Analysing Social Media Chatter About Active Ageing in Ireland

Deeksha Vyas, B.E.

A Dissertation

Presented to the University of Dublin, Trinity College

in partial fulfilment of the requirements for the degree of

Master of Science in Computer Science (Intelligent Systems)

Supervisor: Carl Vogel

August 2022

Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work.

Deeksha Vyas

August 20, 2022

Permission to Lend and/or Copy

I, the undersigned, agree that Trinity College Library may lend or copy this thesis upon request.

Deeksha Vyas

August 20, 2022

Analysing Social Media Chatter About Active Ageing in Ireland

Deeksha Vyas, Master of Science in Computer Science

University of Dublin, Trinity College, 2022

Supervisor: Carl Vogel

This research aims to analyse online chatter about Active Ageing in Ireland available on the social networking platform Twitter. For the requirements of this research, I investigated all the social groups, activities, and meetups favoured by the elderly. It gave me a lot of insight into their social lives. After exploration to nearby places in Dublin, I made a listing of all such events, and from that listing, I collected a set of keywords. Then I used these keywords to scrape tweets or hashtags containing them geolocated to Ireland. This scraped data would offer conversations, particularly about older demographic concerning the social groups. After I had a collection of these tweets, I performed data preprocessing and built a semantic space. I also performed functions provided by LSAfun along with detecting sentiments and POS tags. This research conducts various experiments to answer the question, 'Is there any relation between the text and the Emotions and parts of speech of those texts expressed?'. The research could not give conclusive relation from either of those, however, it provides scope for future works.

Acknowledgments

I wish to express my sincere gratitude to Professor Carl Vogel for guiding me throughout the completion of this research. I appreciate his teaching and admire his sound knowledge.

I would also like to give special thanks to my parents for their constant encouragement and my sister for being there to support me every day. Lastly, I am more than grateful for the friends that I have made on this journey.

Deeksha Vyas

University of Dublin, Trinity College
August 2022

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this research, I will analyse the social media chatter i.e. the thoughts, opinions or discussions posted by individuals on Twitter regarding the topics under the umbrella of Active Ageing in Ireland. Various keywords, such as 'Active Retirement', 'Elderly', etc. have been used to retrieve data from Twitter, specific to the country, Ireland.

This analysis is done using an off-the-shelf tool created by Günther et al. (2015), who uses Latent Semantic Analysis proposed by Landauer et al. (1998), and provides experiments that help retrieve some information about the chatter and from that information, we might be able to determine or infer if there's any introspecting sense in them. To evaluate this research, I will use a random sample of data to manually check if the inferences from the experiments done are true and apt and don't imply something else than inferred.

1.1 Motivation

I have carried out this research to see if there is any relation between the online chatter about active ageing in Ireland to emotions and parts of speech, nouns and adjectives. Another intention was to identify if the active ageing initiative by the government of Ireland is benefiting the older demographic and people's response to such initiatives.

1.2 What is Active Ageing?

Active ageing is a government initiative that helps the elderly engage in multiple activities so they are healthy both physically and mentally. According to Wikipedia contributors (2022a), this concept was deployed by the European Commission of the WHO. It means that people get exposed to connectivity with the community and activities and stay healthy. It also leads to an increase in the retirement age.

Active ageing Programme

According to Ireland Active (2009), the main objective of this program is to maintain a connection with the local community and provide a safe environment for the elderly to give proper time for exercises and personal growth activities, especially during winters when they could get very disconnected with the society. This disconnection was at its peak during Covid-19 because of social-distancing laws.

There are many more initiatives, namely Active Retirement, Age Action, and likes of such. I will use these keywords to perform analysis based on classifications such as Emotion. I will also use these keywords to plot neighbourhoods and try to get some inferences.

1.3 Research Question

This paper aims to answer the following research questions:

- How to harvest data and create a Semantic Space from social media chatter about Active Ageing in Ireland?
- What kinds of experiments to perform to get inferences from an LSA Space?
- Is there any similarity between a set of terms or documents based on that semantic space created?
- Are emotions and parts of speech able to give inferences from a semantic space?

1.4 Structure of the rest of the paper

The rest of the paper includes sections as described here. Firstly, it has a Literature Review that compares closely related projects and provides literature for various tools concerning the creation of semantic spaces and using them. Then it is followed by the Implementation section which focuses on gathering data and creating a semantic space. Further, it is followed by the Experiments and Results section. Finally, a Conclusion is given based on the work of this paper.

Chapter 2

Background Area and Literature Review

2.1 Introduction

It is crucial to know how to fetch a collection of data from the internet and how to use it to perform text analytics. Online platforms provide vast information, from knowledge to opinions to personal details. Because of this much data, data science has evolved to learn about humans and is used to make better day-to-day experiences, such as personalisation services from recommender systems, artificial intelligence for decision-making, and the likes of such. On the other hand, having that much data could make it difficult to get relevant information on a particular topic of interest.

For the same reason, various methods to get a relevant collection of data available online are proposed. This research has attempted to analyse online chatter concerning active ageing and sentiments expressed. To fulfil the purpose of this research, I have explored such techniques and libraries. Similar sets of research exist, although not for this domain. These concepts and explanations of closely related projects are provided in the following sections.

2.2 Background: Concepts and Definitions

2.2.1 Cosine Similarity

Every document can expressed in the form of a vector where each element of the vector represents a word. The scalar coefficient associated with each word represents the frequency of the word in that document. There are many ways to calculate the similarity of two documents. The most common approach is the Euclidean distance. For two document represented by vectors $X = (x_1, x_2...x_n)$ and $Y = (y_1, y_2...y_n)$, the euclidean distance similarity between them can be obtained as:

$$EuclideanDistance(X, Y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_n - y_n)^2}$$

$$EuclideanDistanceSimilarity(X, Y) = \frac{1}{1 + EuclideanDistance(X, Y)}$$

However this measure has a flaw that a word which is repeated quite common could dominate the calculation of the similarity. In this case the words with less frequency will have very little effect on the calculation.

An alternative to Euclidean distance is the cosine similarity. Cosine similarity measures the cosine of the angle between the two vectors. For documents represented by vectors $X = (x_1, x_2...x_n)$ and $Y = (y_1, y_2...y_n)$, the cosine siilarity between them can given as:

$$CosineSimilarity(X, Y) = \frac{X \cdot Y}{|X||Y|}$$

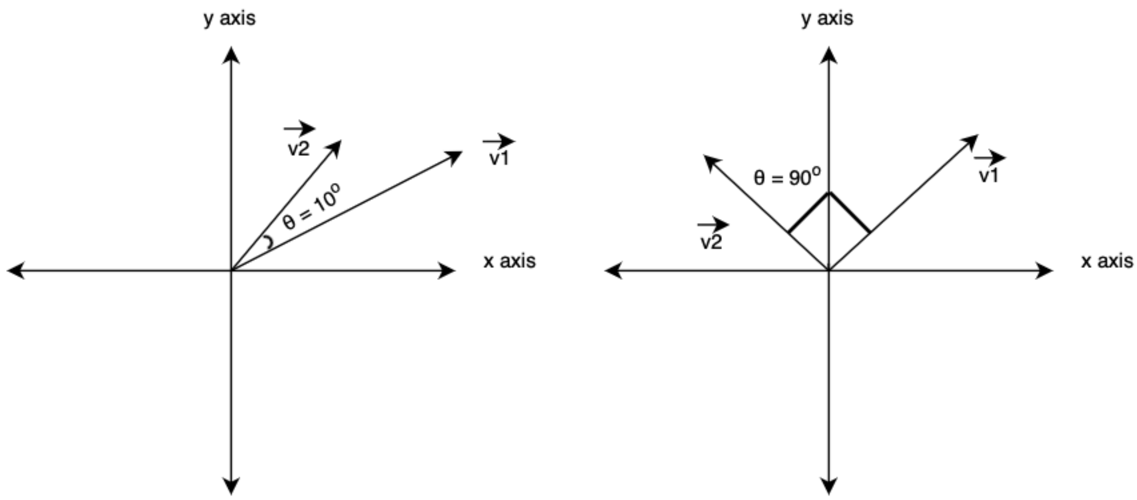

Figure 2.1: The figure displays two different cosine of angles between two vectors. The one of the left have highest cosine similarity than the other because of least angle of cosine in comparison.

The cosine similarity can be understood better by considering two vectors in a 2D space, consider 2.1. If the two vectors are quite opposite to each other, then they will be in opposite directions to each which makes the angle between them to be 180°. The cosine similarity then becomes -1 showing minimum similarity. If the two vectors are similar to each other then they will be in the same direction which makes the angle between them to be 0°. This makes the cosine similarity to be 1 showing maximum similarity.

2.2.2 Latent Semantic Analysis

Latent Semantic Analysis is the process of identifying relationships between documents by analyzing the frequency of the words present in them. The basic assumption is that words having similar meaning will occur in similar contexts. For each document a vector is created using the words present in it. A matrix is then created out of all the document vectors where the columns represents the documents and the rows represent the words. The number of rows are then reduced by using a dimensional reduction technique like Singular Value Decomposition. The documents are then compared by calculating the cosine similarity between the reduced vectors.

2.3 Literature Review

The first step for analysing data is famously creating a semantic space. Secondly, researchers use a similarity measurement to find similarities between terms or documents in the collection, depending on the purpose. Following these two base tasks, multiple approaches ranging from machine learning techniques to linguistics are available. This section discusses a few of those approaches.

2.3.1 Exemplar creation and classification using Support Vector Machine

Shi et al. (2019) were seeking to identify themes from a set of online chatter. They assumed that any ongoing event happening in a particular part of the world would mean that people in that area would often converse about it online. However, using conventional methods such as supervised classification methods gives unsatisfactory results in identifying the latent meaning of the conversation online.

The authors followed a unique approach that has four main steps. They use Latent Semantic Analysis and find cosine similarities between tweets. Then they apply one of the clustering algorithms, Affinity Propagation. According to Wikipedia contributors (2021), Affinity Propagation works on the concept of "message passing" between data points. It means that it treats data points, i.e. Twitter messages, as exemplars, and by using a similarity metric, it pairs them with each other until the formation of a set of Exemplars, i.e. clusters of high quality.

It is one of the clustering algorithms which eventually forms clusters as seen in 2.2.

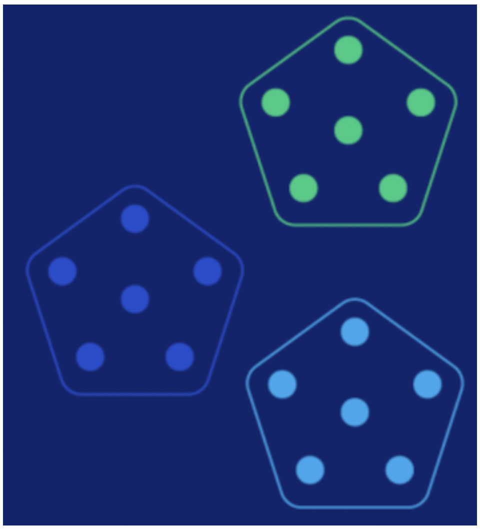

Figure 2.2: The figure depicts the formation of clusters after applying Affinity Propagation. Each Pentagon represents an exemplar consisting of the resulting messages formed.

After this step, cosine similarity is again calculated between the clusters and messages grouped. Finally, tweets have been classified by SVM using these exemplars. SVM, i.e. Support Vector Machine, is a classification approach for supervised learning models. The authors employed a novel approach to SVM utilised with Affinity propagation and successfully gathered insights.

2.3.2 Classification Algorithms used with latent Semantic Analysis

Another example where classification models are used with LSA is Karamitsos et al. (2019). It primarily aims at comparing the opinions of users of cloud computing services. The opinions were expressed on Twitter. It explores of there's any impact of social networking platforms on these cloud services providers. moreover, the authors aimed to observe if giving reviews for one of the services affect the decision of choosing that service. Popular brands like Azure, Amazon Web services and Google Cloud were the points of the experiment for the authors.

Applying sentiment analysis was done using emotion detection based on supervised models. They were done on the collection of tweets to help identify the sentiments of users concerning using the respective web services. The sentiments were mostly tested to be true. This research was able to identify mostly correct sentiments.

2.3.3 Sentiments and Parts of Speech correlation - Automatic Sentiment Analysis

Pak and Paroubek (2010) have tried to identify if there is a correlation of emotions with certain speech parts. Tools like Treetagger have been used for such research. The authors mainly attempted in performing sentiment analysis from data available on Twitter and linguistic analysis on the same.

For the emotion classification, TreeTagger has been used. The research was successful in building an automated corpus and thereby training it using a sentiment classifier.

Automatic sentiment analysis is performed using machine learning by first having a good amount of training dataset with emotions and classifier for parts of speech or similar and then applying classification models to this training data set. Once the model learns the emotions, it can be applied to any data. the performance depends on how was the quality of data fed to train.

One of the applications of the automated sentiment analysis process is handling bad reviews on online platforms for a product. The brand can immediately respond aptly before more users could face the difficulty as it was not detected on time. Moreover, this can result in a bad impression of the product to other potential users who might change their minds about using the respective services.

2.3.4 More on Automatic Sentiment Analysis

In another research focusing on automatic sentiment analysis, Pekar et al. (2021) attempted to detect voting intentions on networking platforms and online opinion polls. The authors aimed to determine the popularity of political candidates. They have used Natural Language Processing (NLP) techniques.

NLP

Natural Language Processing is a field of Computer science, especially Linguistics and Artificial Intelligence. It simply means to program a computer such that it understands human text, both written and spoken discourse, close to how humans understand these texts. Natural language or Ordinary language means natural speech by a human.

The authors have performed a naive approach by building Non-linear Autoregressive models for the opinion polls based on AdaBoost, LSTM models, and Gradient Boosting. They experimented with various machine-learning and neural networking techniques. They were able to get conclusive results with the behavioural intention in text. In addition to detecting voting intentions, they have also predicted the results of opinion polls.

Non-linear Autoregressive Models

An autoregressive model means that the model takes previous forecasts to generate new predictions Versloot (2020). It takes a linear set of past predictions. These Models generate texts as part of predictions. Hence, they require models to know how to learn the language, term relations in a sentence, and semantics. On the other hand, non-linear autoregressive models take a non-linear set of predictions.

AdaBoost

According to Wikipedia contributors (2022b), AdaBoost, also called Adaptive Boosting, is a statistical classification 'meta'-algorithm. 'Meta-learning' is a process of learning how to learn new tasks faster, as said by Brownlee (2020). It is done by learning from the experience of observing all machine learning approaches to learn a multitude of tasks.

AdaBoost aims to increase the efficiency of binary classifiers, likewise. On iterations, it detects errors of weak classifiers and makes them better, according to Kurama (2020).

'Adaptive' gives a sense that weak classifiers are adjusted to increase performance on repeated learnings. Wikipedia contributors (2022b) also points out how increasing the strength of individual learners results in stronger learning by the final model.

LSTM

According to Wikipedia contributors (2022d), Long short-term memory (LSTM) is an AI neural network (NN) with feedback connections. It provides a short-term-memory for Recurrent neural Networks for a longer timestamps. It means that it is capable of learning long-term dependencies in data Tutorialspoint (2022). It is capable to prpcess data in the form of video or speech. Therefore, it can be used in applications like video games, hand-writing recognition, etc. Notingly, it is the most cited NN of the last century.

Gradient Boosting

Gradient boosting, as the name suggests, boosts a Machine Learning process, i.e. minimising error in predictions. It is predicated on the hunch that when prior models are coupled with the best possible upcoming model, the overall prediction error is minimized. Setting the desired results for this subsequent model in reducing errors, is the main idea Hoare (2022).

Chapter 3

Implementation

3.1 Introduction

For the implementation of this research, firstly, I gathered data from Twitter. Then I transformed and preprocessed the data to create a semantic space using LSA. Then, I applied SVD to reduce dimensions in the LSA space. Further, I used cosine similarity for neighbourhood computation using LSAfun, and finally, I added dimensions for classifying words into sentiments and parts of speech. 3.1 depicts the flow of implementation.

The process along with the technology stacks used for the implementation are explained in detail in the following sections.

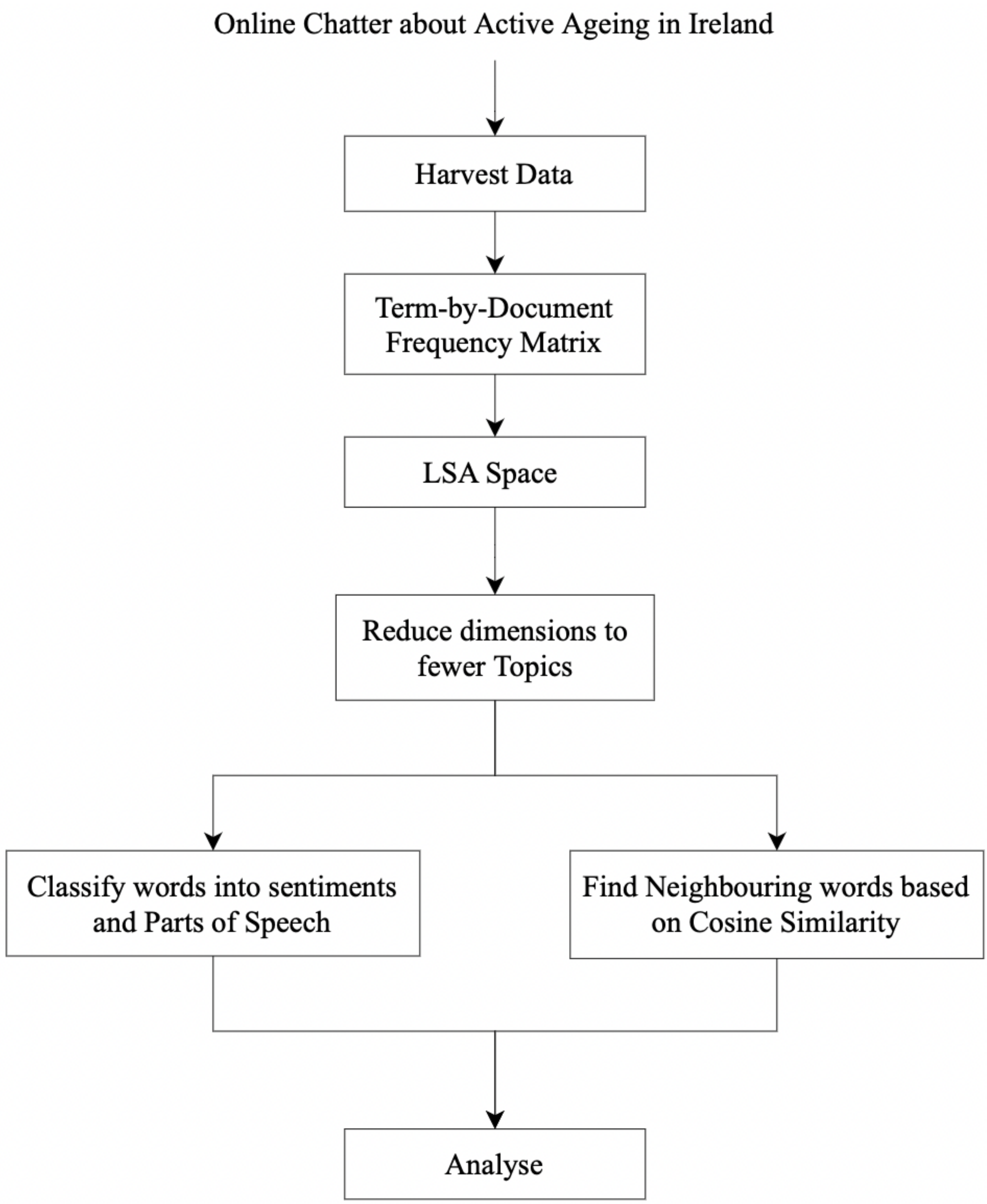

Figure 3.1: Flow diagram explaining the steps for Implementation

3.2 Technology Stacks

I have used two programming languages in this research, Python and R programming languages. The content posted on Twitter (Tweets) are fetched using Snscrape - a scraper used for social networking services. These tweets are fetched in JSON format which is then transformed into CSV format using Python; giving raw data for the research. The cleaning of this raw data is done using some tools in Python, viz. Pandas and Regular Expressions, 're'. Then, R is used to form semantic spaces using that data, and performing multiple functions like applying weighting scheme, SVD and Dimensionality Reduction, etc. The technology version and tool dependencies are given below.

Snscrape (Version 0.4.3.20220106), developed by Python Community (2022), requires a Python version $\geq$ 3.8. After its installation, Snscrape automatically installs Python package dependencies. It also requires Pandas and the tools libxml2 and libxslt for one of its dependencies, lxml. The lxml is a Python library that helps process XML and HTML. One can implement Snscrape using either Command Prompt or Python code.

LSAfun (Version 0.6.3) requires R version $\geq$ 3.1.0 along with tools 'lsa' (Version 0.73.3) and 'rgl' (Version 0.109.6). The latter is a 3D Visualisation tool that uses OpenGL and depends on R version $\geq$ 3.3.0. The former depends on another tool called, SnowballC (Version 0.7.0) used for Stemming. It is one of the arguments for preprocessing the text before creating a term-by-document frequency matrix. This argument is a part of 'textmatrix', explained in the section, 3.6.1.

3.3 Data Harvesting

I have used the social networking platform Twitter to gather data. Unlike other platforms such as Facebook or Instagram, on Twitter, all users publicly share information about certain news or events; not primarily used for staying connected with close ones and sharing personal messages. Hence, Twitter seems like a good starting point for this research.

To get tweets that concern the elderly, their engagement and chatter about Active Ageing, I have used the tool, Snscrape. I have scraped tweets using different keywords, and the tool finds contents or hashtags that contain these keywords and downloads them in JSON format.

3.3.1 What is Web Scraping?

Scraping means extracting data from a website in the form of a required format. It is also called Web harvesting. Many tools have been developed, such as Snscrape, Twitter Intelligent Tool (Twint), developed by Poldi and Community (2019), etc., to extract data from a social media website. One of these websites, Twitter, allows scraping data. Therefore, it is legal to use such tools.

3.3.2 Snscrape - Social Networking Services Scraper

Snscrape is a Scraper tool developed in Python in April 2019. After its 10^{th} upgrade, I have used the latest version, v0.4.3.20220106, which fixes the issue of absent time zones in user profile details.

It can scrape content from multiple social media platforms, such as Facebook, Instagram, Reddit, Twitter, Telegram, etc. It needs to be first installed using pip. The installation guide can be found here, Python Community (2022).

I have chosen this tool because it only follows these four steps:

1. On Terminal, go inside the folder where you want the JSON files to be created.
2. Select a keyword to use (eg. SaturdayBingo) and name of the JSON file to be created (eg. saturday_bingo_tweets.json).
3. Decide the Start date and End date for the retrieval of tweets. The format must be yyyy-mm-dd hh:mm:ss.
4. Run the following command to scrape tweets based on the particular keyword: snscrape −−jsonl −−progress −−since '2021−08−01 00:00:00' twitter−search "SaturdayBingo until:2022−02−28" > saturday_bingo_tweets.json

3.3.3 Keywords Used to Scrape Data from Twitter

To collect online chatter about Active Ageing, I identified all the social groups, Community Centers and activities attended by the elderly by visiting nearby churches and places conducting such social events. After much investigation, I've observed the following to be a few points of interest by the older demographic:

- Age and Opportunity Engage by Age and Opportunity (2022).
- Digital Skill - Training program by Active Retirement Ireland (2022).

- Age Friendly University (2022).
- Age Action - 'Getting Started Computer Training Program' for the elderly.
- Recreational centres.
- Social group in Dublin: Coastline & River Walks.
- Social group in Dublin: Shades of 50 plus.
- Social group in Dublin: Dublin Circle of Friends 50+.
- Social group in Dublin: Lunchtime Walks in Step Aside 36 yrs and Upwards.
- Social group in Dublin: Women 50 +.
- Third Age Ireland.
- Active Ageing Program.
- Umbrella Wellness.
- Events at National Gallery of Ireland.
- Theatres

All the social events, outings, groups and programs that consume most of the time of the elderly, in my opinion, must be one of their topics of conversation on social media. Amidst a myriad of data, finding such topics of interest was crucial to getting relevant information.

In my opinion, these subjects of interest can give us enough online chatter to get inferences about Active Ageing. From these mentioned subjects, I have used the keywords given in table 3.1 to finally get the data using Snscrape. These were taken from 2021-08-01 00:00:0 to 2022-02-28.

The tweets retrieved for a particular keyword are stored in a single file named after the keyword. The empty files are excluded from further consideration.

All the tweets are fetched in JSON format. This collection needs to be transformed into CSV to make it readable to the LSA tool in R. The following section explains how the data is transformed.

Keywords	Number of tweets retrieved
HiDigital	159
LifelongLearning	17691
Ageaction	1050
Darndale Belcamp Recreation	8
Blackhall Recreation	0
Hardwicke Street Community Centre	0
Coastline-River- Walks	6
Shades of 50 plus	124
Circle50	2
Lunchtime walks in step aside	Item 0
Lunchtime walks	1070
Women 50 +	Stopped after 24272
SeniorLine	12
Active Ageing programme	25
Ireland Active	2678
Gallery Tour	9648
whats on stage	15314
Saturday Bingo	1
Aged	10417

Table 3.1: The table lists all the keywords that were used to scrape data from Twitter and a count of number of tweets retrieved from the respective keywords.

3.4 Data Transformation

Data fetched in raw JSON needed to be transformed into a DataFrame which is stored as CSV to use the LSA tool, a library that takes CSV as one of the accepted data files formats.

Each entry in the JSON file represents a tweet. It has 30 fields, as shown in table 3.2. The field 'user' has 23 subfields, one of which is 'location'. If in the field 'location', the name is not an Irish city name or has a 'Null', then that tweet is skipped and not transformed into a DataFrame. It reduced the overall tweets to a count of 3887, as a majority of the users have not specified their location in their Twitter profiles. And, users outside of Ireland were not considered for this research. Moreover, only the text in the field, 'content', is pulled into the DataFrame and saved as CSV.

The final CSV has combined data for all the JSON files extracted from multiple keywords.

JSON fields

_type	url	date	content
renderedContent	id	user	replyCount
retweetCount	likeCount	quoteCount	conversationId
lang	source	sourceUrl	sourceLabel
outlinks	tcooutlinks	media	retweetedTweet
quotedTweet	inReplyToTweetId	inReplyToUser	mentionedUsers
coordinates	place	hashtags	cashtags

Table 3.2: The table lists all the fields in a tweet fetched in JSON format. The fields "user" and "mentionedUsers" have further subfields that give information about a user, such as 'username', 'id', 'displayname', 'verified', 'location' and the likes of such.

3.5 Data Preprocessing

It was necessary to preprocess the data before using it to form meaningful semantic spaces. To preprocess the data, following steps were executed:

1. Import the Combined CSV file into a DataFrame and libraries.

2. Dealing with missing values by checking if text is empty or have only whitespaces or characters other than alphabets.

3. Remove hyperlinks attached in the tweets as these might get used as gibberish words while forming a semantic space.

4. Remove text having language other than English and emojis by removing rows with non-ASCII characters.

5. Remove tagged usernames of people.

6. Removing stop words from the data.

After performing these steps to clean the data, I got 1366 entries in the DataFrame, a Python object to represent a table. Each entry is stored in a different CSV file to create a term-by-document matrix using LSA.

3.6 Creating Semantic Space

There are many ways to create a Semantic Space from a given set of documents. LSA is one of the tools to create a semantic space. LSA works best for small data of a few thousand entries as it does Full-Singular Value Decomposition. Since the data is reduced to 1366 entries after data transformation and preprocessing, using LSA seemed to be a suitable choice.

LSA first creates a term-by-document frequency matrix, then applies weightings (optional) and finally applies SVD and dimensionality reduction. These tasks performed by LSA align well with the requirements for the LSAfun tool. That's another reason I've chosen LSA to create the semantic space (LSA space). The following sections describe each step in creating the semantic space.

3.6.1 'textmatrix'

LSA performs tokenisation of the text from the data and creates a term-by-document frequency matrix from all the documents in a given repository, using 'textmatrix'. The frequency of a term (word) in a given document is input inside the matrix. Each row of the matrix represents the terms, and each column represents the documents.

'textmatrix' takes 14 arguments such as 'stemming', 'language', 'stopwords' and 'vocabulary'. These arguments contribute to preprocessing the text. One of the arguments, 'minWordLength', was set to 2 so that single-letter terms get omitted. Here is a small example of the formation of a textmatrix.

Let three documents contain the following texts:

Document 1: 'I got the results. I passed the exam.'
Document 2: 'The science exam was easy. I passed!'
Document 3: 'I aced the science exam. My first exam!'

3.3 shows the resulting term-by-document frequency matrix.

	D1	D2	D3
got	1	0	0
the	2	1	1
results	1	0	0
passed	1	1	0
exam	1	1	2
science	0	1	1
was	0	1	0
easy	0	1	0
aced	0	0	1
my	0	0	1
first	0	0	1

Table 3.3: The table shows a term-by-document frequency matrix created using the LSA textmatrix for the three documents.

A summary of the textmatrix created using the preprocessed data is shown in 3.4.

vocabulary	*documents*
2428	1222
freqs not '0'	*max term length*
9242	18
non-alphanumerics in terms	
0	

Table 3.4: The table shows a summary of the textmatrix created using the data collected from chatter on Twitter platform.

3.6.2 SVD and Dimensionality Reduction of Topics: LSA Space

Through singular value decomposition (SVD), a complex matrix is factored into three different matrices that give inferences about the matrix. According to Gundersen (2018), if we stretch or compress a matrix after rotating, a new orientation might be present. The transformed matrix has singular values which can tell a lot about the matrix. These singular values are the length and width of the matrix.

For a large matrix, performing SVD takes a lot of computing time. However, it is an essential part of statistical techniques.

LSA library in R has the functionality to compute the SVD of a rectangular matrix. For the matrix created above, I have performed SVD using LSA to reduce dimensionality to 5. It means that the text data has some latent features that are found using LSA, which then reduces the number of terms.

The idea behind reducing dimensions was to observe any relationship between the two classifications: Sentiments and POS tags and the dimensions. A binary setting like positive-negative or adjective-noun, in a two-dimensional topic, is unlikely to be seen if not fully impossible. Hence, I reduced the dimensions to 5 topics to check if one of them could relate to sentiments or the word type.

After Dimensionality reduction, the matrix is converted into three matrices, the term vector matrix (tk, the left singular vectors or term vectors), document vector matrix (dk or document vectors) and a diagonal matrix (sk) which consists of singular values.

The three matrices are reduced to a given number of dimensions to finally form a latent semantic space. Here's how the matrix is denoted, where k is the number to dimensions that needs to be reduced:

$$M_k = \sum_{i=1}^{k} t_i.s_i.{d_i}^T$$

Here's a short snippet of dimensionality reduction to 5 in a term vector matrix 3.2 and document vector matrix 3.3 after SVD:

```
> l$tk
                      [,1]          [,2]          [,3]          [,4]          [,5]
comprehensive 4.414787e-03 -5.848916e-03 -5.482859e-03  3.715903e-03 -3.196393e-03
expeditiously 1.173919e-03 -2.056985e-03 -1.104550e-03  1.509208e-03 -9.869086e-04
framework     3.988486e-03 -7.783150e-03  6.100853e-03  8.217471e-03 -1.681734e-03
human         3.584078e-02 -1.089428e-03  1.201734e-02  3.407646e-03  1.179738e-02
move          7.751081e-03 -1.352597e-03  5.890847e-04  4.167992e-03 -1.232410e-03
need          1.087567e-01 -1.839669e-01 -1.130797e-01  1.100242e-01 -8.831596e-02
report        1.795935e-02 -2.678284e-02  8.518119e-03  4.798090e-04  2.052006e-03
towards       1.102459e-02 -1.496714e-02 -1.061565e-02  7.697109e-03 -6.735434e-03
age           6.935700e-02 -1.045374e-01  1.347410e-01  1.523959e-01 -1.540898e-01
as            7.922024e-03 -1.010966e-02  7.515558e-04 -6.855899e-04 -1.066570e-02
derelict      2.428849e-03 -3.656281e-03  8.319603e-04 -2.987546e-03 -7.708775e-03
easily        2.428849e-03 -3.656281e-03  8.319603e-04 -2.987546e-03 -7.708775e-03
grow          6.809102e-03 -1.011754e-02 -3.302830e-03 -4.357923e-03 -1.987568e-03
old           2.959557e-02 -4.345141e-02  1.342130e-02  1.641996e-02 -4.139977e-04
```

Figure 3.2: The figure shows a Term Vector Matrix post Dimensionality Reduction to 5. The column on the left are a list of terms.

```
> l$dk
                [,1]          [,2]          [,3]          [,4]          [,5]
0.csv     1.497039e-02 -2.239624e-02 -1.066921e-02  1.449521e-02 -9.335068e-03
1.csv     3.097387e-02 -3.980921e-02  8.036175e-03 -2.869392e-02 -7.291651e-02
10.csv    2.243651e-02 -1.908744e-02 -2.699426e-02  1.105995e-02 -1.033196e-04
100.csv   1.413049e-03 -1.210400e-03 -3.941660e-05 -8.042978e-05  8.994399e-04
1000.csv  9.020049e-03 -6.522894e-03 -4.585304e-03  2.519085e-03 -3.467295e-04
1001.csv  3.167310e-03 -4.227088e-03  2.156327e-03 -4.713130e-03  9.859252e-03
1002.csv  2.454771e-02 -4.225491e-02  9.846972e-03 -2.704977e-02  4.171728e-03
1003.csv  1.305307e-02 -2.038675e-02 -9.416967e-03 -1.752538e-03  1.217584e-02
1004.csv  1.542836e-03 -3.391958e-03  4.620759e-04  2.133726e-04  3.086225e-04
1005.csv  5.289270e-02 -9.860254e-02 -1.121287e-02  7.210716e-03  3.787162e-02
1006.csv  1.721006e-02 -4.206252e-02  4.252894e-02 -1.131822e-02  5.253199e-02
1007.csv  2.450013e-02 -1.501635e-02 -1.051257e-02  5.084905e-03 -5.641856e-03
1008.csv  1.217565e-02 -1.109773e-02 -1.169421e-02  6.371081e-03 -5.481984e-03
1009.csv  9.543383e-02 -4.601509e-02  8.654919e-02 -1.964694e-02  8.362882e-02
101.csv   2.689692e-02 -2.880831e-02 -2.528015e-02 -2.039127e-02 -2.574954e-02
```

Figure 3.3: The figure shows a Document Vector Matrix post Dimensionality Reduction to 5. The column on the left indicate names of the documents.

From these figures, we can see that each keyword has been given a value against each of the columns i.e. the dimensions or topics. Higher values indicates higher relation to that topic. Similarly, documents are also given values against each of the topics. I will try to identify if these values will help finding any relations concerning sentiments and word-types.

3.7 Neighbourhood Computation using LSAfun

Certain keywords are used to fetch n number of neighbours based on Cosine similarity, suing the tool, LSAfun. For example, if n=10, here's how neighbours are visualised: 3.4

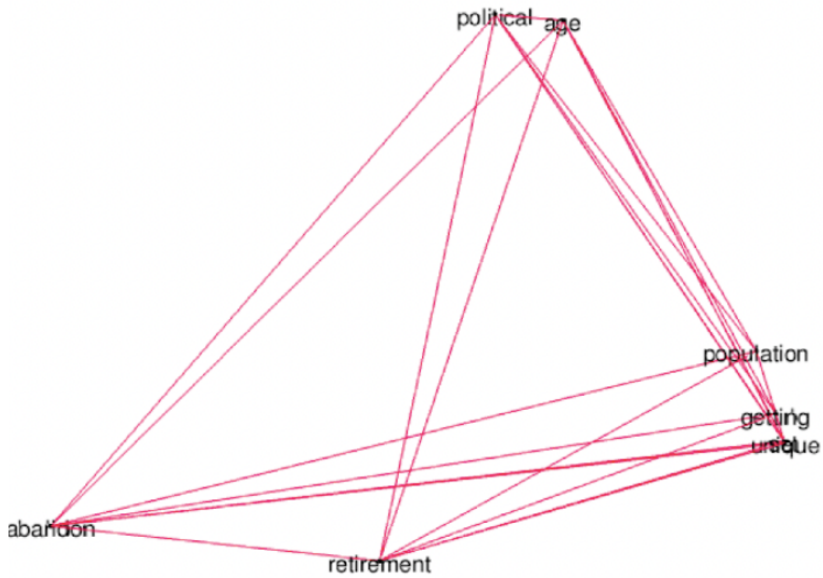

Figure 3.4: The figure shows a neighbourhood computation using Cosine Similarity in LSAfun tool for the keyword: age

The Lsafun can also retrieve the similarity value of each term for the given keyword. However, this visualisation helps in identifying how close each term is without having to read through numerical values. The figure also shows overlapping words, which indicates the closeness of terms. A point to observe is that age has come along with terms like retirement and politics. It might infer a few things. Observations like these result from a semantic space visualisation.

3.8 More Dimensions for Classification: Sentiments and POS tags

To perform analysis on the data, and gather inference, I added two more dimensions to the existing 5 topics, Sentiments and POS tags.

In classification based on Sentiments, terms are classified as Positive, Negative or Neutral. I have used 'SentimentIntensityAnalyzer' Hutto and Gilbert (2014) provided by the NLTK Sentiment Analysis Package to extract sentiments of each term. It returns a float (1.0) for the particular sentiment based on the input term.

For Parts of speech tagging, I have used NLTK is used to identify terms as Adjectives or Nouns.

An updated term matrix after adding two more dimensions can be seen here: 3.5. In the figure, the column names starting with 'V' indicates the five topic fields viz., V1-V5.

	word	V1	V2	...	V5	emotion	word_type
0	comprehensive	0.004415	-0.005849	...	-0.003196	Positive	Noun
1	expeditiously	0.001174	-0.002057	...	-0.000987	Neutral	Other
2	framework	0.003988	-0.007783	...	-0.001682	Neutral	Noun
3	human	0.035841	-0.001089	...	0.011797	Neutral	Noun
4	move	0.007751	-0.001353	...	-0.001232	Neutral	Noun
5	need	0.108757	-0.183967	...	-0.088316	Neutral	Noun
6	report	0.017959	-0.026783	...	0.002052	Neutral	Noun
7	towards	0.011025	-0.014967	...	-0.006735	Neutral	Other
8	age	0.069357	-0.104537	...	-0.154090	Neutral	Noun
9	as	0.007922	-0.010110	...	-0.010666	Neutral	Other

Figure 3.5: The figure shows a new matrix constituting of five topics along with two more dimensions - Sentiments and POS tags.

In the part of speech tagging, I have only considered nouns and adjectives and named the rest of the word types as 'others'. This was done just to experiment with how nouns and adjectives will relate with the rest of the dimensions. Since adjectives are the words that describe a noun, they can give some information that noun doesn't provide. Moreover, in linguistics, it might change the information provided by a noun altogether, according to Wikipedia contributors (2022c). Hence I have chosen these two tags for the experiments.

Chapter 4

Experiments and Results

tapply (R base package)

Firstly, I have calculated the average values of sentiment criteria and word-type classification individually for words of a particular topic. I have performed this calculation using a statistical measuring function, the 'tapply' function, that can operate on jagged arrays.

I have calculated it before all the tests to visualise if any differences are visible from the tests. These don't have significance attached to them. However, this is to visualise where the differences are. The tests would identify if there's any interaction between values of all the topics and word type and/or emotions. Particularly, inspecting the mean values can tell where the difference lies.

As a reference in the following sections, the topics are denoted as V1 - V5, the sentiments are addressed by 'emotions' and the parts of speech are given by 'word-type'.

The listing at 4.1 show the mean values for the three sentiments.

Topics	**Negative**	**Neutral**	**Positive**
V1	0.003111370	0.006841019	0.007845983
V2	-0.001707900	-0.004618902	-0.006152435
V3	-0.0002258408	-0.0018323566	-0.0012091543
V4	0.0010360592	0.0004336136	0.0003916417
V5	-0.0013366357	-0.0008142397	0.0017494242

Table 4.1: The listing denotes Mean - value of sentiments with the topics.

The listing at 4.2 show the mean values for the noted word-types.

As a function of emotion, V1 i.e. Topic 1, has different means as compared to other topics, seen here 4.1. It can be seen that v1 is lowest for negative emotions and largest for positive and neutral is in between. This seems to be a natural monotonic trend. On

Topics	**Adjective**	**Noun**	**Other**
V1	0.007222005	0.005622337	0.009328355
V2	-0.005633251	-0.004319079	-0.004797740
V3	-0.0034869502	-0.0009702311	-0.0025884241
V4	0.0009051726	0.0004271106	0.0003715615
V5	0.0008787915	-0.0012368061	0.0005197059

Table 4.2: The listing denotes Mean - value of word-types with the topics.

the other hand, V2 appears to get further smaller. It greatest vale has negative emotion, middle value for neutral emotion and smallest value for positive emotion, even when all three of the values are negative.

4.1 Experiments

Initially, I have the null hypothesis that the variables i.e. topics are normally distributed. I ran some tests to address this null hypothesis. Some of test also identify if there is any significance of the mean values extracted in relation to emotions and word-type.

4.1.1 Shapiro-Wilk Normality test

Shapiro-Wilk is a normality test used in statistics. Here, I have attempted to answer the question of whether the variable i.e. Topic is normally distributed or not. The results of the test is shown below, in table 4.3. W is the value of the Shapiro statistics. A

Topics	W	**p-value**
V1	0.28043	<2.2e-16
V2	0.31555	<2.2e-16
V3	0.29343	<2.2e-16
V4	0.18108	<2.2e-16
V5	0.24134	<2.2e-16

Table 4.3: The listing denotes outcome of Shapiro-Wilk normality test.

point to identify is the magnitude of that statistic given the number of items.So a thing to pay attention is looking at the p-value. A p-value <0.05, shows that the results are statistically significant, as 0.05 is a typical threshold. Since all p-values in the listing are extremely small, close to zero, we can say that the results are highly significant. Hence, I have rejected the null hypothesis that each of those values, V1 through V5, are normally distributed.

This test was also conducted to determine if a parametric or non-parametric test should be employed. A parametric statistic assumes the population's distribution from which the samples were drawn. Moreover, no assumptions are made when calculating a non-parametric statistic; the information can be gathered from a sample that doesn't have a clear direction.

Both Shapiro Wilk and Kruskal Wallis tests are non-parametric tests and thus well suited.

4.1.2 Kruskal-Wallis Rank Sum Test

Test for Emotion

In order to assess the effective emotions with the topics, I have now ran Kruskal-Wallis tests. This is a non-parametric test that identifies if there is existence of any statistical differences between groups of the variable that depend on a dependent variable.

Since we have mean values of V1 as it distributes acress each ofthe three emotions, this test is trying to answer the wuestion, "Are those differences significant in a statistical sense?"

The resuts of the test can be seen in 4.4. If we observe the p-value from the results of

Topics	**Kruskal-Wallis chi-squared**	**df**	**p-value**
V1	26.403	2	1.848e-06
V2	28.126	2	7.807e-07
V3	1.6725	2	0.4333
V4	0.82922	2	0.6606
V5	8.1407	2	0.01707

Table 4.4: The listing denotes outcome of Kruskal-Wallis rank sum test for emotions.

this test, we can reject the null hypothesis that there is no interaction between the value of V1 and emotion. It confirms that the results are significant for topic V1 as the p-value is less than the threshold. We can also reject the null hypothesis for V2 and V5. However, we don't reject the null hypothesis for V3 and V4.

Therefore, these topic values are sensitive to emotion. They might not be completely determined by emotion but there is some sensitivity.

Test for word-type

Same test is now ran for the word-types. The reuslts acn be seen in 4.5. It is evident from

Topics	Kruskal-Wallis chi-squared	df	p-value
V1	11.472	2	0.003228
V2	6.3808	2	0.04116
V3	1.75	2	0.4169
V4	3.8112	2	0.1487
V5	1.8138	2	0.4038

Table 4.5: The listing denotes outcome of Kruskal-Wallis rank sum test for word-type.

the results that the topics V1 and V2 rejects the null hypothesis. It means that the value of V1 and v2 interacts with the word-type. To visualise where the differences might lie, we can look at the table of means for word-type at 4.2. We can note that adjectives have higher value than noun for V1 and lower value than noun in V2. However, it doesn't tell much about the interactions in comparison to mean values of other topics.

On the other hand, V3, V4 and V5 do not reject the null hypothesis. Another point of observation is that V3 and V4 have shown no interaction with either emotions or word-type.

4.1.3 Interaction between Emotions and Word-type

I have now tried to identify the interaction of word-type and emotions. Again, using 'tapply' in R, an average of emotions and word-type for each topics are calculated. These values can be seen in the listings, 4.6, 4.7, 4.8, 4.9 and 4.10.

	Negative	Neutral	Positive
Adjective	0.003257794	0.007576314	0.007706325
Noun	0.003121976	0.005504930	0.008284702
Other	0.002982621	0.010131049	0.006847968

Table 4.6: The listing shows interaction of V1 with both emotions and word-type.

	Negative	Neutral	Positive
Adjective	-0.002209919	-0.005536710	-0.007708879
Noun	-0.001553114	-0.004446092	-0.005109680
Other	-0.001803772	-0.004685250	-0.007500771

Table 4.7: The listing shows interaction of V2 with both emotions and word-type.

	Negative	Neutral	Positive
Adjective	-0.001077805	-0.0043976171	-0.0009180992
Noun	0.000476715	-0.0009253008	-0.0023179204
Other	-0.001624941	-0.0031638904	0.0013648611

Table 4.8: The listing shows interaction of V3 with both emotions and word-type.

	Negative	Neutral	Positive
Adjective	1.049232e-04	0.0013680519	-0.0006085221
Noun	1.654007e-03	0.0003475780	0.0002709501
Other	-7.172981e-05	0.0002581755	0.0015345197

Table 4.9: The listing shows interaction of V4 with both emotions and word-type.

	Negative	Neutral	Positive
Adjective	-0.0008586884	0.0007628272	0.002207997
Noun	-0.0016394891	-0.0015448802	0.001617229
Other	-0.0008083096	0.0004719320	0.001701932

Table 4.10: The listing shows interaction of V5 with both emotions and word-type.

In one of the topics, V3, the value of Noun has good amount of difference in the mean values for negative and positive emotions. The nouns seem to be highly negative with least positive relations. Similarly in V4, nouns have comparatively more negative emotions than positive.

The interaction of mean values of the two characteristics in relation to topics, failed to give significant results, for inference. An attempt on plotting might show some differences.

4.1.4 Interaction Plots

A better understanding might take place from the visualisation of these interactions. I have used Interaction plots in R to provide a graphical summary of tables generated before. These can be seen in 4.1, 4.2, 4.3, 4.4 and 4.5.

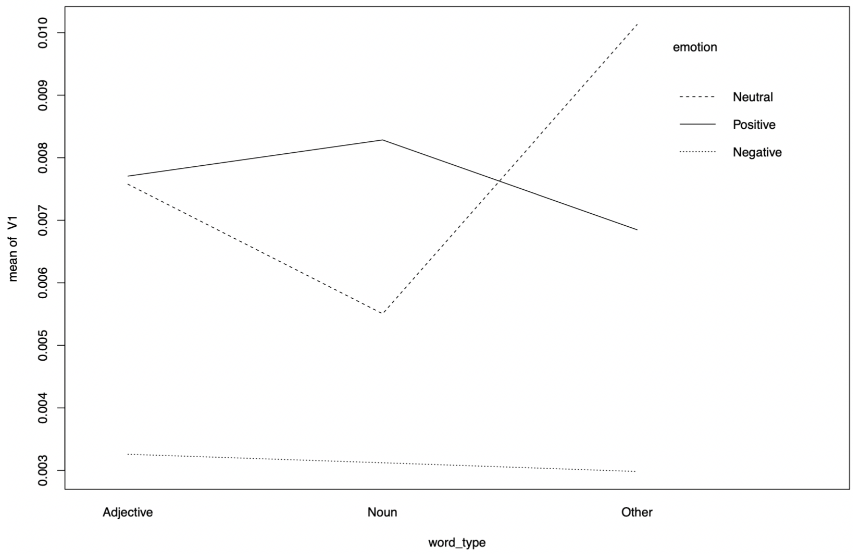

Figure 4.1: Interaction plot for V1

We can see that in V1, i.e. the first topic, there is very less relation to negative emotions. However, most relation is to neutral emotions. It doesn't not give enough information to make any inference.

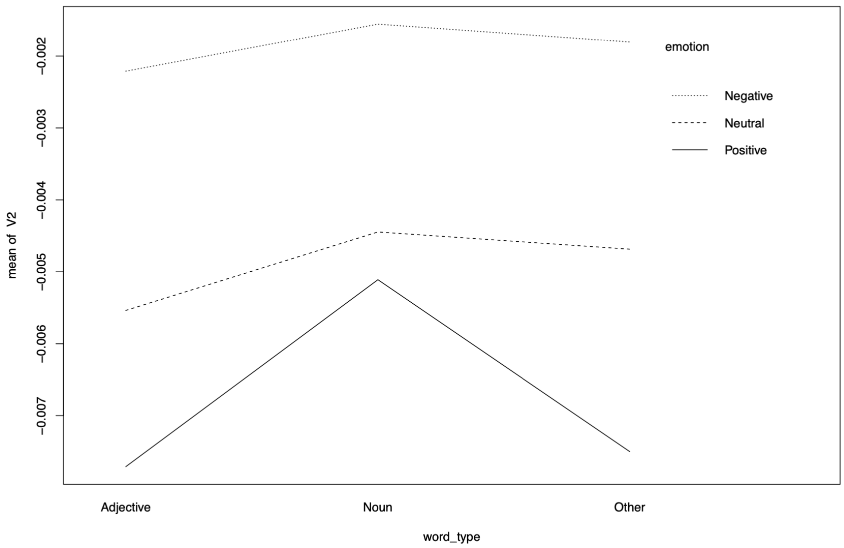

Figure 4.2: Interaction plot for V2

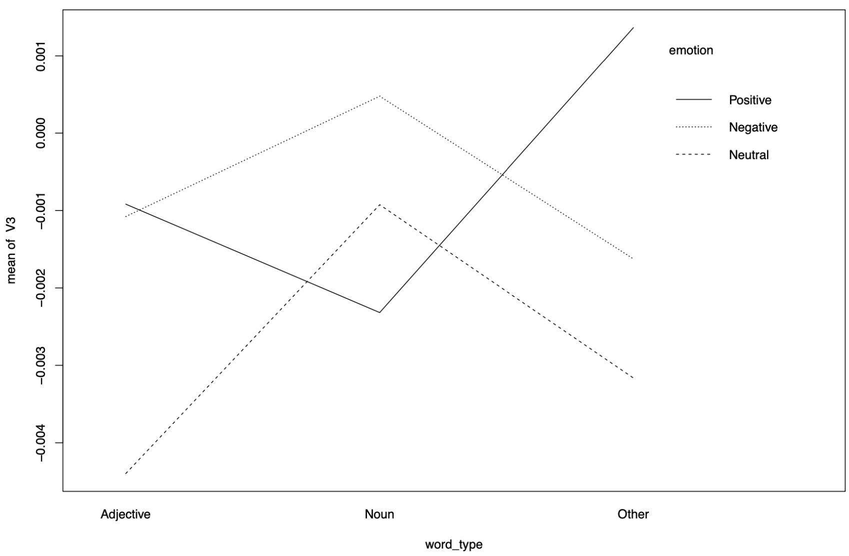

Figure 4.3: Interaction plot for V3

In the second topic, V2, there is high relation to negative emotions and least to positive emotion. This topic consists of high negative emotion Nouns. The third Topic i.e. V3 has highest positive emotions. It has both positive and negative adjectives.

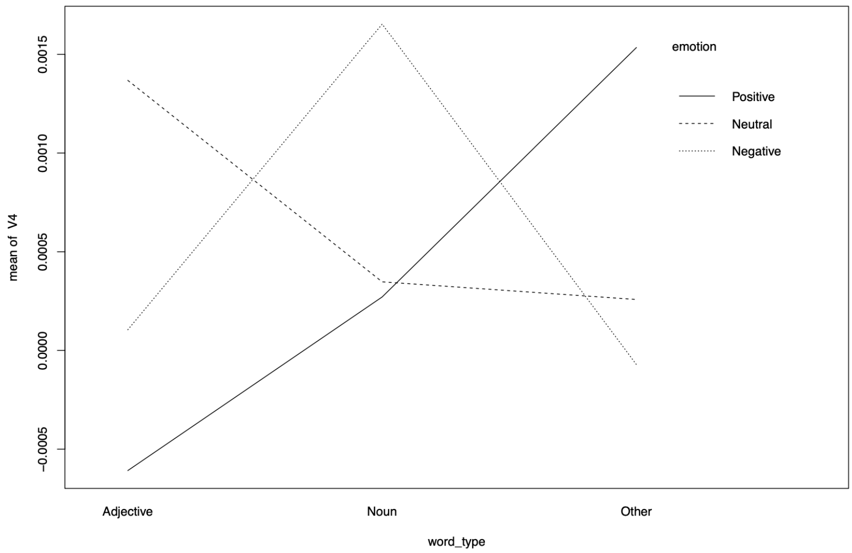

Figure 4.4: Interaction plot for V4

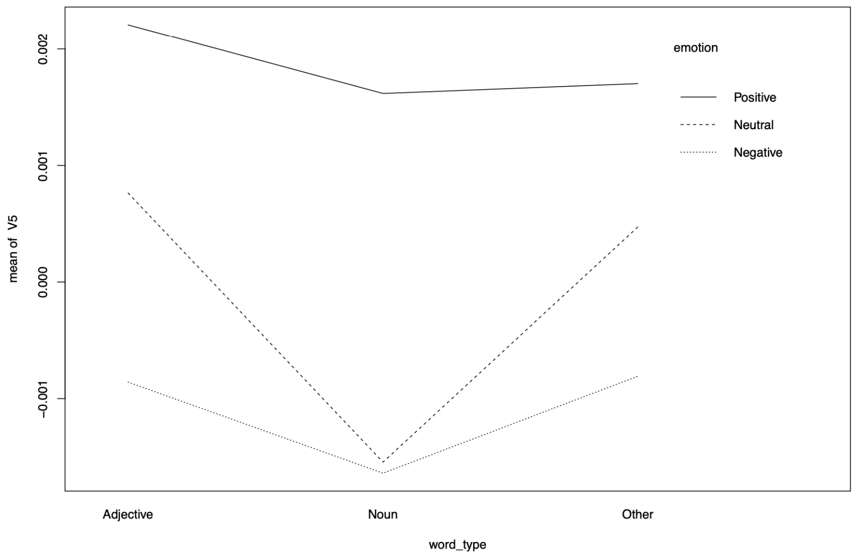

Figure 4.5: Interaction plot for V5

The fourth topic, has no specific relation to either of the characteristics. No inferences can be made using this topic. The fifth topic shows a good relation to positive emotions and have maximum positive adjectives. This topic has least relation to negative emotions. Another point of observation is that all the nouns are also positively related to this topic.

4.2 Results

I have associated the observations seen above with the actual data to find out if there is a confluence of noun with positive or negative emotions. Therefore, knowing the possibility of using this guide for inspecting terms.

On arranging topic 5 into descending order, I have found the following 30 top values: 4.6

	word	V5	emotion	word_type
72	learning	0.543980	Neutral	Other
116	new	0.239227	Neutral	Adjective
137	lifelong	0.173232	Neutral	Noun
120	limerick	0.142796	Neutral	Noun
51	across	0.116084	Neutral	Other
119	inspired	0.110883	Positive	Other
223	promote	0.071232	Positive	Noun
228	adult	0.068180	Neutral	Noun
291	irish	0.066901	Neutral	Adjective
64	education	0.056258	Neutral	Noun
136	engage	0.055032	Positive	Noun
45	formal	0.050272	Neutral	Adjective
113	forward	0.046944	Neutral	Other
114	looking	0.045151	Neutral	Other
105	us	0.038958	Neutral	Other
66	support	0.038026	Positive	Noun
54	event	0.037439	Neutral	Noun
46	informal	0.034210	Neutral	Adjective
22	like	0.033900	Positive	Other
1038	open	0.031223	Neutral	Adjective
117	empower	0.031195	Neutral	Noun
118	inspire	0.030093	Positive	Noun
182	two	0.029972	Neutral	Other
264	year	0.028668	Neutral	Noun
121	participate	0.028499	Neutral	Other
236	local	0.027451	Neutral	Adjective
154	fantastic	0.026704	Positive	Adjective
189	learn	0.026340	Neutral	Noun
312	interested	0.026139	Positive	Adjective
73	look	0.025854	Neutral	Noun

Figure 4.6: Top values of V5

If I find neighbours of these keywords 'limerick', 'learning', and 'lifelong', then I might find the generalisation of what this topic is trying to tell. Here are the neighbours found using cosine similarity: 4.8, 4.7 and 4.9.

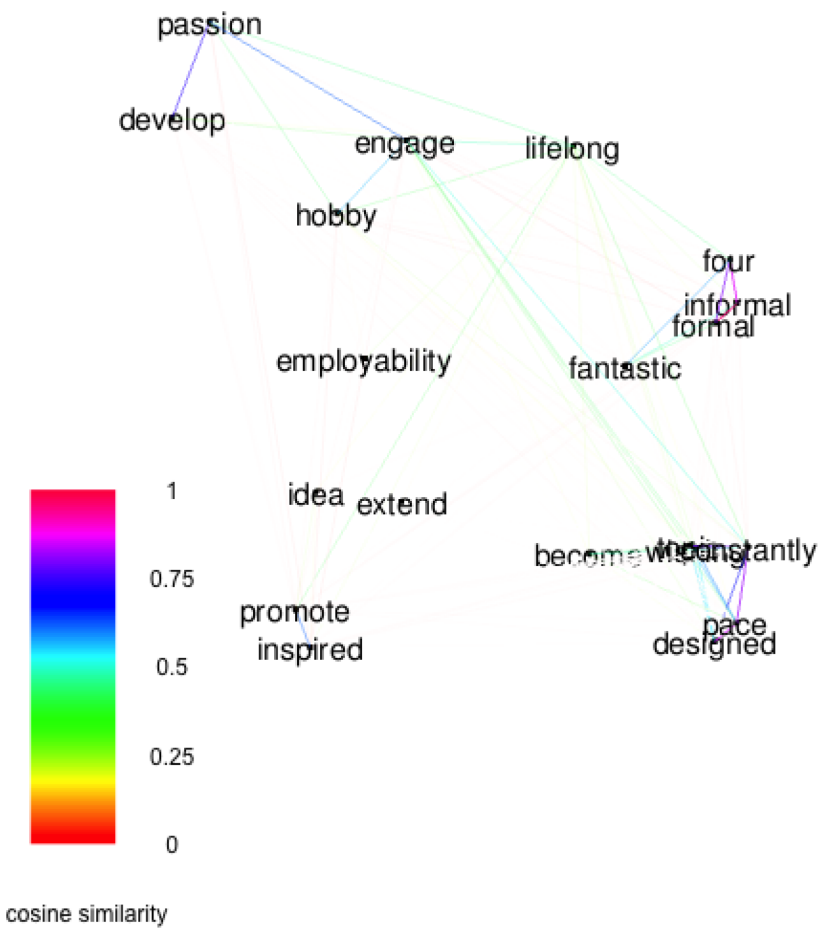

Figure 4.7: Neighbours for keyword - lifelong. On the left, the color range of cosine similarity is given for reference.

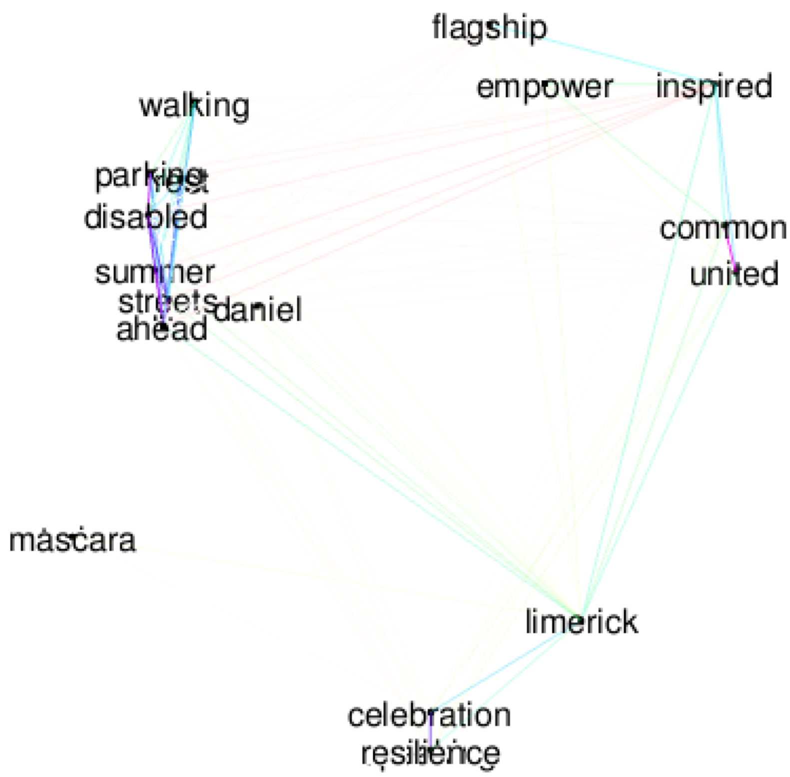

Figure 4.8: Neighbours for keyword - Limerick

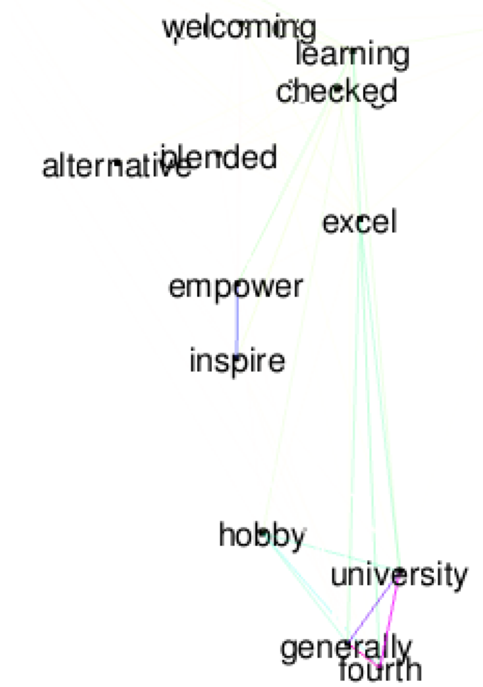

Figure 4.9: Neighbours for keyword - learning

These top keywords and their neighbours could infer that the topic could be in relation to learning and there could be a positive response from the chatter in relation to that topic. On inspecting 'limerick' in the overall collection of data, I get 40 texts, out of which the common five are given below

- "FEATURE FRIDAY This feature Friday, we are featuring ProFi Fitness School who are based in Dublin, Cork, Galway and Limerick. If you would like to find out more visit PFSchool #education #feautrefirdays #qualityassured #irelandactive"

- "Limerick pushes ahead of other counties in Ireland. More homes, well lit streets for walking with family, cycling ahead of the rest for active travel, an disabled app for parking bays for people with mobilty issues. Come visit Limerick this summer !!!"

- "Learn something new in 2022! You can #getinvolved and host an event for the festival. Submit your ideas for events by March 11th at . #unesco #learningcites #lifelonglearning #learningcommunities #limerickfestival #limerickedgeembrace #LLLFestival2022"

- "Learning Limerick was delighted to be profiled and have its case study included in the recent UNESCO Publication 'Entrepreneurship Education for Learning Cities'. See report at - #lifelonglearning #unesco #learningcities"

- "#LearningAmbassadors are Limerick people who promote #lifelonglearning in their workplaces, homes and communities by sharing their own lifelong learning stories. #lovelearning #limerickfestival #learningcommunities"

On plotting neighbours for elderly, we get multiple close keywords. One of the nearby keyword was 'aimee', as seen in 4.10.

On searching from the original tweets in the data, I found only 2 tweets. These are given below:

- "Today is 'Random Act of Kindness Day'.

  Sports mentor Aimee presented Liz from Jigsaw NI with a bouquet of flowers as we facilitated her group of elderly on our active ageing programme.

  Thank you for everything you do Liz. Keep up the good work.

  #RandomActsofKindnessDay"

- "on the train to Limerick and there's a group of six 50+ year old women, big Karen energy. They got less painful to listen to when they started talking about makeup and the new Sculpted By Aimee mascara x"

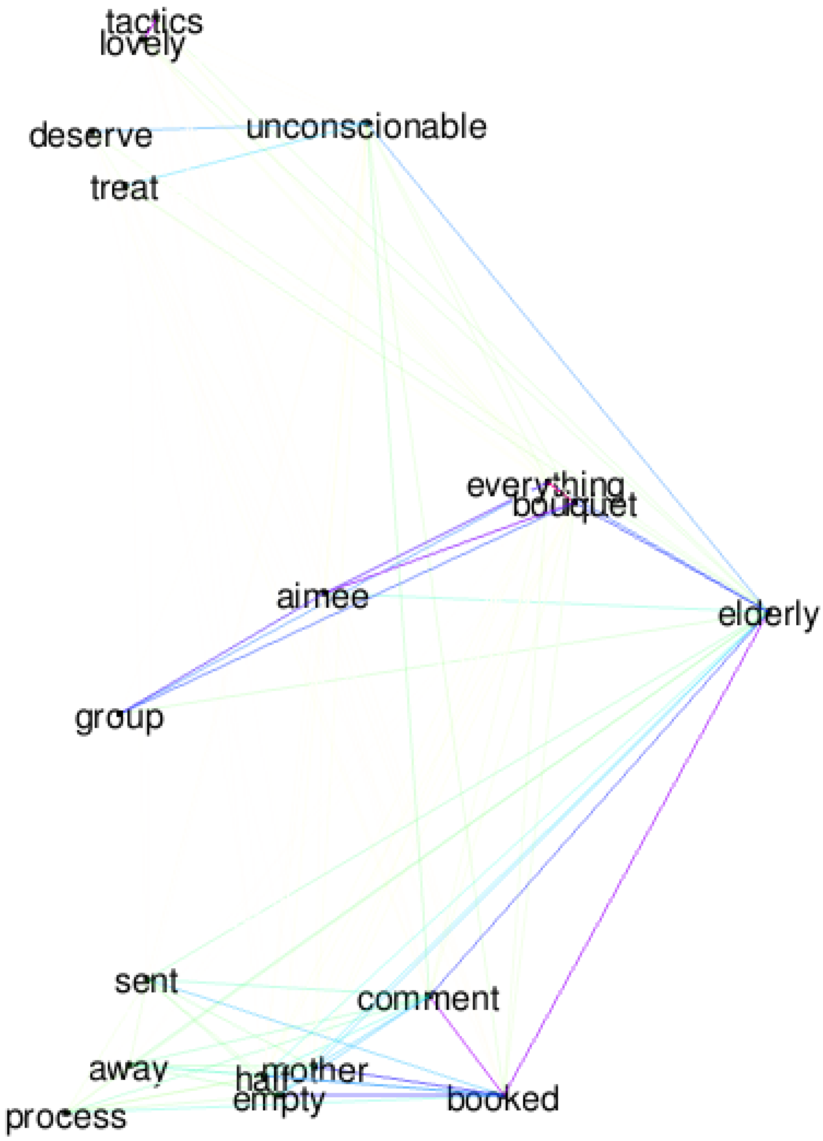

Figure 4.10: Neighbours for keyword - elderly

Coincidentally, both the tweets had that name, however, the first tweet was in fact in relation to the Active Ageing programme.

Chapter 5

Conclusion & Future Work

In this research, I harvested and preprocessed online chatter available on Twitter in an attempt to analyse active ageing in Ireland. The text was tokenised and formed into a matrix. I was able to reduce dimensionality to five topics. These topics had small p-values in the Shapiro-Wilk normality test. It, therefore, rejected the hypothesis that there was a normal distribution of these five topics.

I observed the relation of topics to emotions in the interaction plot for V5 at 4.5; its high relation to positive emotion. The collection of top words in V5, neighbours of keywords in V5 and actual tweets proved that relation was present. However, I could not see a relation in other topics, as there was no distinct interaction. Hence, I cannot conclude that there is, in fact, a relation to emotions. Moreover, I couldn't prove the presence of a relationship between emotions and word type.

Limitation

This research is limited, as I only considered English text for harvesting data. Most users on Twitter have no location details on their accounts. It impacted hugely in collecting the data. The results could be very different in the presence of more data. Collecting data from other networking platforms, such as Facebook, Reddit, etc., can also be considered for better results.

Future work

Identifying a first-person point of view in texts can be one of the future works. It could help analyse how the organisations, such as the Active Ageing Programme, approach online chatter and how other people are responding or conversing about them. First-person text can also represent the elderly, for example. So we can identify what they are trying to express online and how other people engage with these messages. Similarly, seeing how people respond to the younger generation can help us make inferences concerning the behaviour of people towards other people of different demography.

Many people have used emoticons in their tweets. Interpreting them and replacing them with keywords that best describe them could be a way of not ignoring information in the form of emojis. One can assume that if a person has an urge to say something, they would hardly use emojis instead of words to express themselves. So, the use of emojis might be a less important or perhaps sarcastic form of expression. Machine learning approaches in finding out whether the tweets show actual sentiment or an act of sarcasm or euphemism can also help identify the true meaning of the message carrying those emojis.

Furthermore, many tweets had web links inside the messages. A future approach could be analysing content from the attached websites in the tweets. It would further add to the data and help get more relevant results.

Lastly, forming clusters of tweets that refer to a common point of interest, could be another future work. It could help fetch the themes concerning active ageing discussed online.

Bibliography

Active Retirement Ireland (2022). Hi Digital programme. https://activeirl.ie/hidigital/. [Online; accessed 19-August-2022].

Age and Opportunity (2022). Age & Opportunity Engage. https://ageandopportunity.ie/engage/. [Online; accessed 19-August-2022].

Age Friendly University (2022). Age-Friendly Learning Opportunities at DCU. https://www.dcu.ie/agefriendly/age-friendly-learning-opportunities-dcu. [Online; accessed 19-August-2022].

Brownlee, J. (2020). What Is Meta-Learning in Machine Learning? https://machinelearningmastery.com/meta-learning-in-machine-learning/. [Online; accessed 19-August-2022].

Gundersen, G. (2018). Singular Value Decomposition as Simply as Possible. https://gregorygundersen.com/blog/2018/12/10/svd/. [Online; accessed 19-August-2022].

Günther, F., Dudschig, C., and Kaup, B. (2015). Lsafun-an r package for computations based on latent semantic analysis. *Behavior research methods*, 47(4):930–944.

Hoare, J. (2022). Gradient Boosting Explained – The Coolest Kid on The Machine Learning Block. https://www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/. [Online; accessed 19-August-2022].

Hutto, C. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the international AAAI conference on web and social media*, volume 8, pages 216–225.

Ireland Active (2009). Active Ageing Programme. ://irelandactive.ie/active-ageing-programme/. [Online; accessed 19-August-2022].

Karamitsos, I., Albarhami, S., and Apostolopoulos, C. (2019). Tweet sentiment analysis (tsa) for cloud providers using classification algorithms and latent semantic analysis. *Journal of Data Analysis and Information Processing*, 7(4):276–294.

Kurama, V. (2020). A Guide to AdaBoost: Boosting To Save The Day. `https://blog.paperspace.com/adaboost-optimizer/`. [Online; accessed 19-August-2022].

Landauer, T. K., Foltz, P. W., and Laham, D. (1998). An introduction to latent semantic analysis. *Discourse processes*, 25(2-3):259–284.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*.

Pekar, V., Najafi, H., Binner, J. M., Swanson, R., Rickard, C., and Fry, J. (2021). Voting intentions on social media and political opinion polls. *Government Information Quarterly*, page 101658.

Poldi, F. and Community, P. (2019). Twint - twitter intelligent tool. `https://github.com/twintproject/twint`. [Online; accessed 19-August-2022].

Python Community (2022). Snscrape social networking service scraper. `https://pypi.org/project/snscrape/`. [Online; accessed 14-August-2022].

Shi, X., Xue, B., Tsou, M.-H., Ye, X., Spitzberg, B., Gawron, J. M., Corliss, H., Lee, J., and Jin, R. (2019). Detecting events from the social media through exemplar-enhanced supervised learning. *International Journal of Digital Earth*, 12(9):1083–1097.

Tutorialspoint (2022). Time Series - LSTM Model. `https://www.tutorialspoint.com/time_series/time_series_lstm_model.htm`. [Online; accessed 19-August-2022].

Versloot, C. (2020). Differences between Autoregressive, Autoencoding and Sequence-to-Sequence Models in Machine Learning. `https://github.com/christianversloot/machine-learning-articles/blob/main/differences-between-autoregressive-autoencoding-and-sequence-to-sequence-models-i`
`md`. [Online; accessed 19-August-2022].

Wikipedia contributors (2021). Affinity propagation — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Affinity_propagation&oldid=1043691960`. [Online; accessed 18-August-2022].

Wikipedia contributors (2022a). Active ageing — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Active_ageing&oldid=1092192247. [Online; accessed 19-August-2022].

Wikipedia contributors (2022b). Adaboost — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=AdaBoost&oldid=1101642150. [Online; accessed 18-August-2022].

Wikipedia contributors (2022c). Adjective — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Adjective&oldid=1102587889. [Online; accessed 19-August-2022].

Wikipedia contributors (2022d). Long short-term memory — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Long_short-term_memory&oldid=1104694063. [Online; accessed 18-August-2022].