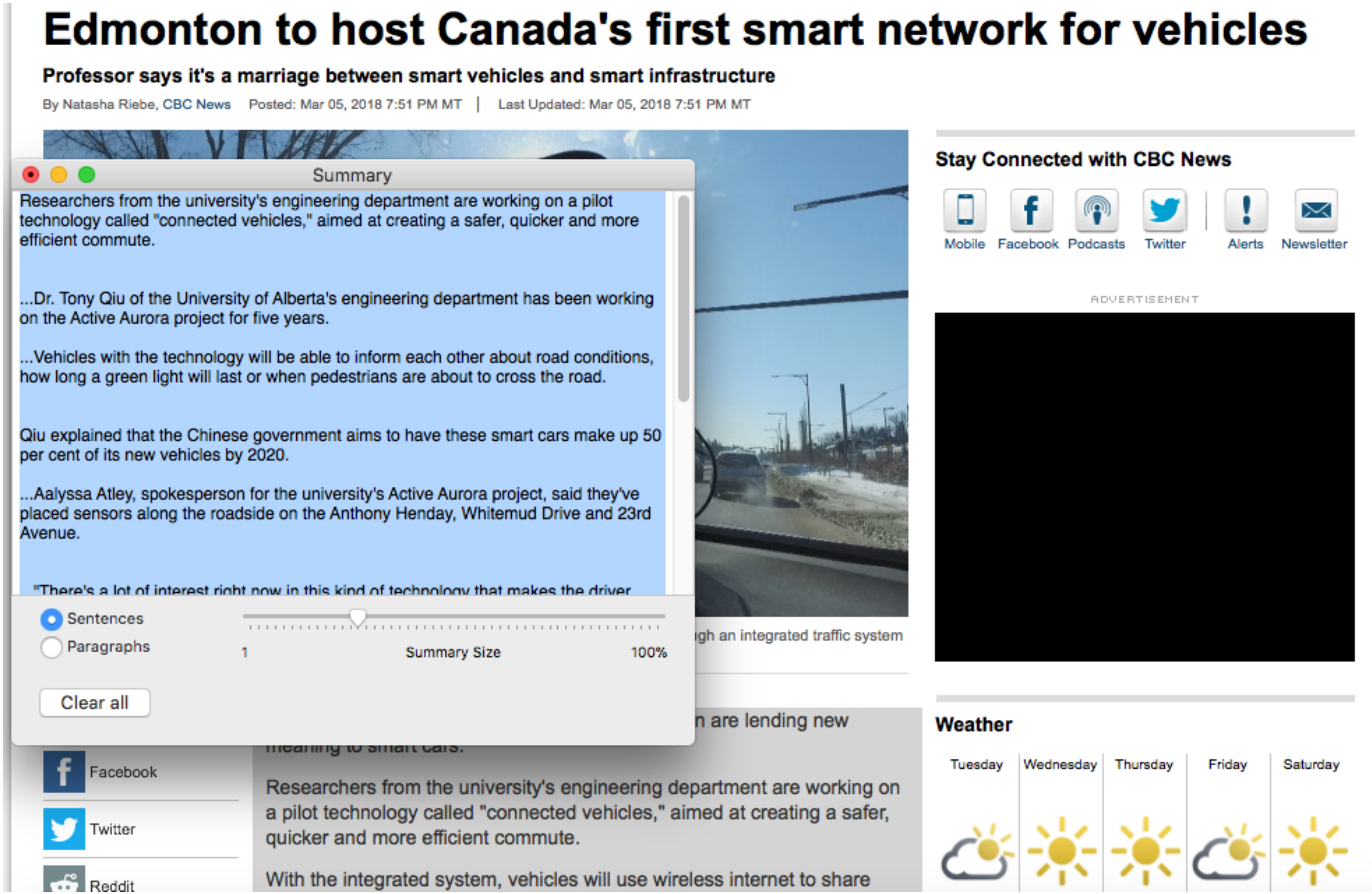
The University of Alberta is developing technology that allows vehicles to talk to each other through an integrated traffic system controlled by the city. (CBC)

The University of Alberta and the City of Edmonton are lending new meaning to smart cars.

Researchers from the university's engineering department are working on a pilot technology called "connected vehicles," aimed at creating a safer, quicker and more efficient commute.

With the integrated system, vehicles will use wireless internet to share

MacOS X Summarize Service



Edmonton to host Canada's first smart network for vehicles

Professor says it's a marriage between smart vehicles and smart infrastructure

By Natasha Riebe, CBC News Posted: Mar 05, 2018 7:51 PM MT | Last Updated: Mar 05, 2018 7:51 PM MT

Summary

Researchers from the university's engineering department are working on a pilot technology called "connected vehicles," aimed at creating a safer, quicker and more efficient commute.

...Dr. Tony Qiu of the University of Alberta's engineering department has been working on the Active Aurora project for five years.

...Vehicles with the technology will be able to inform each other about road conditions, how long a green light will last or when pedestrians are about to cross the road.

Qiu explained that the Chinese government aims to have these smart cars make up 50 per cent of its new vehicles by 2020.

...Aalyssa Atley, spokesperson for the university's Active Aurora project, said they've placed sensors along the roadside on the Anthony Henday, Whitemud Drive and 23rd Avenue.

"There's a lot of interest right now in this kind of technology that makes the driver

● Sentences
○ Paragraphs

Summary Size: 1 – 100%

Clear all

ugh an integrated traffic system

n are lending new meaning to smart cars.

Researchers from the university's engineering department are working on a pilot technology called "connected vehicles," aimed at creating a safer, quicker and more efficient commute.

With the integrated system, vehicles will use wireless internet to share

Sentiment Analysis

- Use of analytics, natural language processing, and computational linguistics to identify and extract subjective information in source materials.
- Also known as opinion mining

Sentiment Analysis Tasks

- **Clean data:** Remove punctuation, stop words, etc.
- **Subjectivity identification:** Classifying a given text into one of two classes: objective or subjective.
- **Classifying polarity:** Determining the opinions, polarity, or sentiments expressed on different features or aspects of entities
- **Classification Result:** Positive, negative, neutral

Sentiment Methods

- **Knowledge-based techniques:**
  - Classify text by affect categories based on the presence of unambiguous affect words such as happy, sad, afraid, and bored.
  - Some knowledge bases not only list obvious affect words, but also assign arbitrary words a probable “affinity” to particular emotions (sad, scared, happy, excited, etc.).
- **Statistical methods**
  - Classify text based on past examples.
  - Certain words, sentences, combinations more likely to be linked to classifications (positive, negative neutral) based on past data.
- **Hybrid Approaches**

Sentiment Analysis Process

Given text (a tweet for example):

- Clean the text (remove punctuation, extra spaces, numbers, etc.).
- Break the text into individual words
- **Stem the words**
  - Remove prefixes and suffixes (e.g. stealing, steals and steal would all be stem to steal.
- Remove stop words (e.g., as, is, the, a)
- Format the words, e.g., convert all text to upper or lowercase
- Iterate over all the remaining words, if the word appears in the keywordList, add the sentiment value of that word to the total sentiment value for the text.
- Calculate the average or total sentiment based on the affect words
- Classify the result (Positive, Negative, Neutral, etc.)

Sentiment Analysis Process

Given text (a tweet for example):

- Clean the text (remove punctuation, extra spaces, numbers, etc.).
- Break the text into individual words
- **Stem the words**
  - Remove prefixes and suffixes (e.g. stealing, steals and steal would all be stem to steal.
- Remove stop words (e.g., as, is, the, a)
- Format the words, e.g., convert all text to upper or lowercase
- Iterate over all the remaining words, if the word appears in the keywordList, add the sentiment value of that word to the total sentiment value for the text.
- Calculate the average or total sentiment based on the affect words
- Classify the result (Positive, Negative, Neutral, etc.)

Sentiment Analysis Process

Given text (a tweet for example):

- Clean the text (remove punctuation, extra spaces, numbers, etc.).
- Break the text into individual words
- **Stem the words**
  - Remove prefixes and suffixes (e.g. stealing, steals and steal would all be stem to steal.
- Remove stop words (e.g., as, is, the, a)
- Format the words, e.g., convert all text to upper or lowercase
- Iterate over all the remaining words, if the word appears in the keywordList, add the sentiment value of that word to the total sentiment value for the text.
- Calculate the average or total sentiment based on the affect words
- Classify the result (Positive, Negative, Neutral, etc.)

Sentiment Analysis Process

Example:

```
I am very HAPPY to be here today!
```

Step 1: Remove punctuation, numbers, etc.

```
I am very HAPPY to be here today
```

Step 2: Remove stop words and stem the words

```
very HAPPY today
```

Step 3: Make lowercase

```
very happy today
```

Sentiment Analysis Process

Example:

I am very HAPPY to be here today!

Step 4: Iterate over words and tally sentiment

very	**Neutral:** +0
happy	**Positive:** +10
today	**Neutral:** +0

Total: 10

Step 5: Classification

Neutral = 0

Positive >= 10

Negative <= 10

Therefore sentiment is Positive

Natural Language Processing

- A field of artificial intelligence and computational linguistics concerned with the interactions between computers and human (natural languages)
- The goal is to derive meaning from human or natural language

Natural Language Processing Methods

- Reduce inflectional forms and sometimes derivationally related forms of a word to a common base, root, lemma, or stem word
- Methods for getting the root, base or stem word
  - Stemming
  - Lemmatization

Stemming

- *Stemming* usually refers to a crude heuristic process that chops off the ends of words in the hope of getting the root word.
- It typically achieves its goal most of the time, and often includes the removal of derivational affixes.
- **Example:**

  speaking and speaks
  stemmed to speak

Stemming

- Uses simple rules based on end of word.
- Some simple examples:

Rule	Example
SSES → SS	caresses → caress
IES → Y	ponies → pony
S →	cats → cat
LY →	friendly → friend
ED →	failed → fail
ING →	looking → look

Stemming

- Not always successful, but often good enough:

Rule	Example
SSES → SS	busses → buss
IES → Y	movies → movy
S →	yes → ye
LY →	only → on
ED →	cried → cri
ING →	wing → w

Lemmatization

Refers to doing things with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the *lemma*.

Stemming vs. Lemmatization

- If confronted with the word *quizzes*, stemming might return *quizze* or *quizz*, whereas lemmatization would attempt to return *quiz*.
- Stemming is often more efficient in terms of computation, but provides less accurate results.
- Stemming is also far easier to implement.

Example

Write a function named RemoveStop that removes the stop words "the", "is", and "at" from a string. Ignore capitalization. Use the following function header:

```
Function RemoveStop(str As String) As String
```

Hint 1: Don't use Replace as this might remove stop words like "at" inside of another word like "cat".

Hint 2: Rather than remove them from str build a new string that does not contain the stop words.

Example

```
Function RemoveStop(str As String) As String
    Dim i As Integer

    Dim cleanStr As String
    cleanStr = ""

    Dim words() As String
    words = Split(str)

    For i = LBound(words) To UBound(words)
        If StrComp(words(i), "the", vbTextCompare) <> 0 And
           StrComp(words(i), "is", vbTextCompare) <> 0 And
           StrComp(words(i), "at", vbTextCompare) <> 0 Then
            cleanStr = cleanStr & words(i) & " "
        End If
    Next i

    RemoveStop = cleanStr
End Function
```

Should be on same line in VBA, shown this way for space reasons

Example

```
Function RemoveStop(str As String) As String
    Dim i As Integer

    Dim cleanStr As String
    cleanStr = ""

    Dim words() As String
    words = Split(str)

    For i = LBound(words) To UBound(words)
        If StrComp(words(i), "the", vbTextCompare) <> 0 And
           StrComp(words(i), "is", vbTextCompare) <> 0 And
           StrComp(words(i), "at", vbTextCompare) <> 0 Then
            cleanStr = cleanStr & words(i) & " "
        End If
    Next i

    RemoveStop = cleanStr
End Function
```

Should be on same line in VBA, shown this way for space reasons