Overview of machine learning task and justification for your methods. An unsupervised machine learning method could be most suitable here (e.g., word2vec).

Text Data

For this project, the chosen dataset was Rotten tomatoes reviews and looked into through a machine learning algorithm after the python code was created. This was selected as it was interesting to see the patterns of movie critics. This Dataset provides reviews of movies rated by rotten or fresh and other adjectives that may be used, including those. Any information, such as movie information, can be obtained instantly in the age of information technology. Every year, a new film is released. It would be preferable if there were a tip to filter movies that people watch. There are many film options. One of the suggestions is rotten tomatoes. It provides the suggestion based on the score calculated by the Expert critic review used in the system. (Sankar et al., 2020). On the other hand, the procedure is not open to the public.

Machine learning is increasingly being used in various fields, ranging from material quality testing to the development of solid computer vision technologies. (Sankar et al., 2020). The recommendation system, a technology that proposes things to consumers based on their tastes, is one such modern use. (Sankar et al., 2020). Multi-task learning (MTL) has lately aided in developing better representations for various NLP tasks. MTL tries to improve a primary task's performance by collaborating on a secondary assignment. ( (Liang & Shu, 2022). This research focuses on a specific recommendation system known as movie recommendation. We use user reviews of movies to form an overall opinion on the film, which we then use to recommend the film to other users. However, sophisticated review systems have been confounded by many available reviews. As a result, a mechanism for extracting useful information from the existing reviews must be developed and implemented in identifying movie reviews and forecasting how each one will be received. A movie review can be good, harmful, or neutral. (Sankar et al., 2020).

One of the approaches for analysing user opinion from a review is sentiment analysis. According to past research, many researchers use two ways in sentiment analysis, the linguistic-based approach and a supervised machine learning approach. ( X. Ouyang, P et al., pg., 2359) The linguistic method involves employing a dictionary with sentiment polarity, such as SentiWordnet.

The defined label is used as a target in a supervised machine learning strategy that uses statistical analysis to predict sentiment polarity. ( Chatterjee et al.,2021) The domain determines the problem of supervised machine learning. Machine learning must be retrained to achieve high accuracy if the data is from a new environment. However, it is word2vec and an unsupervised learning method in this case. This is used by writing python code(python -m gensim. scripts.word2vec2tensor -i OneDrive/Desktop/model/last\_rt-reviews.bin -o OneDrive/Desktop/model/ to tokenise, etc making a bin file. After the bin file using a filer directory path alongside “is run through the terminal, a tensor and metadata file is made for an embedding model and put into https://projector.tensorflow.org/.

Data pre-processing steps.

There were many steps, such as tokenisation(In Python, tokenisation refers to **splitting up a larger body of text into smaller lines, words or even creating words for a non-English language)**. And the lower case, etc.

Adding variable which allows for the tokenisation

sample sentence = """fresh,"

This then allows the text to be tokenised.

Tokenise text.

Code: reviews = [nltk.word\_tokenize(review) for review in reviews]

this looks like this “['manakamana',

'does', "n't", 'answer', 'any', 'questions',

Values in an array changed to a list.

Code: reviews = list(df['Review'].str.lower().values)= “" wilfully offensive and powered by a chest-thumping machismo, but it is good clean fun.",

' It would be difficult to imagine material more wrong for spade than lost & found.',”

Reviews. This allows for creating lists.

All strings to lowercase

Code: reviews = df['Review'].str.lower()=“0 manakamana doesn't answer any questions, yet ...” Making this all-lower case allows for it easier to format and keep code consistent. It also makes it easier to follow and see what the patterns are.

Method

Problems with method

The method was quite complicated due to some problems regarding the laptop used. Everything linked to one drive, so it struggled and was an unfixable error. Said unfixable error was 'Word2VecKeyedVectors' object has no attribute 'index\_to\_key.'. This is due to the key used in code for analysing text ‘review\_model.wv.key\_to\_index['actor'].’ This is meant to discover analogies and patterns within the dataset through machine learning and the code.

Method once fixed for writing code.

After this issue was fixed, the code was written out poorly, starting with sorting folders and writing the path so I could open the CSV for a dataset of rotten tomatoes. After I set up the code for importing packages such as NumPy and panda. Then wrote the tokenising code strings, lower casing, returning rows, and manipulating lines. Lastly, set up code for patterns and analogies.

Method of creating embedding file and creating patterns

Firstly, the code was created to make a bin file which review\_model.wv.save\_word2vec\_format('last\_rt\_reviews.bin'). after that was run, I used terminal to generate the tsv and metadata file. This was able to occur due to the package genism and using word2vec. This now allows me to study the patterns. It runs the dataset and shows patterns between attributes in the dataset, such as how a word is represented, such as ‘woman’, and how many ways it is shown in a sequence of numbers.

Instances of the algorithm

Multiple-instance learning (MIL) is supervised learning in machine learning. Instead of receiving a collection of individually labelled examples, the learner is given a set of labelled bags containing many instances. A bag can be labelled negative if all of its models are negative in the simple situation of multiple-instance binary classification. On the other hand, a backpack is termed positive if it contains at least one positive instance. The learner attempts to either induce a concept that will correctly label cases individuals from a collection of labelled bags or (ii) learn how to label bags without causing the idea from a group of labelled bags. (e.g., a bag of words)

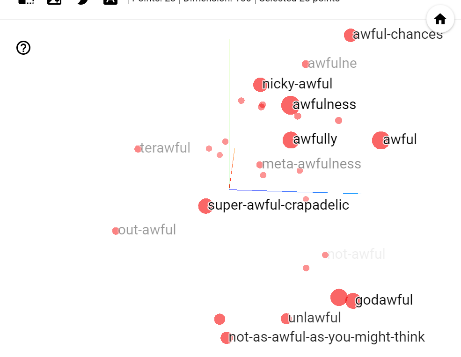
An instance is a specific example from the training data. Several attributes are used to describe a model. A class label is an example of a point. Aspect/Attribute: A fact is a feature of an instance (e.g., temperature, humidity). In Machine Learning, details are commonly referred to as features.

Step 1: Obtain a description of the problem. This step is much more complex than it appears. Step 2: Analyse the situation. Step 3: Develop a high-level algorithm. Step 4: Refine the algorithm by adding more detail. Step 5: Review the algorithm. Lastly, attributes were added, such as analogies like a” man is to woman” or “the king is to the Queen.”

Evaluation and findings

Present any visualisations/results from your analysis

Patterns for awful isolated (1) and terrible of all data (2)

Text

Description automatically generated with medium confidence

This model has three patterns, similar to the above visualisation of the word awful. There are analogy patterns and word representation patterns. The analogy consisted of “analogy = review\_model.wv['actor'] - review\_model.wv['man'] + review\_model.wv['woman']” this showed the different analogies within the dataset for actors and men and their opposites women and actress among other variations. Lastly, there is the word representation pattern that was found. In NLP, word representation, which aims to represent a word with a vector, is crucial. Many vectors are used to display this, in particular regarding the code. The word man in the array is displayed in various ways in an array, such as “array([ 0.22268021, 0.13195364, 3.4917772, 0.17641763, -1.599144,

-0.56681216, -1.6402311 , -0.07568511, 2.3460414 , 2.1069791 ,

1.4401008 , 0.959796 , 0.69077986, 0.71069545, 3.2943869 ,

1.4278432 , -0.15102923, -3.686055 , -0.78208023, 2.5736213 ,])”

Natural Language Processing (NLP) and Artificial Intelligence can be applied for sentiment analysis of this textual data. NLP applications nowadays rely on pre-trained embeddings drawn from massive corpora such as news and web crawlers for text processing. (Yu et al., 2022) This paper compared the accuracy of different word embeddings (pre-trained and untrained) for sentiment analysis of embedding models. Regarding this assignment, the theme seems to be that the patterns are pretty similar to how people write reviews, such as many using words like awful, charming, etc.; this shows that people tend to think the same way when writing reviews. The analogies show how it can assert variations within the assessments and how there is a spectrum of what people write within said reviews. Lastly, as for the word representation, the vectors are pretty close to one another, for example, the word ‘man’. This would mean people will use similar but almost contrasting words far from the original word vector.

Bibliography

Sankar, H., Subramaniyaswamy, V., Vijayakumar, V., Arun Kumar, S., Logesh, R., & Umamakeswari, A. (2020). Intelligent sentiment analysis approach using edge computing‐based deep learning technique. *Software: Practice and Experience*, *50*(5), 645-657. DOI: 10.1002/spe.2687

X. Ouyang, P. Zhou, C. H. Li and L. Liu, "Sentiment Analysis Using Convolutional Neural Network," 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, 2015, pp. 2359-2364, DOI: 10.1109/CIT/IUCC/DASC/PICOM.2015.349.

Yu, A., Lee, H., & Le, Q. (2022). Learning to Skim Text. Retrieved from <https://arxiv.org/abs/1704.06877>

Chatterjee, S., Chakrabarti, K., Garain, A., Schwenker, F., & Sarkar, R. (2021). JUMRv1: A Sentiment Analysis Dataset for Movie Recommendation. Applied Sciences, 11(20), 9381. <https://doi.org/10.3390/app11209381>

Liang, D., & Shu, Y. (2022). Deep Automated Multi-task Learning. Retrieved 20 April 2022, from <https://arxiv.org/abs/1709.05554>