國立臺灣大學社會科學院經濟學系
碩士論文
Department of Economics
College of Social Science
National Taiwan University
Master Thesis

基於歌詞與歌曲音訊特徵之熱門歌曲預測
Lyrics and Audio Features-based Hit Song Prediction

張詠翔
Yong-Xiang Zhang

指導教授：馮勃翰博士
Advisor: Po-Han Fong, Ph.D.

中華民國 107 年 7 月
July, 2018

doi:10.6342/NTU201802042

摘要

本研究目的在於探討單純使用音樂本質的資訊，是否有辦法預測一首歌曲是不是熱門歌曲？研究方法主要使用機器學習的技巧，資料方面使用了 Spotify 所提供的音訊特徵與 KKBOX 所提供的歌詞。研究中我們測試了單純使用歌詞的模型、單純使用音訊特徵的模型與兩者混合的模型，結果顯示使用了歌詞與音訊的模型表現高過單純使用歌詞的模型 3%，而使用了音訊特徵的模型表現高過單純使用歌詞的模型 2%。

關鍵字： 機器學習、深度學習、自然語言處理、音樂資訊檢索、增強方法

JEL 分類代號： C45、C51、C52、C55、L82

doi:10.6342/NTU201802042

Abstract

The purpose of our research is to explore whether we can just use audio and lyrics-based features to predict if a song is a hit or flap. We mainly use the Machine Learning skills to build models and obtain the lyrics and audio data from KKBOX and Spotify, respectively. In our research, we build models using lyrics, audio and the mixture of lyrics and audio. The results show that the performance of the model based on mixture is 3% higher than the model based on lyrics and the performance of the model based on audio is 2% higher than the model based on lyrics.

Keywords: Machine Learning, Deep Learning, Ensemble Methods, NLP, MIR

JEL Classification: C45, C51, C52, C55, L82

doi:10.6342/NTU201802042

目錄

doi:10.6342/NTU201802042

doi:10.6342/NTU201802042

圖目錄

doi:10.6342/NTU201802042

表目錄

doi:10.6342/NTU201802042

章節 1

緒論

假如你是一位歌手、作詞家、作曲家、甚至是一家音樂串流公司的決策者，你可能會很關心一項問題 — 究竟一首怎麼樣的歌曲會賣座？有部分的人相信著一首歌曲會紅是有脈絡可循的，甚至出版了著作教導人們如何寫出一首賣座的歌曲，如：Perricone (2000) 與 Blume (1999)。假如在一首歌發布前能準確的預測一首歌會不會造成賣座，創作歌曲或是發行歌曲的人在發行歌曲前就能有一個參考，讓他們更有效的進行創作與進行成本控管。

然而，分析歌曲的資料不像是經濟學面臨的問題，在資料的處理上就會面臨一定程度的困難，如：從歌詞與音訊中抽取特徵等，需要用到更複雜的算法。成功抽取出特徵後在對於特徵的詮釋上也有一定程度的困難。但是由於近年電腦算力的進步，機器學習、深度學習領域中以往需要強大算力的算法得以在實務與研究中應用，使得預測歌曲是否會賣座的問題也得以進行研究。

早期的研究宣稱了我們能夠使用機器學習的技巧，從歌曲音訊與歌詞中發現一些造成賣座的要素Dhanaraj and Logan (2005)，但是Pachet and Roy (2008) 利用了更大規模的資料來驗證相同的問題，不同的是，他們做出了與Dhanaraj and Logan (2005) 相悖的結論，相關領域的研究便沈寂了幾年。

在近年，Herremans, Martens, and Sörensen (2014) 將過往相關研究關注的問題轉為較為簡單的問題，不關注在預測一首歌的真實排名或是播放量，而是將問題簡化成一首歌是不是一首賣座歌曲，他們的研究中關注在預測一首歌是不是一首排行榜前幾名的歌曲。換句話說，將量化 (Quantitative) 或是多標籤 (Multi-labeling) 的模型轉換成較為簡單的分類 (Classification) 問題。在簡化了研究議題後，他們

doi:10.6342/NTU201802042

得到了熱門歌曲似乎還是得以預測的結論。而在利用歌詞預測的文獻中，Singhi and Brown (2014) 單純利用了歌詞來預測一首歌會不會賣座，得到了模型準確率優於隨機選取的結論。

除了基於歌曲音訊與歌詞的預測以外，也有一些研究者將問題關注在造成歌曲賣座的其餘因素。Bischoff, Firan, Georgescu, Nejdl, and Paiu (2009) 在他們的研究中利用了音樂社群網路和歌曲、專輯與歌手的關係，成功的預測了一首歌是否能成為一首賣座歌曲。Garg, Smith, and Telang (2011) 利用了音樂社群進行了訊息傳播的研究，發現社群中的同好有助於音樂訊息的傳播與發現。Kaplan and Haenlein (2012) 研究了歌手 Britney Spears 在發行單曲的期間與粉絲在 Twitter、Facebook 與 YouTube 的互動情形。Kim, Suh, and Lee (2014) 利用了推特 (Twitter) 的 # now playing 標籤發現了歌曲與歌手的每日轉推 (Daily Tweets) 可以有效的預測一首歌在排行榜的名次。然而，在我們的研究中，我們只會將問題關注在單純利用歌詞與音樂訊號抽取出的特徵來預測歌曲是否會賣座。

在研究方法選擇的部分，我們選擇了使用機器學習的技法，而不使用經濟計量方法。原因在於我們的資料有著高維度、分布偏斜等特性，因此我們需要利用較複雜的模型。除此之外，在建立歌詞的預測模型時，我們所面臨的資料會是一串序列 (Sequence) 而非單個資料點 (Sample Point)，因此必須使用機器學習的方法來處理。

但使用經濟計量方法並非劣於機器學習技法，利用機器學習的技法做出的結果，往往由於過於複雜的模型架構，導致研究者對於結果的詮釋困難重重，反觀經濟計量領域的模型，對於結果能夠有一個比較好的詮釋。在經濟計量領域，對於因果關係造成的效果 (Causal Effect) 非常的重視，也發展出了一些工具來處理這樣的問題，如：工具變數 (Instrumental Variables) 等，但在機器學習的領域對於因果效果的估計則相對比較困難。

如果在建立模型時，我們更關心預測的準確度，則我們應該選擇使用機器學習的技法，反之，如果我們對於解釋變數對於被解釋變數的詮釋、變數間的因果關係更為重視，我們則應該使用經濟計量方法。在機器學習的領域中，有許多方法也能夠被應用在經濟計量模型的建制過程，如：交互驗證 (Cross Validation)、變數選取 (Feature Selection)、超參數調整 (Hyperparameters Tuning) 等方法。有關於經

doi:10.6342/NTU201802042

濟計量方法與機器學習技法的優劣比較，能夠參考Varian (2014)。值得一提的是，在我們的研究問題中，我們相對於變數間的因果關係，更關心模型預測的準確度，因此我們選擇了使用機器學習的技法。

另外，在歌詞預測的模型中，有許多的研究者在做文本分析時，會關注在哪些題材的組合能夠造成歌曲的賣座，因此會使用一些主題模型 (Topic Model)，像是潛在狄立克雷分配 (Latent Dirichlet Allocation，LDA)、潛在語義標注 (Latent Sematic Indexing，LSI) 等，但在我們的研究中，我們希望直接給予模型觀察每首歌曲的代表性字詞後，直接回答歌曲是否會造成賣座。因此沒有使用主題模型來進行分析。

在接下來的章節中，我們會在章節2中對於我們的資料作介紹，並做一些簡單的探索性分析 (Exploring Data Analysis，EDA)、在章節3介紹我們的研究中所使用的算法、章節4介紹實驗方法的設計、在章節5詮釋我們的實驗結果並在章節6闡述我們的結論。

doi:10.6342/NTU201802042

章節 2

資料

我們的資料來源主要分為兩部分，歌詞的資料主要從 KKBOX[1]的網頁上面爬取。而歌曲音訊特徵則主要從 EchoNest[2]所提供的開發者工具呼叫其 API 獲取。我們的資料包含了歌曲的基本資料，如：歌手，發行日期，歌曲名稱，專輯名稱等、EchoNest 所獲取之音訊特徵，如：popularity、danceability 等、與歌詞。發行日期從西元 1982 年 9 月 19 日一直到 2017 年 12 月 20 日，共包含了約 31000 首歌、650 位歌手、11500 首不重複歌曲。我們會在接下來的小節依序針對音訊特徵與歌詞作介紹。

2.1 音訊特徵

我們的歌曲音訊特徵主要從 EchoNest 的 API 所獲得，表2.1為 EchoNest 所提供的說明文件，我們可以從文件中了解每個變數所衡量的層面，但值得一提的是，對於這些變數如何從歌曲中抽取出來的我們無從得知，在文件中也沒有詳細說明。我們的變數中，popularity 為歌曲熱門程度，我們主要會利用此指標轉換為我們模型中的被解釋變數，acousticness、danceability、energy、instrumentalness、liveness、speechiness 和 valence 都為介於 0 到 1 之間的變數，分別衡量了一首歌的情感程度 (快樂-悲傷)、人聲含量 (人聲-旋律)、電子樂器含量 (電子-古典樂器) 等。我們可以利用觀察一首歌的音訊特徵，讓我們在還沒聽到歌曲時大致知道這

[1]KKBOX 為亞洲地區知名音樂串流公司。
[2]EchoNest 為一提供音訊解析、辨識等解決方案之公司，現為 Spotify 子公司。

doi:10.6342/NTU201802042

表 2.1

EchoNest's API 文件

變數	敘述
popularity	track's popularity.
acousticness	from 0.0 to 1.0 of whether the track is acoustic.
danceability	a value of 0.0 is least danceable and 1.0 is most danceable.
energy	from 0.0 to 1.0 and represents a perceptual measure of intensity and activity.
instrumentalness	predicts whether a track contains no vocals.
key	the key the track is in.
liveness	detects the presence of an audience in the recording.
loudness	the overall loudness of a track in decibels (dB).
mode	indicates the modality (major or minor) of a track.
speechiness	detects the presence of spoken words in a track.
tempo	the overall estimated tempo of a track in beats per minute (BPM).
valence	from 0.0 to 1.0 describing the musical positiveness conveyed by a track.

Note: 這份表格描述了 EchoNest 所提供的每個變數所衡量的層面說明。其中值得注意的是，popularity 代表了歌曲的熱門程度，也是我們模型中會用來轉換為被解釋變數的指標，acousticness、danceability、energy、instrumentalness、liveness、speechiness 和 valence 都為介於 0 到 1 之間的變數。

首歌曲聽起來會是怎麼樣的感受。

在歌曲音訊特徵的資料中，我們可以發現 popularity 為一個左偏分配，顯示出大部分的歌曲都是較不紅的歌。而經由 danceability、energy、mode、speechiness、acousticness、instrumentalness、liveness 和 valence 等變數可以看出，在我們資料中的歌曲平均而言屬於調性為大調、用字較少、非現場演奏、非純樂器演奏、較悲情的歌曲。表2.2與圖2.1列出音樂特徵資料的敘述統計與畫出變數的分佈，其中有三個變數，speechiness、instrumentalness 與 liveness 非常集中於接近 0 的位置。

除了每個變數的分配以外，我們可以進一步探索變數之間的關係，由圖2.2可以看出，有些變數的關係相當直覺，如：acousticness 與 energy、loudness 的反向關係，valence 與 energy、danceability 的正向關係等，但值得一提的是，popularity 與音訊特徵並無明顯的關係。

doi:10.6342/NTU201802042

表 2.2

音訊特徵敘述統計

	count	mean	std	min	50%	max
popularity	30928	28.521	18.866	0.000	27.000	96.000
danceability	30928	0.555	0.138	0.000	0.556	0.969
energy	30928	0.528	0.201	0.000	0.509	0.999
key	30928	5.269	3.560	0.000	5.000	11.000
loudness	30928	-8.079	3.307	-39.053	-7.453	1.162
mode	30928	0.768	0.422	0.000	1.000	1.000
speechiness	30928	0.050	0.057	0.000	0.035	0.952
acousticness	30928	0.443	0.302	0.000	0.449	0.996
instrumentalness	30928	0.024	0.131	0.000	0.000	0.989
liveness	30928	0.167	0.137	0.009	0.116	0.985
valence	30928	0.388	0.212	0.000	0.335	0.974
tempo	30928	123.877	27.391	0.000	125.904	216.064

Note: 上表提供了音訊特徵的敘述統計，從數值中我們可以觀察出，歌曲知名度為左偏分配。另外我們可以從音訊特徵中總結出資料中的歌曲平均而言為調性為大調、用字較少、非現場演奏、非純樂器演奏、較悲情的歌曲。

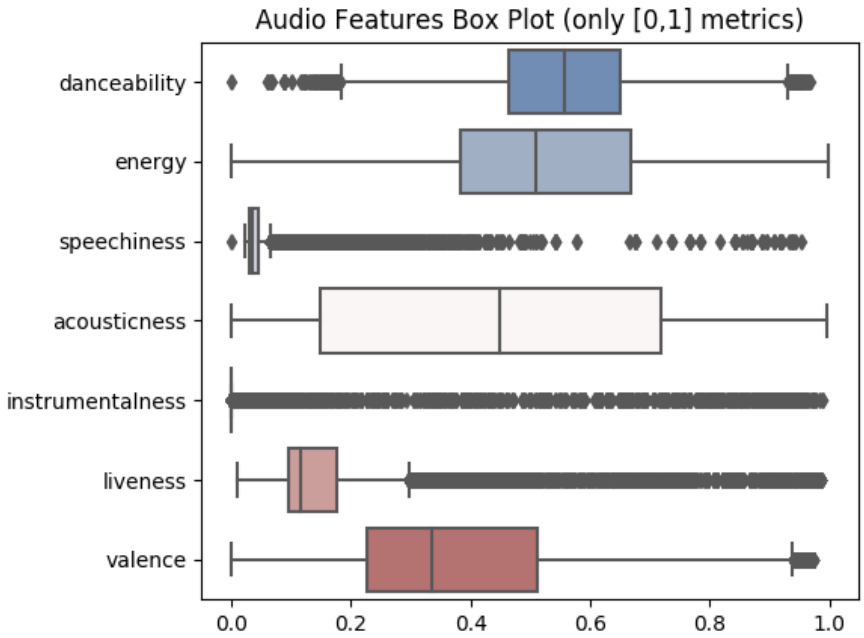


圖 *2.1.* 音訊特徵的箱型圖。從箱型圖我們可以比較容易觀察分布的全貌。比較值得注意的是，speechiness、instrumentalness、liveness 都集中在非常接近 0 的位置，顯示我們的資料可能大多數屬於用字較少、非純演奏、非現場演奏的歌曲。

doi:10.6342/NTU201802042

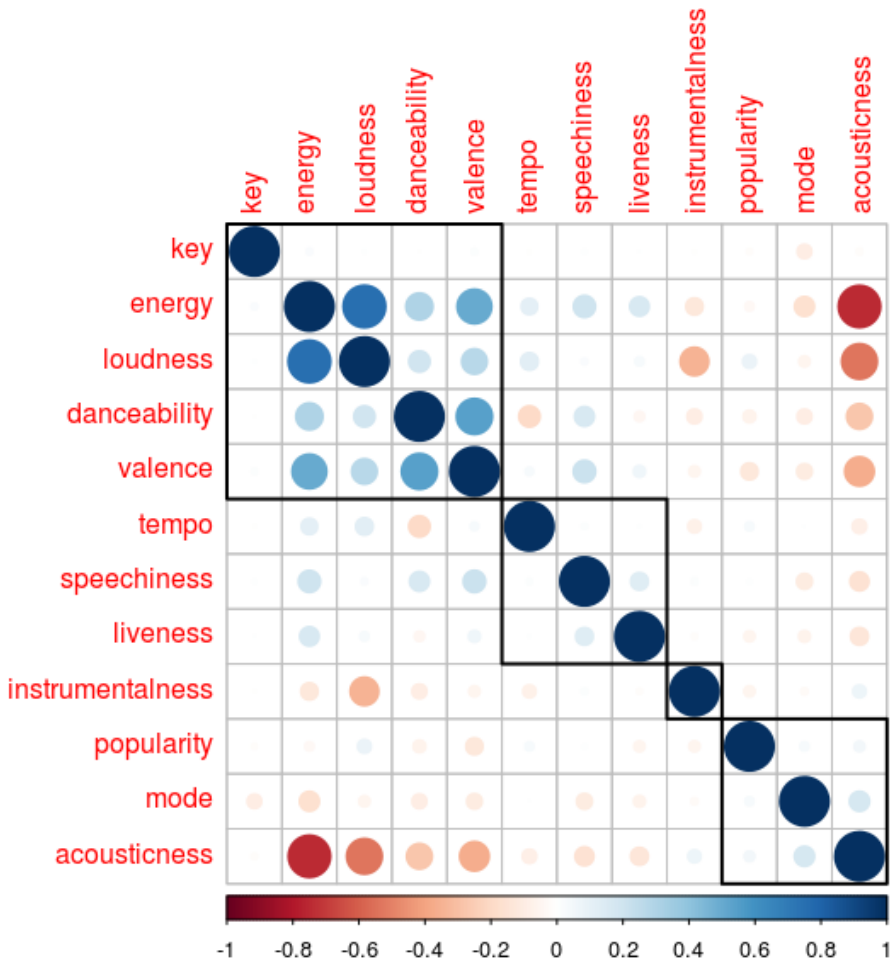


圖 *2.2.* 音訊特徵之相關係數矩陣。我們可以經由觀察變數之間的關係更進一步了解我們的資料，有些變數之間的關係相當符合直覺，如：acousticness 與 energy、loudness 的反向關係，valence 與 energy、danceability 的正向關係等。但 popularity 卻與其他音訊特徵變數沒有特別的關係。

doi:10.6342/NTU201802042

圖 *2.3.* 頻率最高 50 個字之文字雲。我們發現有些頻率過高的字詞但對於建立模型無助，因此我們會進一步將其過濾。

2.2 歌詞

歌詞主要由 KKBOX 的網頁爬取。在我們將資料用於訓練模型前。我們需要對歌詞做一些前處理。歌詞資料包含了許多雜訊，如：特殊符號、寫於歌曲前的作詞、作曲人等、歌詞中的說明文字，像是：間奏、前奏、來賓請掌聲鼓勵等，都會影響到我們的分析結果，我們會利用正規表示式 (Regular Expression) 將此類文字與符號做過濾。圖2.3為我們資料中出現頻率最高的 50 個字，這些字頻繁地出現每首歌裡面，會影響到我們訓練模型所用的資料品質，在我們的模型中，我們會選用每首歌裡面最具代表性的 20 個字來做為我們的訓練資料集，因此我們需要再進一步抽取出每首歌裡面的關鍵字。

為了達到上述目的，我們使用了詞頻-逆向文件頻率 (Term Frequency-Inverse Document Frequency，TF-IDF)，詞頻-逆向文件頻率為一種針對字詞頻率作加權的技術，我們在這邊針對詞頻-逆向文件頻率作簡要的說明。首先我們先定義每首歌 d 裡面每個字詞 t 的數量為 $f_{t,d}$，所以我們可以將詞頻 (TF) 定義為

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t'} f_{t',d}}, \forall t' \in d \tag{2.1}$$

doi:10.6342/NTU201802042

在我們定義了詞頻以後，我們再進一步定義反轉文件頻率 (IDF) 為

$$idf(t, D) = log\frac{N}{|\{d \in D : t \in d\}|} \tag{2.2}$$

其中 N 為我們的歌曲數目，$|\{d \in D : t \in d\}|$ 為每個字詞出現在全部歌曲中的數量。$idf(t, D)$ 在字詞數量極多的時候，會將該字詞的權重調低。我們再進一步組合 TF 與 IDF 為

$$tfidf(t, d, D) = tf(t, d) \times idf(t, D) \tag{2.3}$$

上述式子可以幫我們抓出每首歌曲裡面的代表字詞，直觀地來說，詞頻-逆向文件頻率可以將歌曲中較常出現，但不常出現在所有歌曲中的字詞濾出，我們可以利用計算詞頻-逆向文件頻率來幫助我們處理訓練模型的資料集。

2.3 歌曲熱門度

歌曲熱門度為使用 Spotify 開發者工具獲得，獲取的歌曲熱門度為 2017 年 12 月 20 日當下之 popularity，因此並未考量到跨世代的熱門歌曲。舉例來說，在獲取資料的當下，周興哲的歌曲可能被大眾廣為聆聽，但並不代表張學友的歌曲並不熱門，只是在這個世代較少人聆聽張學友的歌曲。

doi:10.6342/NTU201802042

章節 3

演算法

在我們研究中利用的模型，會使用近年廣為被利用在各個研究與實務作業的增強方法 (Ensemble Methods) 與類神經網路 (Neural Networks)。Zhou (2012) 在書中提到了增強方法可以有效地提高弱學習器 (Weak Learner) 的表現，其中比較常見的弱學習器有決策樹 (Decision Tree)、樸素貝氏分類器 (Naïve Bayes) 等。增強方法可以想成是一種將許多弱學習器做組合來達到更準確、更穩健預測結果的方法。在接下來的小節中，我們會簡潔地介紹我們研究中運用到的三種算法：梯度增強機器 (Gradient Boosting Machine，GBM)、長短期記憶遞迴式類神經網路 (Long Sort-Term Memory Recurrent Neural Networks，LSTM)、隨機森林 (Random Forests)。

3.1 梯度增強機器

在監督式學習 (Supervised Learning) 的問題中，我們有資料集 D 與解釋變數、被解釋變數的組合 $(\mathbf{x}, y) \in D$，我們的目標是在函數集合 F 中尋找一個估計式 $\widehat{f}(\mathbf{x})$ 使得 $\widehat{f}(\mathbf{x}) = f^*(\mathbf{x})$，方法為極小化預期損失 $\mathbb{E}_{y,\mathbf{x}}[L(y, f(\mathbf{x}))]$，我們將上述過程寫為

$$f^*(\mathbf{x}) = \arg\min_f \mathbb{E}_{y,\mathbf{x}}[L(y, f(\mathbf{x}))] \tag{3.1}$$

doi:10.6342/NTU201802042

其中 $L(y, f(\mathbf{x}))$ 為損失函數。根據期望值定理 (The Law of Expectation)，我們可以將式3.1寫為

$$f^*(\mathbf{x}) = \arg\min_f \mathbb{E}_{\mathbf{x}}\Big[\mathbb{E}_{y|\mathbf{x}}[L(y, f(\mathbf{x}))|\mathbf{x}]\Big] \tag{3.2}$$

在尋找最佳估計式時，通常我們會在限定函數形式的條件下做參數的調整，因此我們可以將式3.1改寫為

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} \Phi(\mathbf{P}) \tag{3.3}$$

其中

$$\Phi(\mathbf{P}) = \mathbb{E}_{y,\mathbf{x}} L[y, f(\mathbf{x}; \mathbf{P})]$$

因此式3.1會變為

$$f^*(\mathbf{x}) = f(\mathbf{x}; \mathbf{P}^*)$$

在這邊需要注意的是，我們限定了 $F(\mathbf{x}; \mathbf{P})$ 為可加的形式，換句話說

$$\mathbf{P}^* = \sum_{i=0}^{N} \mathbf{p}_i \tag{3.4}$$

其中 $\mathbf{p}_0$ 為亂數初始化的參數，$\{\mathbf{p}_i\}_1^N$ 為基於前面式子所產生的連續增量 (Successive Increments)，在最佳化的過程中我們使用了梯度下降法 (Gradient Descent)。

接下來我們在更進階地說明上述提到的增量 (Increments)。在優化的過程中，我們會計算基於目前函數狀態下的梯度 (Gradient)

$$\mathbf{g}_i = \{g_{ji}\} = \left\{\left[\frac{\partial \Phi(\mathbf{P})}{\partial P_j}\right]_{\mathbf{P}=\mathbf{P}_{i-1}}\right\}$$

其中

doi:10.6342/NTU201802042

$$\mathbf{P}_{i-1} = \sum_{k=0}^{i-1} \mathbf{p}_k$$

代表 $\mathbf{P}_{i-1}$ 為從之前的狀態操作梯度下降法以後所得結果遞迴加總而來。而新的增量可以表示為

$$\mathbf{p}_i = -\rho_i \mathbf{g}_i$$

其中

$$\rho_i = \arg\min_{\rho} \Phi(\mathbf{P}_{i-1} - \rho \mathbf{g}_i) \tag{3.5}$$

代表在決定梯度以後，我們還需要由演算法決定梯度下降的權重。將式3.3與式3.5組合後可得到

$$f_i \leftarrow f_{i-1} + \rho_i h(\mathbf{x}; \mathbf{P}_i) \tag{3.6}$$

式3.6即為加入新的弱學習器後的更新規則，且須滿足

$$(\rho_i, \mathbf{P}_i) = \arg\min_{\rho, \mathbf{P}} \sum_{j=1}^{M} [-g_i(x_j) + \rho h(x_j; \mathbf{P})]^2$$

其中 M 為資料集的大小、h 為弱學習器。針對梯度增強機器更詳盡的介紹可以在Friedman (2001) 找到。

3.2 長短期記憶遞迴式類神經網路

我們在這一節會簡短地介紹長短期記憶遞迴式類神經網路。長短期記憶遞迴式類神經網路為遞迴式類神經網路 (Recurrent Neural Networks，RNN) 的改寫版本，不同的是，長短期記憶遞迴式類神經網路在記憶單元 (Memory Cell) 上做了特殊的結構設計。遞迴式類神經網路在訓練模型時，所接受的資料為序列 (Sequence) 而非單點，假設 t 為序列中不同的狀態，$\mathbf{x}^{(t)}$ 為不同狀態下的解釋變數，$\mathbf{h}^{(t)}$ 為不同狀態下的隱藏層 (Hidden Layers)，$\boldsymbol{\theta}$ 為模型所儲存的參數。在訓練時，遞迴式類

doi:10.6342/NTU201802042

神經網路會將上個狀態的隱藏層輸入到目前狀態的隱藏層做訓練

$$\mathbf{h}^{(t)} = f(\mathbf{h}^{(t-1)}, \mathbf{x}^{(t)}; \boldsymbol{\theta}) \tag{3.7}$$

因此遞迴式類神經網路是一個有記憶的模型。長短期記憶遞迴式類神經網路更進一步地針對記憶的機制做了修改，設計了三個門閥來控制在狀態 t 時資料的輸入、輸出與記憶，並且利用訓練的方式來學習門閥值的大小，分別為輸入門閥 (Input Gate，$\mathbf{i}^{(t)}$)、輸出門閥 (Output Gate，$\mathbf{o}^{(t)}$) 與遺忘門閥 (Forget Gate，$\mathbf{f}^{(t)}$)，

$$\mathbf{i}^{(t)} = \sigma(\mathbf{W}_{ix}\mathbf{x}^{(t)} + \mathbf{W}_{ih}\mathbf{h}^{(t-1)} + \mathbf{W}_{ic}\mathbf{c}^{(t-1)}) \tag{3.8}$$

$$\mathbf{f}^{(t)} = \sigma(\mathbf{W}_{fx}\mathbf{x}^{(t)} + \mathbf{W}_{fh}\mathbf{h}^{(t-1)} + \mathbf{W}_{fc}\mathbf{c}^{(t-1)}) \tag{3.9}$$

$$\mathbf{c}^{(t)} = \mathbf{c}^{(t-1)} \odot \mathbf{f}^{(t)} + \mathbf{i}^{(t)} \odot \phi(\mathbf{W}_{cx}\mathbf{x}^{(t)} + \mathbf{W}_{ch}\mathbf{h}^{(t-1)}) \tag{3.10}$$

$$\mathbf{o}^{(t)} = \sigma(\mathbf{W}_{ox}\mathbf{x}^{(t)} + \mathbf{W}_{oh}\mathbf{h}^{(t-1)} + \mathbf{W}_{oc}\mathbf{c}^{(t)}) \tag{3.11}$$

$$\mathbf{h}^{(t)} = \mathbf{o}^{(t)} \odot \phi(\mathbf{c}^{(t)}) \tag{3.12}$$

在上述式子中，$\mathbf{W}$ 代表了門閥中個別變數所儲存的權重、$\mathbf{c}^{(t)}$ 為長短期記憶遞迴式類神經網路內的記憶單元 (Memory Cell)、σ 為邏輯函數 (Sigmoid Function)、ϕ 為雙曲正切函數 (Hyperbolic Tangent Function) 在模型中擔任激勵函數 (Activation Function) 的角色。

可以看到在輸入門閥、輸出門閥、遺忘門閥與記憶單元中訓練時都會將上一個狀態的隱藏層與目前狀態的解釋變數輸入模型中，這是與遞迴式類神經網路不同的地方，現今大多數的遞迴式類神經網路也都以使用長短期記憶遞迴式類神經網路為主，因為其相對於遞迴式類神經網路有較好的表現。有關遞迴式類神經網路的介紹詳細可以參考Goodfellow, Bengio, Courville, and Bengio (2016)。

3.3 隨機森林

隨機森林為一個代表性的引導聚集算法 (Bootstraping Aggregating，Bagging)，用一句話來總結引導聚集算法的話，就是使用很多的弱學習器來作表決，但他們

doi:10.6342/NTU201802042

各自都只看到一部份的資料。而隨機森林更進一步地在解釋變數上做了拔靴法(Bootstraping)。以下我們簡短地介紹引導聚集算法的概念。給定弱學習器的數量為 M、弱學習器為 h，我們同時訓練了所有的弱學習器

$$h_t(\mathbf{x}) = \arg\min_h \mathbb{E}_{y_{bs},\mathbf{x}_{bs}}[L(y_{bs}, h(\mathbf{x}_{bs}))], \forall t = 1, \ldots, M \tag{3.13}$$

其中 $(\mathbf{x}_{bs}, y_{bs}) \in D_{bs}$，$D_{bs}$ 為對 D 操作拔靴法後的資料集。在訓練完所有的弱學習器後，我們利用所有的弱學習器進行多數決投票 (Majority Voting)，

$$f^*(\mathbf{x}) = \arg\max_{y\in\mathcal{Y}} \sum_{t=1}^{M} \mathbb{I}(h_t(\mathbf{x}) = y) \tag{3.14}$$

其中 $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\}$，$N$ 是類別的數量。

隨機森林即將引導聚集算法加以修改成在解釋變數上也操作抽樣法即可。有關隨機森林的詳細算法可以參閱Breiman (2001)。

doi:10.6342/NTU201802042

章節 4

實驗設計

我們參考了Herremans et al. (2014) 的設計，將我們的被解釋變數轉換成一個二元的分類問題，換句話說，我們的 popularity 原本為一個實數域上大於 0 的連續變數，我們取其 90 百分位以上的資料，將其標註成熱門歌曲，而低於 30 百分位的資料，我們將其標註成不熱門的歌曲，而位於 30 百分位至 90 百分位的資料我們將不採用。

在我們將被解釋變數轉換後，我們會分別利用音訊特徵資料訓練梯度增強機器、歌詞訓練長短期記憶遞迴式類神經網路和上述兩個模型的預測結果訓練隨機森林，並利用混淆矩陣 (Confusion Matrix)、接收者操作特徵曲線 (Receiver Operating Characteristic Curve，ROC Curve) 與精確度-召回率曲線 (Precision Recall Curve，PR Curve) 來驗證模型的結果。

接下來的小節中我們會詳述上述每個模型的訓練方式與過程。

4.1 梯度增強機器

在梯度增強機器模型中，我們單純利用了音訊特徵來訓練模型，訓練的過程中，我們首先使用了 80% 與 20% 的比率做了訓練/測試集資料的切分，再來我們利用訓練集資料訓練模型，過程中利用網格搜尋法 (Grid Search) 來做參數最佳化，在最佳化的過程中，為了避免過度擬合 (Overfitting)，我們使用了 10 摺分層抽樣的交叉驗證。在訓練完模型後，我們會分別利用混淆矩陣、接收者操作特徵曲線與精確度-召回率曲線搭配測試資料集來驗證我們的結果。圖4.1畫出了我們訓練

doi:10.6342/NTU201802042

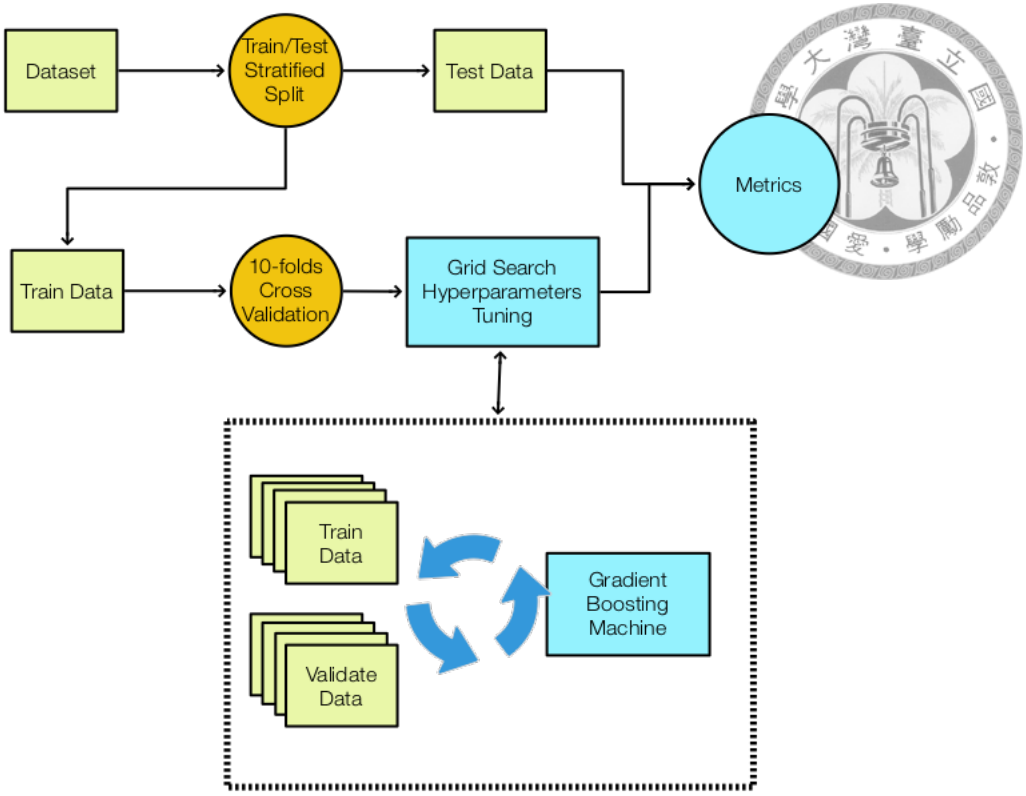


圖 *4.1.* 梯度增強機器訓練流程。我們首先進行了訓練/測試集切分，再利用網格搜尋法進行超參數調整，並搭配了 10 摺分層抽樣的交叉驗證避免過度擬合。訓練完畢以後再以混淆矩陣、接收者操作特徵曲線與精確度-召回率曲線搭配測試資料集進行模型的驗證。

模型的流程。

4.2 長短期記憶遞迴式類神經網路

在長短期記憶遞迴式類神經網路中，我們單純利用了歌詞資料來訓練模型。在訓練模型前，我們需要將每首歌曲中的關鍵字轉換為數值資料，在這邊我們利用了 FastText(Bojanowski, Grave, Joulin, & Mikolov, 2017)，來將我們先前提取每首歌曲的關鍵字轉換為向量，最後再將向量做為我們的訓練資料輸入長短期記憶遞迴式類神經網路。

訓練的過程與梯度增強機器大致相似，我們會先使用 80% 與 20% 的比率做

doi:10.6342/NTU201802042

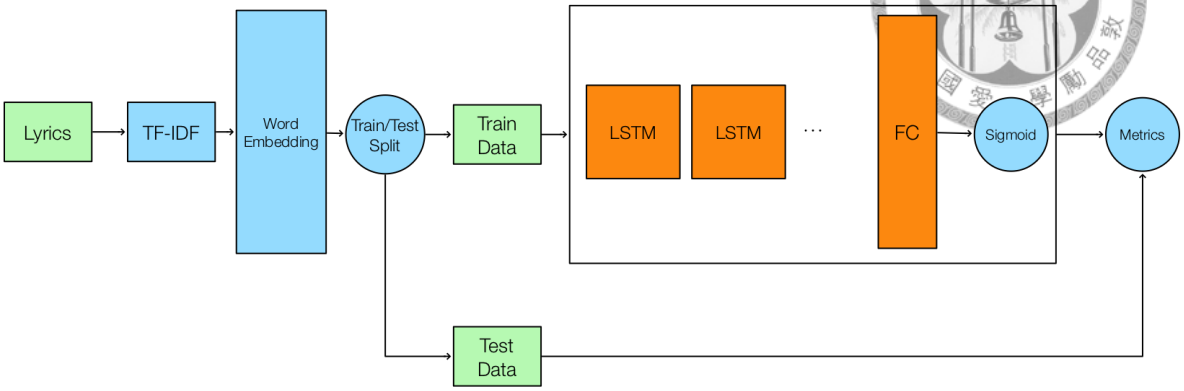


圖 *4.2.* 長短期記憶遞迴式類神經網路訓練過程。我們首先將每首歌關鍵字進行文字轉向量，再進行訓練/測試資料集切分。切分完畢後，再利用訓練資料集訓練模型，利用測試資料集搭配混淆矩陣、接收者操作特徵曲線與精確度-召回率曲線進行模型驗證。

訓練/測試資料集的切分，並利用訓練資料集訓練資料。訓練過程中，在每一期 (Epoch) 我們都操作了訓練/驗證資料集切分與驗證，並在隱藏層中加入了丟棄層 (Dropout) 來防止過度擬合。除此之外，訓練遞迴式類神經網路會有梯度消失的問題 (Vanishing Gradient)，我們還針對了梯度做了梯度剪裁 (Gradient Clipping)。值得一提的是，抽取完關鍵字後，每首歌的歌詞順序變得沒有意義，因此在訓練模型時，我們採用了雙向 (Bidirectional) 的訓練方法，意即使模型在每個序列跑過前往後、後往前兩個方向。訓練完畢後，我們同樣利用混淆矩陣、接收者操作特徵曲線與精確度-召回率曲線搭配測試資料集進行模型的驗證。

4.3 隨機森林

在訓練完前兩小節的模型後，我們可以以上述兩個模型分別利用歌詞與音訊資料產出預測結果，並將結果作為隨機森林的解釋變數再次進行訓練。訓練的過程

doi:10.6342/NTU201802042

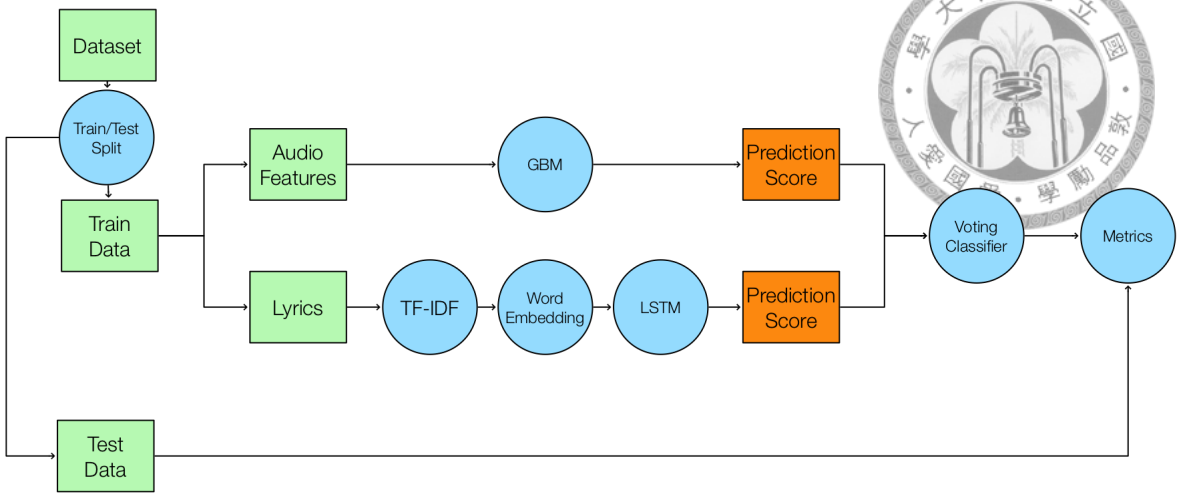


圖 *4.3.* 隨機森林模型結構。我們首先進行了訓練測試集資料切分，再分別將歌詞與音訊資料代入長短期記憶遞迴式類神經網路與梯度增強機器產生預測結果，再以預測結果作為解釋變數訓練模型。訓練完畢後我們將測試資料集做了相同轉換，並利用混淆矩陣、接收者操作特徵曲線與精確度-召回率曲線搭配測試資料集進行模型的驗證。

與上述兩模型雷同，

我們利用 80% 與 20% 的比率進行訓練/測試資料集切分。切分完畢後，分別利用訓練資料集內的歌詞與音訊資料放入長短期記憶遞迴式類神經網路與梯度增強機器產生預測結果，並利用結果作為解釋變數訓練隨機森林，訓練的過程我們同樣採用網格搜尋法搭配 10 摺交叉驗證。在訓練完畢後，我們將測試資料集同樣分別利用長短期記憶遞迴式類神經網路與梯度增強機器進行轉換，再利用轉換後的資料搭配搭配混淆矩陣、接收者操作特徵曲線與精確度-召回率曲線進行模型驗證。附帶一題，我們利用隨機森林針對前兩種模型進行進一步訓練的動作，可以視為利用兩個模型進行表決，因此後面章節我們會將隨機森林稱為表決分類器 (Voting Classifier)。

doi:10.6342/NTU201802042

章節 5

實驗結果

在此章我們會將章節4的實驗結果一一作說明，分別以三個不同模型的三種模型驗證方法之結果做比較，並會利用Herremans et al. (2014) 與Dhanaraj and Logan (2005) 兩篇性質較類似的論文做比較。

5.1 混淆矩陣之實驗結果比較

混淆矩陣是一個常用來衡量分類器的指標，使用的概念很簡單，就是直接拿模型預測的結果與真實的結果作比較並統計數量。根據比較的結果我們可以把所有情況分為四種

- 模型預測為熱門歌曲，真實資料為熱門歌曲，代表模型正確地判斷歌曲為熱門歌曲，我們稱為真陽 (True Positive，TP)
- 模型預測為熱門歌曲，真實資料卻為不熱門歌曲，代表模型錯誤地判斷歌曲為熱門歌曲，我們稱為偽陽 (False Positive，TN)
- 模型預測為不熱門歌曲，真實資料卻為熱門歌曲，代表模型沒有正確的判斷出歌曲為熱門歌曲，我們稱為偽陰 (False Negative，FP)
- 模型預測為不熱門歌曲，真實資料為不熱門歌曲，代表模型正確的判斷歌曲為不熱門歌曲，我們稱為真陰 (True Negative，TN)

doi:10.6342/NTU201802042

在混淆矩陣中，我們希望真陽與真陰的比率越高越好，換句話說，我們希望分類器準確猜中真實資料的情況越多越好。圖5.1畫出了我們在先前小節所訓練的三種模型，利用測試資料集搭配混淆矩陣做驗證後的結果。

在圖5.1中我們可以看出梯度增強機器的偽陰數量的比率高於長短期記憶遞迴式類神經網路，但梯度增強機器的偽陽數量的比率卻低於長短期記憶遞迴式類神經網路。在將兩個模型做組合後，我們發現偽陽與偽陰的數量比率相較於先前兩模型得到較平衡的結果。

我們的模型在進行預測時，雖說產出的預測結果為二元的標籤，但實際上模型產出的預測結果為預測一首歌是否為熱門歌曲的機率，我們是在固定了某一個特定的門檻值後，觀察特定門檻值之下的混淆矩陣。在使用混淆矩陣衡量時，我們使用的門檻值為 0.5，意即在模型預測機率大於 0.5 時，我們將其歸類為熱門歌曲，反之，模型預測機率小於 0.5 時，我們將其歸類為不熱門歌曲。但此並沒有辦法全面性地衡量分類器整體的表現，因此我們在接下來的兩個小節會利用接收者操作特徵曲線、精確度-召回率曲線來評估我們的三個模型成效。

5.2 接收者操作特徵曲線之實驗結果比較

接收者操作特徵曲線也是一個常用於衡量分類器表現的指標，相較於混淆矩陣，接收者操作特徵曲線不會只觀察特定一門檻值之下的模型表現，而是衡量從 0 至 1 的門檻值之下分類器的表現，並將其結果映射到接收者操作特徵空間 (ROC Space) 呈現。

接收者操作特徵空間為一個二度空間平面圖，其橫軸為偽陽率 (False Positive Rate，FPR)，定義如下

$$FPR = \frac{FP}{FP + TN} \tag{5.1}$$

而其縱軸為真陽率，其定義如下

$$TPR = \frac{TP}{TP + FN} \tag{5.2}$$

接收者操作特徵曲線即為計算不同門檻值之下的真陽率與假陽率，並將其點在

doi:10.6342/NTU201802042

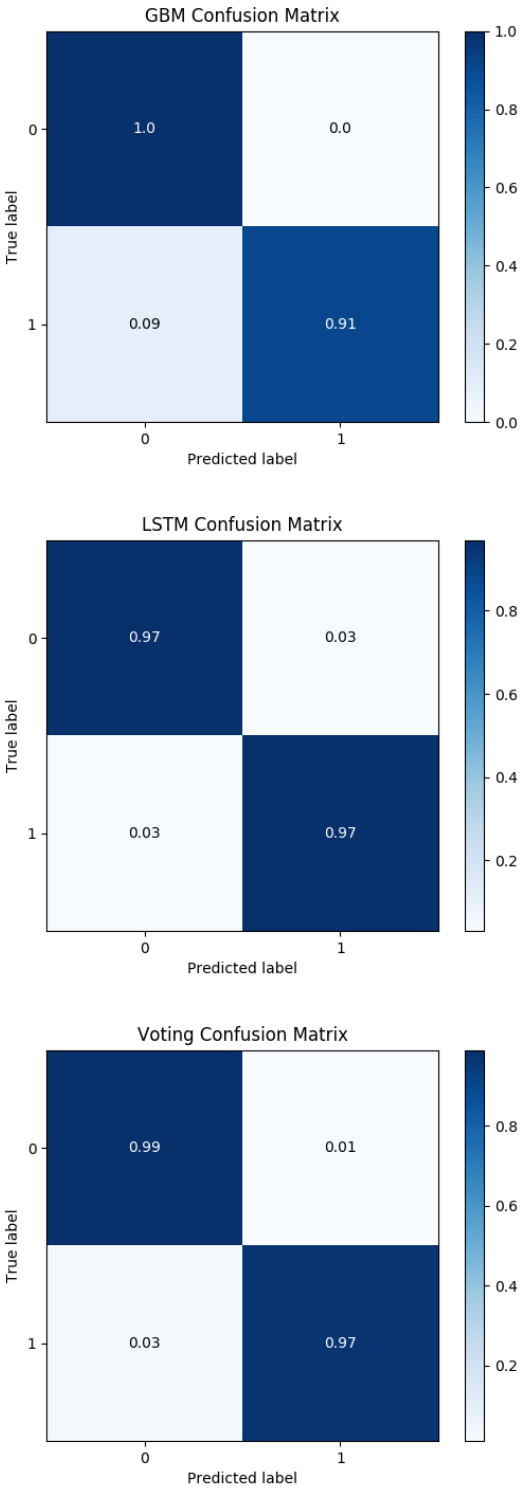


圖 *5.1.* 混淆矩陣比較圖。從圖中我們可以看出梯度增強機器的偽陰數量的比率高於長短期記憶遞迴式類神經網路，但梯度增強機器的偽陽數量的比率卻低於長短期記憶遞迴式類神經網路。在將兩個模型做組合後，我們發現偽陰與偽陽的數量比率相較於先前兩模型得到較平衡的結果。

doi:10.6342/NTU201802042

接收者操作特徵空間之上且連接起來。在接收者操作特徵空間中，接收者操作特徵曲線越接近左上角表示分類器的表現越好。

我們利用圖5.2來說明接收者操作特徵曲線的獲取過程，在使用分類器進行完預測後，利用真實資料將其結果分為陽性與陰性兩個分布，透過移動從 0 至 1 的門檻值計算不同門檻值底下的真陽率與偽陽率，並將其畫上接收者操作特徵空間，即可得到接收者操作特徵曲線。

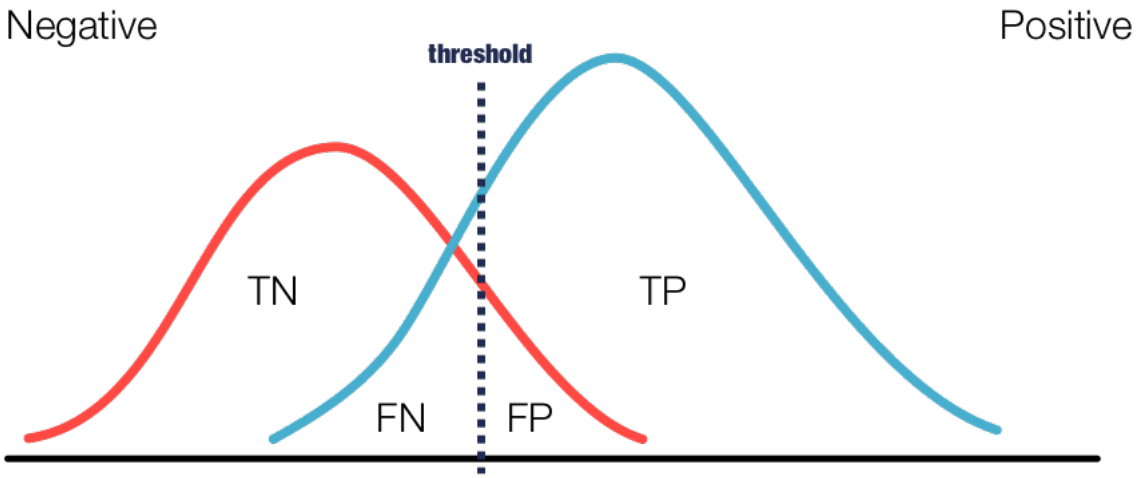
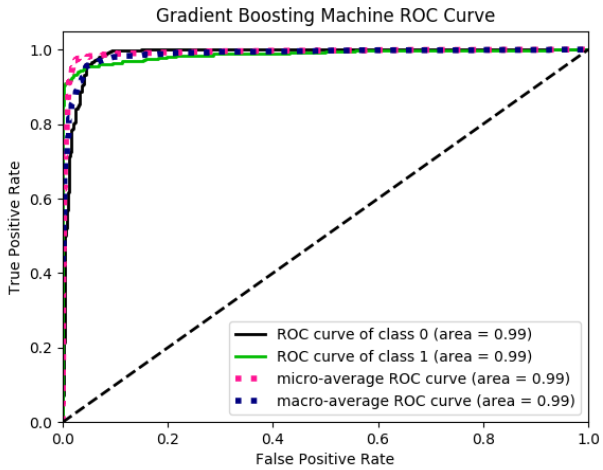


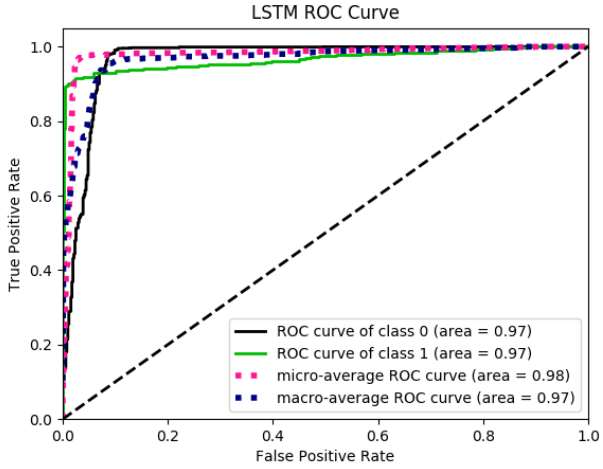
圖 *5.2.* 接收者操作特徵產生機制說明。在使用分類器進行完預測後，利用真實資料將其結果分為陽性與陰性兩個分布，透過移動從 0 至 1 的門檻值計算不同門檻值底下的真陽率與偽陽率，並將其畫上接收者操作特徵空間，即可得到接收者操作特徵曲線。

雖說我們知道接收者操作特徵曲線越接近右上角越好，但利用肉眼觀察並非一個合適的方法，因此我們進一步計算曲線底下面積 (Area Under Curve，AUC)，並透過比較曲線底下面積來衡量模型的表現。

圖5.3畫出了三個模型的接收者操作特徵曲線，我們以肉眼觀察三個模型的接收者操作特徵曲線可以看到長短期記憶遞迴式類神經網路的曲線較其他兩者略為偏離左上角。顯示長短期記憶遞迴式類神經網路較其他兩者表現較差。我們進一步計算三者的曲線底下面積，並列於表5.1。由表中看出梯度增強機器的曲線底下面積高於表決分類器，造成此種結果的方法可能是由於長短期記憶遞迴式類神經網路拖累了表決分類器的表現，也可能是因為我們的資料中兩個種類 (熱門歌曲與不熱門歌曲) 之比率不平衡所造成He and Garcia (2009)。直觀上來說，考慮了歌詞

doi:10.6342/NTU201802042





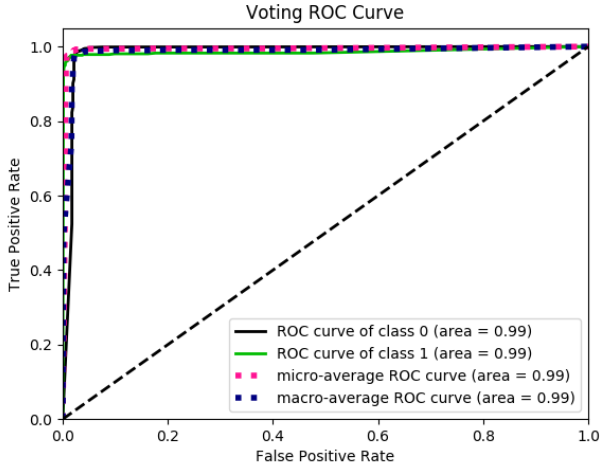


圖 *5.3.* 接收者操作特徵曲線比較圖。我們以肉眼觀察三個模型的接收者操作特徵曲線可以看到長短期記憶遞迴式類神經網路的曲線較其他兩者略為偏離左上角。顯示長短期記憶遞迴式類神經網路較其他兩者表現較差。而在梯度增強機器與表決分類器的比較上，雖說肉眼觀察發現表決分類器的表現優於梯度增強機器，但在計算曲線底下面積後，我們發現梯度增強機器的面積值略微大於表決分類器。

doi:10.6342/NTU201802042

與音訊特徵的模型表現應該會是三者中最好的，但結果卻違反了我們的直覺判斷，因此我們在下一小節會利用精確度-召回率曲線重新衡量三個模型的表現。

表 5.1

模型的曲線底下面積 *(*接收者操作特徵曲線*)*

Model	AUC(ROC)
LSTM	0.966
GBM	**0.987**
Voting	0.986

Note: 此表列出三種模型的曲線底下面積 (接收者操作特徵曲線)，直觀來說，使用了歌詞與音訊特徵的表決分類器之表現應為三者中最好，但在表中卻看到梯度增強機器的表現為三者之中最好，我們推測有可能是由於長短期記憶遞迴式類神經網路導致表決分類器的表現下降，也有可能是由於類別的數量不平衡所造成。

5.3 精確度-召回率曲線之實驗結果比較

在這一小節，我們進一步使用精確度-召回率曲線來衡量三個模型的表現，首先我們先分別定義精確度 (Precision) 與召回率 (Recall)，精確度的定義如下，

$$Precision = \frac{TP}{TP + FP} \tag{5.3}$$

其意義為在給定一個分類器預測一首歌為熱門歌曲的情況下，有多少數量是被分類器正確的預測的。而召回率的定義如下，

$$Recall = \frac{TP}{FN + TP} \tag{5.4}$$

其意義為在給定真實資料為熱門歌曲時，有多少數量是被分類器正確預測的。

我們在繪製精確度-召回率曲線時，繪製方法大致與接收者操作特徵曲線相同，以移動門檻值得方式分別計算不同門檻值之下所算出的精確度與召回率，精確度-召回率曲線在越接近圖中右上角時，模型表現越好。

He and Garcia (2009) 提到，在資料的分布偏斜時，換句話說，我們的類別數量極為不平衡時，運用接收者操作特徵曲線會高估分類器的表現，而精確度-召回率曲線相較於接收者操作特徵曲線，將焦點更放在數量較少的類別。如果我們對於數量少的類別更感興趣時，使用精確度與召回率來評估模型的表現是一個比較合

doi:10.6342/NTU201802042

適的方法。在我們的研究中，我們對於是否能夠判斷一首歌曲為熱門歌曲更為感興趣，加上我們研究所使用之資料的類別數量也不平衡，因此我們會進一步利用精確度-召回率曲線來確認上一小節的分析結果。

圖5.4畫出了三個模型的精確度-召回率曲線，由圖中我們可以看出長短期記憶遞迴式類神經網路的表現如同運用接收者操作特徵曲線衡量時一樣表現最差，而在比較梯度增強機器與表決分類器時，結果卻與使用接收者操作特徵曲線衡量時相反，表決分類器的表現優於梯度增強機器的表現。造成此結果的原因我們猜測大致是因為在使用接收者操作特徵曲線衡量時，由於熱門歌曲的數量較少，造成不同類別的數量比率不平衡而導致，因此在使用精確度-召回率曲線衡量時，梯度增強機器的表現劣於使用接收者操作特徵曲線衡量時的表現，顯示了使用接收者操作特徵曲線衡量模型時，的確有可能高估了模型的表現。精確度-召回率曲線同樣可以計算曲線底下面積，用以更精確的衡量模型的表現，我們將其結果列於表5.2。

表 5.2

模型的曲線底下面積 (精確度-召回率曲線)

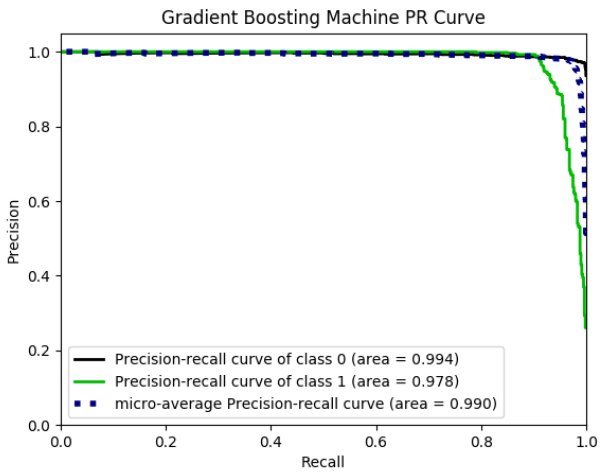
Model	AUC(PRC)
LSTM	0.951
GBM	0.978
Voting	**0.981**

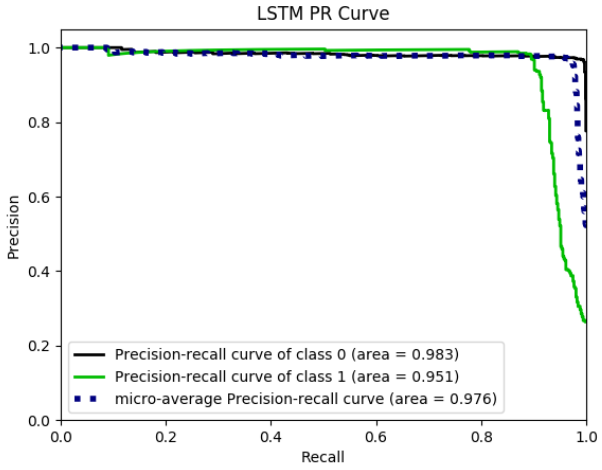
Note: 表中列出了三種模型的曲線底下面積 (精確度-召回率曲線)，在更換衡量指標後，表決分類器的表現變為三者之中最佳的。

5.4 不同研究之間的模型表現比較

我們在這一小節會比較與我們的研究性質相關的兩篇研究，Herremans et al. (2014) 與Dhanaraj and Logan (2005)，我們比較的基準會運用接收者操作特徵曲線，原因在於上述兩篇論文所提供的衡量指標為接收者操作特徵曲線，並未提供精確度-召回率曲線。表5.3列出了三篇論文的模型表現與使用的變數，經由比較後發現我們的模型表現高出其他兩篇論文許多，原因可能在於使用的被解釋變數不同所造成。會做出此結論原因在於，Herremans et al. (2014) 與我們使用了大致相同的解釋變數，但在模型的曲線底下面積上卻相差許多。不幸的是，由於我們的被

doi:10.6342/NTU201802042





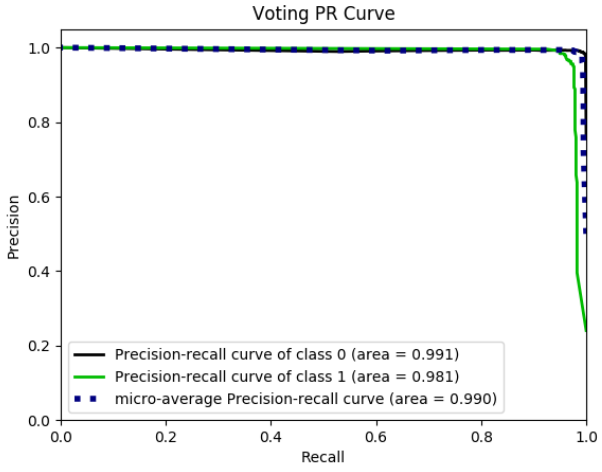


圖 *5.4.* 精確度-召回率曲線比較圖。長短期記憶遞迴式類神經網路的表現在三者之中表現最差，而以肉眼觀察後，表決分類器的表現變為三者之中表現最好的模型。

doi:10.6342/NTU201802042

解釋變數是經由 Spotify 的開發者工具獲得，我們並不知道 Spotify 如何創造出此衡量指標。

表 5.3

不同論文的模型比較

Model	Features	AUC (ROC)
Dhanaraj and Logan (2005)	Lyrics & Audio	0.69
Herremans et al. (2014)	Audio	0.65
Our paper	Lyrics & Audio	**0.986**

Note: 我們使用了Herremans et al. (2014) 與Dhanaraj and Logan (2005) 與我們的研究作比較，我們的模型表現高於其他兩者許多，其主要原因可能為被解釋變數的差異所造成。另外Herremans et al. (2014) 與我們使用了大致相同的解釋變數，但在模型的曲線底下面積上卻相差許多。

doi:10.6342/NTU201802042

章節 6

結論

在最後的章節，我們會針對我們的研究內容作出簡要的結論。首先我們探討了與我們的研究之相關文獻，並說明了研究方法的選擇。再來我們針對了我們的資料集做了簡單的探索性分析，發現單一音訊變數與歌曲熱門度並無明顯的關係，除此之外我們還針對了歌詞做了前處理。做完資料的探索與處理後，我們針對了研究中所運用的三種模型做了簡要介紹並說明了實驗的設計，最後我們分別以混淆矩陣、接收者操作特徵曲線與精確度-召回率曲線衡量三種模型的表現，我們認為以精確度-召回率曲線作為衡量指標為較合適的選擇。若以長短期記憶遞迴式類神經網路作為基準模型，結果顯示梯度增強機器在精確度-召回率的曲線底下面積高於基準模型 2%，表決分類器則高於基準模型 3%。

基於研究過程所產生的分析與結果，我們可以作出以下幾點結論，

- 我們以長短期記憶遞迴式類神經網路為基準模型，若以精確度-召回率曲線作為衡量指標，梯度增強機器之曲線底下面積高於基準模型 2%，表決分類器之曲線底下面積高於基準模型 3%。
- 梯度增強機器可以計算變數重要性 (Feature Importances)，經由計算我們發現前五重要的音訊特徵為 loudness、tempo、speechiness、valence 與 acousticness，圖6.1列出了所有音訊特徵計算出的變數重要性。
- 相對於Herremans et al. (2014) 與Dhanaraj and Logan (2005) 兩篇論文，我們的接收者操作特徵曲線之曲線底下面積高於其約 0.3 左右，主要原因可能為被解釋變數所造成，若要與之比較，應進一步搜集相同的被解釋變數。

doi:10.6342/NTU201802042

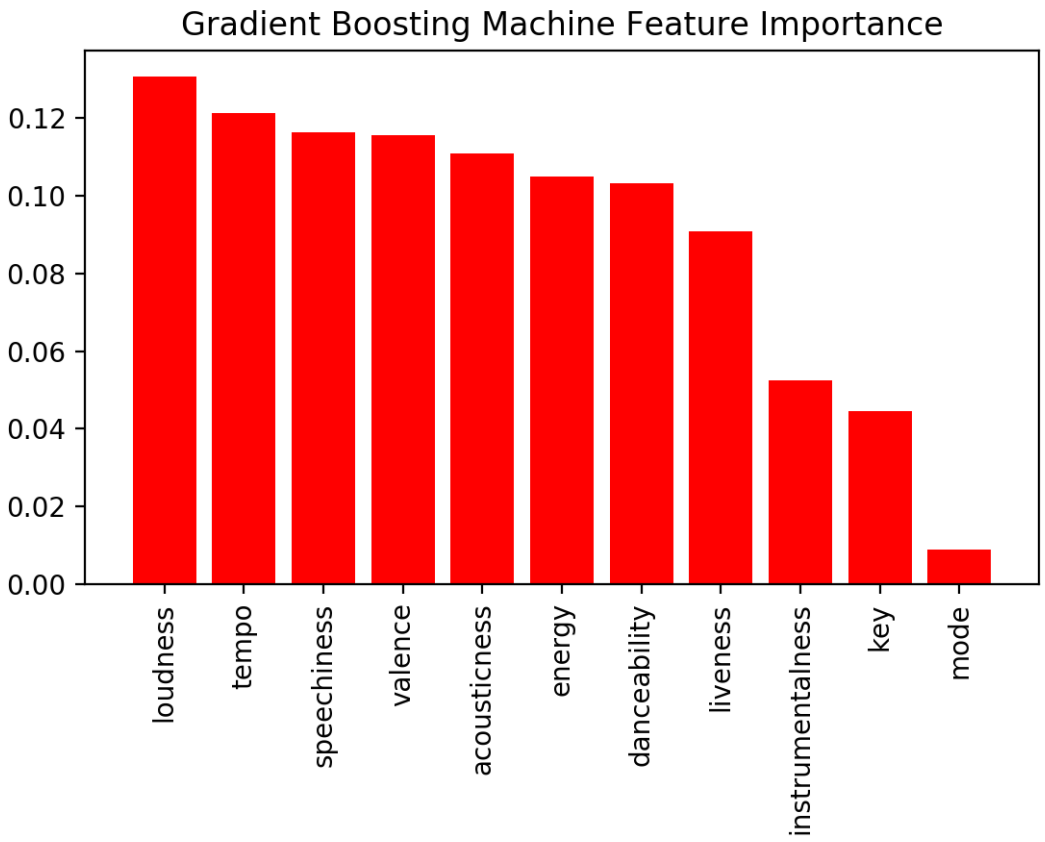


圖 *6.1.* 基於梯度增強機器計算之音訊特徵重要性。我們發現前五重要的特徵分別為 loudness、tempo、speechiness、valence 與 acousticness。

另外我們的研究結果是基於以下的限制與假設而完成，

- Spotify 並未提供 popularity 的創造方式，換句話說，我們並不知道 popularity 實際上是如何衡量。因此若將被解釋變數更換為與Herremans et al. (2014) 等論文較接近的被解釋變數，可能會比較貼近真實數字。
- 我們經由主觀的方式決定將資料切割於百分位 30% 與 90%，假如更換百分位數可能會影響到模型的結果。
- 我們的模型只能簡單地判斷一首歌是否為熱門歌曲，並沒有辦法給出精確的數字，如：排名、點擊率等。
- 使用的資料時間長度過長，可能會使得不同世代間的熱門歌曲無法被客觀的衡量。

doi:10.6342/NTU201802042

未來的研究方向建議可以將資料集橫跨的時間區間縮短，以減少不同世代歌曲衡量不客觀的現象，或是能夠取得不同時間點的歌曲熱門程度。然後將我們的被解釋變數換成與Herremans et al. (2014) 等研究較相符的變數，如：Billboard 排行榜等。獲得的結論可能較為客觀且能夠被驗證。

doi:10.6342/NTU201802042

參考文獻

Bischoff, K., Firan, C. S., Georgescu, M., Nejdl, W., & Paiu, R. (2009). Social knowledge-driven music hit prediction. In *International conference on advanced data mining and applications* (pp. 43–54).

Blume, J. (1999). *Six steps to songwriting success: The comprehensive guide to writing and marketing hit songs*. Billboard.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, *5*, 135–146.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32.

Dhanaraj, R., & Logan, B. (2005). Automatic prediction of hit songs. In *Ismir* (pp. 488–491).

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189–1232.

Garg, R., Smith, M. D., & Telang, R. (2011). Measuring information diffusion in an online community. *Journal of Management Information Systems*, *28*(2), 11–38.

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* (Vol. 1). MIT press Cambridge.

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, *21*(9), 1263–1284.

Herremans, D., Martens, D., & Sörensen, K. (2014). Dance hit song prediction. *Journal of New Music Research*, *43*(3), 291–302.

Kaplan, A. M., & Haenlein, M. (2012). The britney spears universe: Social media and

doi:10.6342/NTU201802042

viral marketing at its best. *Business Horizons*, *55*(1), 27–31.

Kim, Y., Suh, B., & Lee, K. (2014). # nowplaying the future billboard: mining music listening behaviors of twitter users for hit song prediction. In *Proceedings of the first international workshop on social media retrieval and analysis* (pp. 51–56).

Pachet, F., & Roy, P. (2008). Hit song science is not yet a science. In *Ismir* (pp. 355–360).

Perricone, J. (2000). *Melody in songwriting: tools and techniques for writing hit songs*. Hal Leonard Corporation.

Singhi, A., & Brown, D. G. (2014). Hit song detection using lyric features alone. *Proceedings of International Society for Music Information Retrieval*.

Varian, H. R. (2014). Big data: New tricks for econometrics. *Journal of Economic Perspectives*, *28*(2), 3–28.

Zhou, Z.-H. (2012). *Ensemble methods: foundations and algorithms*. CRC press.

doi:10.6342/NTU201802042