國 立 成 功 大 學

資 訊 工 程 研 究 所

碩 士 論 文

使用關係與情境特徵進行社群文章電影預告推薦

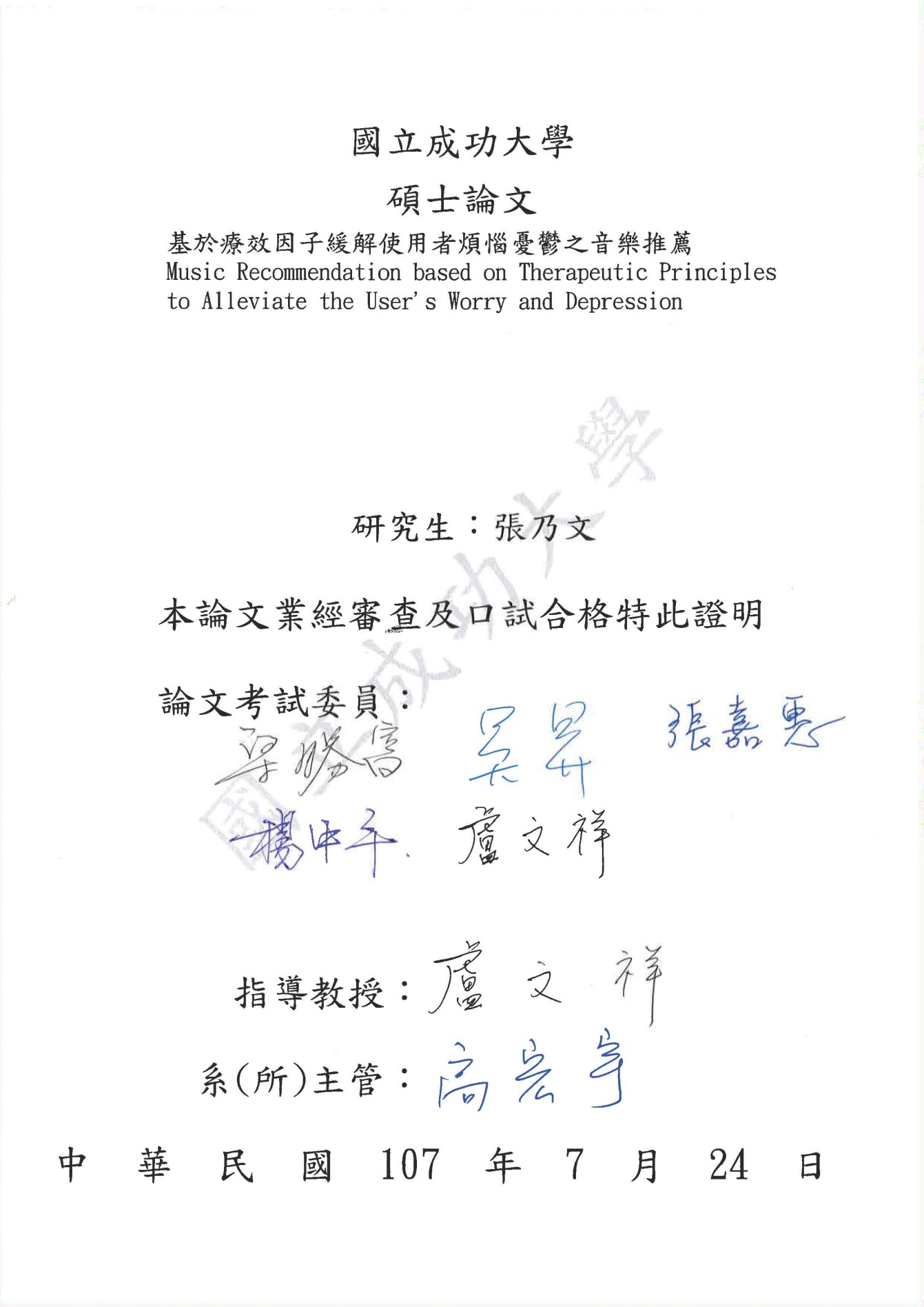
Using Relationship and Scenario Features of

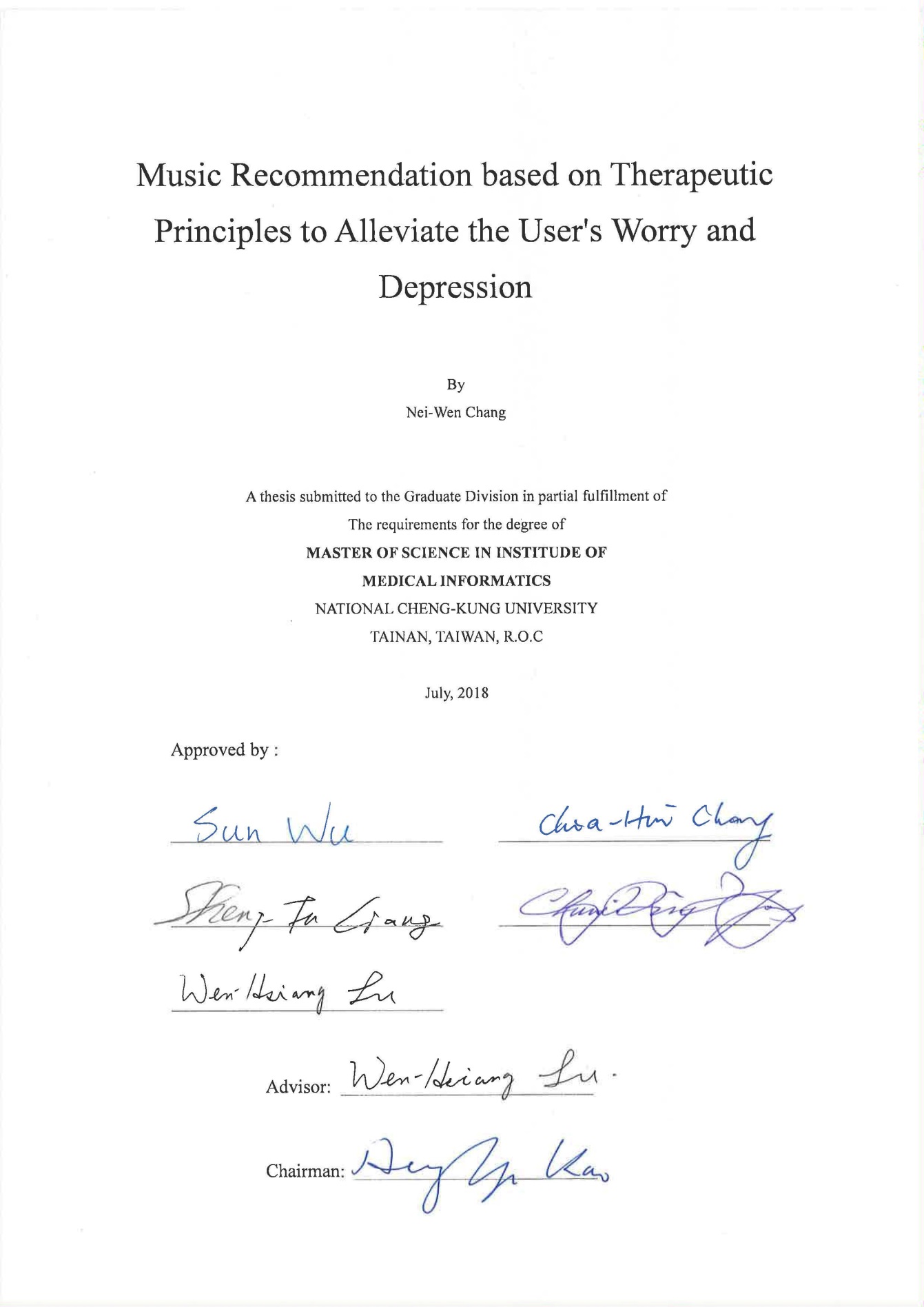
Plot Summaries for Social Article Trailer Recommendation

研究生：簡君聿

指導教授：盧文祥 博士

中華民國 一０八 年 七 月





摘要

推薦使用關係與情境特徵進行社群文章電影預告推薦

簡君聿\* 盧文祥\*\*

中華民國國立成功大學資訊工程研究所

社交平台上發表文章是年輕人最喜歡的活動。隨著電影產業的潛力，開發自動電影推薦引擎成為一個熱門話題。在社交媒體上，在共享相關預告片與關於日常生活在線社交平台的用戶生成文章的場景中，用戶傾向於選擇考慮其抒情主題的預告片。為了解決上述問題，我們提出了一種基於關係 - 場景的預告片推薦系統，該系統可以通過分析抒情主題來推薦預告片列表到輸入文章。我們認為抒情主題是關係和情景的結合，是情節總結的主觀和客觀視角。通過利用關係情景數據庫（Extend-HowNet作為知識庫），我們提取情節摘要和文章的關係和情景特徵。關係特徵表示為人物，情感，事件，地點和時間實體的實現。場景特徵表示為情感和事件實體的實現。

因此，我們表明，使用關係和場景特徵提供更好的推薦結果，而不僅僅考慮其中一個特徵，最後我們的推薦系統在用戶偏好和系統性能的兩個實驗中都優於新的W2V基線。我們還考慮用戶對系統關於不同關係類的偏好。

關鍵字：預告片推薦、情節摘要分析、文章分析、Word2Vec、變壓器雙向編碼器表示、卷積神經網絡、支持向量機、隨機森林分類器

\*作者 \*\*指導教授

Abstract

Using Relationship and Scenario Features of

Plot Summaries for Social Article Trailer Recommendation

Chun-Yu Chien\* Wen-Hsiang Lu\*\*

Institute of Computer Science and Information Engineering

National Cheng Kung University, Tainan, Taiwan, R.O.C

The post articles on the social platform is the favorite activity of young people. With the potential of digital movie industry, developing automatic movie recommendation engines becomes a popular issue. On social media, in the scenario of sharing related trailers with user-generated articles about daily life on line social platforms, users tend to choose trailers considering their lyrical theme.

To solve the above problem, we present a Relationship-Scenario-based Trailer Recommendation System which can recommend list of trailers to an input article by analyzing lyrical theme. We consider lyrical theme as a combination of Relationship and Scenario, the subjective and objective perspective of plot summaries. By utilizing relationship-scenario Database (Extended-HowNet as Knowledge base), we extract relationship and scenario features of plot summaries and articles. Relationship feature is represented as character, emotion, event, location and time entity relation. And scenario feature is represented as emotion and event entity relation.

Consequently, we show that using both relationship and scenario features provide better recommendation results than merely consider one of the features, In the end our recommender system outperforms a novel W2V baseline in both experiments of user preference and system performance. Also we consider user preference on our system about different relationship class.

Keywords: Trailer Recommendation, Plot summaries Analysis, Article Analysis, Word2Vec, Bidirectional Encoder Representations from Transformers, Convolutional Neural Network, Support Vector Machine, Random Forest Classifier

\*The Author \*\*The Advisor

致謝

首先真的非常謝謝我的指導教授盧老師，老師以他心中懷有的崇高教育理念作為他的領導風格，帶領實驗室研究具有創意且與其他教授不同且前衛的觀點，故在我心目中我非常的尊敬他也非常感謝他，也希望接下來的學弟妹能夠繼續秉持堅持著老師帶領我們的精神繼續把自己的研究做到最好。

在學習過程中與論文撰寫部分，非常謝謝在我身邊幫忙的人，包括老師的指導與國豪學長的幫忙跟觀念釐清，還有謝謝朋友佳純與敬濠的協助，過程中除了興奮以外還有感動，當然也謝謝在我的論文實驗中那些幫我標註的朋友們以及在口試期間幫忙很多的學弟們，未來也希望大家都可以順順利利朝著自己的夢想與目標前進，重點是要快樂的做自己。

最後我要謝謝我的父母親及其他愛我的家人，尤其是我的父母親，讓我沒有經濟上的壓力可以順利完成碩班的學業，在我最後的求學生涯上當我最強大的後盾，要感謝的人真的太多了，因此就謝謝上天！

君聿 八月于台南

Table of Contents

[摘要 IV](#_Toc16206655)

[Abstract V](#_Toc16206656)

[致謝 VII](#_Toc16206657)

[Chapter 1 Introduction 1](#_Toc16206658)

[1.1 Background 1](#_Toc16206659)

[1.2 Motivation 3](#_Toc16206660)

[1.3 Method 4](#_Toc16206661)

[1.4 Contribution 6](#_Toc16206662)

[1.5 Organization of this Dissertation 6](#_Toc16206663)

[Chapter 2 Related Work 7](#_Toc16206664)

[2.1 Studies on Sentiment analysis 7](#_Toc16206665)

[2.2 Studies on Film Trailer Topic Detection based on Plot Summaries 7](#_Toc16206666)

[2.3 Studies on Film Trailer Recommendation 8](#_Toc16206667)

[Chapter 3 Method 10](#_Toc16206668)

[3.1 System Framework 10](#_Toc16206669)

[3.2 Preliminaries 12](#_Toc16206670)

[3.2.1 CKIP Parser 12](#_Toc16206671)

[3.2.2 Extended-HowNet 13](#_Toc16206672)

[3.2.3 Data Sets and Preprocessing Steps 14](#_Toc16206673)

[3.3 Feature Generation 16](#_Toc16206674)

[3.3.1 Named Entity Recognition 16](#_Toc16206675)

[3.3.2 Training 20](#_Toc16206676)

[3.3.3 Relationship & Scenario Database 23](#_Toc16206677)

[3.4 Relationship Classification 26](#_Toc16206678)

[3.4.1 Relationship Term Extracting 27](#_Toc16206679)

[3.4.2 Relationship Model Training 28](#_Toc16206680)

[3.5 Scenario Classification 30](#_Toc16206681)

[3.5.1 Scenario Model Training 31](#_Toc16206682)

[3.6 Scenario Analysis 33](#_Toc16206683)

[3.7 Relationship-Scenario based Recommendation 36](#_Toc16206684)

[Chapter 4 Experiments 37](#_Toc16206685)

[4.1 Dataset 37](#_Toc16206686)

[4.2 Experiment of Entity Embedding Quality 38](#_Toc16206687)

[4.2.1 Dataset for Word Embedding 38](#_Toc16206688)

[4.2.2 Evaluation Metrics 38](#_Toc16206689)

[4.2.3 Evaluation Result 39](#_Toc16206690)

[4.3 Experiment of Relationship Classification 41](#_Toc16206691)

[4.3.1 Dataset for Relationship Classification 41](#_Toc16206692)

[4.3.2 Evaluation Metrics 43](#_Toc16206693)

[4.3.3 Experiment Result 44](#_Toc16206694)

[4.4 Experiment of Scenario Classification 45](#_Toc16206695)

[4.4.1 Dataset for Scenario Classification 45](#_Toc16206696)

[4.4.2 Evaluation Metrics 47](#_Toc16206697)

[4.4.3 Experiment Result 47](#_Toc16206698)

[4.5 Evaluation of Relationship-Scenario based Trailer Recommendation 48](#_Toc16206699)

[4.5.1 Evaluation Set and Evaluate User Preference Setting 48](#_Toc16206700)

[4.5.2 Experiment Result 51](#_Toc16206701)

[Chapter 5 Conclusions 52](#_Toc16206702)

[Reference 53](#_Toc16206703)

LIST OF FIGURE

[**Figure 1.1.1 China Cinema Revenue from 2002 To 2017** 2](file:///C:\Users\alex\Desktop\Project\backup\Recommender-System\P76064538.docx#_Toc16206607)

[**Figure 1.1.2 User Need from 2014 To 2017** 2](file:///C:\Users\alex\Desktop\Project\backup\Recommender-System\P76064538.docx#_Toc16206608)

[**Figure 1.2.1 Example of User Post Articles with A Related Trailer** 3](file:///C:\Users\alex\Desktop\Project\backup\Recommender-System\P76064538.docx#_Toc16206609)

[**Figure 3.1.1 System Framework of Relationship-Scenario-based Trailer Recommendation** 10](#_Toc16206610)

[**Figure 3.2.1 The Example of Result Parsed by CKIP Chinese Parser** 13](#_Toc16206611)

[**Figure 3.2.2 A Chip of E-HowNet Taxonomy Structure** 14](#_Toc16206612)

[**Figure 3.3.1 Structure of Feature Generation** 16](#_Toc16206613)

[**Figure 3.3.2 Name Entity Recognition Using CKIP Parser** 17](#_Toc16206614)

[**Figure 3.3.3 Model for BERT** 22](#_Toc16206615)

[**Figure 3.3.4 Model for Skip-Gram** 23](#_Toc16206616)

[**Figure 3.3.5 Relationship Feature** 24](#_Toc16206617)

[**Figure 3.3.6 Scenario Feature** 25](#_Toc16206618)

[**Figure 3.4.1 Structure of Relationship Classification** 26](#_Toc16206619)

[**Figure 3.4.2 CNN Architecture** 29](#_Toc16206620)

[**Figure 3.5.1 Structure of Scenario Classification** 30](#_Toc16206621)

[**Figure 3.5.2 Naïve Bayes** 31](#_Toc16206622)

[**Figure 4.2.1 Word Clouds of Emotion and Event** 40](#_Toc16206623)

[**Figure 4.2.2 Precision and NDCG Result for w2v-sg-150 and w2v-sg-300 and bert-768** 41](#_Toc16206624)

[**Figure 4.5.1 The Clip of Labeling Page for Labeling User Preference** 49](#_Toc16206625)

[**Figure 4.5.2 MAP@k and NDCG@k Results on User Preference and System Performance** 51](#_Toc16206626)

[**Figure 4.5.3 MAP@k and NDCG@k Results for Different Relationship on RSTR** 51](#_Toc16206627)

LIST OF TABLE

[**Table 3.1 Example of Film Storyline** 15](#_Toc16206628)

[**Table 3.2 The Distribution and Example of Character-Object Feature** 18](#_Toc16206629)

[**Table 3.3 The Distribution and Example of Location Feature** 18](#_Toc16206630)

[**Table 3.4 The Distribution and Example of Time Feature** 19](#_Toc16206631)

[**Table 3.5 The Distribution and Example of Emotion Feature** 19](#_Toc16206632)

[**Table 3.6 The Distribution and Example of Event Feature** 20](#_Toc16206633)

[**Table 3.7 Relationship & Scenario Database** 23](#_Toc16206634)

[**Table 3.8 Example of Mood Article** 27](#_Toc16206635)

[**Table 3.9 E-HowNet Relationship Lexicon** 28](#_Toc16206636)

[**Table 3.10 Definition of Scenario Class for Kinship and Percentage of Labeled Articles** 33](#_Toc16206637)

[**Table 3.11 Definition of Scenario Class for Romantic Relationship and Percentage of Labeled Articles** 33](#_Toc16206638)

[**Table 3.12 Definition of Scenario Class for Friendship and Percentage of Labeled Articles** 34](#_Toc16206639)

[**Table 3.13 Definition of Scenario Class for Teacher Relationship and Percentage of Labeled Articles** 34](#_Toc16206640)

[**Table 3.14 Definition of Scenario Class for Business Relationship and Percentage of Labeled Articles** 34](#_Toc16206641)

[**Table 3.15 Definition of Scenario Class for Others Relationship and Percentage of Labeled Articles** 35](#_Toc16206642)

[**Table 4.1 Number of Articles Retrieved from Dcard Board** 37](#_Toc16206643)

[**Table 4.2 Number of Films Retrieved from Movie Web** 37](#_Toc16206644)

[**Table 4.3 Frequent Emotion and Event Count** 40](#_Toc16206645)

[**Table 4.4 Percentage of Labeled Articles in each Different Annotator** 41](#_Toc16206646)

[**Table 4.5 Percentage of Labeled Relationship Articles in E-HowNet** 42](#_Toc16206647)

[**Table 4.6 Guideline for Labeling Relationship** 42](#_Toc16206648)

[**Table 4.7 Percentage of Labeled Articles in each Class of Relationship** 43](#_Toc16206649)

[**Table 4.8 Relationship Genre Classification Results (%) Measured by ACCURACY, PRECISION, RECALL and F-SCORE** 45](#_Toc16206650)

[**Table 4.9 Definition of Scenario Class for Different Relationship** 46](#_Toc16206651)

[**Table 4.10 Percentage of Labeled Movies** 46](#_Toc16206652)

[**Table 4.11 Trailer Scenario Genre Classification Results (%) Measured by ACCURACY, PRECISION, RECALL and F-SCORE** 47](#_Toc16206653)

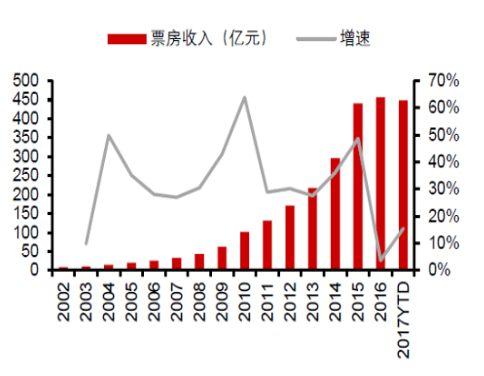
[**Table 4.12 Evaluate System Label Guideline** 49](#_Toc16206654)

# Introduction

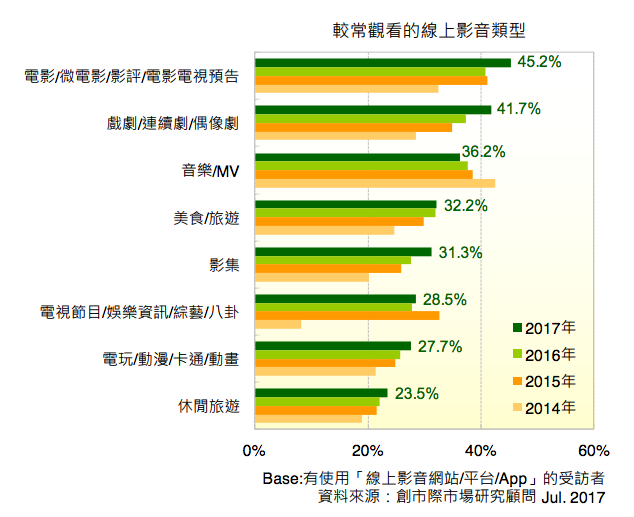
## Background

A recommender system is a type of information filtering system which attempts to predict the preferences of user and make suggestion based on these preferences. There are vast variety of applications for recommender systems. In recent years, with the rapid development and astonished achievement on AI field. More and more researcher using the AI technique to develop more user-friendly and user-closed recommender system. Due to the advances in recommender system, users constantly expect good recommendation.

Trailer has become a phenomenal trend in movie industry over the last decade, according to the report from '中國產業信息網' survey, the china cinema revenue has grown from 1 billion CNY dollars in 2002 to 45 billion CNY dollars in 2017(Fig 1.1.1). On the other hand, With the increased popularity of film, people gradually change their way of watching movie. and watching movies online is now very popular, according to 'ARO / MMX' survey, watching movies field online up to 45.2% in 2017(Fig 1.1.2)



**Figure 1.1.1 China Cinema Revenue from 2002 To 2017**



**Figure 1.1.2 User Need from 2014 To 2017**

## Motivation

With the growth of social network platforms, users can easily share personal stories online. Mood articles often describe various relationship and scenario between characters on daily life of a PO writer, whether it is a PO writer or a reader who likes this article, they want to be involved in the article. The storyline or movie trailer, often describe various relationship and scenario between protagonists in the play, it also the trigger for the audience to want to go to the theater or watch movies online. Sometimes, in order to help the readers to better understand these stories, if also share a trailer, maybe will impact users watching the film and then increase the cinema revenue.

For example shown in Fig 1.2.1, a user posted an article and recommender the trailer "生生". The subject of this article matched the main idea of trailer which process of growth.



**Figure 1.2.1 Example of User Post Articles with A Related Trailer**

Above example, we observe that when users try to convey their stories, if we can recommend the trailer that are relationship and scenario similar, maybe it is a good idea. However, it is not an easy task for users to find the most thematically related trailer of their stories. Actually, this is state of the art idea, it doesn't currently exist. Therefore, in this paper, we introduce a recommendation system to help overcome the difficulty of finding relationship and scenario similar trailer of an article.

## Method

A recent research focus on film review [1, 2], it's direct sentiment analysis or opinion mining, but in this paper, we explore implicit user intention and we utilize relation between articles and film storyline. Two keys of “Storyline Themes” are pointed out in the article, "character relationship express" and “event and emotion description”. Storytelling is said to “be a common device among film”. It can either be in a narrative style of story or be related to a central theme of a person’s life. On the other hand, a film can fit into different moods that help the trailer “keeping the theme intact”.

Every user has different preferences and likes. In addition, even the taste of a single user can vary depending on a large number of factors, such as mood, season, or type of activity the user is doing. according to this problem, if you like this article, also you will like this trailer based on plot. In the domain of mining movie textual contents, recommender system has analysis film review and user preference [3, 4, 5]. However, to the best of our knowledge, the potential of combining both Relationship and Scenario elements for storyline analysis have not been explored yet.

Consequently, we extend the above ideas and further define the “Storyline Themes” of a trailer is composed of two elements: “Relationship” and “Scenario”. “Relationship” is the way in which two or more people or things are connected, or the state of being connected. In this paper, we view 5 features (character, event, time, location, emotion) of each film storyline and article as a probability distribution over several relationships by applying pre-trained model BERT [24] and W2V-SG [6] and then we use CNN Model Architecture [7] that implement classification. “Scenario” is a written outline of a film, novel, or stage work giving details of the plot and individual scenes which is the objective observation of human affairs which include the people and the things that are involved. We represent scenario as a feature vector that composed of several emotion and event concepts to capture the word level meaning of context. we take Extended-HowNet [8, 9, 10, 11, 12] as our knowledge base in our work. Last, we suppose social articles share the same feature space with film storyline and experiment the usability of our analysis result by developing a Relationship-Scenario-based Trailer Recommendation system for social articles.

## Contribution

The main contributions of our work are listed below:

* We use trailer storyline (plot summaries) and article as data source and analyze content in relationship and scenario perspectives.
* We generate relationship features utilizing Extended-HowNet, a Chinese knowledge base.
* We classify trailer storyline (plot summaries) into genres based on a variety of relationship-scenario feature extracted from the storyline (plot summaries).
* We proposed an application of Relationship-Scenario-based Trailer Recommendation system for recommending trailers to social articles using storyline (plot summaries) features.

## Organization of this Dissertation

The rest of the paper is organized as follows. In Chapter 2, we introduce several related works that are prior researches on analyzing movies or film recommender system. In Chapter 3, we give the details of our observation and method. In Chapter 4, we evaluate our system and show analysis results. Finally, In Chapter 5, we draw the conclusion and give some insights for future work.

# Related Work

In the chapter 2, we briefly review a number of researches that are relevant to our topic, including studies on sentiment analysis, studies on film trailer topic detection based on plot summaries and studies on film trailer recommendation.

## Studies on Sentiment analysis

Martineau and Finin (2009) [18] weighted bag-of-words in employing a delta TF-IDF function for training SVMs to classify the reviews. Maas et al. (2011) [19] introduced a model to catch sentiment information and word meanings. Dai Quoc Nguyen (2004) [20] present a new feature type named rating-based feature and evaluate the contribution of this feature to the task of document-level sentiment analysis, however, they only analyze polarity or shallow sentiment, it's very rough. our paper consider reader intention on social article, and then infer to similar movie plots. Learn more about readers' hidden intentions, and further analyze emotion deeply.

## Studies on Film Trailer Topic Detection based on Plot Summaries

Some previous studies showed their interested on analysis plot summaries topics. Hoang, Q et al. (2018) [16] presented a Predicting Movie Genres Based on Plot Summaries, this project explores several Machine Learning methods to predict movie genres based on plot summaries which like utilized Naive Bayes, Word2Vec+XGBoost, Recurrent Neural Networks.

Ali Mert Ertugrul et al. (2018) [17] expected to reflect the genre of movies since many spectators read the plot summaries before deciding to watch a movie and the project perform movie genre classification from plot summaries of movies using bidirectional LSTM (Bi-LSTM). However, considering human understanding on viewing radar chart, they only used normal topics which were too clear for people to tell what storylines were really about. In our work, we do relationship classification and specific scenario classification and further achieve real life expression.

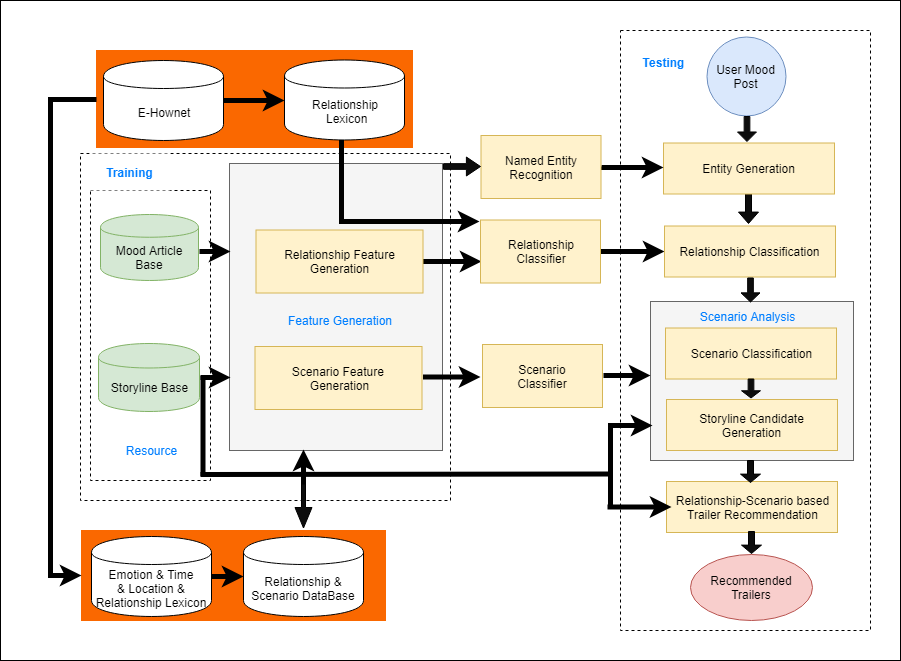
## Studies on Film Trailer Recommendation

Some studies by using movie reviews for recommendation and showing how are they different from our work. XS Vu et al. (2017) [4] presented Mining User/Movie Preferred Features Based on Reviews for Video Recommendation System, the statistic from 70% customers consult reviews or ratings before purchasing. Along with this, viewers also check movie reviews before making decision to buy movie tickets. Apply LDA for finding hidden aspects for addressing user preference aspects and movie feature aspects. After user preferences based on aspects and movie aspects are addressed, KL divergence is used for measuring similarity between movie and user. Top K movies that close to user preferences are recommended to user. it's direct sentiment analysis or opinion mining, it's a polarity research but in this paper, we further explore implicit user intention and we utilize relation between articles and film storyline. Two keys of “Storyline Themes” are pointed out in the article, "character relationship express" and “event and emotion description”. Storytelling is said to “be a common device among film”. It can either be in a narrative style of story or be related to a central theme of a person’s life. On the other hand, a film can fit into different moods that help the trailer “keeping the theme intact”.

# Method

## System Framework

In this paper, we purpose a **Relationship-Scenario-based Trailer Recommendation System** using relationship and scenario features of plot summaries. The proposed system framework is shown in Fig 3.1.1.



**Figure 3.1.1 System Framework of Relationship-Scenario-based Trailer Recommendation**

We defined a set of documents D = {P, S}, where P is a set of social mood articles and S is storyline (plot summaries) dataset. When a user submits a social article p ∈ P, we want to return a recommended list of trailers where each storyline s ∈ S. To complete this task, we want to represent each document d ∈ D as a composition of two features 𝑓 = {𝑓r, 𝑓𝑠}. Feature 𝑓r and 𝑓s denotes the relationship and scenario concepts of the given document. The entire system contains four major processing steps:

(1) Features Generation

Feature Generation is mainly divided into two parts, the first is the relationship feature generation, and the second is the scenario feature generation. Given a corpus c ∈ C, we will extract the character-object, time, location, events and emotions related entity using named entity recognition application and CKIP parser [13]. With the name entity recognition, we will then generate the Entity2Vec Model by applying BERT algorithm or Skip-Gram algorithm.

(2) Relationship Classification

In relationship classification, we will extract the relationship related terms using a predefined relationship lexicon. The relationship lexicon is built with the help of a Chinese knowledge base Extended-HowNet [8]. we will define 7 common classes of article relationships. With the relationship feature 𝑓r, we will build a multi-class relationship classifier. Applying the classifier, we can classify each document into single class of relationship and thus narrow down our film trailer candidates for recommendation output.

(3) Scenario Classification

In scenario classification, according to different relationship we will define several common classes of film scenario. With the scenario feature 𝑓𝑠, we will build a multi-class scenario classifier. Applying the classifier, we can classify each document into single class of scenario and thus narrow down our film trailer candidates for recommendation output.

(4) Relationship-Scenario based Recommendation

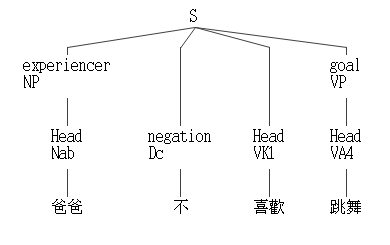
Recommendation step formulates the relevance between a given social article p and each storyline (plot summaries) s ∈ S using both relationship feature 𝑓r and scenario feature 𝑓𝑠. Then our system will return a recommended list of trailers according to the ranking of relevance score.

## Preliminaries

In this section, first of all we are going to introduce CKIP Parser and we give a brief introduction of Extended-HowNet which we use as our knowledge base. We also describe how we prepare our datasets and our preprocessing steps.

### CKIP Parser

Before analyzing the patterns of entity, we must obtain the word segmentation results and semantic features of the articles. We use CKIP Chinese Parser to help us obtain the segmentation results of the articles. CKIP Chinese Parser [13] is a tool developed by the CKIP (Chinese Knowledge and Information Processing Group). It can automatically analyze articles and obtain a lot of useful information, such as word segmentation, POS tag, syntax tree and semantic role. The relevant standards and rules for segmentation, syntax tree and semantic role are documented in the CKIP technical report [13][14][15]. In Fig 3.2.1 we show the result parsed by CKIP Chinese Parser. It can be seen that the sentences are parsed into a tree structure according to the grammar. The words that have the same parent form a phrase. Each word and phrase has a POS tag and semantic role. In following chapter, we will take advantage of POS tag and semantic role to build our pattern of entity. The detailed usages are described in Section 3.2.1.

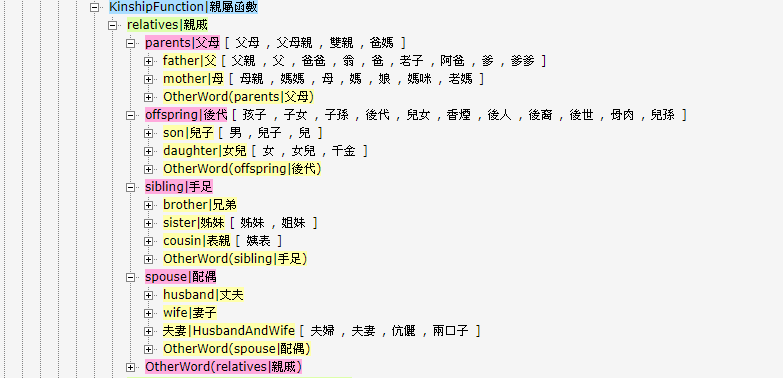


**Figure 3.2.1 The Example of Result Parsed by CKIP Chinese Parser**

### Extended-HowNet

Extended-HowNet (E-HowNet) is a frame-based entity relation model [8, 9, 10, 11, 12] which expended from HowNet. Each word sense can be decomposed into its simplest concepts and defined by E-HowNet. The taxonomies of concepts are organized to hierarchical structure. A part of ontology map of E-HowNet taxonomy structure is shown in Fig 3.2.2.

In this paper, we take the advantage of E-HowNet's hierarchical representation and utilize it in two ways. First, we build lexicons by extracting all terms defined by some concepts. Second, by mapping a term to its corresponding E-HowNet hypernym, we can replace the word level representation of a document to the concept level representation. The detailed usages are described in Section 3.4.1.

****

**Figure 3.2.2 A Chip of E-HowNet Taxonomy Structure**

### Data Sets and Preprocessing Steps

We use two types of datasets which are social mood articles and film storyline (plot summaries). In regard that we want to recommend trailers utilizing storyline features, trailer storyline and social articles are used as our training data. On the other hand, social mood articles are used as our testing data. We describe the characteristics of our datasets as follows.

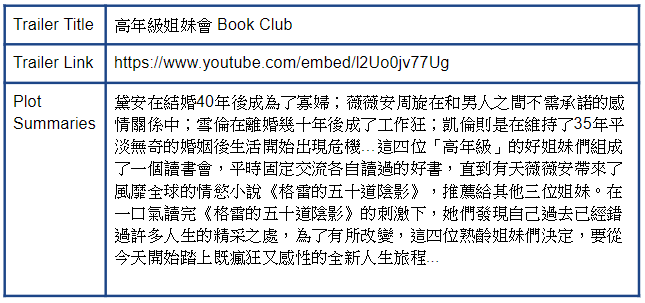
**Social Mood Articles Data**

Social articles are user-generated essays that are published on social websites. We choose social articles from Dcard, it is a popular online social platform for teenager in Taiwan, as our social article source for the reason of its simplicity in article categorization. Since we aim at recommend trailers that are related to user posts about daily life, we main collect posts from “Mood (心情版) ” boards. We don’t use “Movies (電影版)” board because most of the articles on Movie board are about sharing movie review, not sharing one’s daily life. Each social article contains post title and user generated text.

**Plot Summaries Data**

For film storyline data, we extracted from the most popular Mandarin website yahoo.com(奇摩電影) and pixnet.com(痞客邦電影).  Film Storyline are usually composed of short pieces of paragraphs. Some paragraphs may attract movie lovers to watch movies. Table 3.1 shows and example of a film plot summaries. We record film title and plot summaries and link for each trailer.

**Table 3.1 Example of Film Storyline**

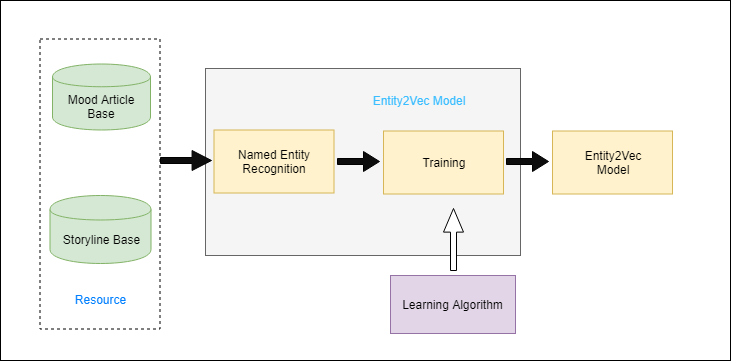


**Data Preprocessing**

A Chinese parser, including word segmentation/POS tagging/parsing/role assignment, has been completed. For parser, both storyline dataset and social mood article set are parsed using CKIP Parser system [13]. Besides, a selected list of stop words is removed from both mood article and film storyline in dataset. Finally, through the observation, a dictionary was also established for filtering.

## Feature Generation

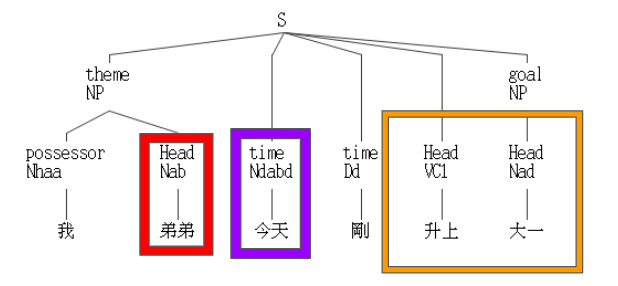
The first part of our system is Feature Generation. In this part, we focus on finding a representation that can showcase the relationship state and scenario state of a given document. The model generating step is shown in Fig 3.3.1.



**Figure 3.3.1 Structure of Feature Generation**

### Named Entity Recognition

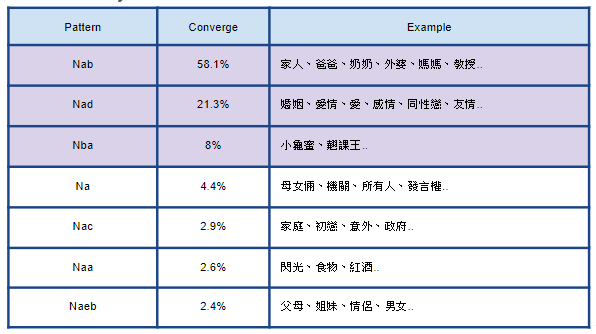
We use articles and storyline for observation and then define POS tagging and semantic role dictionary. The rule based, we can extract the character-object, time, location, events and emotions related entity using named entity recognition application and CKIP parser [13]. For example shown in Fig 3.3.2.



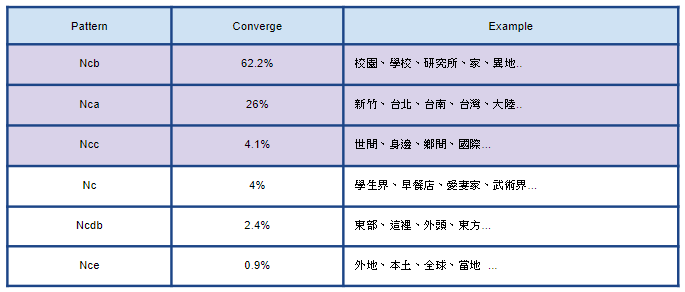
**Figure 3.3.2 Name Entity Recognition Using CKIP Parser**

How we extract character-object candidates, location candidates, time candidates, emotion candidates and events candidates? we use CKIP parser and exploit specific POS tagging and semantic role. It is achieved feature extraction application. We randomly sample 50 articles and 50 storylines for observation and coverage calculation for all data are Shown in Table 3.2, Table 3.3, Table 3.4, Table 3.5 and Table 3.6.

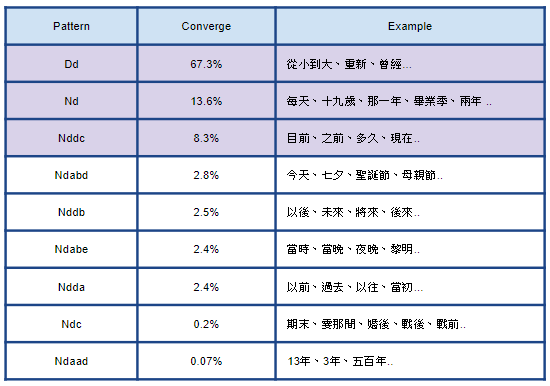
**Table 3.2 The Distribution and Example of Character-Object Feature**

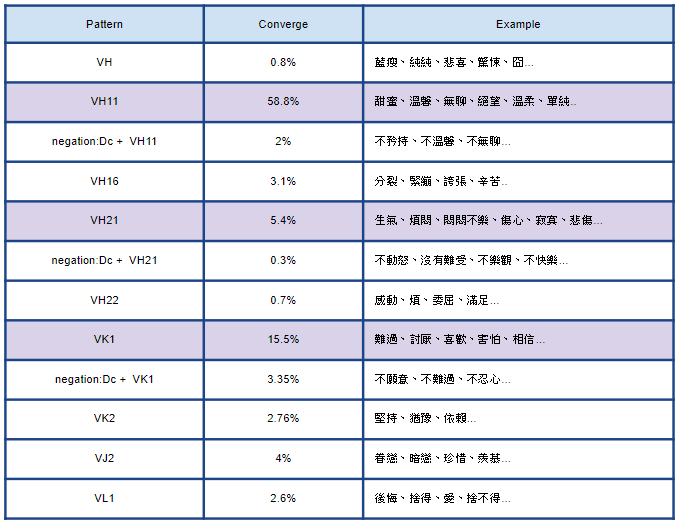


**Table 3.3 The Distribution and Example of Location Feature**

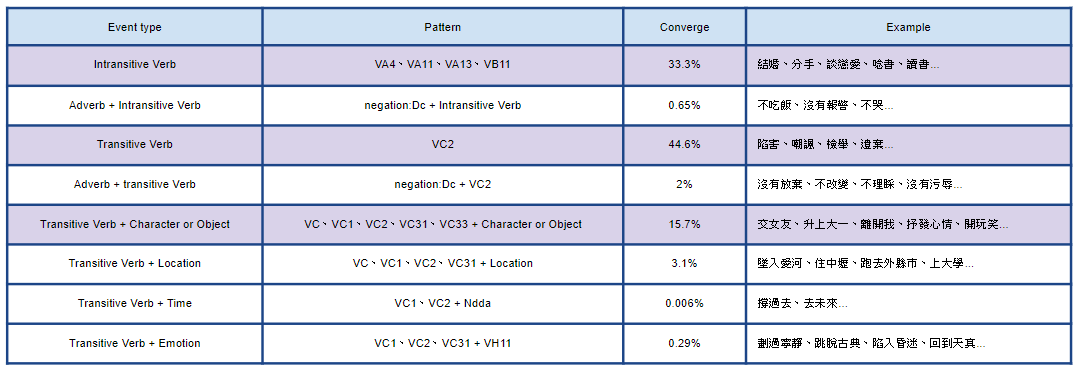


**Table 3.4 The Distribution and Example of Time Feature**



**Table 3.5 The Distribution and Example of Emotion Feature**

**Table 3.6 The Distribution and Example of Event Feature**



### Training

Now having character-object candidates, emotion candidates, time candidates, location candidates, and events candidates of articles and storyline, we want to identify the relationship state and scenario state of documents. We consider the relationship state and scenario state of a document to be represented as a probability distribution over a number of relationship topic related and scenario topic related. And we use BERT and Word2Vec embedding method.

**BERT**

A new language representation model called BERT [24], which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications and BERT architecture shown in Fig 3.3.3.

BERT is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing tasks, including pushing the GLUE score to 80.5% (7.7% point absolute improvement), MultiNLI accuracy to 86.7% (4.6% absolute improvement), SQuAD v1.1 question answering Test F1 to 93.2 (1.5 point absolute improvement) and SQuAD v2.0 Test F1 to 83.1 (5.1 point absolute improvement).

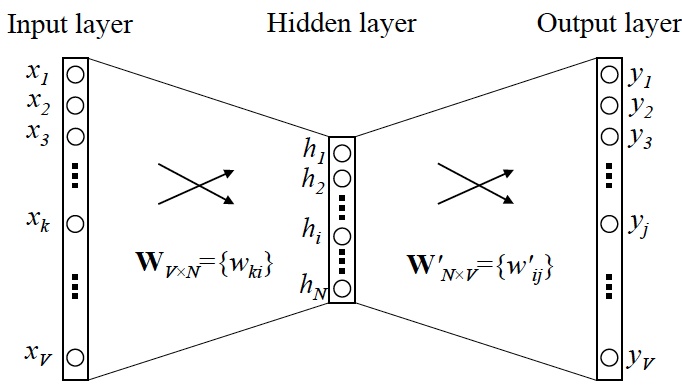


**Figure 3.3.3 Model for BERT**

**Word2Vec**

Word Embedding were revolutionized by Mikolov et al. [6] who proposed the CBOW and skip-gram models. CBOW computes the conditional probability of a target word given the context words surrounding it across a window of size k. On the other hand, the skip-gram model does the exact opposite of the CBOW model, by predicting the surrounding context words given the central target word. The context words are assumed to be located symmetrically to the target words within a distance equal to the window size in both directions. In unsupervised settings, the word embedding dimension is determined by the accuracy of prediction. As the embedding dimension increases, the accuracy of prediction also increases until it converges at some point, which is considered the optimal embedding dimension as it is the shortest without compromising accuracy. Let us consider a simplified version of the skip-gram model where only one word is considered in the context. This essentially replicates a bigram language model.

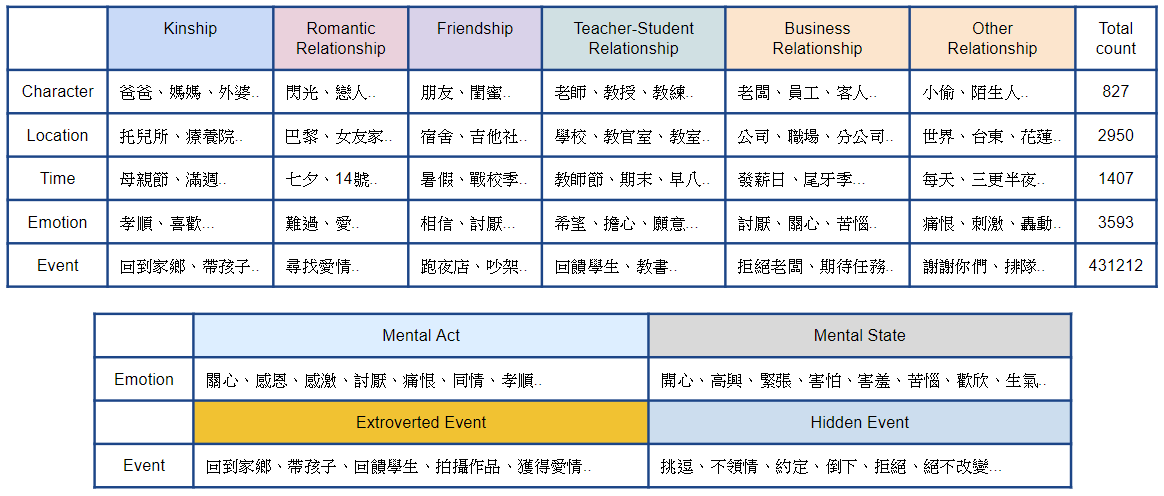
As shown in Fig 3.3.4, the skip-gram model is a simple fully connected neural network with one hidden layer. The input layer, which takes the one-hot vector of target word has V neurons while the hidden layer has N neurons. The output layer is Softmax function of all words in the vocabulary.



**Figure 3.3.4 Model for Skip-Gram**

### Relationship & Scenario Database

**Data Base**

**Table 3.7 Relationship & Scenario Database**

* Character

E-HowNet Relationship Lexicon and One Annotator

* Location

E-HowNet Place Lexicon and One Annotator

* Time

E-HowNet Time Lexicon and One Annotator

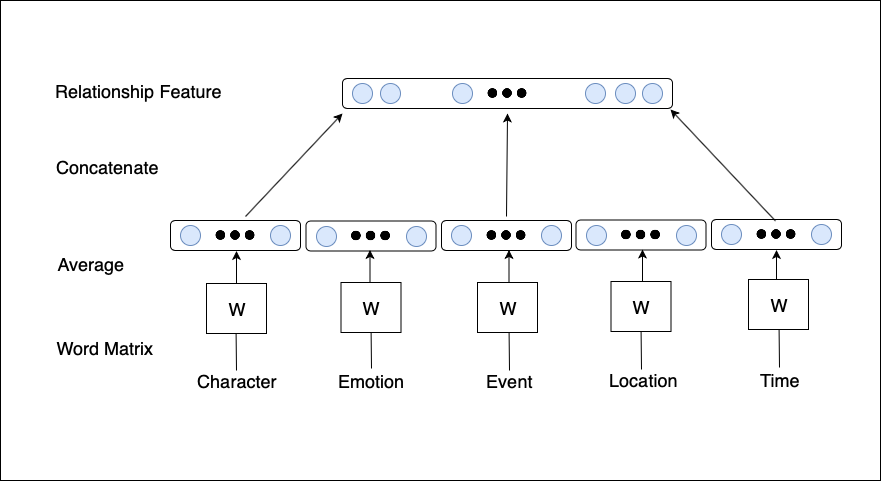
* Emotion

E-HowNet Emotion Lexicon

* Event

Article Event

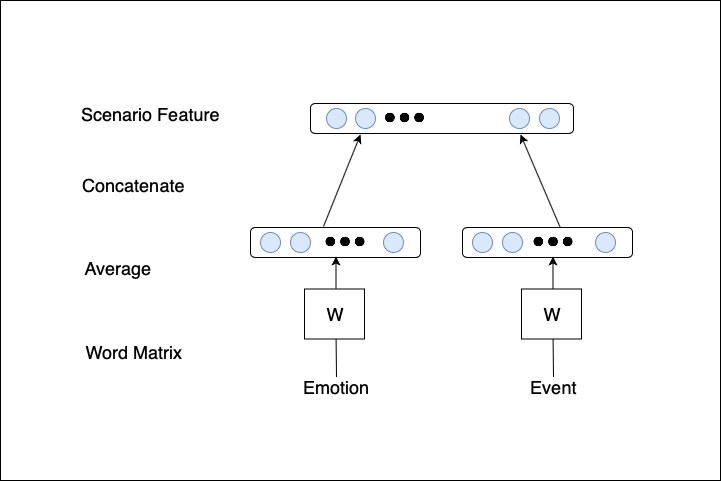
**Relationship Feature**



**Figure 3.3.5 Relationship Feature**

We consider character, emotion, event, location and time entities about relationship feature. In this framework, every entity is mapped to a unique vector, represented by a column in a matrix W. The column is indexed by position of the word in the vocabulary. The average of the vectors is then used as entity feature and then concatenate every entity feature will become relationship feature. As shown in Fig 3.3.5.

**Scenario Feature**

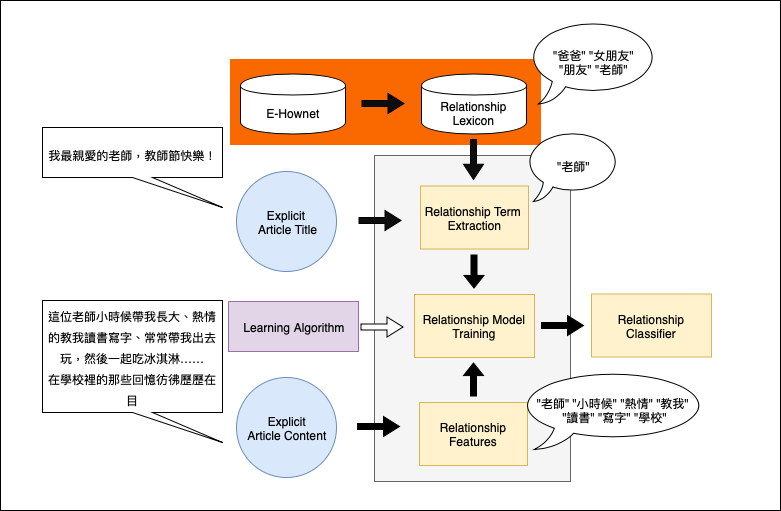
****

**Figure 3.3.6 Scenario Feature**

We consider emotion and event entities about scenario feature. In this framework, every entity is mapped to a unique vector, represented by a column in a matrix W. The column is indexed by position of the word in the vocabulary. The average of the vectors is then used as entity feature and then concatenate every entity feature will become scenario feature. As shown in Fig 3.3.6.

## Relationship Classification

The second part of our system is **Relationship Classification**. In this part, we focus on finding a representation that can showcase the relationship class of a given document. The model generating step is shown in Fig 3.4.1.



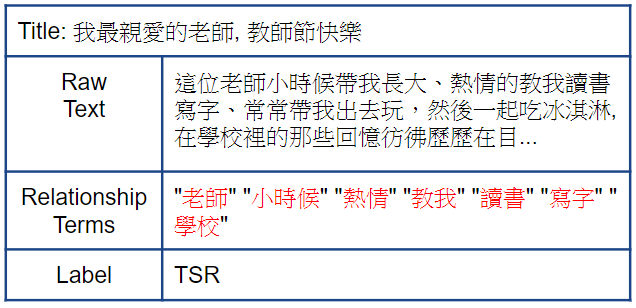
**Figure 3.4.1 Structure of Relationship Classification**

First of all, we define relationship lexicon built from E-HowNet. Using the lexicon, we can extract relationship terms from each explicit title article document for ground truth. Then we apply Convolutional Neural Network (CNN) [7] for relationship model training. Finally, based on the relationship classifier, the relationship class of each document can be represented as a probability distribution over the number of relationship topics, which we called relationship genre.

### Relationship Term Extracting

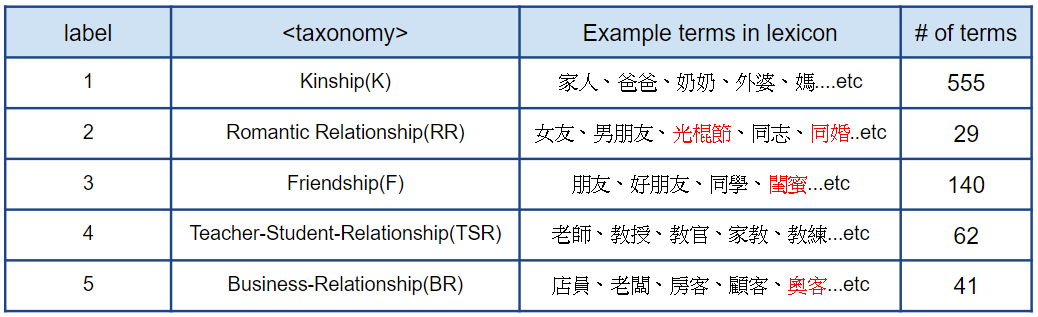
To extract relationship term, first we need to define which terms are related to relationship and further build relationship lexicon. We define seven types of relationships through articles and movie plots, which can be divided into family relationship, romantic relationship, friendship, teacher and student relationship, business relationship, others and no relationship etc.. For examples shown in Table 3.8, It is an example of a teacher-student relationship, we observe that no matter what kind of relationship will be affected by the character-object, emotion, time, location, and events.

**Table 3.8 Example of Mood Article**



On the other hand, the lexicon mentioned in the previous section is mainly used to use the title of the article, that is, to use lexicon for semi-automatic tagging is shown in Table 3.9. It can be seen from the table that there are mainly five types, which are just mentioned in the other Two types of artificial annotator.

**Table 3.9 E-HowNet Relationship Lexicon**

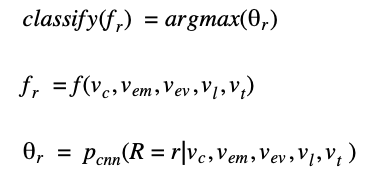


**\* the red word means new words.**

Now having article genres and extracted relationship features (character-object, emotion, time, location, and events) of articles, we want to identify the relationship genre of documents. We consider the relationship state of a document to be represented as a relationship genre using supervised learning method.

### Relationship Model Training

Generate the relationship topic feature which is represents as a distribution relationship topic(types). Apply Convolutional Neural Network(CNN) (Kim, Yoon. 2014. Convolutional neural networks for sentence classification)

 (1)

(2)

(3)

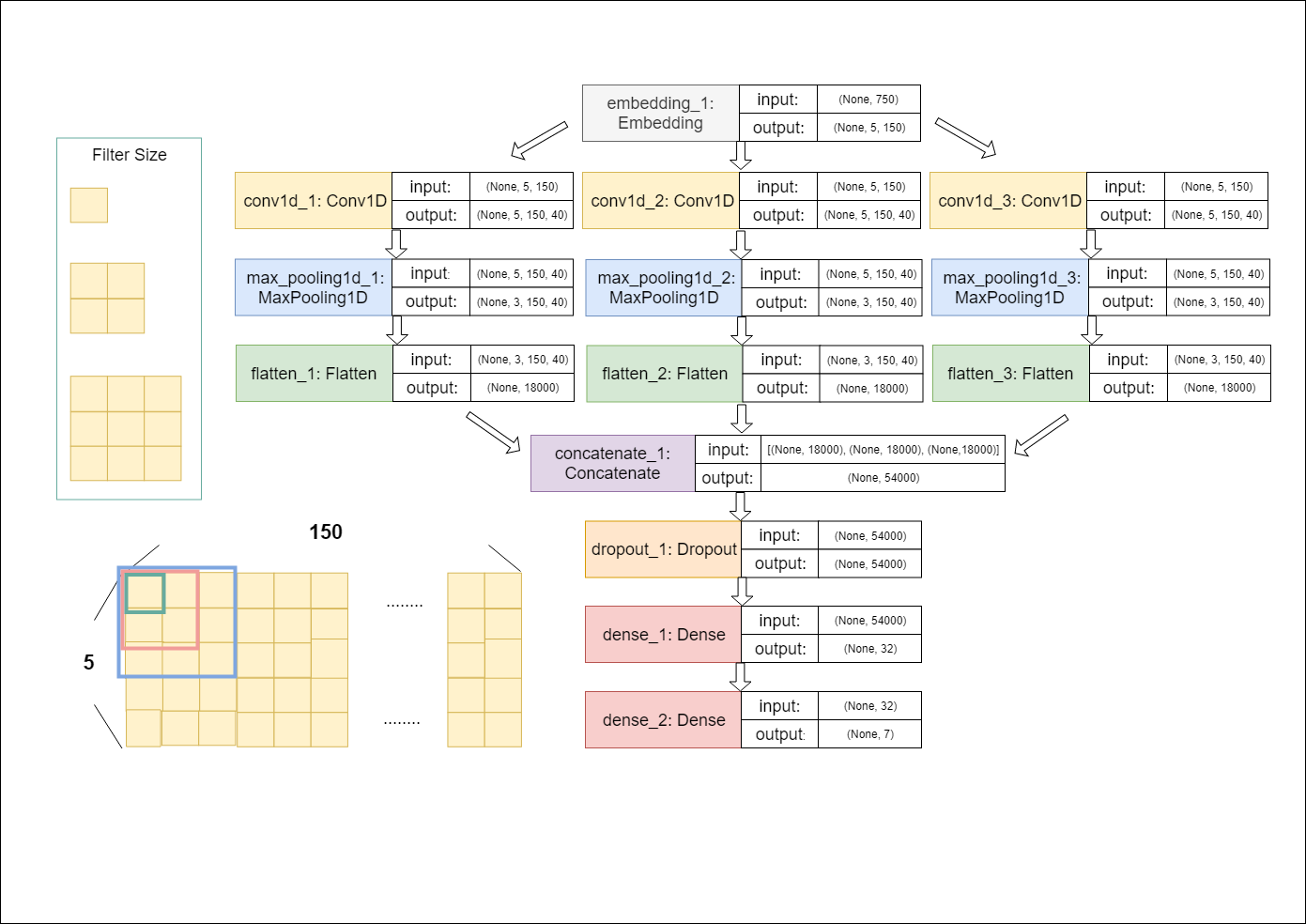
Where are character, emotion, event, location and time entity vector of an article p.

**Convolutional Neural Network**

Convolutional neural networks (CNN) utilize layers with convolving filters that are applied to local features (LeCun et al., 1998). Originally invented for computer vision, CNN models have subsequently been shown to be effective for NLP and have achieved excellent results in semantic parsing (Yih et al., 2014), search query retrieval (Shen et al., 2014), sentence modeling (Kalch-brenner et al., 2014), sentence classification (Yoon Kim et al., 2014) [7], and other traditional NLP tasks (Collobert et al., 2011).

**Model Architecture**

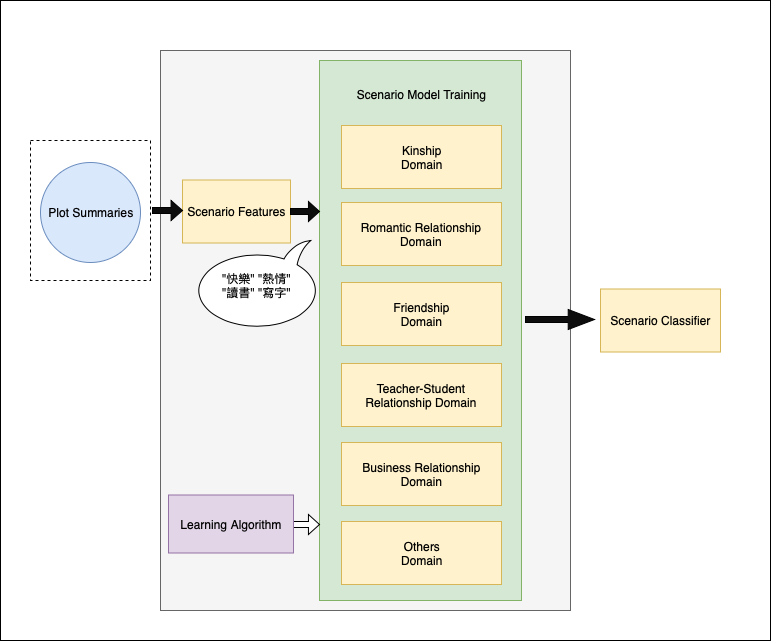
Apply Convolutional Neural Network(CNN) (Kim, Yoon. 2014. Convolutional neural networks for sentence classification), based on this structure, we create fit our case model. As shown in Fig 3.4.2.



**Figure 3.4.2 CNN Architecture**

## Scenario Classification

The third part of our system is **Scenario Classification**. In this part, we focus on finding a representation that can showcase the scenario class of a given document. The model generating step is shown in Fig 3.5.1.

****

**Figure 3.5.1 Structure of Scenario Classification**

First we can obtain scenario features from each document. Then we apply Machine Learning (Random forest, SVM, Naive Bayes) for scenario classification. Finally, based on the scenario classifier in different relationship domain, the scenario class of each document can be represented as a probability distribution over the number of scenario topics, which we called scenario genre.

### Scenario Model Training

Now having trailer genres and extracted scenario features (emotion and events) of trailer, we want to identify the scenario genre of documents. We consider the scenario state of a document to be represented as a scenario genre using supervised learning method.

**Naïve Bayes(Baseline)**

In [machine learning](https://en.wikipedia.org/wiki/Machine_learning), naive Bayes classifiers are a family of simple "[probabilistic classifiers](https://en.wikipedia.org/wiki/Probabilistic_classifier)" based on applying [Bayes' theorem](https://en.wikipedia.org/wiki/Bayes%27_theorem) with strong (naive) [independence](https://en.wikipedia.org/wiki/Statistical_independence) assumptions between the features. Naive Bayes has been studied extensively since the 1960s. It was introduced (though not under that name) into the text retrieval community in the early 1960s, [21] and remains a popular (baseline) method for text categorization.



**Figure 3.5.2 Naïve Bayes**

**Support Vector Machine**

In [machine learning](https://en.wikipedia.org/wiki/Machine_learning), support-vector machines (SVMs, also support-vector networks[22]) are [supervised learning](https://en.wikipedia.org/wiki/Supervised_learning) models with associated learning [algorithms](https://en.wikipedia.org/wiki/Algorithm) that analyze data used for [classification](https://en.wikipedia.org/wiki/Statistical_classification) and [regression analysis](https://en.wikipedia.org/wiki/Regression_analysis). Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic [binary](https://en.wikipedia.org/wiki/Binary_classifier) [linear classifier](https://en.wikipedia.org/wiki/Linear_classifier) (although methods such as [Platt scaling](https://en.wikipedia.org/wiki/Platt_scaling) exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

**Random Forest Classifier**

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees habit of overfitting to their training set.

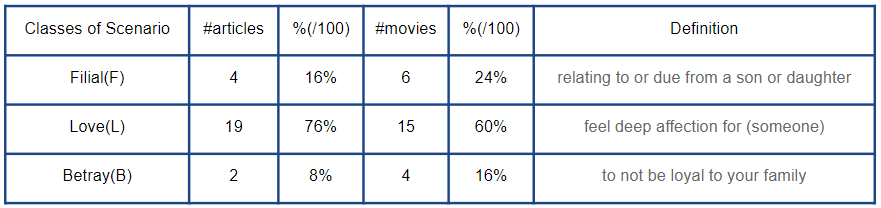
The first algorithm for random decision forests was created by Tin Kam Ho using the random subspace method [23] which, in Ho's formulation, is a way to implement the "stochastic discrimination" approach to classification proposed by Eugene Kleinberg.

## Scenario Analysis

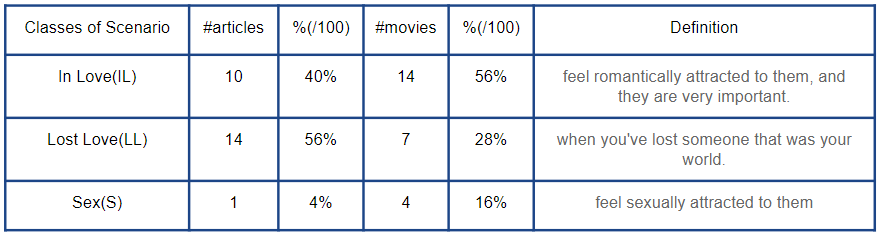
Build scenario classifier to generate storyline candidates for recommendation Q: What kinds of scenarios are most likely for user to want to post a trailer with?

* Separately observe 25 social articles and film storyline about daily life for kinship, romantic relationship, friendship, teacher-student relationship, business relationship and others respectfully.  Shown in Table 3.10, Table 3.11, Table 3.12, Table 3.13, Table 3.14 and Table 3.15.
* Movies by Genre: <https://www.imdb.com/feature/genre/>

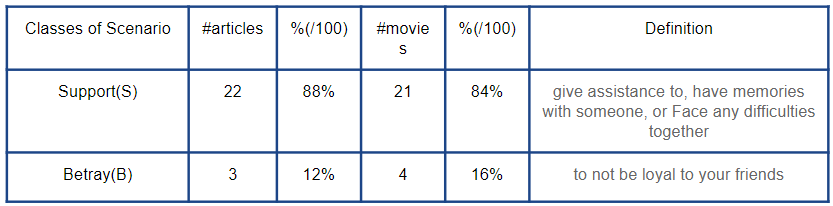
**Table 3.10 Definition of Scenario Class for Kinship and Percentage of Labeled Articles**



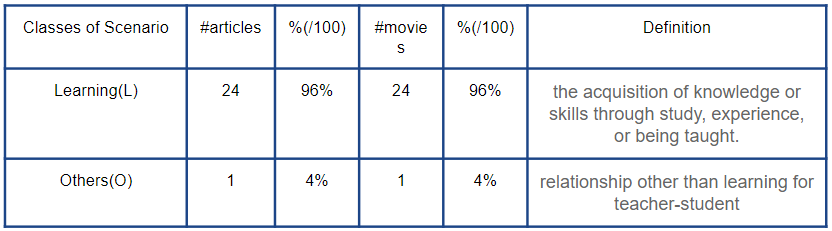
**Table 3.11 Definition of Scenario Class for Romantic Relationship and Percentage of Labeled Articles**



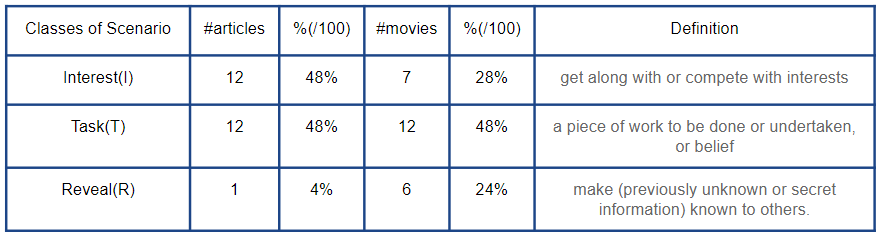
**Table 3.12 Definition of Scenario Class for Friendship and Percentage of Labeled Articles**



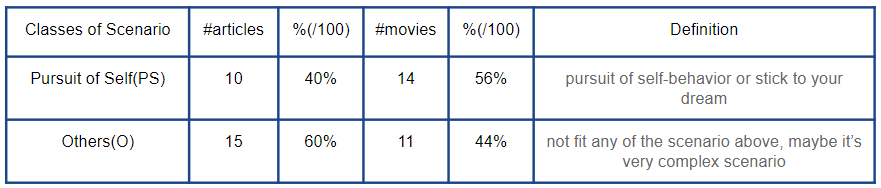
**Table 3.13 Definition of Scenario Class for Teacher Relationship and Percentage of Labeled Articles**



**Table 3.14 Definition of Scenario Class for Business Relationship and Percentage of Labeled Articles**



**Table 3.15 Definition of Scenario Class for Others Relationship and Percentage of Labeled Articles**



**Storyline Candidate Generation**

* https://lh3.googleusercontent.com/gl3PRoYtjVgBz5oDb_M2qOa4mUL86cB1jA9sQEJQdtCcvd-hlCcZW5_Q7XQpl2qIVU91W3hWYrcD0M9tNK5lZDCGPx6EZI_2lYdKPKXBLJeL_wY8UPjDa2W7y3vyQau1FmyLQLVYtQIHaving the relationship classifier and the scenario classifier, the testing article p will be classified into specific relationship class and scenario class, say
* https://lh4.googleusercontent.com/a1nYwDk54lbRDNrEOmJUMkBFGYYhlag_qzxGeSiZuJBlJLno6STvL3zJ_UC-bkfNtKeb37bRn--w78ETc3ohzABih556Q7lVWbUPx-xUg7xNWZX0afmtLyQIH-d8_rQCz9gH4Ths1FEGiven the storyline base, with each storyline in has already classified into the storyline candidates           for recommendation is

https://lh5.googleusercontent.com/dQ4iQ0VrUkAXMYEDAZt82iTtiQ_HtqeYOyFSmkSHAgiPFPbEEQwqEOulFKT676tUFxIVk2tyDzUilQ_YQL4iU40EW4tntwwbzBtz-NjLLAAxVRlzgXYn84mBYZekZYWcsFNN-ftFGzM

(4)

## Relationship-Scenario based Recommendation

Having the relationship features and scenario features for a social article p and a set of trailer storyline s, we now aim at finding trailer that are highly related to article p. In our work, we assume that a storyline s is suitable for article p if they are in similar relationship state and are stories about similar scenario.

The trailer emotions and events is what we value most. Therefore, For each articles(p) state and storyline(s) state, we consider scenario state. To find the best trailers for p in storyline set s, for each storyline s ∈ S, we first calculate the relevance scenario score between p and s. The relevance scores are calculated as follows.

https://lh6.googleusercontent.com/XXhXPaOGWblsZJT0i94n47E2cAWvG4MRx76gb3VvdU-KNEB-BO1e8p6NM4K8a9u0BXNjsgXERBqPnTjAwGkTEbNWnT49_I2r8qOouJoJ5hLdI0SUSeY-xWjNxgb9fonJcxOnW-rd (5)

where https://lh4.googleusercontent.com/kkvqLOmzqA5WOJgpZVxLvu_VeKhvkVAkOb8NCKWWZZJ7dd_x0nbnpoUF4EzdrxSXhlHwzJHPYPSENfL3l-UAOWemJosAvOGUQcD4o0DT0D3vh0KOYE1XtChIenBhtpDB_mcwAHGG is scenario feature of article p, https://lh4.googleusercontent.com/MTo2pWwejAvUm5HW5ndZvE758pBJ89dg-3gUHnv4jLPSCdeE1NaV2EYcDOqnbQue-JxyDRDhE6FMNb-Qz6NFSKgVqNbeKcyEYMqBsPkxjalZZdElUJf6wsjWksLsp0MZ_i7xJCSf is scenario feature of storylines s. In the end, by ranking trailer storyline candidates with their recommendation scores, we can retrieve the recommended trailer list for an article.

# Experiments

## Dataset

(1) Social Articles

For social articles, We crawl social articles from “Mood (心情版)” boards of Dcard. The Social Articles - Dcard, a popular online social platform for young people in Taiwan. Therefore, we consider use it. Table 4.1 shows the number of articles we retrieved from board.

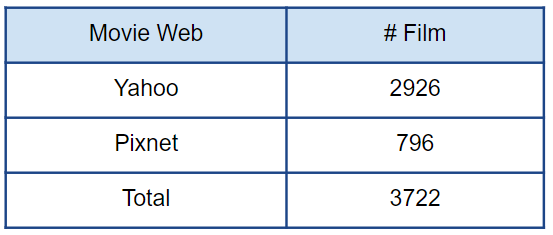
**Table 4.1 Number of Articles Retrieved from Dcard Board**



(2) Storyline (Plot Summaries)

For trailer plot summaries data, we consider yahoo and pixnet movies, an online storyline database.  Table 4.2 shows the number of films we retrieved from movie web.

**Table 4.2 Number of Films Retrieved from Movie Web**



## Experiment of Entity Embedding Quality

In entity embedding. We apply Mikolov et al. [6] who proposed the skip-gram models and Jacob Devlin et al. [24] who proposed BERT.

### Dataset for Word Embedding

**Data Set**

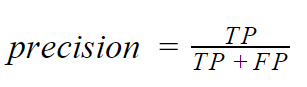
We use about one 3 million of sentences from our articles set and film plot summaries as training data.

### Evaluation Metrics

Two metrics are employed for Entity2Vec performance evaluation: precision (P@1) and normalized discounted cumulative gain (NDCG@k).

* **Precision at 1(P@1):**

The equation of precision is defined as follows:

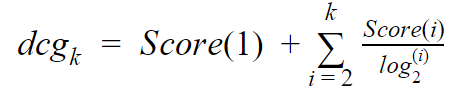


(6)

where the TP here is whether the rank order result retrieved by Enetiy2Vec model is the standard answer score.

* **Normalized Discounted Cumulative Gain at k (NDCG@k):**

With the relevance score, we also calculate NDCG to examine whether the terms of higher scores have the higher rank. We first calculate the DCG score as follows:



(7)

where 𝑘 is the number of top 𝑘 returned results, and https://lh4.googleusercontent.com/jnxGV2A7r0s7iyQXcbgTfmKFr4NMPvRRCLZe3m0e-dU_74Rr0zrViyuo6Oa_-nIuZXw5xfl0N8fzMnnXc6qv7aUdZVlSpN8UJ19RNeV6whCCM2G6Sefh4GuFQkx24jYmPLvRrfks is thehttps://lh3.googleusercontent.com/wkxHZMaHq3v2sKNBYprjF7oz3WP1xKJi_H1JojaHCBMf7Nvlfcujj65JtZC3Yr7e-rOebNe6VpehwUXQBdU75vylTQ-lFqeNGSP1bM3wrZEyZPfPUR94JP78kJXKYw5YTHLaIGfUscore of results. Then, we normalize DCG value with IDCG and forms the NDCG score as follows:

https://lh3.googleusercontent.com/J4LBGKiyLOMA0uA5AmI2F0YbVxZbVuJeDcXeRl3lAj1TFyTvn2cCoc0Yikos0Wq3rEGc4ByW87LIrbzic276HmnjN8eCY9net9aWJdGpRWL4DCVSmwMD0po_Cdvca1zBQO8DpMQr

(8)

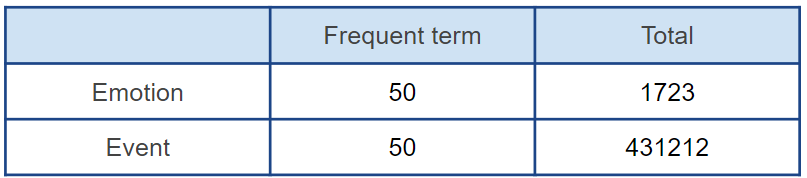
where IDCG refers to the ideal DCG score by sorting the terms by their relative relevance score. On calculating NDCG, we also set k from 1 to 10.

### Evaluation Result

**Experiment Protocol**

* Entity2Vec
  + Skip-gram(W2V-SG-150) and Skip-gram(W2V-SG-300)
    - Training Set: Mood Articles and Storyline
  + Bidirectional Embedding Representations from Transformers(BERT-768)
    - Training Set: wiki data
* Testing Emotion and Event
  + Top 50 frequent unique emotion
  + Top 50 frequent unique event

**Table 4.3 Frequent Emotion and Event Count**

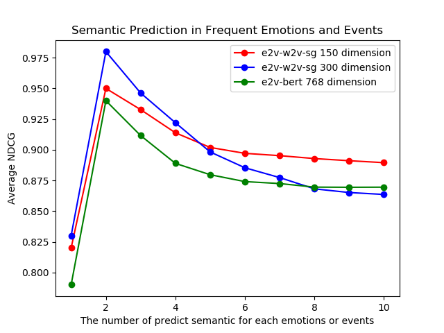
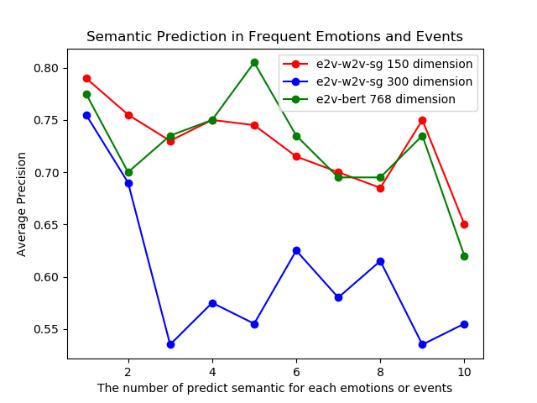




**Figure 4.2.1 Word Clouds of Emotion and Event**

* Three score criteria for entity embedding quality:
  + High relevance (score is 2): semantically correct or highly relevant
    - E.g. Event "聊天" > "抬槓", Emotion "記得" > "不會忘記"
  + Relevance (score is 1): semantically correct or relevant
    - E.g. Event "相處" > "不合拍", Emotion "開心" > "拿到禮物"
  + Not relevance (score is 0): semantically incorrect or less relevant
    - E.g. Event "不知道" > "沒這麼", Emotion "分手" > "復合"

We evaluate the performance of three different dimension: 150(w2v-sg) and 300(w2v-sg) and 768 (bert). The result is shown in Fig 4.2.2.



**Figure 4.2.2 Precision and NDCG Result for w2v-sg-150 and w2v-sg-300 and bert-768**

## Experiment of Relationship Classification

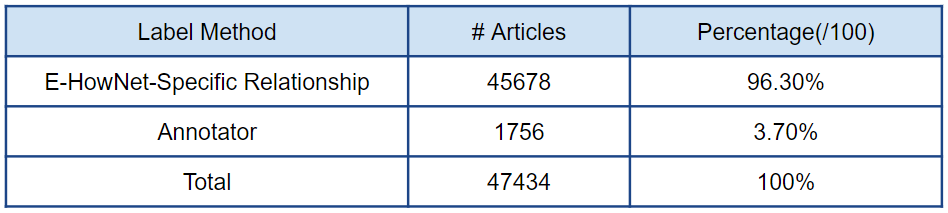
In relationship classification, we aim at finding a better embedding for multi-class classifier which can classify documents into one.

### Dataset for Relationship Classification

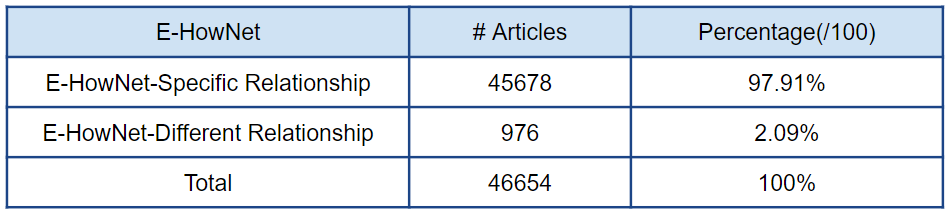
**Dataset**

We use about one fourth of articles from our articles set as training data. Percentage of labeled articles in each different annotator shown in Table 4.4 and Table 4.5.

**Table 4.4 Percentage of Labeled Articles in each Different Annotator**



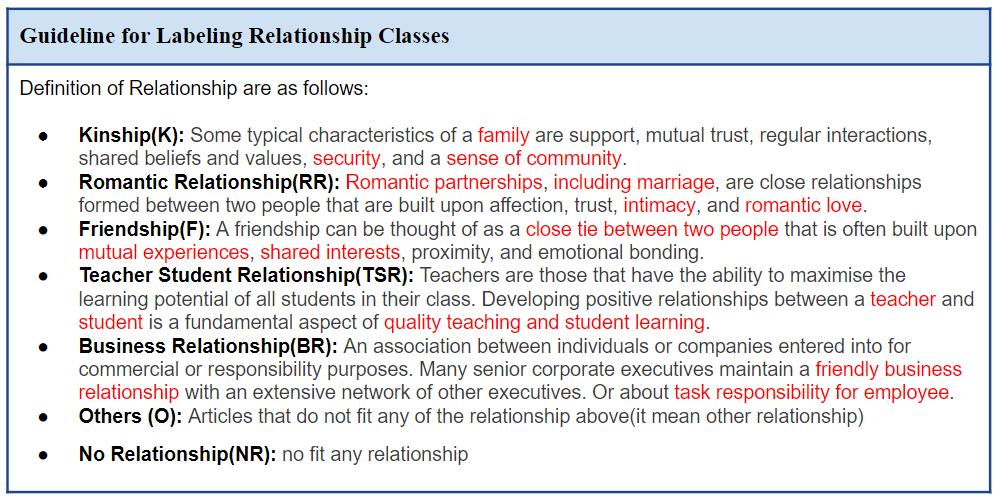
**Table 4.5 Percentage of Labeled Relationship Articles in E-HowNet**

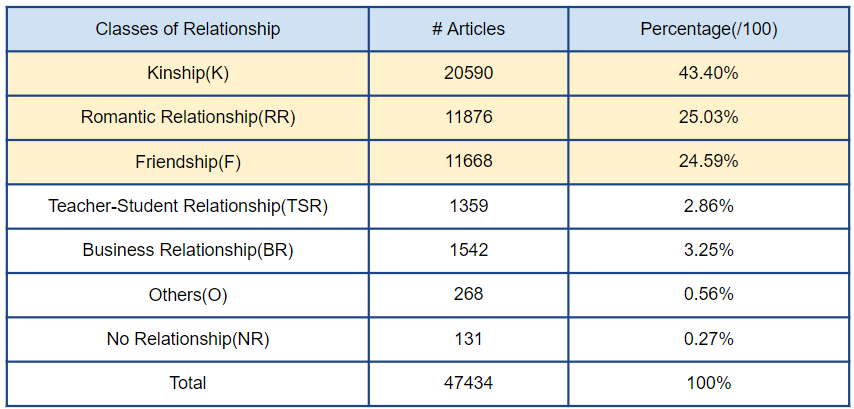


**Label Strategy**

Beside semi-automatic tagging, we have artificial annotator. To label the training data, we employ two annotators who is interested in articles to annotate. Annotators label each article in training data to relationship labels by the following guideline shown in Table 4.6.

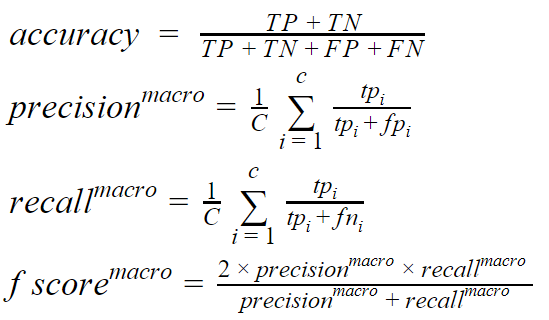
For each article, the relationship labels are then set to be the union set of labels labeled by the two annotators and semi-automatic tagging. Table 4.7 shows the percentage of articles in each class of relationship.

**Table 4.6 Guideline for Labeling Relationship**

**Table 4.7 Percentage of Labeled Articles in each Class of Relationship**

### Evaluation Metrics

To evaluate the performance of multi-class classifiers. We use 10-fold cross validation via our dataset and calculate the accuracy and hyper-parameters are tuned by cross-validation on the task training data and evaluation test data. To further compare performance of different methods and we consider holdout data, I calculate the confusion matrix, accuracy, macro precision, macro recall and F-score for each genre as well as for all of the test data (holdout data). These metrics are defined as:



(9)

(10)

(11)

(12)

where C is the number of target labels, TP, FP, TN and FN are true positive, false positive, true negative and false negative respectively. Note that, since we perform our experiments on a single dataset, micro precision, micro recall and micro f-score values are all equal and they represent the accuracy of the classifier. Accordingly, we only present the macro results.

### Experiment Result

We evaluate the performance of one multi-class classifiers: Convolutional Neural Network (CNN). We use 10-fold cross validation via our dataset and calculate the accuracy for different embedding method (e2v\_bert, e2v\_w2v\_sg and w2v\_w2v\_sg). The result is shown in Table 4.8.

**Table 4.8 Relationship Genre Classification Results (%) Measured by ACCURACY, PRECISION, RECALL and F-SCORE**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 10-fold CV  Acc. | Accuracy |  | Macro  Pre. | Macro  Rec. | Macro  F1 |
| CNN-W2V-W2V-SG | 73.94% | 74.55% |  | 40.84% | 42.63% | 41.69% |
| CNN-E2V-W2V-SG | **84.12%** | **83.87%** |  | **66.75%** | **60.94%** | **62.00%** |
| CNN-E2V-BERT | 74.90% | 74.07% |  | 50.52% | 49.06% | 49.72% |

## Experiment of Scenario Classification

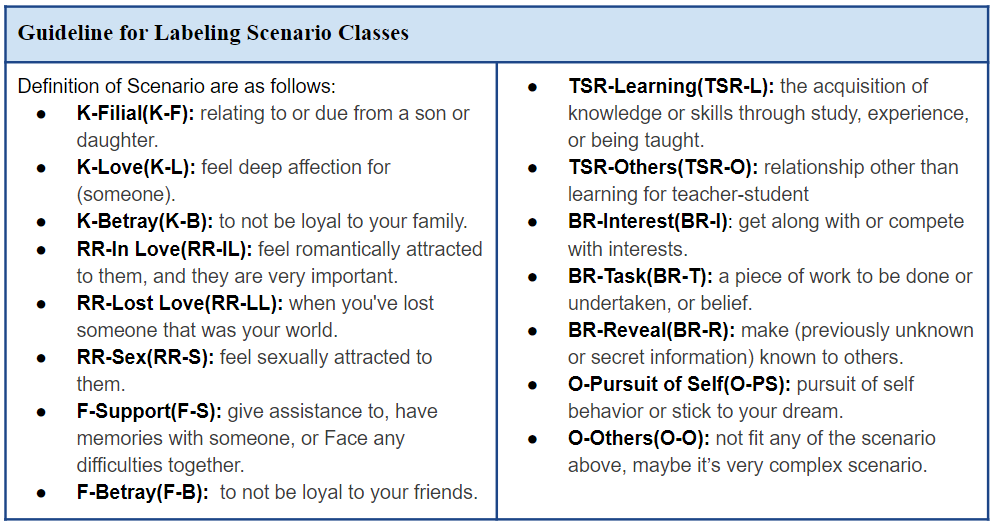
In scenario classification, we aim at finding a better multi-class classifier which can classify documents into one.

### Dataset for Scenario Classification

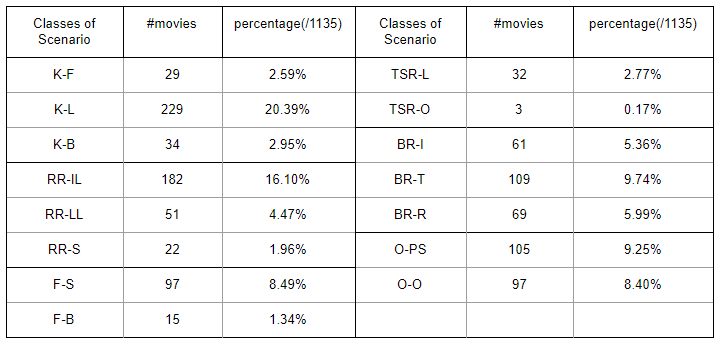
**Data Set and Label Strategy**

We use about 1135 from our storylines set as training data. To label the training data, we employ two annotators who are interested in trailer to annotate. Annotators label each storyline in training data to scenario labels by the following definition shown in Table 4.9. And the percentage of trailer in each class of scenario shown in Table 4.10.

**Table 4.9 Definition of Scenario Class for Different Relationship**



**Table 4.10 Percentage of Labeled Movies**

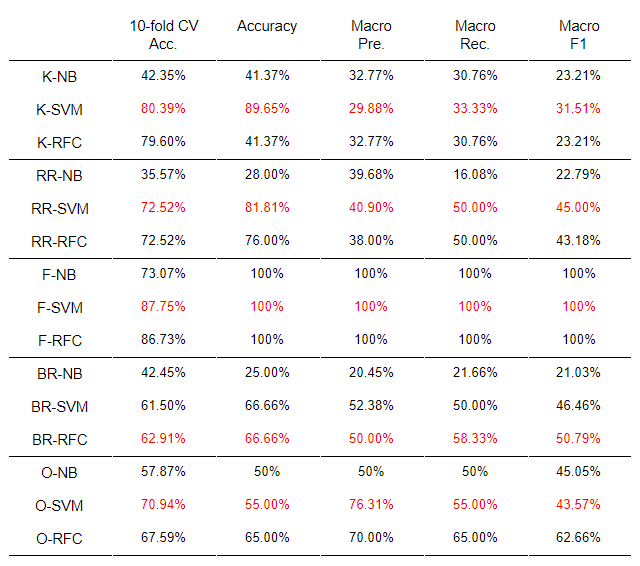


### Evaluation Metrics

We use 10-fold cross validation via our dataset and calculate the accuracy and hyper-parameters are tuned by cross-validation on the task training data and evaluation test data. To further compare performance of different methods and we consider holdout data, I calculate the confusion matrix, accuracy, macro precision, macro recall and F-score for each genre as well as for all of the test data (holdout data). The formulation of accuracy, macro precision, macro recall and F-score is introduced in Section 4.3.2.

### Experiment Result

**Table 4.11 Trailer Scenario Genre Classification Results (%) Measured by ACCURACY, PRECISION, RECALL and F-SCORE**



We evaluate the performance of three different multi-class classifiers on different relationship domain: Naive Bayes(NB), Support Vector Machine(SVM) and Random Forest Classifier(RFC). We use 10-fold cross validation via our dataset. By training our data, validating it, and testing it on the holdout set. The result is shown in Table 4.11.

## Evaluation of Relationship-Scenario based Trailer Recommendation

In relationship-scenario based trailer recommendation system, we want to know user preference about recommended trailer, therefore we compare w2v system(Baseline) with our system(RSTR).

### Evaluation Set and Evaluate User Preference Setting

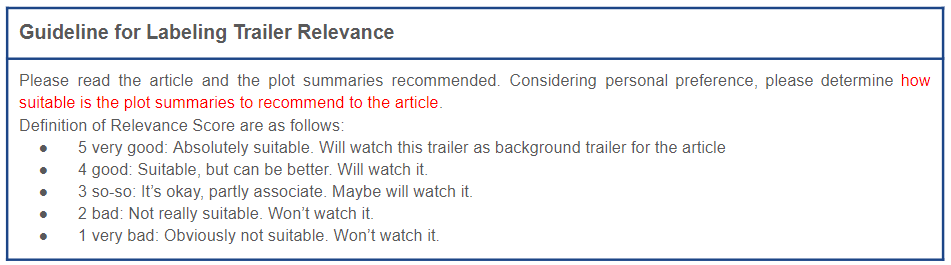
**Evaluation Set and System**

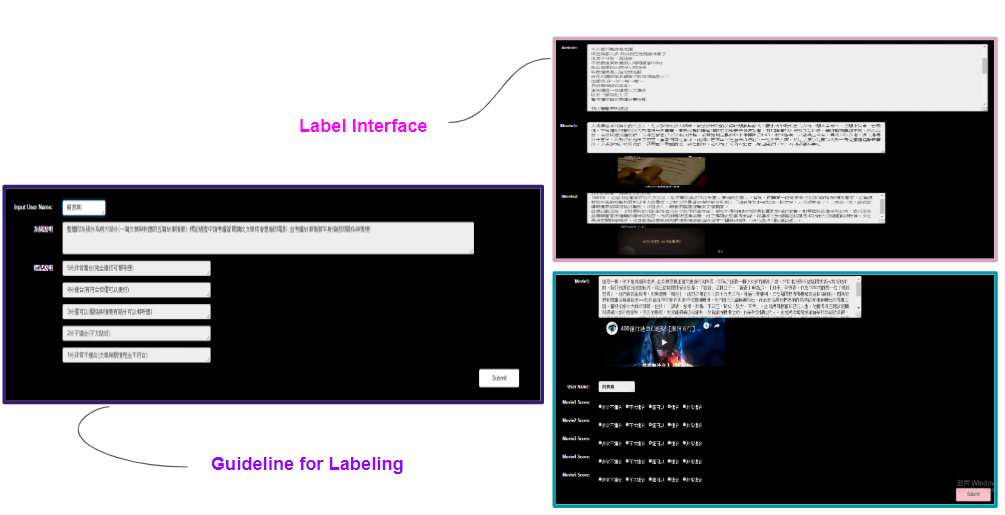
* Evaluation Set
  + Article: Use 60 random articles of our dataset 47434 about different relationship.
  + Trailer: 1135
* Our System
  + Relationship-Scenario based Trailer Recommendation(RSTR)
* Baseline
  + Word2Vec(W2V): Wen, P. C., et al. 2015(about music)
    - Training word vector of 150 dimensions
    - Each article and storyline are represented as sum of word vectors

**Evaluation of User Preference**

* Label Strategy (Table 4.12)
  + For each baseline, generate the set to label from top 5 results
  + 9 annotators are involved (6 males, 3 females, aged between 20-29)
  + Label “how suitable is the recommended plot summaries to the article”

**Table 4.12 Evaluate System Label Guideline**



****

**Figure 4.5.1 The Clip of Labeling Page for Labeling User Preference**

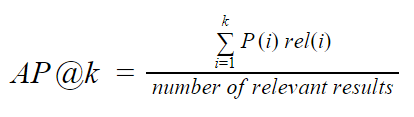
**Evaluation Metrics**

Two metrics are employed for user preference evaluation and system performance: mean average precision(MAP@k) and normalized discounted cumulative gain(NDCG@k).

**Mean Average Precision at k(MAP@k)**

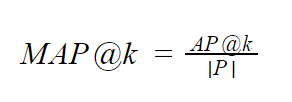
To calculate the precision score, we will transfer the relevance score labeled by annotators into binary indication. If the relevance score of plot summaries is larger than 3, we view the plot summaries as correct answer and label as 1. On the contrary, if the score is less than 3, the label will be 0. The formulation of MAP@k is introduced below:

Given a set of testing data, MAP is the average of the average precision(AP) for each data in the testing set, we first show the average precision for the top k results as follows:



(13)

where P(i) is the precision at cut-off rank i in recommendation list and rel(i) is the binary indicator function which is 1 if plot summaries and article are relevant, 0 vice versa.

****Then, the MAP@k is calculated as follows:

(14)

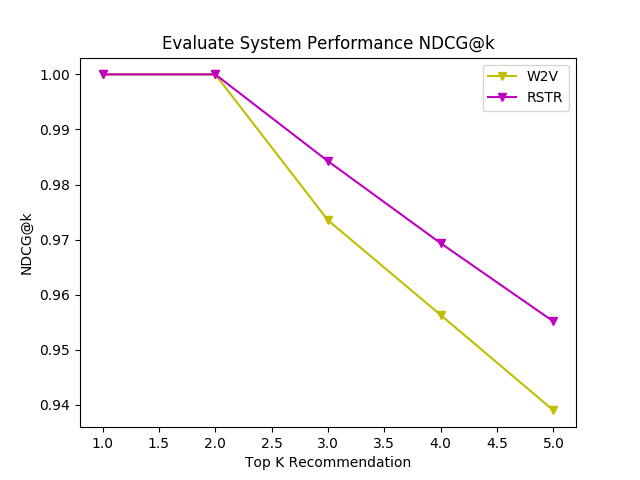
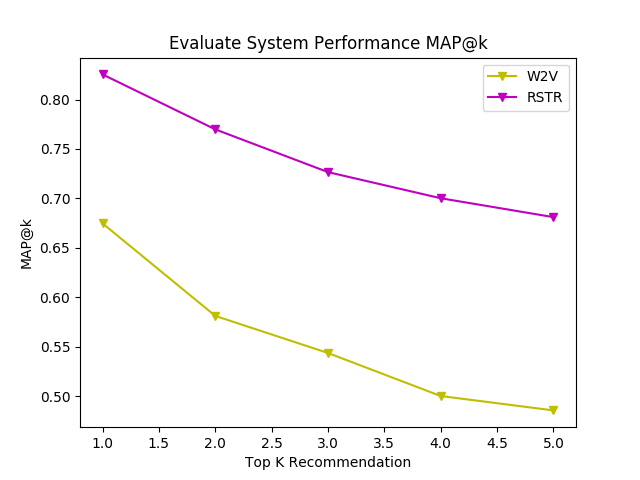
where |P| is the number of testing social article set.

**Normalized Discounted Cumulative Gain at k(NDCG@k)**

With the relevance score, we also calculate NDCG to examine whether the recommended trailers of higher scores have the high rank. The formulation of NDCG@k is introduced in Section 4.2.2.

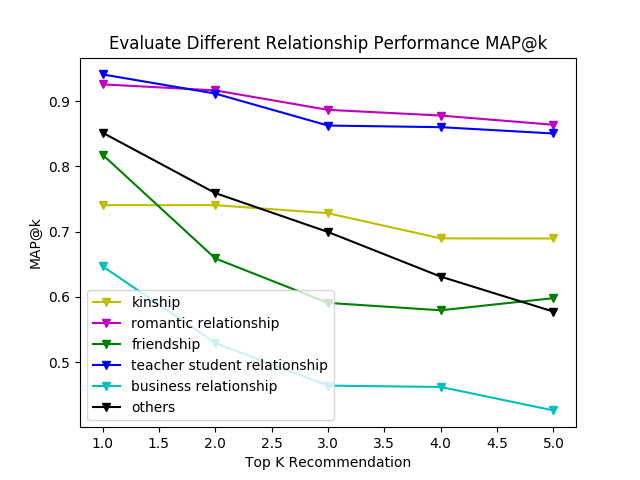
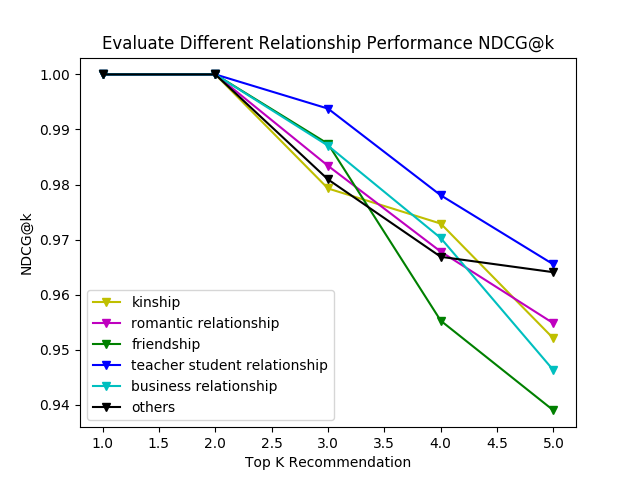
### Experiment Result

**Result of Evaluation on User Preference and System Performance**



**Figure 4.5.2 MAP@k and NDCG@k Results on User Preference and System Performance**

**Result of Evaluation on User Preference for Different Relationship**



**Figure 4.5.3 MAP@k and NDCG@k Results for Different Relationship on RSTR**

# Conclusions

We proposed a relationship-scenario based trailer recommendation system for social articles using both relationship and scenario features of plot summaries. And we utilized Pattern and E-HowNet to build Relationship and Scenario Dataset. And we evaluate extroverted event and hidden event quality. And we compare different pre-trained model for downstream task.

Overall, our system outperformed W2V representation on both user preference and system performance evaluation.

**We suggest some issues which are worth to study for future works**

In near future, we can consider more complete storyline, because plot summaries do not convey all the information about a movie, and are sometimes ambiguous.

Using different Named Entity Recognition method for Event Extraction. And using all data fine tuning on Pre-trained BERT Model. We also can consider more different architecture for training.

# Reference

1. Hadi Pouransari, Saman Ghili , “Deep learning for sentiment analysis of movie reviews ”, 2015
2. Mihaela Sorostinean, Katia Sana, Sentiment Analysis on Movie Reviews , March 1, 2017
3. Gaurav Arora1, Ashish Kumar2, MOVIE RECOMMENDATION SYSTEM BASED ON USERS’ SIMILARITY, IJCSMC, Vol. 3, Issue. 4, April 2014, pg.765 – 770
4. Xuan-Son Vu† Seong-Bae Park†, Mining User/Movie Preferred Features Based on Reviews for Video Recommendation System, 2017
5. Jennifer Golbeck1, Generating Predictive Movie Recommendations from Trust in Social Networks, 2006
6. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. ICLR Workshop, 2013.
7. Kim, Yoon. 2014. Convolutional neural net- works for sentence classification. arXiv preprint arXiv:1408.5882 .
8. Extended-HowNet, <http://ehownet.iis.sinica.edu.tw/index.php>
9. Chen, K. J., Huang, S. L., Shih, Y. Y., & Chen, Y. J. (2005a). Extended-HowNet- A Representational Framework for Concepts. In Processing of OntoLex 2005 - Ontologies and Lexical Resources IJCNLP-05 Workshop, Jeju Island, South Korea.
10. Huang, S. L., Chung, Y. S., & Chen, K. J. (2008). E-HowNet- an Expansion of HowNet. In Proceedings of 1st National HowNet Workshop, Beijing, China.
11. CKIP Group (2013 reversion), Lexical Semantic Representation and Semantic Composition- An Introduction to E- How Net, CKIP Technical Report.
12. Dong Zendong, and Qiang Dong. 2006, HowNet and the Computation of Meaning. World Scientific Publishing Co. Pte. Ltd
13. Y.-M. Hsieh, M.-H. Bai, J. S. Chang, and K.-J. Chen. Improving pcfg chinese parsing with context-dependent probability reestimation. Proceedings of CLP’12, pp. 216-221. 2012.
14. Chinese Knowledge Information Processing Group Academia Sinica Institute of Information Science. Technical Report no. 93-05 中文詞類分析(三版)
15. Chinese Knowledge Information Processing Group Academia Sinica Institute of Information Science. Technical Report no. 13-01 句結構樹中的語意角色
16. Hoang, Q.: Predicting movie genres based on plot summaries. arXiv preprint [arXiv:1801.04813](http://arxiv.org/abs/1801.04813) (2018)
17. A. M. Ertugrul, P. Karagoz, Movie Genre Classification from Plot Summaries using Bidirectional LSTM, 12th IEEE International Conference on Semantic Computing (ICSC), 2018.
18. Justin Martineau and Tim Finin. 2009. Delta tfidf: an improved feature space for sentiment analysis. InProceedings of the Third Annual Conference on We-blogs and Social Media, pages 258–261.
19. Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Vol 1, pages 142–150.
20. Nguyen DQ, Nguyen DQ, Vu T, Pham SB (2014) Sentiment classification on polarity reviews: an empirical study using rating-based features. In: Proceedings of the 5th workshop on computational approaches to subjectivity, sentiment and social media analysis. Baltimore, pp 128–135
21. Maron, M. E. (1961). "Automatic Indexing: An Experimental Inquiry" (PDF). [Journal of the ACM](https://en.wikipedia.org/wiki/Journal_of_the_ACM). 8 (3): 404–417. [doi](https://en.wikipedia.org/wiki/Digital_object_identifier):[10.1145/321075.321084](https://doi.org/10.1145%2F321075.321084).
22. [Cortes, Corinna](https://en.wikipedia.org/wiki/Corinna_Cortes); Vapnik, Vladimir N. (1995). "Support-vector networks". [Machine Learning](https://en.wikipedia.org/wiki/Machine_Learning_(journal)). 20 (3): 273–297. [CiteSeerX](https://en.wikipedia.org/wiki/CiteSeerX) 10.1.1.15.9362. [doi](https://en.wikipedia.org/wiki/Digital_object_identifier):[10.1007/BF00994018](https://doi.org/10.1007%2FBF00994018).
23. Ho TK (1998). "The Random Subspace Method for Constructing Decision Forests"(PDF). IEEE Transactions on Pattern Analysis and Machine Intelligence. 20 (8): 832–844. [doi](https://en.wikipedia.org/wiki/Digital_object_identifier):10.1109/34.709601.
24. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. Google Scholar