Bimodal Data Mining: Integration of Key Data and Semantic Analysis for Text/Audio Datasets

Victor Corja
Volgenau School of Engineering
Applied Information Technology, MS
Fairfax, Virginia
vcorja2@gmu.edu

Austin Crow
Volgenau School of Engineering
Applied Information Technology, MS
Fairfax, Virginia
acrow3@gmu.edu

Nannan Liu
Volgenau School of Engineering
Applied Information Technology, MS
Fairfax, Virginia
nliu6@gmu.edu

Abstract

***There are 2.5 quintillion bytes of data created every day, an amount that would be impossible to process without the assistance of specialized data analytics tools and methodologies. Methodologies such as speech recognition, text data mining, and semantic analysis are used to help turn the 2.5 quintillion bytes of data into usable data to help one's needs. Individually, these tools and methodologies are capable of assisting with functions from automated answering machines to determining the emotional implications of a given text.**

This research was aimed at combining these individual methodologies into a consolidated model that takes in data in both audio and text format. For each desired function, models were selected and ranked by adherence to criteria which determined their applicability to the desired product. Upon selection of the models, they were implemented through libraries into a consolidated program, which took as an input a combinatorial text and audio dataset, and provided a report of the analysis resulting from data mining. The program was tested using data from TED talks, performed text mining and semantic analysis, and provided a structured output of the generated statistics.

Keywords— Data Mining, Speech Recognition, Text Mining, Semantic Analysis

I. Introduction

With the surge of social media, data creation is growing daily, and considering the amount of data available for use, there is a large gap between the potential usability and the current usage. It would not be feasible for any team of analysts to manually look at and analyze all of the data, and effectively determine its possible uses, which has led to the growth in interest in text and auditory data mining. Data mining has become more and more popular with the integration of technology into people's daily lives. Text-based data mining is a tool with a vast array of uses, from collecting specific filtered data representations, to identifying the preferences of particular groups of people based on metadata generated by their actions. A common use for text data mining is a spam filter, which email providers implement to determine whether an email is likely to be spam, as opposed to having value to the user. These filters take into account not only the text of the message, but also user preferences and behaviors with respect to similar messages from the sender [1].

Aside from text-based data mining, the same technique can be applied to audio data. Ever since the creation of Siri for iOS in 2011, speech recognition technology has become increasingly prevalent, especially as Amazon's Alexa and Google Home created a wave of mainstream attention into the technology. The uses for speech recognition are widely varied, and can be implemented in parallel with data mining, vastly improving the potential worth of data, as well as the generation of metadata, for audio resources [2].

In addition to data mining, semantic analysis can be applied to provide further insight into the data's meaning and potential. Semantic analysis enables a computer to derive the meaning of data and understand it with the context of its surrounding sentences, paragraphs, documents, and even entire datasets. Conversely, the same technique can be applied to sentence and word structures, as well as grammatical relationships, thereby providing an analysis both specific and broad in scope [3].

*The code related to this research can be found at https://github.com/VictorCorja/bimodal-data-mining

Individually all of these tools can be remarkably useful, but a combinatorial implementation can significantly enhance their usability. An implementation of a comprehensive model that takes in raw data and performs speech recognition, text data mining, and semantic analysis could drastically improve on the performance of these models on their own. Throughout this paper the aim is to help better understand what models were chosen as well as how this implementation will be used in the future.

II. Literature Review

There are a plethora of models that are used in order to pull information from any given data., but there are few that work together for a total data extraction to semantic analysis workflow [4]. The consequent sections outline models for speech recognition, textual keyword mining, and semantic analysis. Existing research was identified regarding various methods of key data extraction and semantic analysis, for both audio and text resources. The paper's focus is on determining models that could be integrated into the project, and for this purpose studies which utilized open-source models were selected, and had significant documentation. In order to separate this paper's analysis of the existing literature in the data analysis field of research, the review is split into three sections, each focusing on a particular aspect of the paper.

The first section is focused on publications which discuss models enabling the analysis of audio resources through the conversion of the audio data into text. Speech recognition is the process of converting a piece of the speech signal into text information, consisting mainly of four parts: feature extraction, acoustic model, language model, dictionary, and decoding. The final text after decoding will enable us to later perform keyword extraction and semantic analysis on the total combined data. The initial research looked into the possibility of directly analyzing audio data, as well as converting the audio resources into text. While direct analysis would potentially be more effective for audio resources, considering the complexity of existing approaches, as well as the significant difference in both the requirements and the outputs of those approaches, this paper's focus is only on integrating models dealing with speech recognition for audio-to-text conversion.

The second section identifies papers in which methodologies are proposed for the identification of keywords in text resources, such as topic mining and keyword extraction. While data analysis is usually performed on structured data, in order to retrieve all possible relevant information this paper's aim is to perform the analysis on the unstructured text data contained within the datasets. This will require data pre-processing in order to account for the enormous amount of information, followed by keyword analysis and topic mining.

The third section is focused on research into the semantic analysis of data, including the determination of key themes and moods within text and audio data. Data is composed of words, sentences, and paragraphs, so semantic analysis can also be divided into lexical-, sentence-, and paragraph-level semantic analysis to provide insight into the emotional and contextual composition of text resources. These analyses determine word sense disambiguation, the links between text entities such as location, time, and reason, and overall themes within the text, respectively.

Speech Recognition:

- There are many methods of performing speech recognition on audio data, a few of which are listed and ranked by Toshniwal [5], who suggests that combining the traditionally separate automatic speech recognition (ASR), learned acoustic model, pronunciation model, and language model (LM) into the same single network is the best and most effective way of working with speech data. The focus of the paper is on the differences between language models, which can be categorized into shallow, deep, and cold fusion, and have different integration timings and training times. Per the authors' analysis of the models based on tests performed on two datasets, it was determined that shallow fusion is generally the best approach until the "second pass rescoring", in which cold fusion takes the lead.

Hidden Markov Models (HMM)

- One of the oldest and most prevalent models for speech recognition, used for sequence analysis. Prior to the use of the model, feature extraction is required, primarily in the form of Mel Frequency Cepstral Coefficients (MFCC), since Markov chains require discrete states. After using MFCC, the extracted features are converted into discrete variables, and can be

analyzed using HMM, which uses a generative probabilistic model to determine the next character based on the relationships between two sets of variables. In order to identify the next most likely character, the model requires an input with specific information about the language, which can be used for training purposes [6].

Listen, Attend, Spell (LAS)

- Listen, Attend, Spell (LAS) is an end-to-end model for automatic speech recognition, differing from HMM in that it does not make assumptions about the output sequence. LAS works by transcribing the audio sequence signal to a word sequence one character at a time. The operation is performed in two sequences, the "Listen" and the "Attend and Spell" operations. The first operation transforms the original signal into a high-level representation," while the second takes the high-level representation and produces the probability distribution over character sequences [7].

Recurrent Neural Network (RNN)

- The third model looked at for speech recognition used Recurrent Neural Networks (RNN), a variation on standard neural networks focusing on differentiating phonemes. RNN does not require prior training in the language being analyzed, making it far more adaptable than models such as HMM. RNN focuses on networks with multiple feedback connections, creating nodes that contain deep-seated memory which can be queried [8]. By using backpropagation, each network can iterate through the provided data and create weights for likely values which can be shared across networks.

Textual Keyword Mining:

- The second section identifies papers in which methodologies are proposed for the identification of keywords in text resources, such as topic mining and keyword extraction. Keyword extraction algorithms are generally divided into two types: supervised and unsupervised. The supervised keyword extraction method is mainly carried out by classification, by constructing a vocabulary, and then judging the matching degree of each document with each word in the vocabulary, in a similar way of labeling. The advantage is that the accuracy is high, but the disadvantage is that a large batch of labeled data is required, and the labor cost is too high. Unsupervised methods have low data requirements. Currently, the commonly used keyword extraction algorithms are based on unsupervised algorithms. Such as TF-IDF algorithm, TextRank algorithm and topic model algorithm (including LSA, LSI, LDA, etc.). "Topic mining as a scientific literature can accurately capture the contextual structure of a topic, track research hotspots within a field…" [9]. By grouping key features from the data, the clusters then can be quantified in the number of relationships there are between topics and features, thus giving a strong visual analysis of what the data is about, all done with limited loss of the textual implications in the data.

Latent Dirichlet Allocation (LDA)

- Use the LDA model that comes with gensim. The principle of the usage method is: the candidate keywords and the extracted topics are calculated and sorted to obtain the final keywords. The key, how to calculate the similarity between candidate keywords and extracted topics? The idea is: each topic is represented by the set of N words multiple by probabilities. Each text belongs to k topics, and the words contained in the k topics are assigned to the document, and the candidate word keywords of each document are obtained. If the words obtained after document segmentation are among the candidate keywords, they are extracted as keywords. (Candidate keyword, generally refers to the word obtained after the document word segmentation, here refers to the word contained in the subject of the document).

Term Frequency-Inverse Document Frequency (TF-IDF)

- One commonly used approach to text analysis is topic mining. Another approach to text analysis, described by Lee and Kim [10], uses term frequency (TF) in metadata analysis, an identification of a word or word pattern that appears most frequently in the article, while ignoring common stop words - terms that do

not add to the value of the article. The authors implemented an importance adjustment coefficient to measure whether a word is contextually relevant, using the Inverse Document Frequency (IDF) as the weight of the commonality of a term. The product of the TF and IDF is equivalent to the importance of the word within the article, and this method has the advantage of being simple and fast, and the result is more in line with the actual situation. The model involves a combination of topic mining and term frequency analysis and begins by creating a matrix of unique words within a dataset, then removing the highest and lowest frequency terms based on their TF-IDF product to account for both extremely common words like “and” and highly uncommon words. Once the matrix has been normalized to account for term frequency outliers, the authors then used the software R to randomly choose a distribution over topics and determine the topic proportions for certain topics within each document in the dataset. Each topic was randomly assigned to every nth word in each document, resulting in “the high-probability terms that define a topic in the corpus”, and once the model had determined 20 topics for each document, the process was repeated a total of 10 times, diminishing the impact of determining topics from a lower-frequency word.

TextRank

- The TextRank model is a graph-based ranking algorithm for determining term relationships within text-based datasets, based on Google's PageRank algorithm [11]. The modelling begins by displaying the significant words within the dataset as nodes, and creates edges between the nodes based on the degree of correlation within a close proximity. A TextRank score is then computed, based on the number of correlations and the significance decay by order of correlation, and the list of nodes with the highest scores is returned as the words of highest importance.

Semantic Analysis:

- Semantic analysis is the process in which a computer understands the sequence and meaning of words in the same way a human would, including a contextual understanding of colloquialism and homographs. In Wang, Wu, and Zhou’s article they look into finding the reasoning for a high rate of registration but low rate of completion amongst Massive Open Online Courses. They use the Semantic Analysis Model (SMA) to track emotional tendencies of Learners in order to analyze the acceptance of the courses based on big data from homework completion, comments, forums, and other real-time information [12]. Semantic classification technology plays an important role in intelligent information processing services, identifying themes within the data and increasing the metadata which can be extracted from collected data. The sentiment being derived is emotional (Happy, Sad, Angry, Disappointed, Surprised, Proud, In Love, and Scared), requiring lexical semantics to use the characteristics of different content to classify lexical items. The task of semantic analysis is to conduct context-sensitive relation and classification reviews of data.

N-gram Model

- Tripathy et al. proposed the N-gram model, which presumes that the appearance of any given word is correlated with a selection of other words [13]. Using a set of words with a given length, the N-gram model attempts to determine the overall contextual sentiment based on the emotions contained within. A typical implementation process would break the given text into predefined sections by word length (grams), and analyze the individual contents, before proceeding to analysis with a gram of greater size. A typical example would be to analyze the sentence "The movie is not a good one."
    - Its unigram: "'The','movie','is','not','a','good','one'", would provide an overall positive result due to the presence of the word “good”.
    - Its bigram: "'The movie','movie is','is not','not a','a good','good one'", which considers a pair of words at a time, would still provide the same result as the unigram.

    - Its trigram: "The movie is", "movie is not ", "is not a", "not a good", "a good one", which considers three words at a time, would provide an overall *bad* result, since it would take into account the presence of “not” before “good”, and identify that as a negation.

Latent Semantic Analysis

- LSA looks for patterns in the way words cluster together to give further background meaning to particular clusters. This clustering is done through singular value decomposition (SVD) of the term-document matrix. The basic idea behind LSA is to take advantage of implicit higher-order structure in the association of terms with documents ("semantic structure") in order to improve the detection of relevant documents, on the basis of terms found in queries [14]. LSA keywords are related to the primary keyword, providing word sense disambiguation such that “iPhone” is a keyword of "Apple", while "Apple" is a keyword of both electronic products and fruits.

Convolution Neural Network (CNN)

- Emotion recognition is an important interdisciplinary research topic in the fields of neuroscience, psychology, cognitive science, computer science, and artificial intelligence. Convolutional Neural Networks (CNN) is a statistical learning model inspired by biological neural networks, the goal of which is to automatically mark the text with defined labels. Common text classification tasks include emotion recognition, email filtering, intent identification, and data classification. Two-dimensional signals such as image and voice are hard to be modelled well by traditional models like SVM, so the ability of CNN to characterize two-dimensional signals makes it far more usable in bimodal data analysis. CNN can also adaptively extract features to eliminate the dependence on human subjectivity or experience [15].

III. PROBLEM STATEMENT

The main challenge faced by data analysts after any significant event in which public sentiment is a key factor of analysis is the volume of information available. An occurrence affecting a large group of people, such as some form of a natural disaster, or an election, results in the creation of massive amounts of data without any standard format or medium, and this makes the job of the data analyst that much harder. The project aims to identify the most effective models currently developed for the semantic evaluation of text and speech data and provide a model of this project which would permit the input of any text and speech datasets, separately or combined, and return a comprehensive overview of the key data points and semantic identifiers contained within. This model would allow future researchers to bypass the issue of determining which medium to focus on, as the metadata analysis within would combine the efficacy of existing text and audio data parsing algorithms.

The initial evaluation process was based on a literature review of existing proposals and ratings of models by prior researchers. These evaluations were used to guide decisions as to the capabilities built into the final implementation. Having determined the general composition of the model, a number of well-known and trusted Python-based libraries were identified that followed the modeling methodologies, and a framework of code was created to implement these libraries and generate the desired output from a bimodal dataset input. Finally, a comprehensive analysis of the capabilities and limitations of the model was performed, as well as an analysis of future work required to increase its effectiveness in a greater range of situations.

IV. METHODOLOGY

The methodology consisted of three key steps, starting with the identification of nine existing models for key data and semantic analysis. These models were then compared against one another in their respective categories, and the best from each category was used in the development implementation. The models were selected from among:

- 3 speech recognition models
- 3 key data mining models for text data
- 3 semantic analysis models for text and audio data

The second step was to determine which existing Python libraries were used to implement the models. The libraries were selected based on their popularity within other data analytics projects, as well as based on their performance evaluations. Throughout the implementation process, the Amazon review dataset[16] was used to perform an ongoing analysis, and assist in identifying areas of improvement.

The third step of the development was to combine the output from the library implementations into a single, cohesive file, with an output that could demonstrate the results of the model's analysis. The code was designed to perform a speech-to-text conversion of all audio files within the specified input folder, followed by a general text and sentiment analysis on the entire text dataset. Once analyzed, the statistics gained is output into a PDF document, as well as an interactive HTML file providing additional insight into the topic structure of the data.

Once the development and testing phases were complete, the chosen datasets were run through the model and the final output analyzed to determine the useability and limitations of the model.

V. Ranking Mechanisms Method and Evaluation Criteria

In Section 2, three different models for speech recognition, text mining, and semantic analysis were determined. In order to narrow down these models to the best from each respective category, they were ranked based on a set of criteria: compatibility, dimensionality, accuracy, and documentation (see Figure 1). Each model was ranked on a scale between one and ten for each criterion, and the total score was then used to determine what models would be used for the final implementation (see Figure 2).

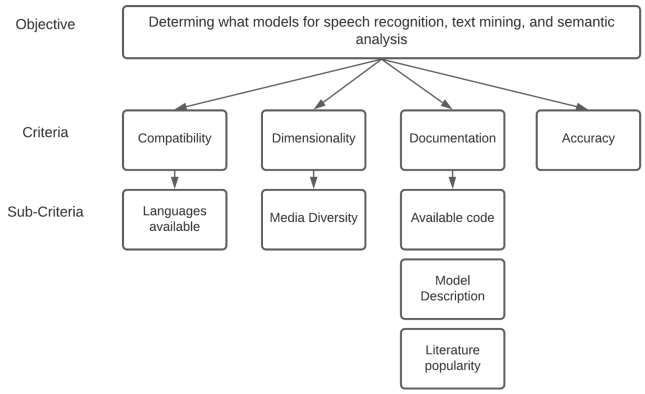

Figure 1: Analytical Hierarchy Process Diagram

- *Compatibility:*

This objective refers to how easily and with how much extra effort it takes to work with other models. This includes the language(s) in which the model can be used in as well as the amount of customization the model requires. The lowest ranking being "impossible to write this model in one of the common data focused languages (i.e. Java, Python, R), or has some barrier to being used together with other models". With the highest ranking being "Easily works with other models and existing code for major languages (Python, R, JAVA)

- *Dimensionality:*

The dimensionality refers to being capable for both text and audio/ multiple functions, such as keyword analysis and semantic analysis. The lowest score being "can only perform one function, and can be used on only one type of media". The highest score being "can do multiple kinds of analyses and can be used on multiple forms of media".

- *Accuracy:*

Accuracy refers to the measurement of how closely the model's output matches the estimated or desired output, and was devised largely from the documentation and reviews of the models, as well as some testing with the Amazon dataset.

- *Documentation:*

This criteria refers to how much literature there is for the particular model as well as documentation on both how to build and how to use the models. The lowest score being "little to no documentation, no usable code, vague description for the model, not popularly used in literature". The highest score being "Plentiful and strong documentation, variety of existing code, detailed model description, and very common amongst academic literature".

		Compatability	Dimensionality	Accuracy	Documentation	Total
Speech Recognition	HMM	8	4	6	10	28
	LAS	7	4	8	7	26
	RNN	5	8	5	5	23
Textual Keyword Mining	LDA	7	8	7	6	28
	TF-IDF	7	5	4	7	23
	TextRank	4	7	6	6	23
Semantic Analysis	LSA	7	5	6	8	26
	N-Gram Model	4	3	6	7	20
	CNN	4	7	7	5	23

Figure 2: Analytic Hierarchy Process Scoring Table

VI. Models Chosen for Implementation

HMM

- Compatibility:
  - This criteria scored an 8 as it has been developed into working with many models and can be used in a variety of languages such as Python, R, and MatLab.
- Dimensionality:
  - A score of 4 was given as HMM primarily works with Speech recognition and text analysis and cannot be used for much more.

- Accuracy
    - A score of 7 was given for this model as when the data is pre processed accurately the results become very accurate.
- Documentation
    - The highest score was given for documentation as HMM was developed in the late 1900's and has an incredible amount of documentation and resources that refer to it.

LDA

- Compatibility:
    - A score of 7 was given for the implementation of the LDA process, because this model has been developed and used many times. Compared with the TextRank model, the LDA model has simple operation and fast calculation speed.
- Dimensionality:
    - A score of 8 was given to the LDA model. The traditional method of judging the similarity of two documents is to look at the number of words that appear in the two documents, such as TF-IDF. This method does not focus on semantic association. For example, "It's winter now", "Will summer clothes be discounted?" These two sentences do not have common words, but the two sentences are similar. If you judge the two sentences according to the traditional method, they are definitely not similar, so when judging the relevance of the document, you need to consider the semantics of the document, so LDA is one of the more effective models.
- Accuracy
    - The highest score was given for this model. The TF_IDF model means "The TF-IDF value increases when a specific keyword has high frequency in a document and the frequency of documents that contain the keyword among the whole documents is low"[17]. For the same topic, the keywords may be the same, but due to the Inverse Document Frequency, the keyword score will not be very high, so the IF_IDF model is not a good choice. LDA can distinguish the same topic well and find out the keywords accurately.
- Documentation
    - A score of 6 was given for the documentation as the visualization of LDA has been greatly developed in the past ten years, so it has more research papers.

LSA

- Compatibility:
    - A score of 7 was given as concise Python implementations are easy to find and many implementations have other functionality included (i.e. pre-processing and TF-IDF implementations)
- Dimensionality:
    - The middle score of 5 was given as LSA can only really be used on text, and generally only for getting main topics. It is not great for identifying emotional sentiments.
- Accuracy
    - A score of 6 was given for this model as LSA is accurate within reason to get the main topics of the data - not as accurate as latent Dirilecht analysis, and does not provide emotional data.
- Documentation
    - An 8 was received in documentation as there is a copious amount of data referring to theory however, not as much about how to actually implement the model.

VII. Implementation

For the implementation of the chosen models - HMM, LDA, and LSA - existing code found on GitHub was used, as given the time constraints for this project, as well as the complexity of the models, original code could not be implemented for all three models [18][19][20]. The approach to the implementation was to identify code repositories that

did not include an abundance of extraneous functionalities beyond what the project desired for the final product, and that could work completely independently without dependencies on other repositories or non-standard modules. Given the prevalence of Python for the implementation of text- and audio-mining models, the final implementation was also written in Python.

The code implementation of the HMM model was based on the Python library speech_recognition [21], which provides a variety of functions relating to usability of audio files. Of these many functionalities speech_recognition was used to transcribe the audio file to a text format. In order to do so a speech recognition engine had to be decided, and in the final implementation the Google API was chosen, having the most robust audio transcription out of the analyzed models. The model identified phonemes (multi-letter units of speech) within the speech signals after accounting for noise within the recording, and applied an algorithm to determine the most likely word composed of the recorded phonemes. The implementation of the LDA model used the Python gensim library for data preprocessing, as well as for the implementation of TF-IDF, on which the LDA model is partially dependent [22]. The gensim library is used widely for natural language processing (NLP) and facilitates the use of machine learning for unsupervised topic modeling. In the model testing phase, code from implementation [19] was used, slightly modified for the project, to analyze a dataset consisting of user reviews of products in Amazon's "Fine Dining" department. The model was able to generate the following topic compositions for the dataset, ranking from the highest probability to the lowest:

1. Score: 0.5362482070922852
   Topic: 0.014*"order" + 0.013*"amazon" + 0.012*"price" + 0.011*"store" + 0.011*"ship" + 0.010*"product" + 0.008*"great" + 0.007*"arriv" + 0.007*"purchas" + 0.007*"local"
2. Score: 0.2492866963148117
   Topic: 0.012*"water" + 0.007*"bottl" + 0.005*"tast" + 0.004*"drink" + 0.003*"like" + 0.003*"gummi" + 0.003*"product" + 0.003*"wine" + 0.003*"matcha" + 0.003*"good"
3. Score: 0.1334620863199234
   Topic: 0.010*"cereal" + 0.010*"butter" + 0.010*"popcorn" + 0.009*"peanut" + 0.009*"gluten" + 0.007*"free" + 0.007*"love" + 0.007*"oatmeal" + 0.007*"tast" + 0.007*"great"
4. Score: 0.06383361667394638
   Topic: 0.019*"food" + 0.019*"treat" + 0.015*"dog" + 0.010*"cat" + 0.010*"love" + 0.008*"chew" + 0.005*"train" + 0.005*"like" + 0.005*"chicken" + 0.005*"puppi"

The implementation of the LSA model was based on the text2emotion library [23], which analyzed the composition of any text input based on its component words, and determined the breakdown into the five emotions of happiness, sadness, anger, surprise, and fear. Additionally, the library provided a measurement of polarity for each input, which measures the negativity, positivity, or neutrality of the contents.

The final repository of code obtained was able to determine whether the input data was a text or audio file, and either apply or omit the HMM speech recognition model to transcribe it depending on the determination. It then performed an analysis of the dataset's topic composition, as well as it's general semantic statistics, and presented this as a report, in addition to several other pieces of data derived during the script execution, such as the results of the TF-IDF calculations.

VIII. Conclusions

In recent years, artificial intelligence has developed rapidly, and multi-modal data analysis has become a research hotspot. Semantics, a vital aspect of natural language processing (NLP), has attracted more and more attention, due to its complexity and its importance in understanding social trends as well as advancing marketing technology. Multi-modal recognition methods that consider audio, video, text, biological, and other types of data can improve the efficacy of common data analysis methods, so the focus of this research was to identify the best NLP modeling approaches to implement with a bimodal dataset.

There are many different approaches to combining text and audio data into a single analysis process and considering the relative novelty of speech recognition and audio semantic analysis technologies, the most efficient and effective method is to transcribe audio data into text and perform a comprehensive analysis on the joint data in text form. Through experimentation and a review of the literature surrounding models in

the areas of speech-to-text transcription, topic and text mining, and semantic analysis, three models were selected which were most applicable for the stated purpose. These models and algorithms have their own advantages and disadvantages and are dependent on many variables including the field of application and the type of data provided, so for the purpose of the research they were selected based on the models' compatibility, dimensionality, accuracy, and level of documentation.

The model chosen for transcribing audio data into text was the Hidden Markov Model (HMM), which outperformed other models in its field based on the numerous implementations written for it, and the accuracy of the results of the transcription, which present a significantly lower Word Error Rate than other models such as Recurrent Neural Networks (RNN). For text data mining and topic analysis, the Latent Dirichlet Allocation model was selected, based on its operational efficiency, its inclusion of context and semantics in the analysis process, and its high level of documentation. Finally, for the purpose of semantic analysis, the Latent Semantic Analysis (LSA) model was implemented, a well-documented and multi-purpose model which provides a more accurate analysis of emotional content than more basic models such as the N-gram model.

Through experimentation and the use of these models to analyze two datasets involving data based on TED talks, with both audio recordings and text transcriptions, an output was generated providing basic statistics of the data, as well as more intricate information through data mining. Following a transcription of the audio recordings, which were all output from the program, an analysis of the major keywords and topics within the datasets, as well as the topic distribution, was performed and written into a report. Additionally, a measurement of the key emotions within the data and its polarity was derived, providing significant insight into the makeup of the data. While the lack of access to more viable datasets prevented the research from demonstrating the combined model's true usability, the output demonstrated that bimodal and multimodal data analysis should be a significant focus of data analytics, considering its immense potential and the amount of data currently untapped within audio, video, photographic, biological, and a myriad of other data formats.

IX. Limitations

There are a few limitations that the model faces, one of which lies in the lack of compatibility with a variety of data types for speech recognition. The code used to convert audio files into text works only with a limited file selection (WAV, FLAC, and AIFF), which required other data types to be converted externally prior to the use of the model (the authors used a tool called *Audacity* for conversion of MP3 files into the WAV format). Another limitation related to the audio transcription library is the maximum permitted duration of audio files for analysis – the use of the Google API introduces this limit, which the authors bypassed through additional code designed to separate longer audio files into segments of acceptable length, using the audio_segment Python package.

Another limitation identified during the final implementation of the model was its general inefficiency. When used with a dataset containing around 24GB of audio, and 350MB of text data, the model required nearly 24 hours to complete the initial generation of the CSV used to contain the transcribed audio and emotion data. While the model is more optimized for text data, it would still be prudent to optimize the audio transcription process prior to use with a significantly larger dataset.

X. Future Work

Due to the time constraints of the research, there are several improvements and alterations that could be implemented by future researchers, such as an implementation of real-time data analysis. This could be an extremely useful tool, providing a live analysis of an audience's emotional perception of an event or proposal. Using a wider variety of data sources such as streaming data and photographic data could also increase the usability of the model, potentially providing analyses which are imperceptible in a textual or auditory context.

While the initial intention of the research was to perform initial baseline tests with models of social media activity during and after a natural disaster, in combination with emergency response call logs for the same period, the lack of access to such datasets meant the true efficacy of the final model for such a use case remained unmeasured. Provided access to this data, future research could center on enhancing the model with additional capabilities making it more useful for retrospective analysis of the social impact of natural disasters and emergency response efficiency.

Another enhancement which could greatly impact the model's usability would be to review the code implementation and attempt to write original code to replace the use of Python's libraries and packages. While these were key in the initial implementation, they provided several limitations on variables such as input and output data types, the efficiency of execution, and the size of the input that could drastically increase performance were they removed.

References

[1] Wu, Yu, et al. "New anti-spam filter based on data mining and analysis of email security." Data Mining and Knowledge Discovery: Theory, Tools, and Technology V. Vol. 5098. International Society for Optics and Photonics, 2003.

[2] Tan, Zheng-Hua. "Audio and speech processing for data mining." Encyclopedia of Data Warehousing and Mining, Second Edition. IGI global, 2009. 98-103.

[3] Gautam, Geetika, and Divakar Yadav. "Sentiment analysis of twitter data using machine learning approaches and semantic analysis." 2014 Seventh International Conference on Contemporary Computing (IC3). IEEE, 2014.

[4] Shoumy, Nusrat J et al. "Multimodal Big Data Affective Analytics: A Comprehensive Survey Using Text, Audio, Visual and Physiological Signals." *Journal of network and computer* applications 149 (2020): 102447–. Web.

[5] Toshniwal, Shubham et al. "A Comparison of Techniques for Language Model Integration in Encoder-Decoder Speech Recognition." (2018): n. pag. Print.

[6] Chavan, Rupali S., and Ganesh S. Sable. "An overview of speech recognition using HMM." *International Journal of Computer Science and Mobile Computing* 2.6 (2013): 233-238.

[7] W. Chan, N. Jaitly, Q. Le and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4960-4964, doi: 10.1109/ICASSP.2016.7472621.

[8] Venkateswarlu, R. L. K., R. Vasantha Kumari, and G. Vani JayaSri. "Speech Recognition by Using Recurrent NeuralNetworks." *International Journal of Scientific & Engineering Research* 2.6 (2011): 1-7.

[9] Zhang, Tingting et al. "Multi-Dimension Topic Mining Based on Hierarchical Semantic Graph Model." *IEEE access 8* (2020): 64820–64835. Web.

[10] Sungjick Lee, and Han-Joon Kim. "News Keyword Extraction for Topic Tracking." *2008 Fourth International Conference on Networked Computing and Advanced Information Management*. Vol. 2. IEEE, 2008. 554–559. Web.

[11]Zhang, Mingxi et al. "An Empirical Study of TextRank for Keyword Extraction." *IEEE access 8* (2020): 178849–178858. Web.

[12] Wang, Ling, et al. "Semantic Analysis of Learners' Emotional Tendencies on Online MOOC Education." Sustainability, vol. 10, no. 6, 2018, p. 1921., https://doi.org/10.3390/su10061921.

[13] Tripathy, Abinash, Ankit Agrawal, and Santanu Kumar Rath. "Classification of Sentiment Reviews Using n-Gram Machine Learning Approach." *Expert systems with applications* 57 (2016): 117–126. Web.

[14] Zhang, Wen, Taketoshi Yoshida, and Xijin Tang. "A Comparative Study of TFIDF, LSI and Multi-Words for Text Classification." *Expert systems with applications* 38.3 (2011): 2758–2765. Web.

[15] B. Zhang, C. Quan and F. Ren, "Study on CNN in the recognition of emotion in audio and images," 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), 2016, pp. 1 -5, doi: 10.1109/ICIS.2016.7550778

[16] Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering R. He, J. McAuley *WWW*, 2016

[17] Kim, SW., Gil, JM. Research paper classification systems based on TF-IDF and LDA schemes. Hum. Cent. Comput. Inf. Sci. 9, 30 (2019).

[18] wblgers, hmm_speech_recognition_demo, (2018), GitHub repository, https://github.com/wblgers/hmm_speech_recognition_demo

[19] bjherger, Easy-Latent-Dirichlet-Allocation, (2016), GitHub repository, https://github.com/bjherger/Easy-Latent-Dirichlet-Allocation

[20] susanli2016, NLP-with-Python, (2020), GitHub repository, https://github.com/susanli2016/NLP-with-Python

[21] Zhang, A. (2017). Speech Recognition (Version 3.8) [Software]. Available from https://github.com/Uberi/speech_recognition#readme

[22] Rehurek, Radim, and Petr Sojka. "Software framework for topic modelling with large corpora." *In Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks*. 2010.

[23] aman2656, text2emotion-library, (2020), GitHub repository, https://github.com/aman2656/text2emotion-library

[24] Panayotov, Vassil, et al. "Librispeech: an asr corpus based on public domain audio books." 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2015.

[25] Ni, Jianmo, Jiacheng Li, and Julian McAuley. "Justifying recommendations using distantly-labeled reviews and fine-grained aspects." Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019.

[26] Rounak Banik. TED Talks, version 3 (2017). https://www.kaggle.com/rounakbanik/ted-talks