LUISS

Degree Program in
Management and Computer Science

Course of Advanced Coding for Data Analytics

Artificial Intelligence in the News: A Time Series-Based Sentiment Analysis of Media Perception of A.I. (2013 – 2024)

Prof. Alessio Martino

SUPERVISOR

Janhvi Goje - 286971

CANDIDATE

Academic Year 2022/2023

Abstract

This thesis investigates the evolution of media sentiment towards Artificial Intelligence (A.I.) over the past decade, with the goal of understanding how public perception has shifted alongside advancements in A.I. technologies. Using a dataset of news articles collected from The New York Times API, the study applied sentiment analysis using advanced models like RoBERTa to gauge the emotional tone of media coverage. The results revealed key sentiment trends, including a significant peak of optimism around 2018, followed by a sharp decline in 2020 due to growing concerns over ethical issues like job displacement and privacy. Time series analysis, using XGBoost forecasting models, demonstrated that media sentiment towards A.I. is stabilising, with predictions indicating moderate fluctuations around neutral values over the next five years. The research highlights the increasing normalisation of A.I. technologies in public discourse, reflecting society's gradual adaptation to its presence. The study also acknowledges limitations, such as the reliance on a single news source and restricted access to full article content. Future research could benefit from a broader range of data sources and a focus on specific events influencing sentiment shifts, as well as a comparative analysis of media versus public opinion on A.I. technologies.

Table of Contents

I. Introduction

In today's world, the media's coverage of A.I. and its subfields has become pervasive, reflecting the impact of these technologies across various industries and aspects of life. From academic research to professional environments and further into personal life, A.I. tools and systems are progressively being integrated at a rapid rate. As more people engage with A.I. technologies, the usability of these technologies increases as well and consequentially, complexities surrounding their capabilities and limitations also come to the forefront. While many appreciate A.I. for its transformative potential and ability to streamline everyday tasks, there is an undeniable growing concern about the ethical dilemmas, risks, and unintended consequences of its use (Peña-Fernandez et al., 2023).

Applications like ChatGPT, that have quickly become a go-to tool for students and professionals, allow users to automate mundane tasks such as drafting emails and demonstrate how A.I. can significantly enhance productivity. In 2023, MIT researchers conducted a comparative study to measure the time efficiency of ChatGPT. Participants were divided into a group that was permitted access to ChatGPT and a group that did not. The findings showed that the participants with access to the A.I. powered chatbot completed their tasks 40% faster than those who did not with higher output quality of 18% (Winn, 2023). On the other hand, some applications use A.I. can be used to create 'deepfakes', raising serious ethical concerns. Deepfakes, which involve media that is altered or entirely generated using A.I. to depict real or non-existent individuals, have emerged as one of the most prominent concerns due to their ability of spreading misinformation. One such case occurred in 2019. In the UK, a mother allegedly used manipulated audio recordings to falsely portray her ex-husband as violent and irresponsible in an ongoing child custody case. Fortunately, the experts noted inconsistencies that uncovered deepfakes being used to distort the footage. Nevertheless, this case raised serious concerns about the potential for deepfakes to undermine legal proceedings by fabricating (or altering) evidence in the court of law (Edwords, 2022).

Given this ever-evolving landscape, the motivation behind this work is to unravel how media sentiment towards A.I. changes, in particular over the last decade. The relationship between media and public opinion is often characterised by a mutual influence. Media can shape public opinion by setting agendas and public opinion can affect media coverage, as media outlets respond to public interests and concerns to maintain relevance (Stern, Livan and Smith, 2020). Therefore, by studying the shifts in media opinion over the years, we gain a broader insight into how the public's opinion has also evolved about A.I. technologies.

The remainder of this research is structured as follows: Chapter II (Theoretical Background) sets the study into motion reviewing all concepts, models and theory that serves as necessary background information to Chapter III, the Methodology. Chapter III mentions the step-by-step approach that was taken in the construction of the dataset, the sentiment analysis, the time series analysis and ultimately, the forecasting. It comprehensively explains the technical approaches that were taken in this study. Chapter IV comments on the Exploratory Data Analysis (EDA) and all intriguing findings made throughout the research via graphs, data analysis methods and the forecasting results. Finally, Chapter V briefly discusses the learning outcomes, limitations and area of further research. Lastly, we proceed with the final thoughts and summarise the research's main observation in Chapter VI, the Conclusion.

II. Theoretical Background

The first step of this project is to collect news data with relevant keywords to filter out what articles talk about A.I. topics. One method to do so is through web scraping. The process of retrieving and gathering a lot of data from websites—often automatically—is known as web scraping. This method is frequently applied to data mining and analysis, where the extracted information is kept for further use and analysis in a structured format, such as a database or spreadsheet (Mitchell, 2015). However, different restrictive data use policies of various news outlets give rise to several challenges in data collection. Many websites have terms of service or legal restrictions that limit or prohibit web scraping to protect intellectual property, user privacy, and the website's resources. For example, LinkedIn has a strict policy against web scraping. In the case of LinkedIn Corporation v. hiQ Labs, Inc., LinkedIn argued that hiQ's scraping activities violated the Computer Fraud and Abuse Act (CFAA), leading to a legal battle over the right to scrape publicly available data (Wallace, Berzon and Berg, 2022). This case highlights how data use policies can create hurdles, making it essential to comply with a website's terms of service before engaging in web scraping. Thus, although introducing varied news resources would help account for organisational biases, it is important to keep data collection limited to use a platform that allows data access for research purposes.

The New York Times API is a valuable resource in this context, as it provides access to a vast archive of articles, making it an optimal choice for this project. The New York Times offers APIs specifically designed to retrieve articles, metadata, and other relevant information. Meanwhile, other platforms either required expensive subscriptions or presented challenges that made data collection more difficult and less feasible. The Article Search API was selected as a primary tool as it is specifically designed to enable search through the New York Times' extensive archives and offers several key features:

- *Keyword Search:* The API allows users to search for articles using specific keywords, which is crucial for filtering content related to A.I.
- *Date Range Filtering*: To ensure the relevance and timeliness of the data, the API provides options to specify date ranges for the search, enabling collection articles within a particular timeframe.
- *Structured Data Access*: The API returns data in a structured format, including metadata such as headlines, abstracts, publication dates, and article URLs. This structured format facilitated easy integration into a database or spreadsheet, simplifying further analysis and reporting.

In order to understand the ‘opinion’ of media in relation to each article, sentiment analysis can be conducted. Sentiment analysis, or opinion mining, is the process of analysing text with the goal to determine the emotional tone behind it (Liu, 2020). Sentiment analysis algorithms typically involve the following steps to determine the sentiment expressed in a piece of text: Text Preprocessing, Feature Extraction, Sentiment Classification and Sentiment Scoring (Liu, 2020). Let’s discuss each of these steps further to better understand their functioning.

1. Text Preprocessing: Text preprocessing standardises raw text into a format that is more interpretable by sentiment classification models. This is crucial in natural language processing and hence, the forthcoming sentiment analysis (Liu, 2020). The following are the steps:

- *Tokenization:* Tokenization is the process of breaking down a piece of text into individual components, usually words, known as tokens. For example, the sentence "Computer Science

is awesome" would be split into ["Computer", "Science", "is", "awesome"]. Sentiment analysis models usually treat each word as a feature, so breaking down the text is essential for further analysis. Modern tokenization routines, such as those used in BERT and its variants, go a step further by incorporating positional encoding, which takes into account the order of tokens in the text. For example, the sentences "Mary loves John" and "John loves Mary" contain the same tokens but differ in meaning due to the position of the tokens. This positional awareness is crucial for models to capture the context better.

- *Stop Word Removal:* Next, common words that serve grammatical purposes, but do not contribute any sentiment-relevant information (or Stop Words) are removed. In our example, “is” would be removed. By removing them, we reduce noise and make the remaining words more meaningful for the model.
- *Lemmatization (or Stemming):* Words (verbs) are then reduced to their base or root form (known as the lemma) so that variations of the same word are treated as the same. Without reducing them to their base form, the model might treat the variations as completely different words, leading to a fragmented understanding of sentiment. There are two main methods- Stemming or simplifying words by cutting off suffixes, and Lemmatization or mapping words to their proper base form. It is important to note that while these methods help eliminate redundant text elements, they are excluded in modern sentiment analysis models (e.g., BERT and its variants) intentionally as these models directly process raw text, using their advanced architecture to preserve the context, which might otherwise be lost.

2. Feature Extraction: With feature extraction, the goal is to represent the text in a manner that captures crucial information, such as word frequencies, relationships between words, and the context of use. Depending on the model, feature extraction methods range from traditional frequency-based approaches to advanced, context-based approaches (Liu, 2020). The key characteristics of feature extraction are as follows:

- *Lexicon Matching:* Words or phrases are matched against predefined sentiment scores in a dictionary or lexicon.
- *Contextual Embeddings:* Sentences or text are converted into dense vectors representing the meaning of words in their specific context.
- *Attention Mechanism:* The relationships between words are weighed through self-attention, helping the model focus on the most important words for understanding sentiment.
- *Handling of Negations, Modifiers, and Intensity*: Adjustments are made based on words like "not" or intensity modifiers like "very" or "extremely."

3. Sentiment Classification: Once the features are extracted from the text, the next step is to classify the sentiment. Here, sentiment labels (positive, negative, neutral or other more specific sentiments) are assigned to each input. There are several approaches we can take with sentiment classification, usually under the branches of Lexicon-based approaches and Machine Learning approaches:

- *Lexicon-based approaches:* This approach relies on a predefined list of words (a sentiment lexicon) associated with positive or negative sentiment and matches words in the text to this lexicon to determine sentiment (Liu, 2020).
- *Machine Learning approaches:* Machine learning approaches often branch into Supervised Learning and Unsupervised Learning (Liu, 2020).
    - *Supervised Learning:* Algorithms like Support Vector Machines (SVM), Naive Bayes, or deep learning models are trained on labelled datasets where the sentiment uses supervised learning.

- *Unsupervised Learning:* This approach is used when the data is not labelled data. A common method is clustering which is used to uncover trends in the text that may correspond to a variety of sentiments.

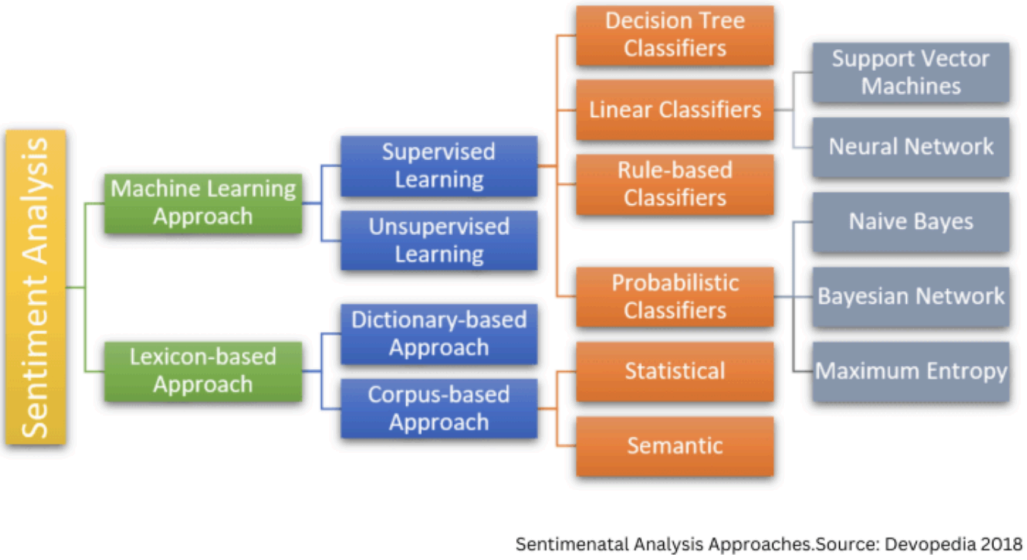

Fig. 1 - Sentiment Analysis Approaches (Gitome, 2023).

4. Sentiment Scoring: The sentiment is classified as positive, negative, or neutral (advanced models can also detect the intensity of the sentiment). Several sentiment analysis algorithms already exist in python that have been known to perform very well (Liu, 2020). These models include:

- *VADER (Valence Aware Dictionary and sEntiment Reasoner):* VADER is a lexicon and rule-based sentiment analysis tool specifically attuned to sentiments expressed in social media. It uses a sentiment lexicon that is a list of lexical features (e.g., words) labelled according to their semantic orientation as positive or negative. What makes VADER unique is that it considers contextual polarity and intensity. For instance, the sentiment score is adjusted if the word "not" precedes a positive word (e.g., "not good"), or if intensity modifiers are present (e.g., "extremely good" vs. "good"). VADER outputs sentiment as a polarity score that categorises text into positive, negative, and neutral, along with a compound score that represents the overall sentiment (Hutto and Gilbert, 2014).
- *RoBERTa Pretrained Model:* RoBERTa (Robustly Optimised BERT Pre Training Approach) is a variant of the BERT (Bidirectional Encoder Representations from Transformers) model. It is a transformer-based architecture trained on a large corpus of text data, designed to understand context in a bidirectional manner. In sentiment analysis, a RoBERTa pretrained model can be fine-tuned on sentiment-labelled data. RoBERTa is capable of handling subtle exceptions, such as sarcasm or irony, that are often challenging for traditional models. Its performance is generally superior to traditional machine learning methods, particularly in tasks requiring deep contextual understanding. The model outputs a probability distribution over possible sentiment labels (positive, negative, neutral) for a given input text (Liu et al., 2019).

- *Transformer's Pipeline:* Transformer's pipeline refers to a streamlined process of utilising transformer-based models like BERT, RoBERTa, or GPT for various NLP tasks, including sentiment analysis. Transformers work by leveraging self-attention mechanisms to weigh the importance of different words in a sentence relative to each other, which is crucial for understanding context and sentiment. This approach is highly flexible and can be used with various transformer models depending on the specific requirements and available resources (Vaswani et al., 2017).
- *TextBlob:* TextBlob is a Python library for processing textual data that includes a simple API for diving into common natural language processing (NLP) tasks, including sentiment analysis. While primarily lexicon-based, TextBlob can incorporate basic machine learning techniques for more complex tasks. However, it is generally considered less sophisticated than transformer-based models and is suitable for basic sentiment analysis needs. TextBlob is highly easy to use and integrates well with applications where efficiency and simplicity are priorities (TextBlob, 2018).

To proceed with analysing the change in this sentiment over the years, we conduct a Time Series Analysis. Time series analysis is the process of analysing chronologically ordered data points to extract relevant statistics and identify characteristics of the data as a trend of change. It often involves the use of models to understand underlying patterns, trends, seasonality, and to make forecasts about future values based on historical data (Chatfield and Xing, 2019). There are a few key terms that are comprised in time series analysis, important to help us better in analysing and commenting on our analysis:

- *Trend:* The long-term movement in a time series. It represents the underlying direction or pattern in the data, which could be upward, downward, or flat, often due to factors like economic growth, technological advancements, or population changes.
- *Seasonality:* Regular, repeating patterns or fluctuations in a time series at specific intervals, such as daily, monthly, or yearly. For example, retail sales may increase during holiday seasons.
- *Cyclic Patterns:* Similar to seasonality but not fixed in period. These cycles are longer-term fluctuations often related to economic or business cycles, which do not follow a strictly regular schedule.
- *Noise or Irregular Component:* The random variability in the data that cannot be explained by the trend, seasonality, or cyclical components. This component is often modelled as white noise or residuals in time series analysis.

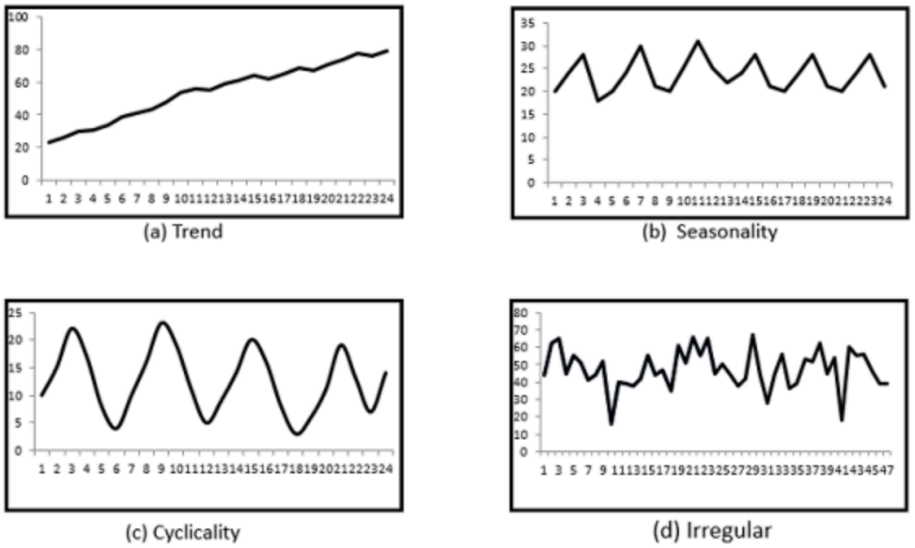

Fig. 2 - Time Series Components (Shukla, 2023)

Recognising these trends ultimately helps us in forecasting. Forecasting in time series analysis involves predicting future values based on previously observed data points. In the context of this project, this would translate into an attempt to predict how the media's opinion regarding A.I. would shift in the future. We can use several models to do so, including Random Forests, XGBoost and GBM. Let's define these models and their features.

- *Random Forests:* Random Forests are ensemble learning methods that combine multiple decision trees to improve accuracy and control overfitting. The model further provides insights into the importance of different features. Random Forests can also capture non-linear relationships between features, which is beneficial when dealing with the complex nature of textual data and sentiment analysis (Hastie, Friedman and Tibshirani, 2001).
- *XGBoost*: Extreme Gradient Boosting (or XGBoost) has mechanisms to handle imbalanced datasets, which can be crucial if certain sentiments (positive or negative) are less frequent in the media. It performs well with large datasets and complex features, which helps in effectively capturing patterns and trends over time. XGBoost is known for its high performance and efficiency in terms of speed and accuracy which is why it often yields better results in prediction tasks (Hastie, Friedman and Tibshirani, 2001).
- *GBM:* Gradient Boosting Machines (or GBM) is a type of boosting algorithm that builds models sequentially, where each new model corrects the errors of the previous one. This is an iterative process which ultimately helps improve model accuracy and performance. It further allows customization of the loss function and the model's complexity, making it adaptable to different types of problems and datasets. GBM shares similarities with Random Forests and XGBoost in that it is also an ensemble method and can handle complex datasets. However, its sequential learning approach and flexibility in defining the loss function offer it a presumed edge over the other two (Hastie, Friedman and Tibshirani, 2001).

In order to evaluate these models against each other, the Root Mean Square Error (or RMSE) score can be used. The score is calculated by taking the average difference between observed and predicted values, statistically interpreted as the standard deviation of the residuals.

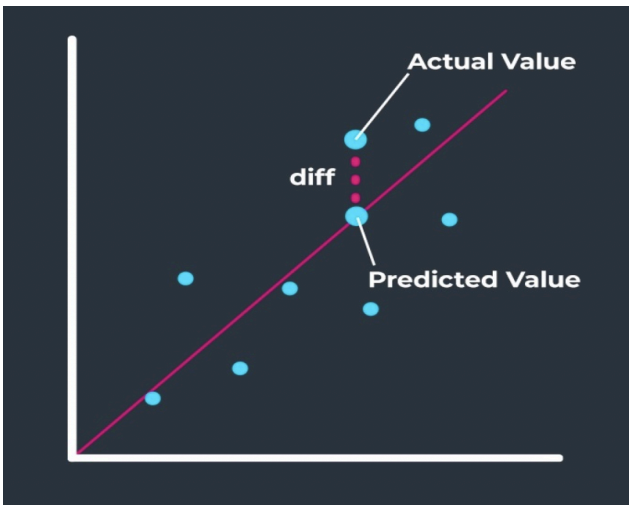

Fig. 3 - Difference between observed and predicted values (Olumide, 2023).

Mathematically, the formula is represented as follows:

$$RMSE = \sqrt{\frac{\Sigma(y-\hat{y})^2}{N-P}}$$

Where:
y is the actual value of the ith observation.
ŷ is the predicted value for the ith observation.
P is the number of the parameter estimated, including the constant.
N is the number of observations.

RMSE is a valuable metric for judging model performance, as it provides an estimate of the average magnitude of errors between predicted and actual values. RMSE is a negative-oriented metric, meaning that lower values indicate better model performance. The lowest possible value for RMSE is 0, which corresponds to a perfect fit between the model's predictions and the actual values. As RMSE increases, it indicates a larger average error and a poorer fit (Olumide S., 2023).

III. Methodology

I. Data Collection

The first step towards this project is data collection using the New York Times API. Due to the lack of a pre-built dataset of already-scraped articles, we leveraged the API to query the New York Times archive and gather news articles. A connection to the NYT API is made using a registered API key and base URL to proceed with extracting the articles. These articles were acquired by performing a Query search of A.I. terminologies and subfields (Fig. 4) throughout the restricted time frame of the past decade (2013-2024). The necessary features extracted were- Query, Headline, Publication Date (as a string), Snippet (or sub-headline) and the URL (Fig. 5).

```
def main():
    queries = [
        "A.I.",
        "Artificial Intelligence",
        "Augmented reality",
        "Automation",
        "Chatbot",
        "Data Science",
        "Deepfake",
        "GPT",
        "M.L.",
        "Machine Learning",
        "Natural Language Processing",
        "NLP",
        "Virtual Reality"
    ]
```

Fig. 4 - A.I. Terminology (Appendix 1.1)

Further, some preliminary data cleaning and organising was performed where the 'Publication Date' was converted (from string) to a Date-and-Time format and the entries were rearranged in a chronological order to make it well-organised, especially considering that our ultimate goal is a time series analysis (Fig. 5).

	A	B	C	
1	**Query**	**Headline**	**Publication Date**	**Snippet**
2	**Artificial Intelligence**	Today's Scuttlebot: Tech Design Comics and Artificial Intelligence Money	2013-01-04 09:15:58+00:00	The technology reporte
3	**Artificial Intelligence**	A Motherboard Walks Into a Bar ...	2013-01-04 21:02:48+00:00	Researchers are teachi
4	**Virtual Reality**	A Bull in Stocks, but a Bear for Free Speech	2013-01-08 16:43:12+00:00	Expectations among in
5	**Virtual Reality**	Taking in Paris Any Day, Any Century	2013-01-09 22:25:45+00:00	"Paris 3D" lets viewers
6	**Virtual Reality**	Some future gadgets I'd maybe buy (aka a realist's guide to CES)	2013-01-10 00:00:00+00:00	Eight Wirecutter writers
7	**Virtual Reality**	Unpacking the Pandora's Box of Technology	2013-01-11 02:38:31+00:00	"Welcome to the Mach
8	**Virtual Reality**	Reflecting on War and Its Tentacles	2013-01-14 22:40:05+00:00	David T. Little's "Soldie
9	**Automation**	Critical Infrastructure Systems Seen as Vulnerable to Attack	2013-01-18 02:49:14+00:00	A computer security co
10	**Virtual Reality**	The Missing President	2013-01-23 01:29:21+00:00	Venezuelans are stuck
11	**Automation**	Robot Makers Spread Global Gospel of Automation	2013-01-24 00:07:33+00:00	Manufacturers of robot
12	**A.I.**	Ray Kurzweil Says We're Going to Live Forever	2013-01-25 16:31:24+00:00	He just isn't sure how t

Fig. 5 - Dataset compiled (Appendix 1.2)

II. Sentiment Method Evaluation

Now that we have compiled the (preliminary) dataset, we move to adding a 'Sentiment' column to this dataset. There are multiple methods to do this and since this step is highly crucial in this project, we must ensure that we use the most accurate method to do so. Therefore, we must evaluate these methods initially using a labelled dataset to measure the performance of each method better.

The dataset used for this was uploaded on Kaggle (Appendix 2.1). The file contains a variety of insights on social media's user generated comments. It further includes sentiment labels, timestamps, platform information, trending hashtags, user interaction data, and geographical origins. Our focus primarily remains on the 'text' and 'sentiment' column, and hence, we keep our dataframe limited to them (Fig. 6). Next, it was noted that the dataset used a diverse set of emotions which added much complexity as most models cannot classify sophisticated emotions (gratitude, awe, etc.) (Fig. 7) and therefore, the sentiments were manually classified into the umbrella sentiment of positive, negative and neutral.

```
[93]  df.head()
```

	Text	Sentiment
0	Enjoying a beautiful day at the park! ...	Positive
1	Traffic was terrible this morning. ...	Negative
2	Just finished an amazing workout! 💪 ...	Positive
3	Excited about the upcoming weekend getaway! ...	Positive
4	Trying out a new recipe for dinner tonight. ...	Neutral

Fig. 6 - Dataframe (Appendix 2.2)

```
[94]  df['Sentiment'].unique()

array([' Positive  ', ' Negative  ', ' Neutral   ', ' Anger       ',
       ' Fear        ', ' Sadness     ', ' Disgust     ',
       ' Happiness   ', ' Joy         ', ' Love        ',
       ' Amusement   ', ' Enjoyment   ', ' Admiration  ',
       ' Affection   ', ' Awe         ', ' Disappointed ',
       ' Surprise    ', ' Acceptance  ', ' Adoration   ',
       ' Anticipation ', ' Bitter     ', ' Calmness    ',
       ' Confusion   ', ' Excitement  ', ' Kind        ',
       ' Pride       ', ' Shame       ', ' Confusion ', ' Excitement ',
       ' Shame ', ' Elation      ', ' Euphoria     ', ' Contentment  ',
       ' Serenity     ', ' Gratitude    ', ' Hope         ',
       ' Empowerment  ', ' Compassion   ', ' Tenderness   ',
       ' Arousal      ', ' Enthusiasm   ', ' Fulfillment  ',
       ' Reverence    ', ' Compassion', ' Fulfillment  ', ' Reverence ',
       ' Elation  ', ' Despair       ', ' Grief         ',
       ' Loneliness     ', ' Jealousy       ', ' Resentment     ',
       ' Frustration    ', ' Boredom        ', ' Anxiety        ',
       ' Intimidation   ', ' Helplessness   ', ' Envy           ',
       ' Regret        ', ' Disgust        ', ' Despair      ',
       ' Loneliness ', ' Frustration ', ' Anxiety   ', ' Intimidation ',
       ' Helplessness ', ' Jealousy    ', ' Curiosity      ',
       ' Indifference   ', ' Confusion      ', ' Numbness       ',
       ' Melancholy     ', ' Nostalgia      ', ' Ambivalence    ',
       ' Acceptance     ', ' Determination  ', ' Serenity       ',
       ' Numbness ', ' Zest ', ' Contentment ', ' Hopeful ', ' Proud ',
...
```

Fig. 7 - Varied Emotions in the dataset (Appendix 2.2).

Now that our dataset is simplistic enough, we are ready to use our models. In particular, we use VADER, RoBERTa's Pretrained Model (by Hugging Face), Transformer's Pipeline and Textblob. Each model is applied to the 'text' column, and its result is stored as a dedicated column. Eventually. Our dataframe is updated to Fig. 8 with the predicted sentiments in addition to Fig. 6.

	Text	Sentiment	Sentiment_VADER	Sentiment_Roberta	Sentiment_Pipeline	Sentiment_Blob
0	Enjoying a beautiful day at the park! ...	Positive	Positive	Positive	Positive	Positive
1	Traffic was terrible this morning. ...	Negative	Negative	Negative	Negative	Negative
2	Just finished an amazing workout! 💪 ...	Positive	Positive	Positive	Positive	Positive
3	Excited about the upcoming weekend getaway! ...	Positive	Positive	Positive	Positive	Positive
4	Trying out a new recipe for dinner tonight. ...	Neutral	Neutral	Neutral	Negative	Positive
...	...	...	...	...	...	...
727	Collaborating on a science project that receiv...	Positive	Positive	Positive	Positive	Positive
728	Attending a surprise birthday party organized ...	Positive	Positive	Positive	Positive	Positive
729	Successfully fundraising for a school charity ...	Positive	Positive	Positive	Positive	Positive
730	Participating in a multicultural festival, cel...	Positive	Positive	Positive	Positive	Positive
731	Organizing a virtual talent show during challe...	Positive	Positive	Positive	Positive	Positive

732 rows × 6 columns

Fig. 8 - Updated dataframe with the predicted sentiments from each method (Appendix 2.2).

We proceed to measure the accuracy of each method by comparing its results to the original 'sentiment' column and measuring the correct matches as a percentage of the whole (Fig. 9).

	Method	True (%)	False (%)
0	VADER	77.459016	22.540984
1	RoBERTa	80.601093	19.398907
2	Pipeline	79.644809	20.355191
3	TextBlob	48.224044	51.775956

Fig. 9 - Table showing accuracy results of all methods (Appendix 2.2).

After looking at the results, it is evident that Hugging Face's RoBERTa Pretrained Model performed the best with an approximate 80% accuracy rate compared to others. Therefore, we proceed with selecting the same as our model of choice for our news articles' dataset.

III. An Attempt to Filter Misclassified Entries

Before the application of the model, it is crucial to clarify why there exist entries that were falsely classified as 'A.I.' related. For example, the title- "Gang R*pe Defendants at Risk in Tihar Jail, Lawyers and Family Say" mentions "M.L. Sharma" as an acronym for an Indian lawyer relevant in the case discussed in the article, not in the context of machine learning. An attempt was made to clean this data through a filtering loop with conditional statements that checked whether the 'Headline', 'Snippet' or 'URL' (for keywords) contained the queries (Fig. 4) or not (Fig. 10).

```
# Iterate over each row in the input file
for row in reader:

    for query in queries:
        # Check if the query is in the Headline or Snippet
        if query.lower() in row['Headline'].lower() or query.lower() in row['Snippet'].lower() or query.lower() in row['URL']:
            dataa_writer.writerow(row)
            break
        elif query=="End":
            datan_writer.writerow(row)
            break
        else:
            pass
```

Fig. 10 - Iteration attempt to 'clean' the data (Appendix 3.1).

However, after examining the dataset created from this method, it was noticed that there were significantly more false positives i.e. articles that are relevant with conversations of A.I. but were classified as unimportant. Some of the examples of such articles being:

- "Confronting the Fact of Fiction and the Fiction of Fact"
- "At Google Conference, Cameras Even in the Bathroom"
- "Yes, Economics Is a Science"

After all, this method does make the bold assumption that all headlines (or snippets or URLs) explicitly mention the main topic of discussion as most of the time, the title's purpose is to act as a hook that catches the readers' attention and peaks their curiosity. We thereby choose to ignore context for the sake of not losing any relevant data.

IV. Constructing 'Sentiment' column - RoBERTa's Model

We proceed with applying the RoBERTa model to our dataset, specifically targeting the 'Headline' and 'Snippet' columns to generate two new sentiment-related columns: 'Sentiment_H' and 'Sentiment_S'. The model outputs sentiment scores across three categories: positive, negative, and neutral. To consolidate these scores, we compute the difference between the positive and negative sentiment scores ('roberta_pos' - 'roberta_neg'), in order to return the sentiment as a single variable for both the headline and snippet (Fig. 11). Then, we calculate the weighted average of 'Sentiment_H' (20%) and 'Sentiment_S' (80%) to create a composite 'Sentiment' column. The weighted average ensures that the snippet, containing more descriptive text relative to the articles, is given more importance than the heading which merely serves as a catchy hook. This column serves as the dependent variable for the subsequent time series analysis (Fig. 12).

```python
def classify_sentiment(text):
    encoded_text = tokenizer(text, return_tensors='pt')
    results = model(**encoded_text)
    scores = results[0][0].detach().numpy()
    scores = softmax(scores)
    scores_dict = {
        'roberta_neg' : scores[0],
        'roberta_neu' : scores[1],
        'roberta_pos' : scores[2]
    }

    return scores_dict['roberta_pos']-scores_dict['roberta_neg']


df['Sentiment_H'] = df['Headline'].apply(lambda x: classify_sentiment(x) if isinstance(x, str) and x.strip() else None)
df['Sentiment_S'] = df['Snippet'].apply(lambda x: classify_sentiment(x) if isinstance(x, str) and x.strip() else None)

df.head()
```

Fig. 11 - Applying RoBERTa's model to our dataset (Appendix 3.1).

	Sentiment_H	Sentiment_S	Sentiment
	0.012737811	0.310969740152359	0.2513233542442320
	0.010713909	0.7168263792991640	0.5756038852967320
	-0.22306831	0.1972092092037200	0.11315370425581900
	0.13412958	0.48133912682533300	0.4118972193449740
	0.60551864	0.2280540019273760	0.3035469278693200
	-0.06305005	0.03046586364507680	0.011762681044638200
	0.020014115	-0.03347357362508770	-0.022776035871356700
	-0.38942042	-0.48997631669044500	-0.46986513882875400
	-0.40798435	-0.5231510996818540	-0.5001177534461020

Fig. 12 - Sentiment containing columns (Appendix 3.2).

The final dataset therefore consists of the following columns:

- *Query* (String) - A categorical variable consisting of 'A.I.' and other related subfields, signifying the query that the news article contains.
- *Headline* (String) - The title of the news article.
- Publication Date (Date&Time) - The date on which the news article was published to the NYT website.
- *Snippet* (String) - The sub-heading with a further description, more elaborate than the title, about the news articles and the topic being discussed.
- *URL* (String) - The URL of the NYT website of the news article.
- *Sentiment_H* (Float) - A numeric variable ranging from -1 to +1, representing the result of the sentiment analysis on the '*Headline*'.
- *Sentiment_S* (Float) - A numeric variable ranging from -1 to +1, representing the result of the sentiment analysis on the '*Snippet*'.
- *Sentiment* (Float) - An average of the two sentiment score columns '*Sentiment_H*' and '*Sentiment_S*', similarly ranging from -1 to +1.

V. Time Series Analysis

Now, the dataset is ready for time series analysis. We proceed with plotting the data on a chronological time scale (Fig.13). We further aggregate the news articles' sentiment (the dependent variable or y-axis) by month (Fig. 14) and then, by year (Fig. 15). For every month, the average

sentiment is computed and stored in a dataframe and the result is then plotted as a time series, where the x-axis represents the monthly publication dates, and the y-axis represents the aggregated (mean) sentiment values for those months. These trends are way more interpretable and thus, we choose to perform the predictive forecasting on them.

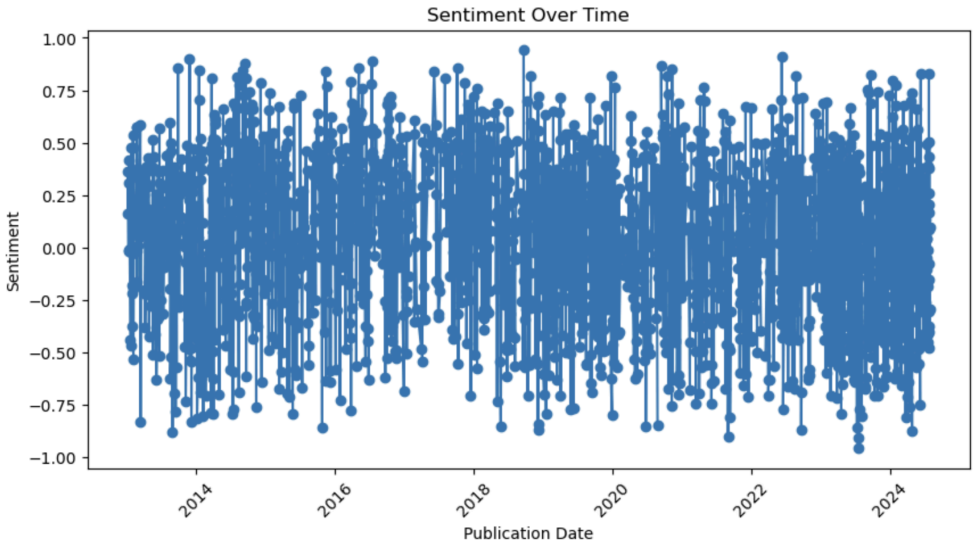

Fig. 13 - Sentiment over Days (Appendix 4).

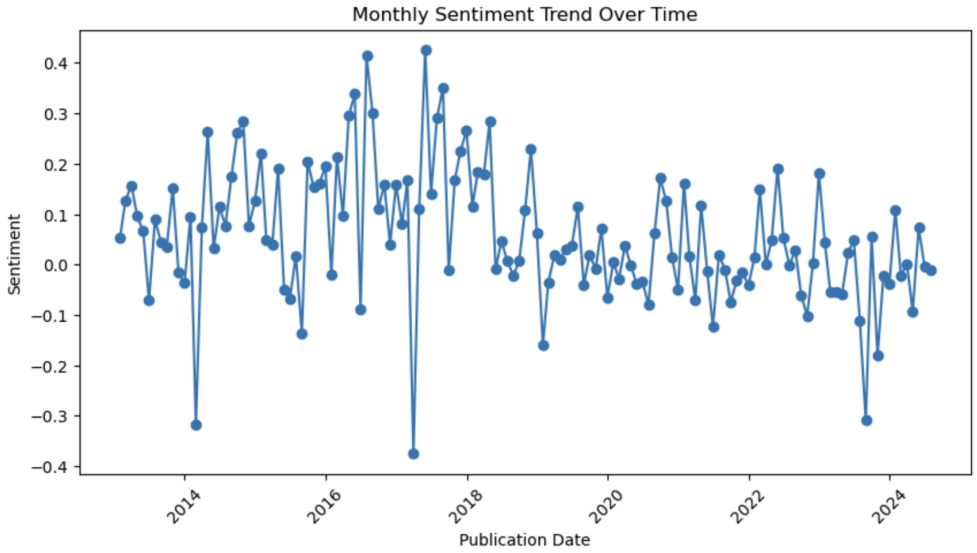

Fig. 14 - Sentiment over Months (Appendix 4).

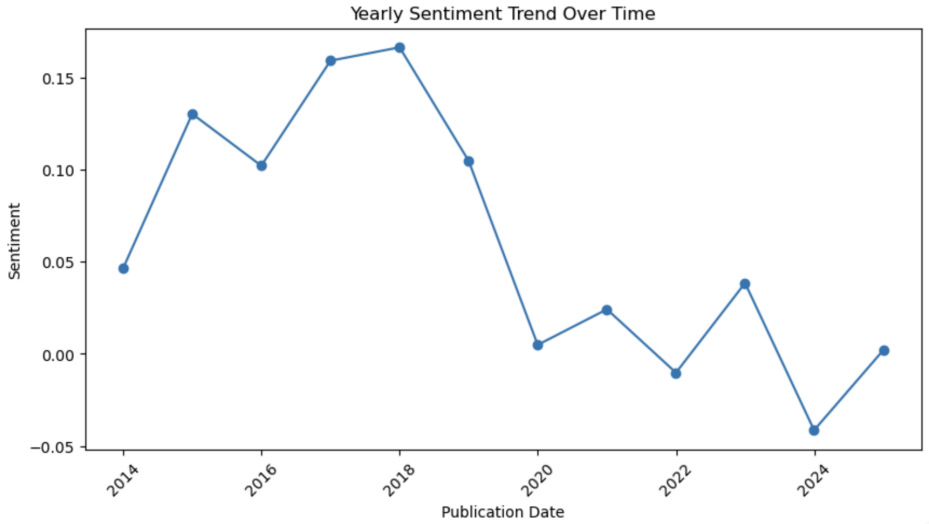

Fig. 15 - Sentiment over Years (Appendix 4).

VI. Forecasting

In order to proceed with the forecasting, three models (Random Forests, XGBoost and GBM) were evaluated by splitting the data into a training set and test set. The split was done using an 80-20 ratio, where 80% of the data was used for training and the remaining 20% for testing, and was performed based on the publication date, ensuring that the earlier part of the time series was used for training, and the later part was used for testing to maintain temporal consistency. After applying these predictive models on the training set, the RMSE was calculated for the test set respectively. This was done for the daily data, monthly data and yearly data (plotted on Fig. 13, 14 & 15 respectively) and the results showed that the RMSE for XGBoost was lowest i.e. approximately 0.04 (Fig.16).

	Data Frequency	Random Forests RMSE	XGBoost RMSE	GBM RMSE
0	Daily	0.413566	0.213566	0.414271
1	Yearly	0.054264	0.036857	0.053422
2	Monthly	0.158326	0.172288	0.175501

Fig. 16 - RMSE score of all models (Appendix 4).

Therefore, we use XGBoost in order to forecast the trend on a daily basis for the next 365 days, monthly basis for 12 months and yearly basis for the next 5 years as follows.

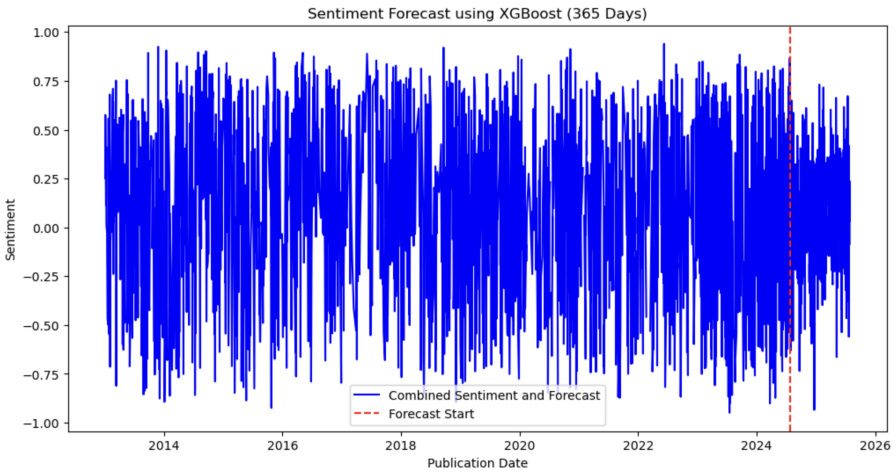

Fig. 17 - Forecasting (Daily) (Appendix 4).

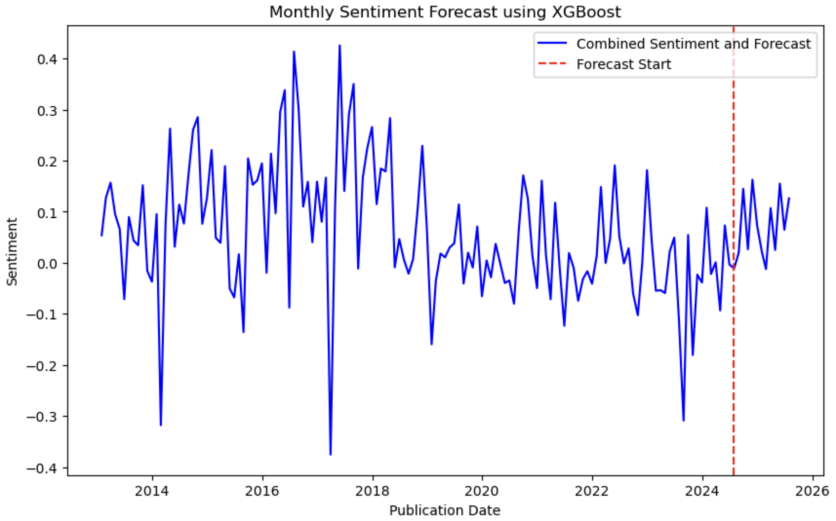

Fig. 18 - Forecasting (Monthly) (Appendix 4).

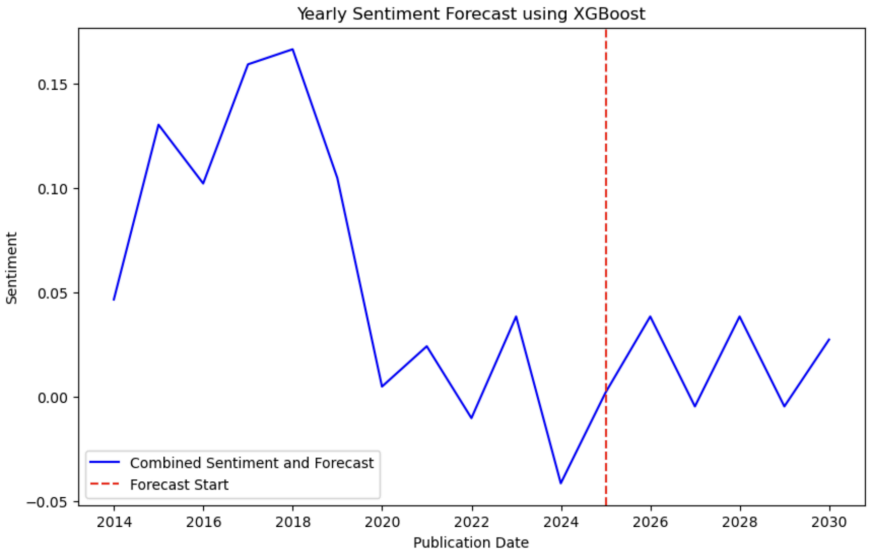

Fig. 19 - Forecasting (Yearly) (Appendix 4).

IV. Results and Finding

I. Query-based Exploratory Data Analysis

We proceed to comment on the insights we gained from our EDA and time series analysis to better understand the representation and nature of all queries in the dataset. Fig. 19 represents the A.I. queries on the y-axis and the frequency of occurrence in the compiled dataset on the x-axis. Each bar corresponds to a specific A.I. topic or query, indicating how frequently it has appeared in the dataset. The graph vividly presents a view of the dominant A.I. topics in media coverage over the past decade and help identify which aspects of A.I. are gaining the most attention, both in terms of innovation and ethical debate.

It is interesting to observe how the frequencies in the graph reflect the age and familiarity of different A.I. topics. For example, 'Automation', with a frequency of around 400, has been discussed in many different contexts over the years. This makes sense, as automation was one of the earliest concepts associated with the idea of machines or systems performing tasks for humans. On the other hand, terms like 'Chatbot', 'Deepfake', and 'Artificial Intelligence' are mentioned less often. These are more recent developments in the field, and though they are important, they haven't been around as long as automation. Finally, queries such as 'Natural Language Processing' and 'GPT' have lower frequencies and have recently received mainstream attention as topics of discussion. Therefore, the graph has two interpretations- the topics that have been 'talked about the most' in the media and also, the topics that 'we know the most about' due to the fact that our data is collected over the past decade.

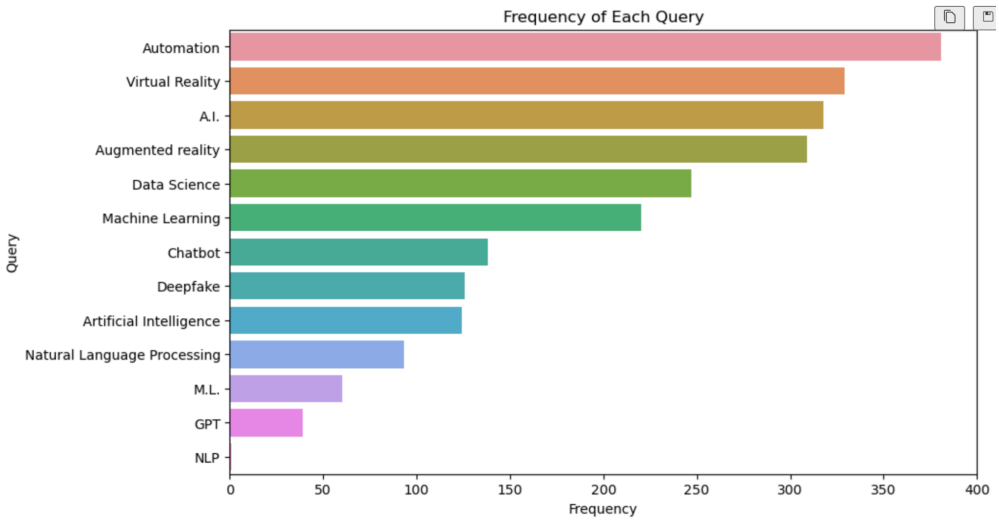

Fig. 20 - Frequency of each Query (Appendix 4).

This argument is further highlighted in Fig. 20 where we notice that when segmented on the basis of years, the respective queries have stark differences in frequency when compared to Fig. 19 where there was no yearly segmentation. In 2013, 'Virtual Reality' reached a significantly high peak due to the introduction of Oculus Rift VR Headset and a developer kit (DK1), allowing developers to create VR applications and games (James, 2016). On the other hand, we see a peak in 'Natural Language Processing' in 2023, starkly contrasting to its overall general frequency in Fig. 21, due to the launch of OpenAI's GPT-4 and its ability to better understand natural language (Lynch, 2023).

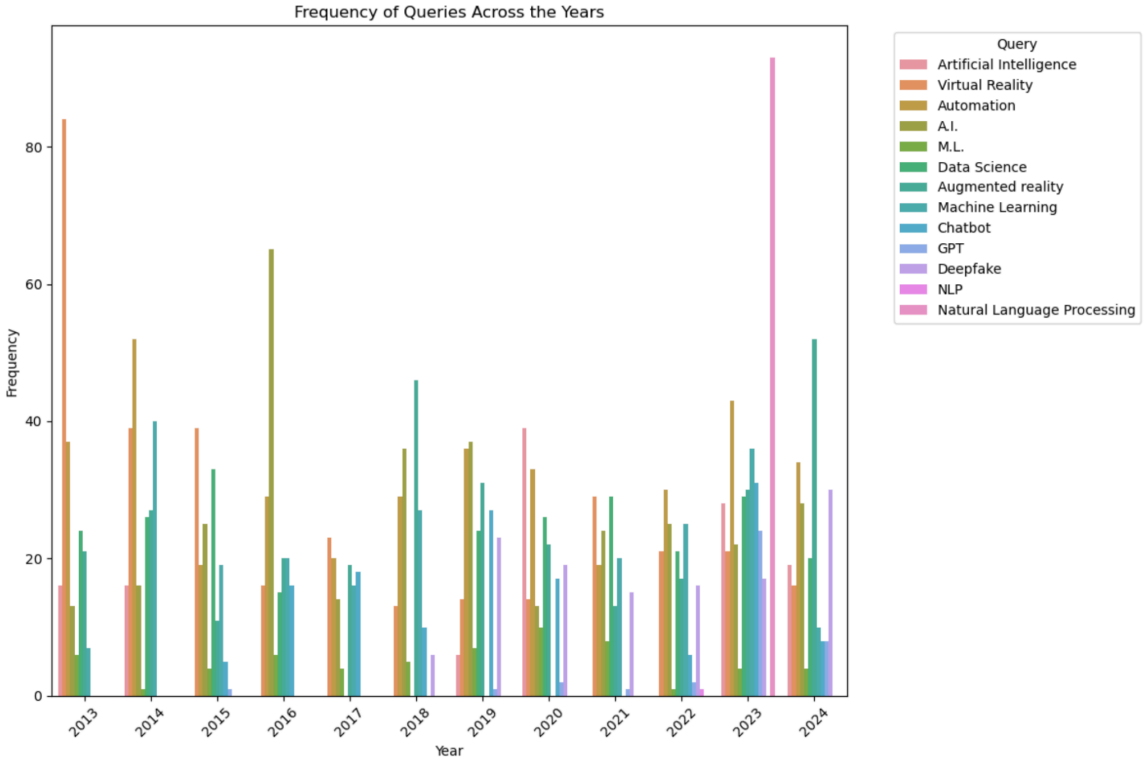

Fig. 21 - Frequency of each Query across Years (Appendix 4).

Additionally, the word cloud helps us to better understand what are the most common words that occur in these titles, irrespective of their queries (Fig. 22). We notice that in addition to the queries used, 'Google' dominates the headlines, followed by 'Microsoft' and 'Apple', which indicates how integral these companies have been in driving and popularising the development of technologies like virtual reality, A.I., and smart home devices across the years.

Fig. 22 - Wordcloud (Appendix 4).

II. Commenting on the Time Series

As evident from Fig. 13, the interpretation of the time series data on a daily basis is extremely difficult. The interpretation is far easier for the yearly data where the peaks and troughs are clearer

(Fig. 23). It is important to consider the scale of the sentiment since it has been aggregated on an yearly basis and thereby, the values are less extreme. However, the graph does provide a general understanding of the trend throughout the past decade.

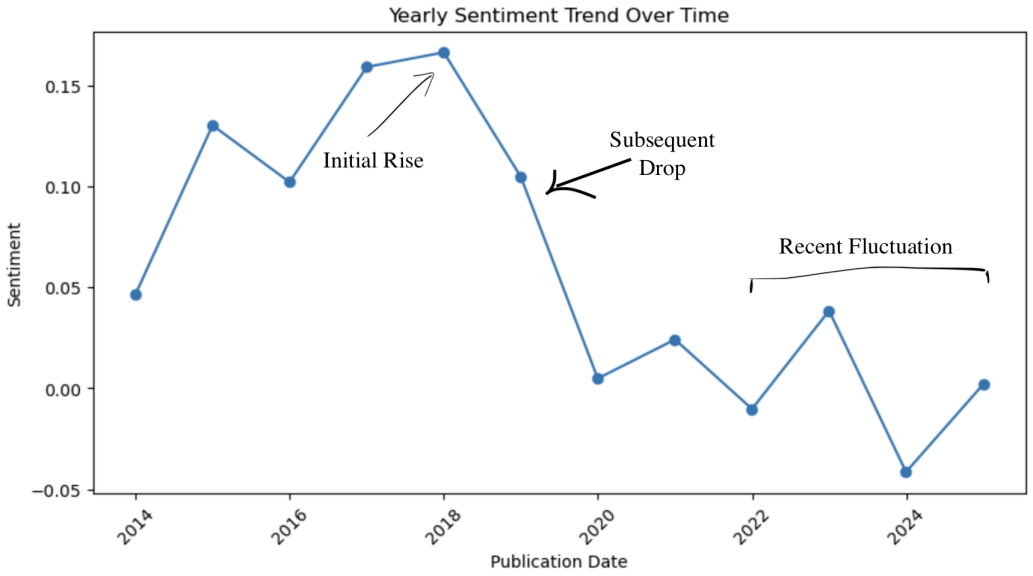

Fig. 23 - Labelled Yearly Trend (Appendix 4).

The sentiment shows a steady increase from 2014, reaching a peak around 2018. This period may signify the initial spark of A.I. with optimistic coverage surrounding the potential new technological advancements, such as the rise of machine learning applications and early breakthroughs in A.I. There is a significant drop from 2018 to 2020 which could be linked to rising public scepticism or concerns regarding A.I., possibly due to ethical issues of job displacement, or even privacy concerns related to the use of A.I. in surveillance. After the dip in 2020, we notice a brief rise in sentiment around 2021-2023, and another drop in 2023. These dilemmas in sentiment shows throughout the recent fluctuation could be attributed to the diversified application of A.I. across industries with the rise of GPT-3 and contrastingly, rising concerns about the increasing dominance of A.I. for malpractices such as the generation of deepfakes as A.I. technologies are more accessible to the general public, including the naive practitioners of these unethical activities, than ever before.

Thus, we can infer that there existed an initial rise in sentiment around 2018, with a stark drop in 2020. Since then, the media sentiment has remained fairly neutral, with less aggressive peaks and troughs after 2020.

III. Commenting on the Seasonality

We further attempt to understand any possible seasonalities that exist in the sentiment. It is possible to do so by calculating the average sentiment across the months. However, the extreme rises in sentiment in the first half of the decade might introduce biases in the interpretation and thus, we interpret seasonality in the first half (2013-2018) of the decade separately from the second (2019-2024).

Month	
January	0.098844
February	0.114224
March	0.088242
April	0.198882
May	0.118081
June	0.009536
July	0.139684
August	0.110385
September	0.105831
October	0.178468
November	0.114289
December	0.098318

Fig. 24 - Average Sentiment (2013-2018) (Appendix 4)

Month	
January	0.033292
February	-0.016638
March	-0.020610
April	-0.018701
May	0.039033
June	0.003024
July	-0.031521
August	-0.041432
September	0.047032
October	-0.045064
November	0.011620
December	-0.006014

Fig. 25 - Average Sentiment (2019-2024) (Appendix 4)

In Fig. 24, we notice that April has the highest average sentiment at 0.198, followed by October which also shows a relatively high sentiment at 0.178. June has the lowest sentiment at 0.0012. Therefore, we notice that the overall trend suggests a more positive media sentiment in spring and autumn months (April, October), while the summer month of June was the least positive. January, March, and December also hover around similar values (0.08–0.1), indicating relatively moderate sentiment.

For the second half of the decade in Fig. 25, we notice that September has the highest average sentiment at 0.047. May also stands out with a slightly positive sentiment of 0.0312 and June at a slightly lower 0.003. October shows the most negative sentiment at -0.045, highly contrasting to the first half of the decade where it was the month with the second most positive media coverage. Further, the media sentiment during the mid-year period leans more negatively. The spring months of March and April are not particularly positive either. Winter months of January and December hover around neutrality with January at 0.033 and December at -0.006.

Overall, it is difficult to make a general inference on seasonality. The only inferences that can be made are the following:

- New year (or the month of January) shows a positive average sentiment throughout the decade, relative to the rest of the months.
- Spring and autumn months showed similar trends- positive in the first half of the decade and negative in the second half of the second half.
- Summer months mostly tend toward neutrality.

IV. Commenting on the Forecasting Results

The yearly and monthly forecasting using XGBoost effectively captures sentiment trends and fluctuations over time. The yearly forecast projects a continued pattern of moderate fluctuations over the next five years, reflecting a stabilisation in sentiment as extreme variations have progressively decreased. (Fig. 26). This cyclical behaviour aligns with the observed trend of diminishing extremes in aggregated sentiment. In contrast, the monthly forecast focuses on shorter-term fluctuations, revealing higher volatility. The fluctuations similarly remain less extreme (Fig. 27) which especially

seems relevant considering the extreme high and lows between 2016 and 2018 and how they become less aggressive after 2020.

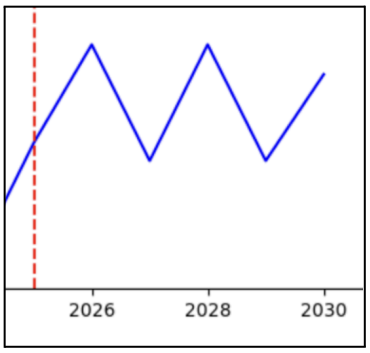

Fig. 26 - Yearly Sentiment Forecast (Appendix 4)

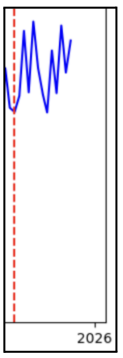

Fig. 27 - Monthly Sentiment Forecast (Appendix 4)

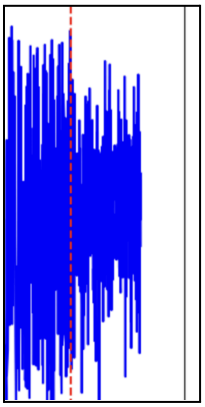

Fig. 28 - Labelled Daily Trend (Appendix 4).

In both cases, our models predict that A.I. related sentiment will no longer receive extreme values and aggressive fluctuation and would rather stabilise on average, mostly adjusting itself around neutral values in the next 5 years (Fig. 26) with more positively inclined sentiment for the next year (12 months) in particular (Fig. 27). While the interpretations become less instinctive in the case of daily predictions (Fig. 28), we can still see a clear reduction in extreme fluctuations on the sentiment, supporting our overall inference of eventual stabilisation in media sentiment.

V. Discussion

I. Learning Outcomes

The main learning outcomes acquired from this research were the observations made after analysing the evolving nature of media sentiment towards A.I. We learned that sentiment peaks during periods of optimism, such as the rise of deep learning in 2018, and declines when ethical concerns, like job displacement, privacy issues emerge later on. In particular, the COVID-19 pandemic of 2020 might further justify this decline as all news outlets shifted their focus to the outbreak.

Moreover, the predictive models demonstrated that sentiment is stabilising, suggesting a shift towards a more neutral and balanced view of A.I. This research further emphasised the importance of data-driven sentiment analysis in understanding broader trends in public discourse.

II. Limitation

The research has a few limitations that must be recognised to better interpret the outcomes of the analysis.

- Organisational Bias: As the only News API used was New York Times, any organisational biases are not accounted for in this research. However, it is interesting to notice that there does exist a variety in the sentiment which might suggest minimal bias.
- Limited Access: Since the API does not give us complete access to the articles, our sentiment analysis purely depends on the heading and the snippet which might give insufficient, or even misleading, information in order to peak the reader's curiosity and get them to click on the article.
- Forecasting: Our forecasting is purely done on the basis of machine learning models and does not account for unprecedented events that significantly increase, or decrease, the sentiment around A.I. This is very likely as more and more industries introduce A.I. technologies in their fields and more companies start exploring the boundless innovative potentials of A.I. tools.

III. Area of Further Research

An area for further research could involve a deeper exploration of the specific factors driving shifts in media sentiment towards A.I. technologies. While this study captured broad sentiment trends, future research could focus on isolating and analysing the impact of specific events, such as major A.I. product launches, regulatory changes, or ethical debates, on public perception.

Additionally, expanding the dataset to include other media sources beyond The New York Times, such as social media or paid international news outlets, could provide a more comprehensive view of global sentiment. Investigating sentiment across different industries and sectors where A.I. is applied, or examining sentiment differences across various demographic groups, could also yield more nuanced insights into how A.I. is perceived and understood by diverse populations.

Further, the methodology of this study can be applied to direct public opinion in the form of tweets, reviews of A.I. services or Reddit threads to assess how the sentiment of the public changes overtime. A comparative analysis can be conducted to highlight if there exist correlations between media sentiment and public sentiment.

VI. Conclusion

This research set out to explore the evolving media sentiment towards Artificial Intelligence (A.I.) and its subfields over the past decade, seeking to understand how public perception has shifted alongside the advancements and controversies surrounding A.I. technologies. The sentiment analysis, conducted using advanced models like RoBERTa, effectively captured these trends, and the time series forecasting suggests a future where media sentiment around A.I. stabilises. The findings portray a field of technology that has experienced aggressive fluctuation in sentiment—from the early optimism and excitement around breakthroughs like machine learning in 2018, to the scepticism and concern that emerged with issues such as job displacement and ethical dilemmas, especially in 2020.

XGBoost's ability to handle non-linear relationships and its robust performance with large datasets made it ideal for forecasting. The results highlighted that A.I.-related sentiment is expected to stabilise over the next five years, with moderate fluctuations around neutral values and a slight positive inclination, particularly over the next two years. This suggests a gradual tempering of extreme public opinion, as A.I. continues to integrate into various sectors.

Conclusively, we discuss a few limitations that have emerged such as organisational biases of the New York Times, limited access to news data and forecasting inability to account for innovations, or scandals, that might cause extreme sentiment changes. The study also serves as a basis for further research into the causes of these trends and provides a comprehensive methodology that can be applied to any other dataset of public or media opinion.

In the span of the last decade, we have witnessed the ‘Age of A.I.’—a period marked by extreme peaks of optimism and equally steep troughs of scepticism. Much like the crowd's cheers to Steve Jobs unveiling the first iPhone, people were captivated by the potential of A.I. in the early years. Yet today, as smartphones have seamlessly woven into our daily lives, so has A.I. and it is growing to become more and more of a steady presence in both our everyday lives, and in the media.

Bibliography

Chatfield, C. and Xing, H. (2019). *The Analysis of Time Series : an Introduction with R*. Milton: CRC Press LLC.

Edwords, L. (2022). *Deepfakes in the courts*. [online] Counsel Magazine. Available at: https://www.counselmagazine.co.uk/articles/deepfakes-in-the-courts.

Gitome, N. (2023). *Getting started with Sentiment Analysis and Implementation*. [online] DEV Community. Available at: https://dev.to/njerigitome/getting-started-with-sentiment-analysis-and-implementation-2g4d [Accessed 12 Sep. 2024].

Hastie, T., Friedman, J. and Tibshirani, R. (2001). *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*. New York, Ny Springer New York Imprint: Springer.

Hutto, C. and Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1), pp.216–225. doi:https://doi.org/10.1609/icwsm.v8i1.14550.

James, P. (2016). *3 Years Ago the Oculus Rift DK1 Shipped, Here's a Quick Look Back*. [online] Road to VR. Available at: https://www.roadtovr.com/3-years-ago-the-oculus-rift-dk1-shipped-heres-a-quick-look-back/ [Accessed 5 Oct. 2024].

Liu, B. (2020). *Sentiment analysis : mining opinions, sentiments, and emotions*. Cambridge ; New York: Cambridge University Press.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M.S., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv (Cornell University)*, 1. doi:https://doi.org/10.48550/arxiv.1907.11692.

Lynch, S. (2023). *13 Biggest AI Stories of 2023*. [online] hai.stanford.edu. Available at: https://hai.stanford.edu/news/13-biggest-ai-stories-2023.

Mitchell, R. (2015). *Web Scraping with Python*. 'O'Reilly Media, Inc.'

Olumide, S. (2023). *Root Mean Square Error (RMSE): What You Need To Know*. [online] Arize AI. Available at: https://arize.com/blog-course/root-mean-square-error-rmse-what-you-need-to-know/#:~:text=The%20RMSE%20values%20have%20the.

Peña-Fernandez, S., Meso Ayerdi, K., Larrondo Ureta, A. and Díaz-Noci, J. (2023). Without journalists, there is no journalism: the social dimension of generative artificial intelligence in the media. *El profesional de la información*, 32(2). doi:https://doi.org/10.3145/epi.2023.mar.27.

Shukla, S. (2023). *Various Techniques to Detect and Isolate Time Series Components Using Python*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2023/02/various-techniques-to-detect-and-isolate-time-series-components-using-python/.

Stern, S., Livan, G. and Smith, R.E. (2020). A network perspective on intermedia agenda-setting. *Applied Network Science*, 5(1). doi:https://doi.org/10.1007/s41109-020-00272-4.

TextBlob (2018). *TextBlob: Simplified Text Processing — TextBlob 0.15.2 documentation*. [online] Readthedocs.io. Available at: https://textblob.readthedocs.io/en/dev/.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł. and Polosukhin, I. (2017). *Attention Is All You Need*. [online] Available at: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Wallace, J., Berzon, M. and Berg, T. (2022). *LinkedIn Corporation v. hiQ Labs*. [online] Available at: https://cdn.ca9.uscourts.gov/datastore/opinions/2022/04/18/17-16783.pdf [Accessed 12 Sep. 2022].

Winn, Z. (2023). *Study finds ChatGPT boosts worker productivity for some writing tasks*. [online] MIT News | Massachusetts Institute of Technology. Available at: https://news.mit.edu/2023/study-finds-chatgpt-boosts-worker-productivity-writing-0714.

Appendix: Overview of Files

This appendix provides an overview of the key files included in the GitHub repository and the Kaggle dataset associated with this project.

A. Appendix 1.1
   - A Jupyter notebook detailing the steps of Data Collection using the New York Times API and Sorting to create the dataset of news articles.

B. Appendix 1.2
   - A CSV file containing the dataset of news articles constructed in the Jupyter notebook of Appendix 1.1.

C. Appendix 2.1
   - A CSV file containing the labelled dataset downloaded from Kaggle to proceed with the application and evaluation of Sentiment Analysis Methods
   - Kaggle Link

D. Appendix 2.2
   - A Jupyter notebook detailing sentiment analysis method evaluation of VADER, RoBERTa's Pretrained Model, Transformers Pipeline and Textblob.
   - The evaluation is done by measuring the correctness of model predictions relative to the labellend dataset in Appendix 2.1 and determination of the best model.
   - The algorithm adds three additional columns to the pre existing dataset- 'Sentiment_H', 'Sentiment_S' and 'Sentiment'.

E. Appendix 3.1
   - A Jupyter notebook detailing an attempt to clean the dataset and application of the best model (RoBERTa's Pretrained Model) to the news articles dataset in Appendix 1.2
   - The algorithm adds three additional columns to the pre existing dataset- 'Sentiment_H', 'Sentiment_S' and 'Sentiment'.

F. Appendix 3.2
   - A CSV file containing the results of the Jupyter notebook in Appendix 3.1.
   - The file consists of all contents from the initial news articles dataset in Appendix 1.2 + the three columns- 'Sentiment_H', 'Sentiment_S' and 'Sentiment'.

G. Appendix 4
   - A Jupyter notebook using the final dataset compiled in Appendix 3.2 to perform Exploratory Data Analysis (EDA) and conduct time series analysis on the dataset.
   - The dataset is split into a training set and a test set using 80-20 ratio in order to evaluate the predictive models of Random Forests, XGBoost and GBM.
   - The models are evaluated using RMSEs and the best model is determined (XGBoost).
   - The XGBoost Model is used to forecast trends in the sentiment on a daily basis for 365 days, monthly basis, making the prediction of the next year (or 12 months) and yearly basis, making the prediction of next 5 years.

Repository Link