Impression Estimation of Short Sentences and Images Using Adjectives

August 2016

Nguyen Thi Thu An

Abstract

Semantic association is an essential concept that is used in various fields such as artificial intelligence, natural language processing, information retrieval, relation extraction, document clustering and automatic data extraction. The work in this thesis focuses on developing a method which explores the possibility to find the semantic association strength between adjectives and words.

The proposed method first queries co-occurrence frequencies of the adjectives and keywords, and the lexical patterns (phrases connecting the adjective and the word) using templates and Google N-gram corpus. K-means clustering method is then employed to cluster similar lexical patterns. The semantic similarity scores computed by several modified computational measurements and the lexical pattern frequencies are used for the training not only to classify adjectives into two classes (association and non-association), but also to get the association scores. A two-class SVM is employed using vector features including pattern clusters and co-occurrence measures to classify association and non-association pairs.

In order to evaluate the efficiency of the proposed method and also examine the contribution on real applications, we proposed two applications: estimation of the impression of short sentences, and estimation of the impression of images. In both applications, for the first step, keywords are extracted. They are then used to measure the level of association with adjectives. After obtaining the association scores of keyword-adjective pair, a rank aggregation method, Borda's method, which is used to generate an acceptable ranking for a given set of adjective ranking list with each keyword is employed. Besides, keywords and adjectives' sentiment analysis are also used to estimate their sentiment orientations. The top n_a adjectives (in this thesis, n_a is 5) having the highest score and the same orientation with keywords are chosen according to the estimated values. The main contribution of this method is to design effective systems to estimate the impression of the sentences and images. We eval-

uated the proposed approach from two viewpoints: fundamental performance of the semantic similarity measurement, and the effectiveness of the proposed measurement in two applications. The evaluation for association classification on 5,000 pairs of words shows that the average accuracy is 82.0% for noun-adjective pairs and 78.0% for verb-adjective pairs. Also, we carried out subjective experiments and obtained 3.0 (5 levels of measurement from strongly disagree (1) to strongly agree (5)).

Acknowledgements

Firstly, I would like to express my sincere gratitude to my advisor Prof. Masafumi Hagiwara for the continuous support of my Ph.D study and related research, for his patience, motivation, and immense knowledge. My advisor was there throughout my preparation of the proposal and conceptualization of its structure. His guidance helped me in all the time of research and writing of this thesis. I would not have been able to do the research and achieve learning in the same manner without his help and support. His recommendations and instructions have enabled me to assemble and finish the dissertation effectively.I would also like to thank my committee members, professor Hiroaki Saito, professor Akito Sakura, professor Michita Imai for serving as my committee members even at hardship. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions, thanks to you.

I would also like to thank all of my instructors and teachers, who throughout my educational career have supported and encouraged me to believe in my abilities. They have directed me through various situations, allowing me to reach this accomplishment.

I thank my fellow labmates for the stimulating discussions, supports and for all the fun we have had in the last five years. Also I thank my friends in the school who are always by my side in any condition and situation.

In addition, I like to thank the participants in my survey, who have willingly shared their precious time during the process of interviewing.

I also thank my friends - Luisa Gonzlez, Nene Sandvicensis, Tri Nguyen, Quy Nguyen, Thu Nguyen, Hue Le and my students in Can Tho University who have supported spiritually and helped me along the course of this dissertation, and my life in general. I would like to thank Janaka Wijekoon, Kazuki Tanida, Rajitha

Tennekoon, Luisa Gonzalez, and Arnaud Rachez for being supportive and for helping me with proofreading my thesis.

Last but not least, I especially thank my mom, dad, and my brother for always believing in me and encouraging me to follow my dreams. I know I always have my family to count on when times are rough.

To my parents.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Impression of text or visual content has recently attracted a great deal of attention. It has been widely investigated along various aspects over the past few decades. One of the most popular studies is sentiment analysis or opinion mining, which aims to identify and categorize subjective opinions expressed in news, stories, texts, or reviews. Besides text-based usage, recently social media users are increasingly using more and more images and videos to express their opinions and share their experiences. Sentiment analysis of such large scale visual contents can contribute to extract user sentiments toward events or topics. Thus, estimation of sentiment from visual contents is complementary to textual sentiment analysis [2–4]. However, such studies still limit the variety of expressions that individuals can use because the selectable impression words are restricted to the polarity (positive, negative or neural), intensity (degree to which a sentiment is positive or negative) of the sentiment, and emotions (sad, happy, angry, etc.). Therefore, it is important to provide a flexible human interface to deal with a greater diversity of impression words.

In this thesis, we address a particular aspect of Kansei engineering, estimation of impression, i.e. impressions which are experienced when reading texts or observing images. In fact, this work is highly subjective and difficult to explain, so it is very

necessary to address the issue of how to map textual and visual content on impression words. In order to solve the problem, first we propose a method to measure how strong the connection between adjectives and words is. This approach is further utilized to estimate the impression of sentences and images.

This chapter will first describe the objectives and motivations of this thesis. We will then answer why it is a challenge and what we have achieved. The outline of this thesis is finally given.

1.1 Objective

This thesis proposes a method for the automatic estimation of impression of words, sentences and images. First, the possibility of finding the semantic association between adjectives and words is mainly explored. Then, we address three sub-goals: adjective-word pair collection, adjective-word lexical pattern collection, and adjective-word pair semantic similarity measurement. Specifically, these tasks are defined as follows:

- **Adjective-word pair collection** aims to create associated pairs and non-associated pairs.
  - Association pairs: pairs of a word and an adjective which arise in human's mind when reading or saying the word, or just thinking about the word.
  - Non association pairs: pairs of a word and an adjective that tend not to be associated to the words.
- **Adjective-word lexical pattern collection** addresses gathering connection phrases between words and adjectives which provide valuable information indicating their semantic relationships in the local context. For example, considering the text "*flower is the most beautiful*" obtained by Google N-gram for

the query "*flower*"and "*beautiful*", the phrase "*is the most*" shows a semantic relationship between "*flower*" and "*beautiful*".

- **Adjective-word pair semantic similarity measurement**: which measure the strength of association between two words by comparing the corpus-level N-gram frequency of a word pair to some function of the unigram frequencies of the individual words.

We also propose an application exploiting the above proposed method to estimate the impression of sentence and image. The goal is to develop a framework that can automatically estimate the impression of sentences and images using adjectives. To achieve the target, this thesis address the following issues: keyword extraction, impression word estimation, and impression word selection are considered. The following definitions explain the targets:

- **Keyword extraction** aims to find the most important information of the sentence or image.
- **Impression word estimation** addresses the adjectives related to image or sentence, and measures the level of association between them.
- **Impression selection** outputs the best match adjectives.

1.2 Motivation

1.2.1 Impression estimation in text

The initial motivation for working with impression estimation of a sentence was originated from the need for a better conversational agent. Although natural language processing has rapidly advanced, especially in the last three decades, dialog processing has not yet achieved a revolutionary impact on the society. Since most of the

conversational systems have limits in terms of sense, feeling, and emotion, it is still very difficult to give users a real chatting impression compatible to have a conversation with human. Hence, the problems have been the subject of numerous studies, and a number of solution techniques have been proposed [5,6]. Desiring to contribute to address these issues, this thesis presents some tasks to illustrate new challenges and opportunities for this subject.

1.2.2 Impression estimation of image

Image impression understanding plays a very important role in the social interaction on the Internet. Recently, common social web environments like Facebook, Twitter, or Flickr have been becoming more and more popular. They contain a huge amount of uploaded images which can evoke strong impressions, emotions, or feelings in the viewers. By finding ways to give the artificial systems the ability to be empathic using adjectives, we believe that the interaction between users and computers becomes much more natural.

Indeed, in the last few years, there has been a growing interest in using language to aid image understanding or to express emotions. Many researches have proposed to combine annotated tags associated with images and computer vision techniques to generate textual description [7]. Although these texts are able to explain the content of the images accurately, studies on the issue of the impression of the image are still lacking.

Thus, in this research, we address the challenge of quantifying *"What comes into our mind after seeing a photo?"* as a goal.

1.2.3 Semantic similarity

Semantic similarity based on the likeness of meaning or semantic content of words plays a vital role in natural language processing. Quite recently, this field of research

has been utilized in many applications including information retrieval, text understanding, paraphrase recognition, text summarization and annotation, and lexical selection [8,9]. However, even though the study of semantic similarity measurements has attracted considerable attention, the approaches only focused on word-to-word and item-to-item belonging to the same part of speech or concepts. In addition, it has been shown that current research of semantic similarity measures in a text understanding study leads to the inability of giving users a real feeling of the text in terms of sense, emotions, or impressions.

Therefore, proposing a solution to improve the drawbacks of current studies is not only desirable but also necessary. Moreover, we make use of the knowledge given by the associations between words and adjectives providing impressions that help interpretation of texts and images. The sub-goals of recognition that we address in this thesis are collection, measurement and selection. Wishing to make a contribution to improve important applications such as a conversational agent, image understanding or image impression estimation.

1.3 Approach

1.3.1 Impression estimation of a short sentence and an image

Natural and intuitive communication with a computer is a primary research goal in human-computer interaction. In recent years, this goal has frequently led to the employment of humanoid agents as interlocutors that are able to both understand and use communicative means to human users. One of the important tasks to enhance current systems is taking adjectives as impression words into account. Particularly, in order to make the conversation natural, responses using adjectives associated to the texts or images are often used. For example, in a conversation, if someone heard that *"it snows"*, normally they would think that *"outside, it is very cold"* or *"the city*

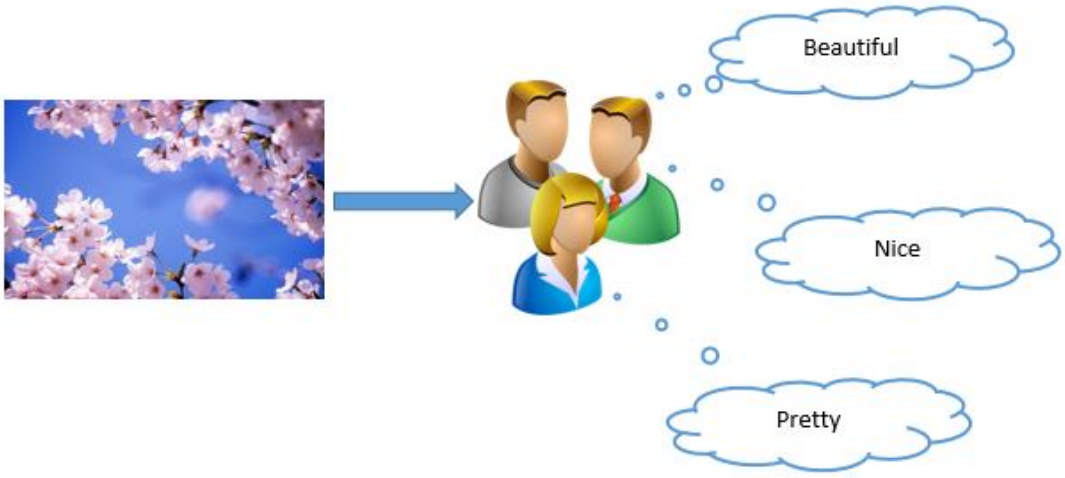


Figure 1.1: Image impression attracts viewers.

is covered with white color". If someone says *"summer is coming"*, the listener would imagine that *"it will be hot or humid"*. In fact, adjectives are used not only in texts, but also popular in showing the impression of images. Referring status by a popular English idiom "*A picture is worth a thousand words*"is an English idiom [10], a single image conveys the meaning or essence more effectively than words do. Images are considered important in attracting attentions and making someone have a particular feeling or emotion. Particularly, the beauty of cherry blossoms shown in in Figure 1.1 is intended to invoke viewers' comments like "*it is so beautiful*"and "*it is so pretty*". Thus, we attempt to adopt a method to show the impression estimation of using adjectives. We believe that in a conversation agent, this research would help answer such a question what is the impression given by texts or images.

In this study, in Chapters 4 and 5, we focus on showing the impression of the texts and images after the system viewing. The main goal of this work is exploring the possibility of utilizing semantic similarity measurements between adjectives and words (nouns, verbs) for the task of impression estimation. The proposed system discovers keywords (general content) in input such as sentences or images. Then, in

order to estimate the impression, measurements of the association between adjectives and the keywords using machine learning algorithm is shown.

1.3.2 Semantic similarity between an adjective and a word

Previous studies [11–15] indicate that the adjective is one of the most important elements in a sentence for sentiment analysis. Adjectives are characterized as expressions or descriptions of feelings, and opinions which are very important in speaking and writing. They also help people understand others' opinions. However, they have not attracted much attention from researchers. There exist only a few works addressing adjectives specifically and most of them have focused either on specific applications such as sentiment analysis, semantic orientation of adjectives or on specific types of adjectives. Therefore, we take the approach of using adjectives showing the impression. We consider lexical patterns which connect adjectives and words as feature vectors, aiming to learn from a set of training association pairs, and the non-association pairs. We focus on two particular types of words - nouns and verbs using Google N-gram corpus to get the frequency and patterns of pairs as well. To this end, we employ SVM machine learning techniques, specifically various discriminative classifiers to get the scores showing the level of connection between adjectives and images or texts.

1.4 Contribution

The primary contributions of this thesis are:

- Proposing a new concept for the association between adjective and forms of words. In our thesis, we mainly focus on the relationship between noun-adjective, and verb-adjective.

- Proposing a new method to measure the association strength between words and adjectives using machine learning algorithm and the similarity measurements.
- Applying the similarity measurements to two applications: estimations of the impression of short sentences and images.
- Combining an image processing method and an image annotation method to discover the topic of the image and proposing an adjective-based system to estimate the impression of the image.

1.5 Organization of the Thesis

The organization of this thesis is as follows:

- **In Chapter 2**, we first present a literature review on semantic similarity measurements. We give a short overview of the methodologies, and challenges to measure the similarity. Next, we discuss the benefits of the usage of adjectives in Kansei engineering research. Major challenges for this research are also introduced at the end of this chapter.
- **In Chapter 3**, we describe the proposed methods. An overview of common methods to preprocess and analyze the data to identify patterns, or features which are related to the impression of texts or images. Besides basic preprocessing measures, widespread methods for feature extraction are explained. Finally, the impression adjectives are briefly output and described.
- **In Chapter 4**, an iterative implementation is applied on short sentences. The evaluation of the developed system by participants is explained.
- **In Chapter 5**, an impression estimation system relying on the proposed method is proposed. Together with the experimental results, we conducted different several evaluations to show the performance of the proposed system.

- **In Chapter 6**, we discuss the conclusions, highlight our contributions and describe further research directions.

Chapter 2

Related Works

Impression estimation is an open research issue, relevant for numerous natural language processing (NLP) and computer vision [16, 17]. In this chapter, we will review the most recent and significant works in the literature on impression estimation of both sentences and images. Also, semantic similarity measurement which is our main method in this research, is discussed as well. First, in Section 2.1, we briefly highlight some of the works regarding impression in texts which are closely related to concepts like sentiment analysis and emotion, followed by discussing their approaches and models and also their weaknesses. In Section 2.2, we continually pay special attention to studies focused on impression analysis of images evoked by visual contents and their limitations in usages which aim at matching the model with the perception humans have. Last but not least, we discuss different ways to measure the semantic similarity between words. Pioneering works used taxonomy, corpus-based, and web-based approach with supervised learning methods. The problem of these methods has been studied in a text processing field in many years. Later works proposed to use lexical patterns together with previous proposed method are explained. The purpose of these methods is to consider lexical semantic association between words. These methods are reviewed in Section 2.3.

2.1 Impression of a sentence

Impression estimation of a short sentence which focuses on understanding human's impressions with features characterizing certain texts has received much attention. Existing approaches in the context of impression estimation can be broadly divided into several categories. The following categories are closely related to our work: sentiment analysis and emotion analysis.

2.1.1 Sentiment analysis

The most related attempt to recognize impression using textual data and a categorical model focused on determining the semantic orientation of words using two base categories: positive and negative. It generally has two research directions: lexicon-based approaches and machine learning approaches.

Lexicon-based approaches

The approaches [18–20] use the positions of words, and linguistic analysis to discover the patterns to determine the polarity of opinions. For lexicon-based approaches, a set of words labeled with sentiments is often required. The work for automatically generating the set of words can be categorized as corpora-based approach [21] and thesaurus-based approach [8]. They first proposed corpora-based word level sentiment analysis. Adjectives are extracted from a large document corpus by using conjunction rules. Turney [22] took into account the other parts-of-speeches which are considered to be responsible for the expressiveness, such as adverbs. Therefore, based on the concept that a positive semantic orientation denotes praise (e.g. "honest") and a negative semantic orientation indicates criticism (e.g. "disturbing"), their goal was to determine the intensity (mild or strong), and a term brings to the class determined by valence (positive or negative). They first defined a set of positive seed terms and

negative seed terms, then searches the target term and seed terms to measure their point-wise mutual information (PMI). The orientation of the target term is the sum of weights of its semantic association with positive seed terms minus that with negative seed terms. This study raised criticism due to language variation, words potentially having multiple meanings [23]. For the thesaurus-based word level sentiment analysis, Kim and Hovy [20] expanded seed sentiment words on WordNet with synonym and antonym relations. The polarity of a term is determined by observing the number of its neighbors that are positive or negative. Refs. [8, 24] assigned a polarity or a strength to subjective expressions (words and phrases that express opinions, emotions, sentiments, etc) by exploiting orientation of adjectives. In order to decide the objectivity or subjectivity orientation of a document [25, 26] or the positive, negative and neutral polarity of an opinion sentence within a document, additional work has focused on the strength of an opinion expression where each clause [27] or phrase [28] within a sentence can have a neutral, low, medium or a high strength. Esuli and Sebastiani [29] built ternary classifiers on the WordNet synsets, a small set which are manually labeled and extended into the final training sets. Kamps et al. [30] link terms on the WordNet with synonym relationships to generate a graph, with the polarity of a term computed by measuring its shortest distance to "*good*"and "*bad*".

The machine learning approach

Leonid Velikovich [31] proposed a fully unsupervised method, which aimed at building large polarity lexicons semi-automatically from the web using a graph propagation algorithm. Another approach [23] took a large amount of web data, a corpus of 200,000 online reviews, in order to try and determine the impact of higher-order n-grams and the most suitable classifier for estimation of the reviews' polarity. Higher-order n-grams are required in polarity analysis due to the larger context they can capture. Both generative and discriminative classifiers were taken into account for categoriza-

tion of text fragments into positive and negative, their estimation accuracy being influenced by the features given as input. For the generative approach, a language modeling based classifier (CMU Cambridge Language Modeling Toolkit [32]) was considered, whose estimation is based on the probability of generating a word sequence, while the discriminative approach has been evaluated using the passive aggressive algorithm [33], which is a margin based online learning classifier. Winnow [34] was the third classifier under testing, which is an online linear classifier for sentiment analysis. Due to the significant differences between classifiers, some preliminary characteristics should be signaled. For example, the generative classifier may present data sparseness due to the limitation of training data, while the discriminative one is very adequate to large data because it uses an online learning pattern and it has a theoretical loss bound which makes its performance predictable. Lun Wei Ku, Yong Sheng Lo, and Hsin Hsi Chen [35] used polarity scores of words for sentence-level opinion extraction. In this subject, the adjective keeps an important role and also is used as a method to check the polarity [6]. [36, 37] used the adjective to identify opinion polarity. The method extracts adjectives and their frequencies from the given review, predicts the polarity of each adjective using the learned classier, and classifies the review based on the polarity of the adjectives.

There have been attempts of utilizing different approaches with encouraging results, though such approaches are, so far, limited to using binary or ternary word impression.

2.1.2 Emotion analysis

This part reviews some emotion analysis studies. The concept that each word has a given emotional state that can vary upon context and the large amount of available textual data, two major text-based emotion representation models have emerged. The first model consists of a dimensional representation of each word, consistent with the

psychological studies, and the second one is based on a categorical model, which is more suitable for computational linguistics, consisting in generating subsets of words around certain associated labels. Most notable categorization models are the following: Ekman's six basic emotions [38] (anger, disgust, fear, joy, sadness and surprise), subjectivity (subjective vs. objective), polarity (positive vs. negative vs. neutral) and stubbornness (opinionated vs. non-opinionated). The categorical model benefits from the existing emotional thesaurus, like WordNet-Affect [39], ConceptNet [40] and SentiWordNet [29], but it can also generate important affective word lists using techniques like bag-of-words, keyword spotting and lexical affinity. SemEval-2007 Task 14 introduced a task named "Affective text" [41]. The task aims at the emotion classification of news headlines which are commonly written with the intention to "provoke" emotions so as to attract the attentions. In this task, given the characteristics of news headlines, the indirect emotional words which depend on the context (e.g. words that imply possible emotional causes such as "killer" or emotional responses such as "cry") were taken into account. Being similar to the task of SemEval-2007, the influence of indirect emotional words was also taken into consideration. They chose words, polarity of subject, verb and object of the sentences and semantic frames as feature sets to classify the sentences into four categories: disgust, fear, happiness and sadness. Bao et al. [17] proposed a joint emotion-topic model to predict the emotion. The model connected emotions with the terms via a set of latent topics. Through comparing with the emotion-term model, the experimental results showed the emotion-topic model improved the estimation performance significantly.

At present, the problems and deficiencies of the research about emotion estimation still exist. One is that previous researches on emotion analysis restrict use of a variety of expressions concerning each text because the selectable impression words are fixed on the six main emotions: happy, joy, sad, angry, surprise, and amuse. To achieve

information systems based on emotion or impression, it is important to provide a flexible human interface to deal with greater diversity of impression words.

2.2 Impression estimation of an image

Recently, uploading and sharing photos have been becoming one of the most popular activities on sociable websites such as Facebook, Google+, etc. These activities open up researches of understanding images which have been gaining importance and attracting considerable attention. So far, most of the previous works on image processing concentrated on emotion expressions from images, and description generation [7, 10].

2.2.1 Emotion expression from visual content

Affective computing or sentiment analysis plays an important role in behavioral sciences, which aims to understand and predict human decision making [42] and enables applications. As mentioned in section 2.1, sentiment and emotion analysis are a very challenging task. Researchers from natural language processing and information retrieval have developed different approaches to solve this problem, achieving promising or satisfying results. So far, the computational analysis of sentiment mostly concentrates on the textual content [8]. However, in the context of social media, there are several additional unique challenges. First, there are huge amounts of data available. Secondly, messages on social networks are by nature informal and short. Thirdly, people use not only texts, but also visual sentiment analysis. Siersdorfer [43] proposes a machine learning algorithm to predict the sentiment of images using pixel-level features.

In predicting the impression, emotion or responses evoked by an image, many researchers have proposed various methods. These methods analyze several levels of

image semantics. From the highest level to the lowest level, this hierarchy can be described as: 1) Abstract semantics, which contributes to our interpretation of the senses (e.g. like, happy) 2) Semantic templates (i.e. semantic categories), which contribute to our accumulated semantic knowledge (e.g. blue color induces sad feelings). 3) Semantic indicators, (i.e., image elements, which are characteristic for certain semantic categories), for example, the large blue region or the dark region. 4) Low level image features, i.e. measurable image attributes. Based on the above semantic model, we can understand some general semantic templates and their indicators of the images, and implement feature detection algorithms to capture these properties as well. The relationship between abstract semantics and semantic templates can be bridged by human experience and knowledge through users' interactive or predefined knowledge base. Motivated by the fact that sentiment involves high-level abstraction, which may be easier to explain by objects or attributes in images, both Borth [3] and Yuan [4] proposed the methods to employ visual entities or attributes as features for visual sentiment analysis. In these studies, 1,200 adjective noun pairs (ANP), which may correspond to different levels of different emotions, are extracted. These ANPs are used as queries to crawl images from Flickr. Next, pixel-level features of images in each ANP are employed to train 1,200 ANP detectors. The responses of these 1,200 classifiers can then be considered as mid-level features for visual sentiment analysis. The results obtained by refs. [2, 44] suggest that color analysis is very important to express specific emotions. Ref. [45] proposed Kobayashi theory on the association of adjectives and color schemes of images. The research indicated that the associations of colors of images to image words are based on the semantic axes e.g., cool - warm and soft - hard. Many researches [46, 47] adopted this theory to build the relation between affective words and color themes. However, these approaches may fail when the emotional semantic is carried by objects in images, such as a child crying, or a whale dying on a beautiful beach. Therefore, additional information is needed. An-

(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 2.1: Examples of black white photos.

other solution is described in [48] which adopted machine learning method to achieve emotional semantic identification based on the effects of the extracted color, texture and shape features of image. Recent approaches have also turned towards web portals like Flickr and YouTube as information sources for visual learning, employing user generated tags as an alternative to manual labels [49, 50].

These approaches are more accurate than the other methods; however, the performance depends heavily on the accuracy of image processing technique. In addition, most of the previous studies have not taken into account the analysis of the topic of the images whose emotional expressions are mainly focused. For example, in Figure 2.1, the white and black colors show the sad emotion of the image in general, but if we consider the topic of the image, the shot focusing on face, the sad emotion expression seems to be wrong in this situation.

2.2.2 Description generation

Refs. [10, 51–57] proposed systems to automatically generate natural language descriptions from images. A number of approaches pose the task as a retrieval problem, where the most compatible annotation in the training set is transferred to a test

image [58], or where training annotations are broken up and stitched together [59]. Several approaches generate image captions based on fixed templates that are filled based on the content of the image [60] or generative grammars [61], but this approach limits the variety of possible outputs. Compared with content-based image understanding, annotation based one is more practical in some applications. Information on users and the semantic contents of images are represented in textual information by users more accurately and precisely. Owing to the advantages of the user annotation tags, recently, Ningning Liu, etc. [16] proposed a method to make use of textual information describing the image that is often provided by photo management and sharing systems. Even though, they could approach text processing into image description to enhance the performance, usage of semantic distance matrix between the text and emotional dictionary leads a problem that it relies on the notion that all links in the taxonomy represent a uniform distance.

Although these descriptions can be useful for a variety of applications, including image retrieval, and automatic video surveillance, the resulting descriptions often lack of the kind of impression, emotion, and feeling typically found in human thinking after viewing an image. Very few works deal with the affective level which can be described as identifying the emotion that is expected to arise in humans when looking at an image, or called affective image classification.

2.3 Association measurement

As introduced in the previous chapter, semantic relatedness, which is our main method in this thesis, has been a very active research field in natural language processing, hence many approaches have been proposed to solve this problem [62–64]. These different approaches can be classified in several different ways and from dif-

ferent points of views, however, in this scope of this dissertation, we consider the following 3 main approaches:

- Measures that exploit WordNet's semantic network.
- Corpus-based measures of distributional distance.
- Web Search Engine-based content measures.

2.3.1 Measures that exploit WordNet's semantic network

WordNet [65,66] is described by its creators as "a large, electronically available, lexical database of English". It is a semantic network in which each node, called a synset, represents a fine-grained concept or word sense. Each synset is composed of a gloss and a set of near-synonymous words which refer to that concept. The synsets are connected by lexical relations such as hyponymy, meronymy, and so on. The shortest path in the network between the two target concepts (target path) is determined. The more edges there are between two words, the more distant they are. Elegant as it may be, the measure hinges on the largely incorrect assumption that all the network edges correspond to identical semantic distance.

Nodes in a network may be connected by different kinds of lexical relations such as hyponymy, meronymy, and so on. Edge counts apart, [67] takes into account the fact that if the target path consists of edges that belong to many different relations, then the target concepts are likely more distant. The idea is that if we start from a particular node c1 and take a path via a particular relation (say, hyponymy), to a certain extent the concepts reached will be semantically related to $c1$. However, if during the way we take edges belonging to different relations (other than hyponymy), very soon we may reach words that are unrelated. The measure of semantic relatedness is below:

$$HS(c1, c2) = C - path\,length - k \times d \tag{2.1}$$

where $c1$ and $c2$ are the target concepts, d is the number of times an edge pertaining to a relation different from that of the preceding edge is taken, and C and k are empirically determined constants.

More recently, Yang and Powers [68] proposed a weighted edge-counting method to determine semantic relatedness using the hypernymy/hyponymy, holonymy/meronymy, and antonymy links in WordNet. Leacock and Chodorow [64] used just one relation (hyponymy) and modified the path length formula to reflect the fact that edges lower down in the *is-a* hierarchy correspond to smaller semantic distance than the ones higher up. For example, synsets pertaining to sports car and car (low in the hierarchy) are much more similar than those pertaining to transport and instrumentation (higher up in the hierarchy) even though both pairs of nodes are separated by exactly one edge in WordNet's *is-a* hierarchy. The similarity measure is:

$$-\log\left(\frac{length}{2\times D}\right) \tag{2.2}$$

where $length$ is the length of the shortest path between the two concepts (using node-counting) and D is the maximum depth of the taxonomy.

Resnik [69] suggested a measure that combines corpus statistics with WordNet. He proposed that the related value is equal to the information content (ic) of the least common subsumer (LCS) (most informative subsumer):

$$Res(c1,c2) = ic(lso(c1,c2)) = -\log P(lso(c1,c2)) \tag{2.3}$$

where $lso(c1,c2)$ is the lowest common subsumer of synset c1 and synset $c2$ and ic is the information content.

This means that the value will always be greater-than or equal-to zero. The upper bound on the value is generally quite large and varies depending upon the size of the

corpus used to determine the information content values. To be precise, the upper bound should be $\ln(N)$ where N is the number of words in the corpus.

Observe that usage of information content has the effect of inherently scaling the semantic similarity measure by depth of the taxonomy. Usually, the lower the lowest super-ordinate, the lower the probability of occurrence of the lso and the concepts subsumed by it, and hence, the higher its information content is.

As Resnik's formula, given a particular lowest super-ordinate, the exact positions of the target nodes below it in the hierarchy do not have any effect on the semantic similarity. Intuitively, we would expect that word pairs closer to the lso are more semantically similar than those that are distant. Jiang and Conrath [70] and Lin [71] incorporate this notion into their jcn measures which are arithmetic variations of the same terms. The similarity value returned by the jcn measure is equal to $jcn\_distance$, where $jcn\_distance$ is equal to $ic(concept1) + ic(concept2) - 2 \times ic(lcs)$.

There are two special cases that need to be handled carefully when computing similarity; both of them involve the case when $jcn\_distance$ is zero.

In the first case, we have $ic(concept1) = ic(concept2) = ic(lcs) = 0$. In an ideal world, this would only happen when all three concepts, $concept1$, $concept2$, and lcs, are the root node. However, when a concept has a frequency count of zero, we use the value 0 for the information content. In this first case, we return 0 due to lack of data.

In the second case, we have $ic(concept1) + ic(concept2) = 2 \times ic(lcs)$. This is almost always found when $concept1 = concept2 = lcs$ (i.e., the two input concepts are the same). Intuitively this is the case of maximum similarity, which would be infinity, but it is impossible to return infinity. Instead we find the smallest possible distance greater than zero and the multiplicative inverse of that distance is returned.

The final semantic distance between the two concepts is then taken to be the sum of these differences. Lin [71] (like Resnik) points out that the lso is what is

common between the two target concepts and that its information content is the common information between the two concepts. The similarity value returned by the Lin measure is the number equal to $2 \times ic(lcs)/(ic(concept1) + ic(concept2))$. Where $ic(x)$ is the information content of x. One can observe, then, that the similarity value will be greater than or equal to zero and less than or equal to one.

If the information content of any of either *concept*1 or *concept*2 is zero, then zero is returned as the similarity score, due to lack of data. Ideally, the information content of a concept would be zero only if that concept were the root node, but when the frequency of a concept is zero, we use the value of zero as the information content because of a lack of better alternatives.

Budanitsky and Hirst [9,72] show that the Jiang-Conrath measure has the highest correlation of noun pairs (0.85) with the Miller and Charles noun pairs and performs better than all of the other measures considered in a spelling correction task. Patwardhan et al. [73] get similar results using the measure for word sense disambiguation.

Many researchers have explored the similarity of nouns using a variety of methods including methods based on WordNet. Nonetheless, little attention has been paid to verbs, there is no standard evaluation set, and it is not clear that the WordNet verb hierarchy is rich enough to support verb similarity assessment. Thus, Refs. [68, 69] presented and adapted a successful noun similarity method based on WordNet to the verb similarity task by cross mapping into the noun hierarchy and back.

However, the limitation of these research is that it relies heavily on the notion that all links in the taxonomy represent a uniform distance and most of taxonomies are built for groups of words having the same part of speech tagging only (noun-noun, verb- verb, adjective-adjective, or adverb-adverb). Although strong conclusions cannot be drawn, since comparison with the corresponding corpus based quantities is missing, it is still apparent that contextual similarity measures have a tendency to detect semantic relations beyond mere synonym.

2.3.2 Corpus-based measures of distributional distance

Corpus-based measures of word semantic similarity try to identify the degree of similarity between words using information exclusively derived from large corpora. Some popular metrics [74–76], namely: (1) cosine similarity - TF-IDF, (2) pointwise mutual information [22], and (3) latent semantic analysis (Landauer, Foltz, & Laham 1998) [77],

Cosine Similarity and TF-IDF

One of the most commonly used similarity measure is cosine similarity [74,78], which is used as their baseline in their studies. In the approach, each document (or a sentence) is represented using a vector space model. The cosine similarity between two vectors (D_1, D_2) is:

$$\cos(D_1, D_2) = \frac{\sum_i t_{1i} t_{2i}}{\sqrt{\sum_i t_{1i}^2} \times \sqrt{\sum_i t_{2i}^2}} \tag{2.4}$$

where $t_1 i$ and $t_2 i$ are the term weight for a word w_i in D_1 and D_2 respectively, for which we use the TF-IDF (term frequency, inverse document frequency) value, as widely used in information retrieval. The IDF weighting is used to represent the specificity of a word: a higher weight means the word is specific to a document, and a lower weight means the word is common across many documents. IDF values are generally obtained from a large corpus. One widely used method for the IDF value for a word w_i is

$$IDF(w_i) = \log(N/N_i) \tag{2.5}$$

where N_i is the number of documents containing w_i in a collection of N documents.

In Ref. [79], Murray and Renals compared different term weighting approaches to rank the importance of the sentences (simply based on the sum of all the term weights in a sentence) for meeting summarization, and showed that TF-IDF weighting is

competitive. Therefore, Ref. [80] uses TF-IDF for term weighting and focuses on the problem of how to calculate the similarity between two documents in the Maximum Marginal Relevance (MMR) framework.

Latent Semantic Analysis (LSA)

LSA [77, 81, 82] is the most popular technique of Corpus-Based similarity. LSA assumes that words that are close in meaning will occur in similar pieces of text. A matrix containing word counts per paragraph (rows represent unique words and columns represent each paragraph) is constructed from a large piece of text and a mathematical technique called singular value decomposition (SVD) is used to reduce the number of columns while preserving the similarity structure among rows. Words are then compared by taking the cosine of the angle between the two vectors formed by any two rows.

2.3.3 Web search engine-based measures

Normalized google distance (NGD)

NGD [83–85] is a semantic similarity measure derived from the number of hits returned by the Google search engine for a given set of keywords. Keywords with the same or similar meanings in a natural language sense tend to be "close" in units of *Google distance*, while words with dissimilar meanings tend to be farther apart. Specifically, the *Normalized Google Distance*(NGD) between two search terms x and y is :

$$NGD(x,y) = \frac{\max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log M - \min\{\log f(x), \log f(y)\}} \tag{2.6}$$

where M is the total number of web pages searched by Google; $f(x)$ and $f(y)$ are the number of hits for search terms x and y, respectively; and $f(x,y)$ is the number of web pages on which both x and y occur. If the two search terms x and y never

occur together on the same web page, but do occur separately, the Normalized Google Distance between them is infinite. If both terms always occur together, their NGD is zero.

Actually, as the authors indicate, the discussion about the Google Distance is independent of the particular search engine they use to access the Web. Different search engines use different indexes and retrieval methods, thus providing different results in page counts. Jorge Gracia and Eduardo Mena [84–86] tried to apply this method into existent web search engines, in order to compare their behaviors.

Web-based approach measures

Some of the other researchers define the semantic relatedness between the words using Web. Various measures have been proposed to compute the degree of association of objects of different terms and documents from different corpora as resources. The proposed measures can work with 2 variables. An application and usage of two variables are: Islam and Inkpen [87] introduced Partial Mutual Information (PMI) as a measure of semantic similarity using the British National Corpus (BNC) [88]. Some researchers define the semantic relatedness between words using Web. Mehran Sahami, Timothy D. Heilman [89] has proposed a method that exploits page counts and text snippets returned by a Web search engine to measure semantic similarity between words. Michael Strube and Simone Paolo Ponzetto [90] also investigated the use of Wikipedia for computing semantic relatedness measures. Ref. [91] has determined the semantic similarity by a number of information sources which consist of structural information from a taxonomy and information content from a corpus. The most interesting work is certainly proposed by Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka [92]. They used the World Wide Web as the database, and Google as the search engine. They first found relationships between synonymous pairs based on snippet counts and then they applied a support vector machine classifier to de-

termine whether a new pair shows a relation of synonymy or not based on a feature vector of lexical relationships.

Although, they got convincible results, the research just tackles the relatedness between pairs of words having the same part of speech which seems not sufficient if we consider realistic applications.

Chapter 3

Semantic Similarity between Adjective and Different Forms of Words

In this chapter, we propose a new concept of association based on previously proposed measures to connect the concepts of adjectives with nouns and verbs. We also propose a new approach for automatically estimating the impression of words. Our approach is based on utilizing semantic similarity measurements and lexical pattern between words, which allows estimating the semantic similarity between different types of words and adjectives. The proposed measurement investigates the association between adjectives and words in the corpus in order to offer an impression of a given word.

The rest of this chapter is organized as follows. In Section 3.1, we present some background information. In Section 3.2, we introduce the proposed measurement for computing the semantic relatedness between adjectives and various types of words. An evaluation of the effectiveness of the method is presented in section 3.4. Finally, the chapter is concluded in Section 3.5.

3.1 Background

This section provides a background of our work that helps the rest of the thesis make sense. Section 3.1.1 gives a short description of Google N-gram corpus, stopword removal and part of speech tagging. Section 3.1.2 presents some overview of technical preliminary about NLP. Section 3.1.3 mentions some concepts about semantic relatedness and sentiment.

3.1.1 Google *N*-gram corpus

In the fields of statical computational linguistics and statistics, an N-gram is a sequence of N consecutive items appearing in a text. The items can be phonemes, syllables, letters, or words according to the application. The N-gram typically are collected from a text or speech corpus. In some applications (e.g. defining product reviews polarity) bigrams and trigrams show better polarity classification. Two main types of N-grams are: character level N-grams, and word level N-grams. Character level N-grams use each character of the string as a token. Word level N-grams use each word of the string as a token. The number of N-grams in a set can be calculated as following (Urbansky [93]):

$$Ngrams = \#Tokens - N + 1. \tag{3.1}$$

Google N-gram [94] is a collection of N-grams from web pages contributed by Google Inc. It contains English word N-grams and their observed frequency counts. N can range from 1 to 5, so the maximum string that can be analyzed is five words long such as *"was as long as possible"*, *"is a nice time to"*. The Google N-gram Corpus is used throughout this study.

3.1.2 Stopwords removal

Stopwords removal is a process of removing words that may have little lexical meaning, or may not change the semantics of a text or a query. They are usually articles, prepositions, pronouns, etc. Common English stopwords include: the, on, of, with, a, about, what, when, where, that, this, by, be, etc. The reason why stop words are critical to many applications is that they have no important contribution to the text representation, and low discriminative potential to differentiate instances from different classes.

3.1.3 Part of speech tagging

- **Tokenization** is a process of breaking up a stream of text into words, phrases, symbols, or the other meaningful elements called tokens. The list of tokens becomes input for further processing such as parsing or text mining. It is useful both in linguistics (where it is a form of text segmentation), and in computer science, where it forms part of lexical analysis.

- **Part of speech (POS) tagging** is a special application of natural language processing. A POS is a category used in linguistics that is defined by a syntactic or morphological behavior of a word. The traditional English language grammar classifies parts-of-speech in the following categories: verb, noun, adjective, adverb, pronoun, preposition, conjunction, and interjection. However, some other sources admit more categories, like articles or even based on variations of the aforementioned ones. The reason why POS tagging is so important to information extraction is the fact that each category plays a specific role within a sentence. For instance, nouns give name to objects, beings or entities from our world, verbs give actions, and an adjective qualifies or describes nouns, etc. The following example (Figure 3.1) shows a tagged sentence:

– *A lot of trees were blown down in the recent storms*

After classifying POS of words, tagged sentence is

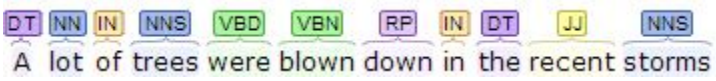


Figure 3.1: A tagged sentence is classified by POS [1].

where DT is determiner; IN is preposition; NN: noun; NNS: noun, plural; VB: verb base form; VBD: verb past tense; VBN: verb past participle; RP: particle; and JJ: adjective.

- **Stanford Parser** [1] is a probabilistic natural language parser; it works out the grammatical structure of sentences and uses knowledge of language gained from hand-parsed sentences to try to produce the most likely analysis of new sentences. In this study, we use this parser to analyze sentences.

3.2 Semantic similarity between adjectives and inputs

As stated in Chapter 2, current approaches for semantic similarity face significant problems to narrow the semantic gap (i.e the mismatch of the association between words). Indeed, these approaches allow to adequately describe the similarity between words having the same part of speech but are unable to work with pairs of words owning different parts of speech. Moreover, they are subject to the limitations due to the fact that those applications mainly focused on text classification or paraphrase. Therefore, a new trend to overcome the aforementioned problems is to propose a method which can show association between different types of words.

3.2.1 Overview of the proposed method

Before detailed explanation, Figure 3.2 illustrates a flow of the process in the proposed method. Although different measurements are used, the quality of the measured similarity of pairs of words is usually unreliable [89]. In addition, it is intuitively clear that adjectives which are used to show impression or have semantic association with words, often appear in the same context of the words under particular lexical patterns. It is therefore important to include lexical patterns which show the connection links among words into the research. Thus, we adopted the method of Bollegala [92] that takes into account semantic relationships (patterns connecting words together) between two words as features of the association. The method mainly has two main steps: Pattern-based relation extraction, and similarity measurement. Specifically, the frequencies are queried for the individual words and their conjunctive. Various co-occurrence measures such as Jaccard, Dice, Overlap, and PMI are calculated. N-grams are also extracted from Google N-gram for the conjunctive query which represents the local context in which two words co-occur in the sentence. Numerous lexical syntactic patterns are identified and the frequency of each pattern is counted. The lexical patterns, that are conveying the similar semantic relations, are clustered together to effectively represent the semantic relation between words. The clusters of retrieved patterns are done using K-means. Both co-occurrence measure based similarity scores and clusters of lexical patterns define numerous features that identify the relation between words. Then a SVM is trained to measure the semantic similarity. We used LIBSVM as the SVM implementation. The training dataset of the proposed system is a collection of human-labeled association and non-association pairs of word and the adjective collected from sites and labeled by participants. Then, we consider the orientation of adjectives and inputs, and sort adjectives. The top n_a having the highest similarity scores and the same orientation with the input are displayed as the result.

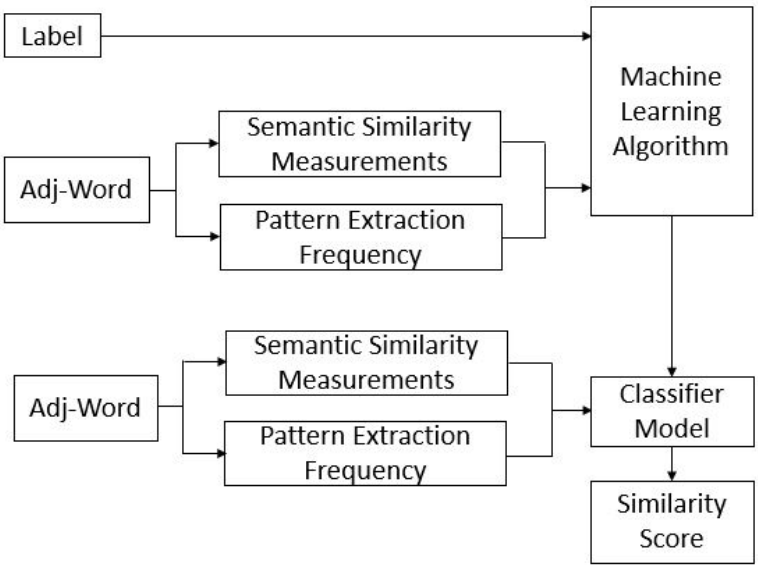


Figure 3.2: Flow chart of semantic similarity measurement.

3.2.2 Lexical pattern-based relation extraction

Adjectives serve a number of purposes in written and spoken language. Like many other words in English, the purpose of an adjective relies heavily on the way it is used by speakers and writers. Adjectives may be used before, or after the words they modify, depending on how the statement is constructed. For example: *"This show is exciting"*, or *"We should run as fast as possible to catch the bus"*. Because of the flexible ways of using adjective, sometimes words and adjectives appear at the same time but they do not have any connection. In particular, this case *"I went into the room and saw a pale pink shirt"*, *pale pink* is used to modify *shirt* only, even though *room* also appears in this context. Hence, it is better to consider phrases between words and adjectives which indicate their semantic relation. In order to extract patterns of pair of word and adjective, we show the flow of our process in Figure 3.3.

Specifically, we first create templates like "*Word*****Adjective*" where *"***"* means a relation from the query corpus to find occurrences. Depending on the forms of the

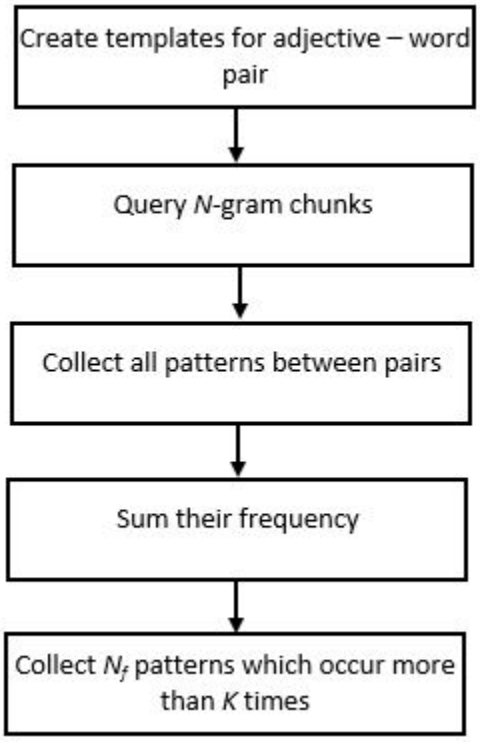


Figure 3.3: Flows of association measurements. All semantic relationship between the adjectives and the keywords are collected in the steps: collect patterns from the N-gram chunks and count the frequency.

words, we change the queries to retrieve the lexical patterns. Table 3.1 shows some examples of query templates between a noun *car* and an adjective *beautiful* or a verb *rain* and an adjective *wet*. The order of the queries is free. Next, to query N-gram chunks and their frequencies, we use Google N-gram as the main corpus in our method. Chunks queried from the corpus contain n words. Since in ref. [92] they claim that 95% of the lexical patterns extracted contain less than 5 words, in this study, we set n as 5, to be able to conduct many experiments in a reasonable time.

Obtaining groups of words, we generate their subsequences that satisfy all of the following conditions:

1. Each *word* and *adjective* have to exist in the subsequence.

2. The length of a subsequence is L words.

Table 3.1: Queries to retrieve patterns between adjectives and input. *** stands for a relation.

No	**Queries of Nouns**	**Examples**
1	Singular noun***adjective	car***beautiful
2	Capitalized singular noun***adjective	Car***beautiful
3	Plural noun***adjective	cars***beautiful
4	Capitalized plural noun***adjective	Cars***beautiful
No	**Queries of verbs**	**Examples**
1	Present tense***adjective	rain***wet
2	Present tense and 3rd singular***adjective	rains***wet
3	Past tense*** adjective	rained***wet
4	Present continuous tense***adjective	raining***wet
5	Present perfect tense***adjective	rained***wet

Table 3.2: Some examples of the elements in popular patterns of nouns and adjectives.

No	Pattern
1	' '
2	,
3	and
4	is
5	of
6	in
7	was
8	of the
9	is a
10	as

No	Pattern
11	to
12	with
13	for
14	for the
15	at
16	of a
17	will be
18	has been
19	is very
20	at the

Table 3.3: Some examples of the elements in popular patterns of verbs and adjectives.

No	Pattern
1	to
2	a
3	and
4	the
5	,
6	of
7	or
8	an
9	is a
10	as

No	Pattern
11	in
12	-
13	for
14	to the
15	with
16	as
17	is
18	by
19	to be
20	on

3. The total number of words skipped in a subsequence is limited at G words.

We extract patterns between inputs and adjectives from satisfied subsequences, Table 3.2 and Table 3.3 show some examples of popular patterns of association patterns between association nouns, verbs, and non association nouns and verbs respectively.

Finally, we count the frequency of all patterns, and then choose the one whose frequency is more than T times.

3.2.3 Clustering lexical patterns

In fact, there are many similar patterns to show the semantic relation. For instance, to say that someone is beautiful, we can use pattern *'is'* in the sentence *'she is beautiful'* and '*is very*' in *'she is very beautiful'*. In other words, it will be wasteful and less meaningful if we treat these patterns differently. Therefore, it is better to apply the clustering method to group similar patterns into a group which helps lessen the distribution of patterns and fasten our further applications.

Generally, clustering [95] is a method which groups patterns into clusters, where patterns within each clusters have high degree of similarity, but are dissimilar to the patterns in the other clusters. Clustering involves dividing a set of data points into non-overlapping groups or clusters of points where points in a cluster are more similar to one another than the points presented in the other clusters. So, a good clustering method would exhibit high similarity in a single cluster and a very less similarity with the other clusters.

Here, we use K-means [96, 97] which is an unsupervised algorithm used in clustering. It chooses the centroid and compares centroid with the data points based on their intensity and characteristics and finds the distance, the data points which are similar to the centroid are assigned to the cluster having the centroid. New *'K'* centroids are calculated and thus K-clusters are formed by finding out the data points nearest to the clusters. The main steps of K-means algorithm are shown in Figure 3.4.

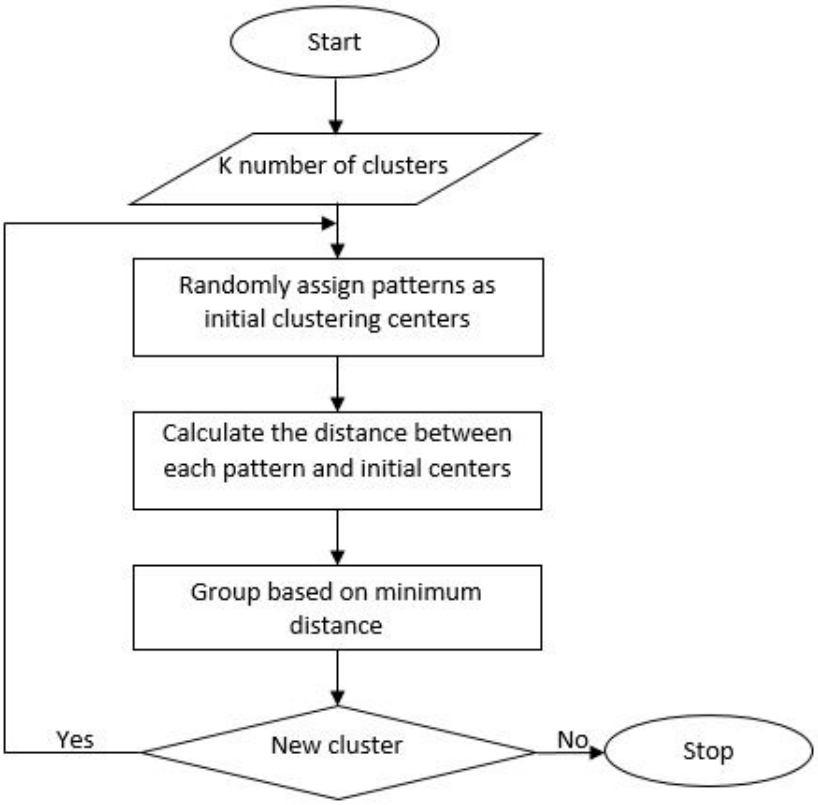


Figure 3.4: Flow chart of K-means algorithm.

Specifically, K-means can be explained as the following steps:

1. Choose K number of points randomly and make them initial centroids.

2. Select a pattern from the collection, and calculate the distance between the pattern and initial centers. Each pattern is represented by a real-valued vector of tf-idf (term frequency-inverse document frequency) weights where the inverse document frequency (idf):

$$idf_t = \frac{\log N}{df_t} \tag{3.2}$$

and tf-idf weight of a term is:

$$w_{t,d} = \begin{cases} 1 + \log tf_{t,d}, & if tf_{t,d} > 0 \\ 0, & otherwise \end{cases} \tag{3.3}$$

Table 3.4: Notation.

Notation	*Description*
$F(w_1)$	frequency of word 1 in corpus
$F(w_2)$	frequency of word 2 in corpus
$F(w_1, w_2, w_3)$	co-occurrence frequency of 3 words
N	total number of words in corpus.
$F(w_1, w_2)$	$\sum$(co-occurrence frequency of word 1 and word 2)
$F(w_1, w_3)$	$\sum$(co-occurrence frequency of word 1 and word 3)
$F(w_2, w_3)$	$\sum$(co-occurrence frequency of word 2 and word 3)
$P(w_1)$	$\frac{F(w_1)}{N}$
$P(w_2)$	$\frac{F(w_2)}{N}$
$P(w_3)$	$\frac{F(w_3)}{N}$
$P(w_1, w_2)$	$\frac{F(w_1,w_2)}{N}$
$P(w_1, w_3)$	$\frac{F(w_1,w_3)}{N}$
$P(w_2, w_3)$	$\frac{F(w_2,w_3)}{N}$

Here, df_t is the number of patterns that contain term t, N is total number of patterns, $tf_{t,d}$:the number of times term t occurs in a pattern d.

If the pattern is found to be the most similar to a centroid then assign it into the cluster of that centroid. The formula to calculate is:

$$\cos(\vec{p}, \vec{q}) = \frac{\vec{p}\cdot, \vec{q}}{||\vec{p}|| \cdot ||\vec{q}||} = \frac{\sum_{i=1} p_i q_i}{\sqrt{\sum_{i=1} p_i^2} \times \sqrt{\sum_{i=1} q_i^2}}. \tag{3.4}$$

3. When each data point has been assigned to one of the clusters, re-calculate the value of the centroids for each K number of clusters.

4. Repeat steps 2 to 3 until no data point moves from its previous cluster to some other cluster (termination criterion has been satisfied).

3.2.4 Similarity measurement between adjective and input

Co-occurrence based similarity measurement

In this work, together with lexical patterns, we also use the popular co-occurrence based measures to measure the semantic similarity between word and adjective.

This approach estimate the similarity between two words using the frequency of co-existence within larger lexical units (sentence, documents). The underlying assumption is that terms that co-exist are often likely to be related semantically. One popular method to estimate co-occurrence is to pose conjunctive queries to a web search engine; the number of returned hits is an estimate of the frequency of co-occurrence. Co-occurrence based methods do not rely on annotated language resources like ontologies nor require downloading documents or snippets, as is the case for context-based semantic similarities. Here, we investigate the performance of four different co-occurrence based measurements: Jaccard, Overlab, Dice, and PMI, defined next. Notations are summarized in Table 3.4. In the equations that follow, words in the list of inputs and adjective play like parameters.

1. **Jaccard Coefficient** Jaccard Coefficient is derived from information retrieval. The measure was originally designed for binary vectors. It divides the number of equal features with the number of features in general. The Jaccard coefficient measure for two words is computed as:

$$Jaccard(w_1, w_2) = \frac{F(w_1, w_2)}{F(w_1) + F(w_2) - F(w_1, w_2)}. \tag{3.5}$$

2. **Dice Coefficient**

   Dice Coefficient is very similar to the Jaccard measure and is also introduced from information retrieval. It is computed as:

$$Dice(w_1, w_2) = \frac{2F(w_1, w_2)}{F(w_1) + F(w_2)}. \tag{3.6}$$

3. **Overlap Coefficient**

Overlap Coefficient is computed:

$$Overlap(w_1, w_2) = \frac{F(w_1, w_2)}{\min(F(w_1), F(w_2))}. \tag{3.7}$$

4. **Pointwise mutual information**

   Pointwise mutual information computes how often two words co-occur, compared with what would be expected if they were independent. This measure is computed as

$$PMI(w_1, w_2) = \log_2 \left(\frac{P(w_1, w_2)}{P(w_1)P(w_2)} \right). \tag{3.8}$$

   The main advantage of this measure comparing to the probability measure is that it penalizes co-occurence with features not specific for the lexeme of interest.

Combine Corpus-based and bag of patterns in measuring similarity using Machine Learning

1. **Feature Selection**

   In many researches, selection of features that contain maximum information about the training data is essential in finding a good classifier. In document classification, a bag of words (BOW) is a sparse vector of occurrence counts of words; that is, a sparse histogram over the vocabulary. Taking that idea into this research, we believe that in expressing impression, people often use some of the patterns to show the impression of an object, an event, a fact, etc. Therefore, instead of using BOW, we consider Bag of Patterns (BOP) and the four co-occurrence based similarity measurements as a dimensional feature vector of this method. On the other words, we consider N_c+4, where N_c is a number of clusters of patterns.

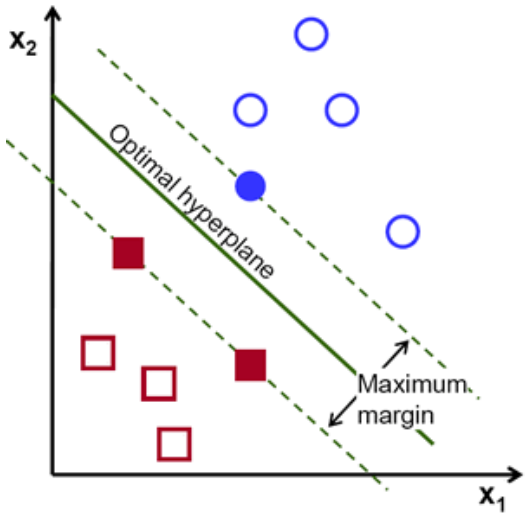


Figure 3.5: The optimal decision surface with maximal margin vs. a non-optimal decision surface.

2. SVM

We use SVM [98], which is a generally applicable tool for machine learning used to discriminate between two classes association and non-association classes. The dataset includes both association pairs and non-association pairs which are considered as positive class and negative class, respectively. Detailed experimental conditions are shown in the experiment part. They are generated from different sources and judged by participants in this study. Obtained training dataset $\{x_i\}$ are then labeled $y_i \in \{-1,1\}$ which aims to indicate the type of the pairs - association ($y_i = 1$)or non association ($y_i = -1$). SVM searches for a separating hyperplane, which separates positive and negative examples from each other with maximal margin, in other words, the distance of the decision surface and the closest example is maximal. Figure 3.5 shows the optimal decision surface with maximal margin.

In SVM, the class of an input vector X can be decided by evaluating the sign of $y(X)$.

$$y(X) = w_T \phi(X) + b \tag{3.9}$$

If y(X) > 0 we assign X to class $+1$ and if y(X) < 0, we assign it to class -1. Here $\phi(X)$ is a feature-space transformation, which can map X to a space of higher, possibly infinite, dimensions.

Given a data set comprising N input vectors $X_1,\ldots,X_n$ and their corresponding labels $t_1,\ldots,t_n$ where $t_n \in \{-1,+1\}$, we would like to find w and b such that it explains the training data: y(X_n) ≤ 1 when t_n=+1 and y(X_n) when $t_n = -1$. This can be rewritten in a single constraint:

$$t_n(w^T\phi(x_n) + b) \leq 1, n = 1, \ldots, N \tag{3.10}$$

In addition, X and b are chosen so that the distance between the decision boundary $w^T\phi(X) + b = 0$ (a line in the 2-D case in the 3-D case, a hyperplane in the n-D case) and the closest points to it is maximized. This distance is called the margin, hence the name *maximum margin* classifier. Geometrically, the margin is found to be $2/\parallel w \parallel^2$ and so the maximum margin problem can be equivalently expressed as the minimization problem:

$$\arg min_w \frac{1}{2} \parallel w \parallel^2 \tag{3.11}$$

subject to constraint (3.8).

3.3 Result and evaluation

As part of this chapter, we will only illustrate the obtained results using the above described method on the manual data, and its impact. An experimental result is evaluated based on the different lengths of the feature vectors, and measurements.

Table 3.5: Some examples of associated noun-adjective pairs.

No	Noun	Adjective		No	Noun	Adjective
1	flower	colorful		11	food	hungry
2	headache	sick		12	hug	tight
3	joy	joyful		13	landscape	scenic
4	leader	influential		14	lesson	useful
5	love	romantic		15	morning	bright
6	mountain	steep		16	nature	natural
7	night	dark		17	park	spacious
8	perfume	fragrant		18	progress	increasing
9	rest	tired		19	risk	cautious
10	accident	terrible		20	baby	cute

Table 3.6: Some examples of non-associated noun-adjective pairs.

No	Noun	Adjective		No	Noun	Adjective
1	christmas	deep		11	actress	coastal
2	perfume	volcanic		12	hobby	noisy
3	love	hungry		13	tree	trusty
4	slope	happiest		14	surprise	hot
5	smile	tasty		15	mountain	crying
6	friendship	cloudy		16	damage	cute
7	gift	deep		17	morning	big
8	juice	frightful		18	snow	expensive
9	sea	competitive		19	road	youthful
10	cousin	autumnal		20	holiday	zoological

3.3.1 Experimental setup

Dataset

For the experiments, we used our own dataset, since to the best of our knowledge, there is no publicly available dataset of pairs of words and impression adjectives. In order to evaluate the performance of our algorithm, we created the dataset from collected list of adjective - word pairs. There are two types of pairs considered in this scale of research: noun-adjective, and verb-adjective which require both association and non association word pairs. Although the common definition of an adjective is a word that describes or clarifies a noun, in the conversation, there are several adjective linked to verbs that have a special function when it comes to show the impression. For example: "*She swam, and got tired*". The adjective "*tired*" somehow modifies

Table 3.7: Some examples of non-associated verb-adjective pairs.

No	Noun	Adjective
1	cut	awake
2	discuss	smoky
3	adjust	lonely
4	help	graphic
5	sink	calculating
6	hate	rapid
7	swim	forgiving
8	develop	fried
9	permit	transported
10	laugh	clean

No	Noun	Adjective
11	cook	bright
12	escape	fruitless
13	wait	wide
14	contribute	dry
15	cost	creative
16	teach	tasteless
17	teach	unprotected
18	die	singing
19	approve	slippery
20	scratch	sweaty

the word "*swam*". Therefore, instead of doing an experiment only focusing on the adjective-noun pair, we conducted an experiment on the database of verb-adjective pair collection as well.

Association word pairs are extracted from different resources such as the database made by Douglas L. Nelson and Cathy L. McEvoy [99], and web page [100] where people share impressions of words. For non-association pairs, we manually generate the database from the collected association pairs. From the set of associated pairs, we shuffled and generated non-associated pairs. To be more precise, we asked 10 participants (4 native speakers and 6 non native speakers) to check and label all of the pairs two times. Next, we calculated their frequency based on their queries using Google N-gram and the collected patterns. In the present case, our final dataset involves 5,000 adjective-noun pairs: 2,500 association and 2,500 non-association and the same number of adjective-verb pairs as well. Table 3.5 and 3.6 show some examples of pairs between noun and adjectives, also Tables 3.7 and 3.8 illustrate some examples of verb -adjective pairs.

Table 3.9 shows the number of patterns extracted for associated and non-associated of both noun-adjective and verb-adjective. After obtaining the pairs of words, we used the proposed BOP in Section 3.2 to deal with lexical pattern collection. We experimentally set the length of N-grams at 5. Figure 3.7 shows the

Table 3.8: Some examples of verb-adjective pairs.

No	Noun	Adjective
1	adapt	new
2	afford	sufficient
3	complain	dissatisfied
4	create	new
5	drink	thirsty
6	fail	sad
7	hate	jealous
8	kick	painful
9	laugh	hilarious
10	rain	wet

No	Noun	Adjective
11	admire	esteemed
12	buy	cheap
13	cry	sad
14	dance	joyful
15	encourage	stimulated
16	feed	hungry
17	invest	financial
18	kiss	sweet
19	pull	tight
20	relax	enjoyable

Table 3.9: Number of lexical patterns collected by the proposed method.

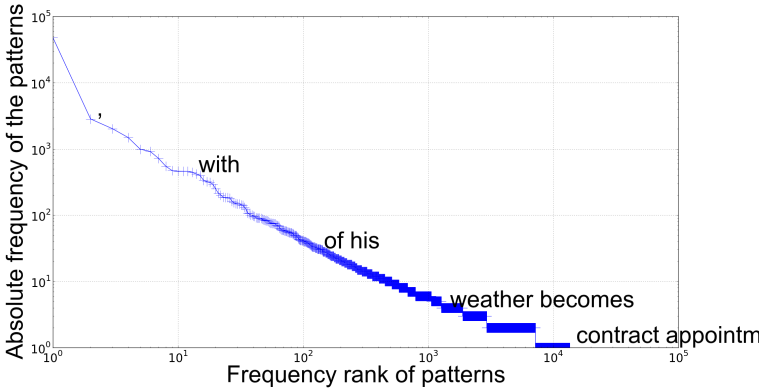
Word pairs	Associated	Non-associated
Noun-adjective	2,500	2,500
Verb-adjective	2,500	2,500
Selected noun extracted patterns	142,612	87,181
Selected verb extracted patterns	94,598	42,633

frequency of types of pairs. To avoid noisy patterns, we ignored all that appeared less than k times. We set k at 5 in this research. Next, we applied our clustering method to group patterns.

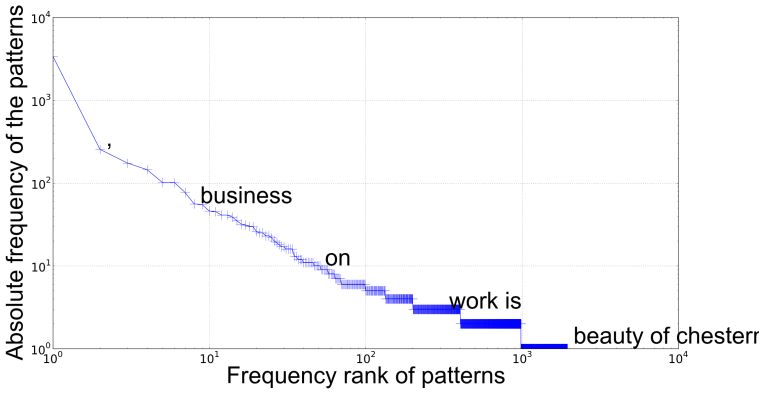
We tested the performance of different methods on our dataset and different length of the vector features.

Vector features

As introduced above, we have collected a number of words collected from different sources. From the collected words, we generated lexical patterns as features of vectors along with measurements. Figure 3.8 shows the non-linear distribution of patterns where each symbol represents different types of pattern. Depending on the size of the patterns, we cluster them into groups. The number of groups we try in this evaluation is range from 10% of the full size to non-cluster (100% of size). In other words, the vector length is from *(10% of the size of patterns + 4)* to equal as *the size of the collection+4*.



(a) The frequency rank of associated noun and adjective pairs

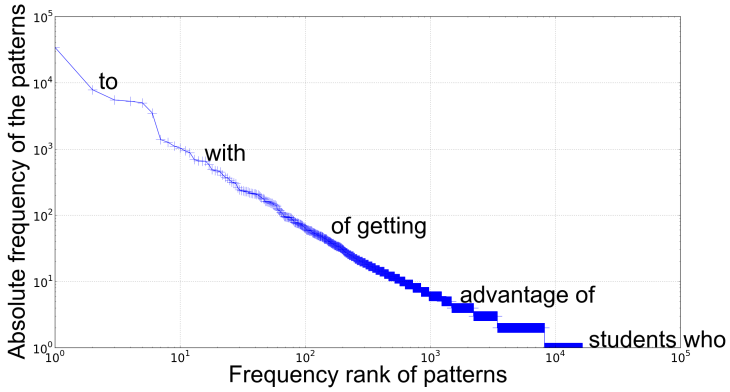


(b) The frequency of patterns of non-associated noun and adjective pairs

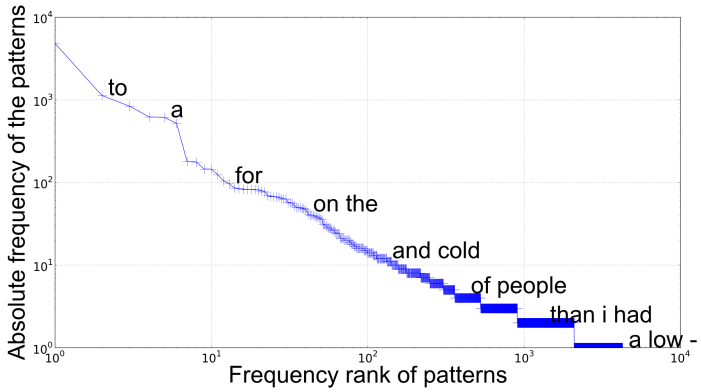
Figure 3.6: The distribution of patterns of Noun and Adjective pairs.

3.3.2 Results

Here, we present the experimental results of the proposed algorithm that allows the estimation of impressions of different forms of words. In the thesis, patterns are clustered into K groups. The original size of the bag of the patterns of nouns and verbs are 229,793 and 137,231 respectively. The number of groups is range from 1% of the original size of the bag to 100% (no cluster). Each form of words was run and evaluated separately. Tables 3.10 and 3.11 show the experimental results of noun and verb.



(a) The frequency rank of associated verb and adjective pairs



(b) The frequency rank of non-associated verb and adjective pairs

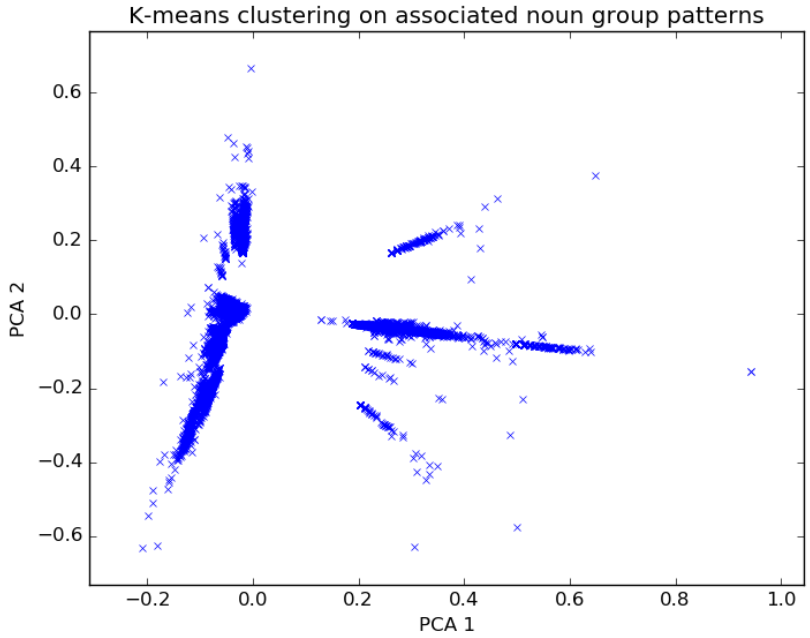
Figure 3.7: The distribution of patterns of verb and adjective pairs.

3.3.3 Evaluation

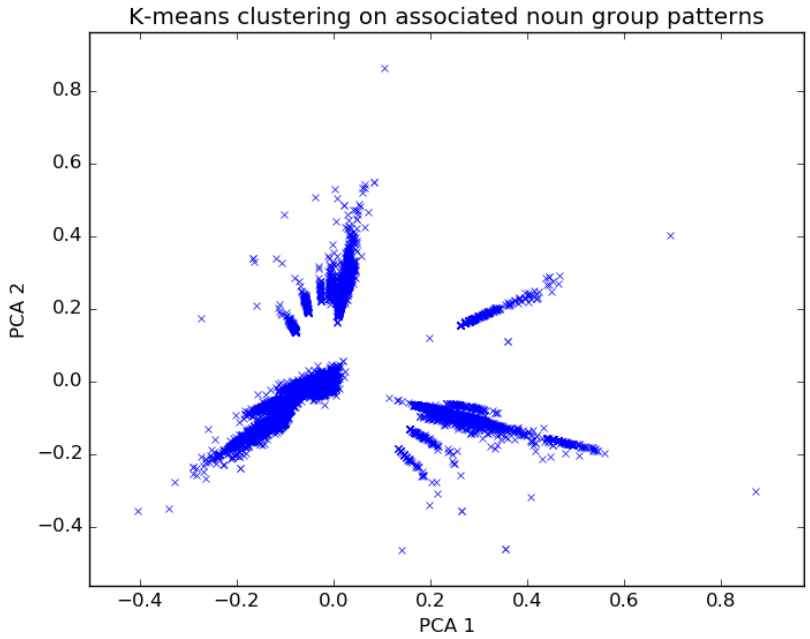
To show the performance of the proposed method, precision under different testing situation are calculated on our database by changing the size of the feature of the vectors and forms of words and different methods.

Parameters for evaluation

In the context of classification, True Positives (TP), True Negatives (TN), False Negatives (FN) and False Positives (FP) are used to compare the class labels assigned to documents by a classifier with the classes the items actually belongs to. True positives (TP) are the examples that the classifier correctly labeled as belonging to



(a) Lexical pattern distribution of Noun-Adjective Pair



(b) Lexical pattern distribution of Verb-Adjective Pair

Figure 3.8: Lexical Pattern Distribution of word-adjective pairs.

the positive class. False positive (FP) are the examples which were not labeled by the classifier as belonging to the positive class but should have been. True Negatives (TN)

Table 3.10: Noun-Adjective pair similarity Scores.

Pairs	**Jaccard**	**Dice**	**Overlap**	**PMI**	**No_clus**	Proposed
food _hungry	0.716	0.757	0.726	0.714	0.714	**1.000**
christmas _ merry	1.000	0.995	0.996	0.995	0.995	**1.000**
tree _ planted	0.198	0.197	0.197	0.196	0.219	**1.000**
love _ sweet	0.312	0.302	0.305	0.321	0.325	**0.950**
culture _ great	0.047	0.059	0.052	0.049	0.079	**0.888**
noise _ annoying	0.045	0.044	0.044	0.043	0.045	**0.750**
career _ important	0.084	0.098	0.098	0.081	0.091	**0.504**
friendship_close	0.019	0.018	0.018	0.017	0.017	**0.401**
fruit _ juicy	0.053	0.054	0.049	0.053	0.050	**0.377**
winter _ severe	0.798	1.000	1.000	1.000	1.000	**0.371**
attack _ violent	0.018	0.017	0.017	0.017	0.016	**0.338**
career _ ambitious	0.058	0.048	0.048	0.058	0.059	**0.335**
restaurant_ fabulous	0.028	0.029	0.028	0.028	0.027	**0.325**
growth _ rapid	0.097	0.122	0.094	0.111	0.148	**0.209**
student _ comparative	0.045	0.046	0.048	0.044	0.041	**0.149**
competition efficient	0.025	0.024	0.024	0.024	0.024	**0.141**
couple _ classified	0.000	0.000	0.000	0.000	0.000	**0.140**
opportunity_ suitable	0.024	0.024	0.024	0.023	0.023	**0.097**
restaurant_luxurious	0.034	0.031	0.033	0.031	0.030	**0.094**
chocolate _ delicious	0.108	0.107	0.108	0.108	0.107	**0.078**
apartment _ cozy	0.018	0.018	0.018	0.018	0.018	**0.074**
cycle _ dangerous	0.017	0.016	0.016	0.016	0.015	**0.004**
flower _ higher	0.013	0.012	0.012	0.012	0.011	**0.000**

are the examples that the classifier correctly labeled as belonging to the negative class. At last there are False Negatives (FN), which are the examples not labeled by the classifier as belonging to the negative class but should have been. Other evaluation measures like precision and accuracy can be easily calculated from these four variables.

- **Accuracy:** Accuracy simply measures how often the classifier makes the correct prediction. It's the ratio between the number of correct predictions and the total number of predictions (the number of test data points).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{3.12}$$

Table 3.11: Verb-Adjective pair similarity Scores.

Pairs	**Jaccard**	**Dice**	**Overlap**	**PMI**	**No Clus**	**Proposed**
handle_easy	0.033	0.742	0.755	0.276	0.362	**1.000**
listen_willing	0.000	1.000	1.000	0.004	0.002	**0.765**
cut_short	0.002	0.208	0.102	0.787	1.000	**0.750**
assume_responsible	0.026	0.031	0.021	1.000	0.039	**0.569**
concern_future	1.000	0.318	0.298	0.610	0.874	**0.549**
select_choice	0.240	0.199	0.373	0.474	0.302	**0.498**
hear_loud	0.600	0.334	0.039	0.190	0.480	**0.445**
pull_trying	0.006	0.577	0.577	0.010	0.035	**0.438**
enter_permitted	0.001	0.567	0.566	0.003	0.039	**0.428**
teach_training	0.346	0.131	0.057	0.574	0.644	**0.362**
apply_appropriate	0.113	0.261	0.223	0.284	0.531	**0.351**
purchase_paid	0.249	0.030	0.200	0.704	0.460	**0.272**
feel_guilty	0.018	0.144	0.219	0.382	0.524	**0.251**
report_executive	0.104	0.168	0.114	0.308	0.478	**0.230**
kill_hurt	0.007	0.021	0.015	0.368	0.021	**0.227**
wake_sleeping	0.087	0.071	0.049	0.255	0.100	**0.203**
hold_collected	0.054	0.021	0.014	0.289	0.119	**0.158**
change_seasonal	0.005	0.151	0.140	0.031	0.307	**0.150**
wait_difficult	0.000	0.144	0.135	0.001	0.110	**0.127**
claim_legal	0.033	0.048	0.034	0.135	0.077	**0.109**
spread_fun	0.002	0.040	0.023	0.136	0.097	**0.098**
invest_diverse	0.000	0.016	0.040	0.129	0.294	**0.089**
win_decisive	0.138	0.091	0.021	0.006	0.021	**0.077**
report_classified	0.021	0.037	0.013	0.090	0.032	**0.069**

- **Precision and recall:** Precision and recall are two widely used metrics for evaluating performance in text mining, and in other text analysis field like information retrieval. They can be seen as extended versions of accuracy, and by using a combination of these measures the problem with skewed data for classifiers dissipates. Precision is used to measure exactness, whereas recall is a measure of completeness. Precision is the number of examples correctly labeled as positive divided on the total number that are classified as positive, while recall is the number of examples correctly labeled as positive divided on the total number of examples that truly are positive. This is shown in the following formulas.

$$Precision = \frac{TP}{TP + FP}. \tag{3.13}$$

$$Recall = \frac{TP}{TP + FN}. \tag{3.14}$$

- **F-Measure:** F-Measure is the harmonic mean of precision and recall. This gives a score that is a balance between precision and recall. F-Measure combines them into one score for easier usage. This is important because it might be better to optimize the system to favors either the precision or the recall if one of these has a more positive influence on the final result of the trading simulation than the other.

$$F = \frac{2 \times precision \times recall}{precision + recall}. \tag{3.15}$$

The results are shown in the Figure 3.9. The accuracy of noun is range from (31%-78%) and the accuracy of verb is range from (36%-82%). As can be seen, in both two cases, the best results are obtained when the number of the feature vector of clusters is from 10% to 30% of the total number of the patterns. This is because there are many noisy collected lexical pattern, so if we can collect those together, we can avoid noisy data and can improve the result.

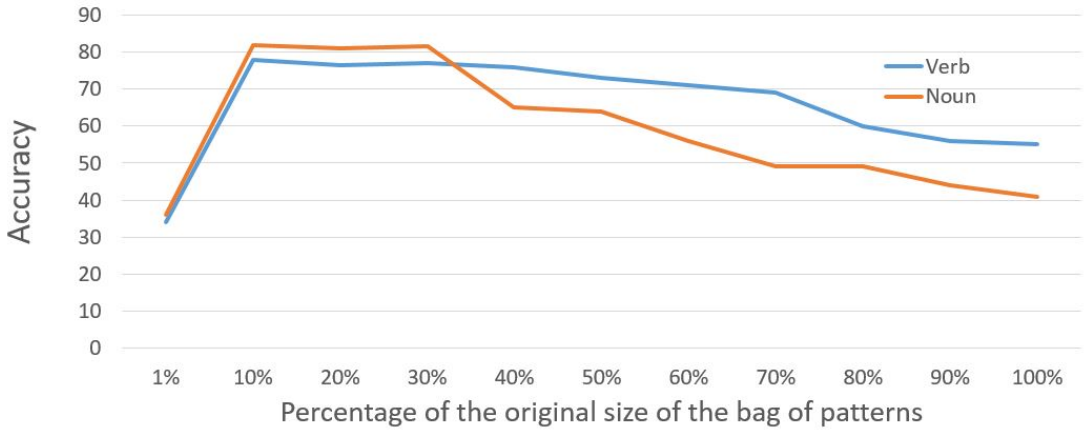


Figure 3.9: The cluster-based accuracy comparison.

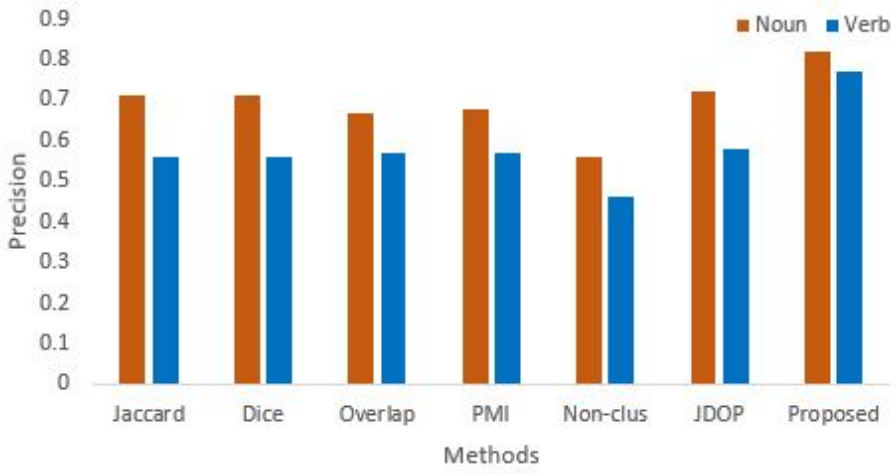


Figure 3.10: Performance comparison.

Comparing the result to the other methods that use only measurements or non clustering, we got the result as shown in Figure 3.10. We made a comparison between seven methods. Overall, the performance of the verbs are always worse than the noun's. The reason of this result is because the usage between verbs and adjectives in the corpus is less popular than nouns. Moreover, the graph shows that combining semantic similarity measurements with patterns helps gain better performance of pairs. Our proposed method achieved the precision as high as 0.82 and 0.77 for noun pairs and verb pairs, respectively. It seems that obtaining patterns is helpful in such kind of measurements.

3.4 Summary

In this chapter, we proposed a new approach to automatically measure semantic similarity between noun- adjective pairs and verb-adjective pairs. Our approach takes the co-occurrence and lexical connections of words and adjectives into account in order to identify their semantic relationship. As illustrated in our experiments, patterns are proposed for reasoning about concepts relatedness, and to show the link between words and adjectives that are semantically related. Instead of the usage of the raw

patterns to improve performance and speed, we propose a new method using clustering. After obtaining the feature vectors, a two-class SVM is used to train the data and measure the strength of connection of pairs. Our experiments showed that the proposed method works well with both noun-adjective pair and verb-adjective pair. Experimental results showed that the performance of noun-adjective pair is higher than verb-adjective pair.

Chapter 4

Impression Estimation of a Sentence

In Chapter 3, we have proposed a new method for measuring semantic similarity between adjectives and words. As aforementioned, showing the relatedness between adjectives and words is actually very useful for building systems which are able to show impressions of sentences. This chapter proposes an application to estimate the impression of short sentences. The proposed approach firstly analyzes the sentences to obtain important information about the sentence. Association between adjectives and sentences through the proposed method is explained in Chapter 3. We also have chosen to use sentiment analysis and ranking aggregation in order to select the best of adjectives. The rest of this chapter is organized as follows. In Section 4.1, we present an overview of the proposed approach. Section 4.2 introduces the experiment and the evaluation. This chapter is concluded in Section 4.3.

4.1 Proposed system

Figure 4.1 illustrates the architecture of the proposed sentence impression estimation system. Keys to the proposed framework are keywords extraction, adjectives

collection, and adjective selection. Each input sentence is preprocessed to extract keywords which show "*What the user want to mention?*". Selected keyword candidates are used to query the N-gram utterances ($N = 5$ in this thesis) which are used in the next step - adjective candidates extraction. After obtaining keywords and related adjectives, we apply our proposed method in Section 3 to measure the similarity strength between each collected adjective and keyword. Consequently, we sort adjectives for each keyword based on the scores, and then combine those orders together. In this research, we adopt the classical positional method - Borda's rank aggregation method to generate an acceptable ranking from computed rankings in the previous steps. They are continually analyzed to get the orientation using sentiment analysis. The top n_a adjectives having the highest similarity scores and the same orientation with the input sentence are displayed.

4.1.1 Keyword extraction

The task of a keyword extraction is to automatically identify words that best describe the input. The approach is to use a frequency to select the important words. As can be seen from the processing flow in Figure 4.2, first, we do part of speech (POS)

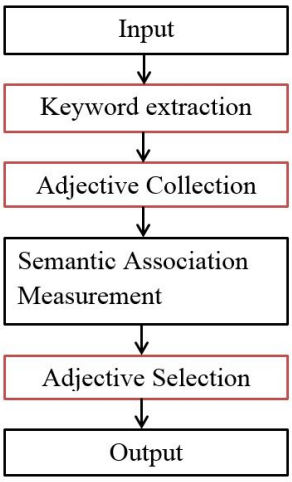


Figure 4.1: Overview of proposed system framework.

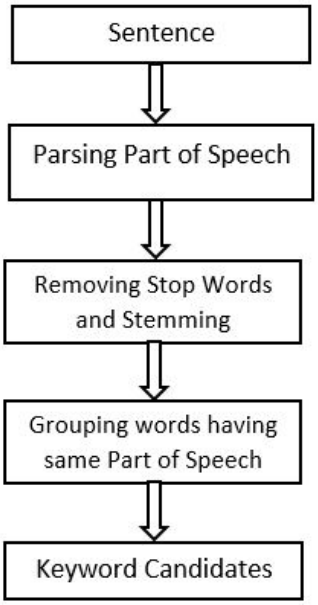


Figure 4.2: Overview of sentence topic extraction.

tagging for the input sentence. In other words, we tokenize and label each word of the input sentence in part as "noun", "verb" using part of speech tagger. Because there are some unnecessary words with little lexical content, we continue to filter out of the given sentence based on Wordnet corpus of stopwords. After removing all noisy data, the rest of the words are grouped by POS tagging and outputted as keywords.

4.1.2 Adjective collection

Figure 4.3 shows the processing flow of the adjective collection step. Each keyword in the list of keywords is then used to create templates to query Google N-gram. The resulting queries are continually processed. The order of the words in the query is random. The result of the query is a list of N-gram chunks. To have an effective retrieval, keywords are changed to forms of noun, verb and adjective. We finally collect words whose part of speech is "adjective". The final list is a collection of adjectives of keywords. Tables 4.1 and 4.2 give a particular template to query adjectives from keywords.

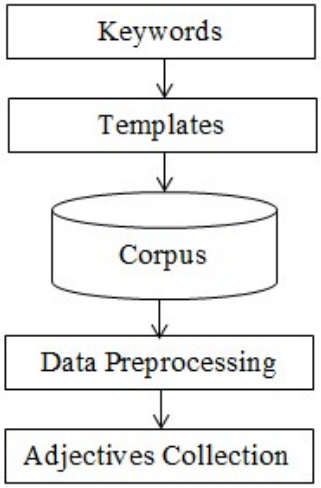


Figure 4.3: The architecture of Adjective Collection.

Table 4.1: An example of templates to query adjectives for *"sunrise"*.

N	**Template**	**Examples**
1	as Noun	as sunrise
2	noun	sunrise
3	noun is	sunrise is
4	Noun is	Sunrise is
5	nouns are	sunrises are
6	Nouns are	Sunrises are

Table 4.2: An example of templates to query adjectives for *"rain"*.

N	**Tense of verb**	**Examples**
1	Present	rain
2	Present and 3rd singular	rains
3	Past	rained
4	Present continuous	raining
5	Present perfect	rained
6	Capitalize	Raining

4.1.3 Semantic similarity measurement

It is important to show the co-occurrence between the keywords and the collected adjectives. Each adjective in the collected adjective list is used to combine with each keyword. These pairs are then inputted into the proposed method explained in Chapter 3 to get the score of similarity.

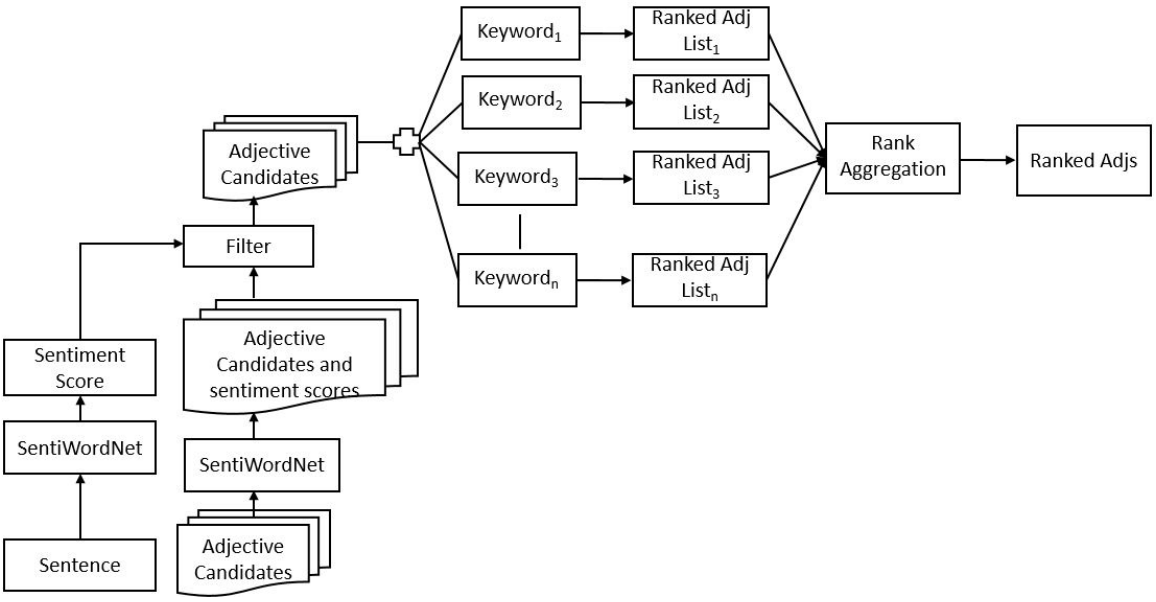


Figure 4.4: The Adjective Selection Framework task.

4.1.4 Adjective selection

In this step, we aim at selecting the top n_a adjectives that are the most related to the sentences. To that aim, we use a rank aggregation technique discussed below.

Rank aggregation

Rank aggregation can be thought as the unsupervised analog to regression, in which the goal is to find an aggregate ranking that minimizes the distance to each of the ranked lists in the input set. Rank aggregation has also been proposed as an effective method for nearest-neighbor ranking of categorical data [4], and gives a robust approach to the problem of combining the opinions of experts with different scoring schemes, as are common in ensemble methods. There exist various methods for merging rank-ordered lists. Basically, they use information that is readily available from ranked lists of items. In most cases, the strategies rely on the following information: (i) the ordinal rank assigned to an item in the rank list; and (ii) the score assigned to an item in the rank list. In the score based methods, items are ranked in order of the assigned scores in the rank lists, or some transformation of those scores [101–107],

while in rank based merging methods, items are ranked in order of the assigned ranks in the rank lists, or some transformation of those ranks [101, 102, 108]. Another orthogonal distinction of rank fusion methods is whether the methods rely on training data (e.g., the Bayes-fuse method [101], the linear combination method [102] and the preference rank combination method [103]) or not. Another class of methods is based on the content of ranked items. In these methods, the ranked documents are downloaded and analysed in order to produce the final ranking.

Borda's method

Borda's method [109, 110] is a "positional"method, in that it assigns a score corresponding to the positions in which a candidate appears within each voter's ranked list of preferences, and the candidates are sorted by their total score. A primary advantage of positional methods is that they are computationally very easy: they can be implemented in linear time. They also enjoy the properties called anonymity, neutrality, and consistency in the social choice literature [111].

Borda's method assigns ranks to items based on a total Borda score, which is computed on each list by showing that the most preferred item in a universe of U items gets $|U|$ points, the next gets $|U| - 1$ points, and so on.

More formally, Borda's method on a set of complete rankings R is computed as follows. For each item i and list $r_k \in R$, let $B_{rk}(i)$ equal the number of items j in r_k such that $r_k(j) > r_k(i)$. The total Borda score for the item i is given by $B_t(i) = \sum_{r \in R} B_r(i)$. Ranks are assigned by sorting scores B_t from highest to lowest rank. When R includes partial rankings, one proposal is to assign any "leftover"score from a given list equally among all remaining unranked items in the list. That is, for a list r_k, where $|r_k| = |U| - d$, and d is a distance function used to compute the $B_r k(i)$ as usual for all items $i \in r_k$, and assign $B_{rk}(j) = \frac{(d+1)^2-(d+1)}{2d}$ for all items $j \notin r_k$. The Borda scores B_t and rankings are assigned as above.

Table 4.3: First five SentiWordNet entries for cold.

POS	Offset	POS(s)	Neg(s)	SynsetTerms
a	1207406	0.00	0.75	cold#a#1
a	1212558	0.00	0.75	cold#a#2
a	1024433	0.00	0.00	cold#a#3
a	2443231	0.13	0.38	cold#a#4
a	1695706	0.63	0.00	cold#a#5

Adjective Orientation

SentiWordNet [29] is a lexical resource for opinion mining. SentiWordNet assigns to each synset of WordNet three sentiment numerical scores, positivity, negativity and objectivity, describing how positive, negative and objective the terms contained in the synset are. Each of the three scores ranges from 0.0 to 1.0, and their sum is 1.0 for each synset. This means that a synset may have nonzero scores for all the three categories, which would indicate that the corresponding terms have, in the sense indicated by the synset, each of the three opinion-related properties only to a certain degree. SentiWordNet word values have been semi-automatically computed based on the use the semi-supervised method described [112]. In 4.3, the first 5 senses of *cold#a* present all possible combinations, included mixed scores *cold#a#4*, where positive and negative valences are assigned to the same sense. Intuitively, mixed scores for the same sense are acceptable, as in "*cold beer*"(positive) vs. "*cold pizza*"(negative).

4.2 Experiment and evaluation

In this section, we show and discuss the results of the proposed method evaluation. In this part, we would like to answer two questions. Firstly, how much effect does the proposed method have on real sentences? Secondly, what is the extent of the agreement do people think between the proposed method and the user's opinion.

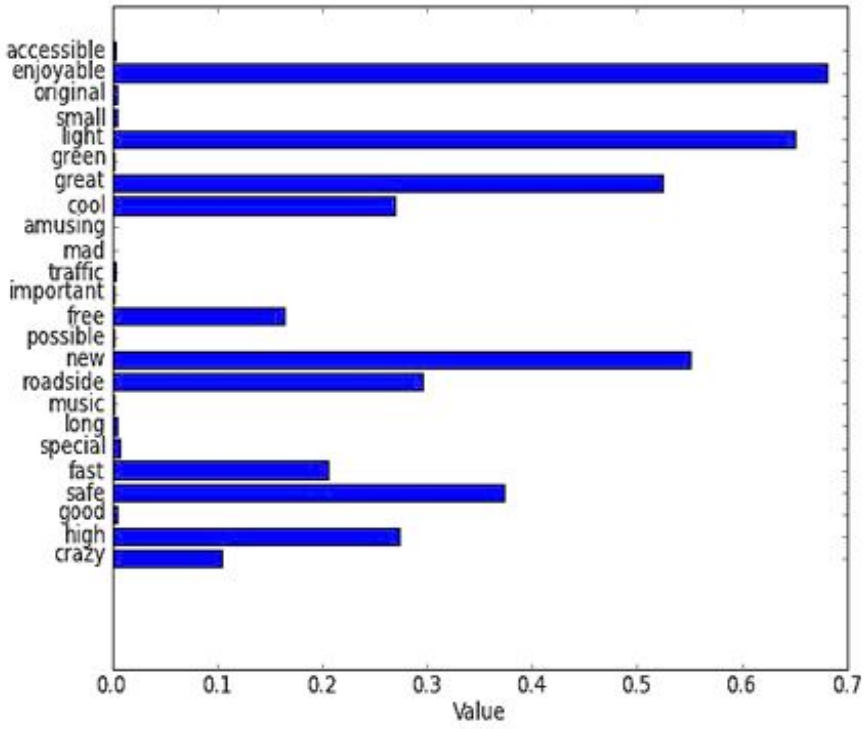


Figure 4.5: An example of the result for the input "She is driving".

4.2.1 Experiment

In order to evaluate the effectiveness of the proposed method, we conducted an experiment with sentences extracted from [113], and analyzed the performance of the proposed method. After obtaining the result, we carried out a subjective experiment to let the participants judge the output. Given some input sentences *"She is driving "* as an example, the results showing how the proposed method works is shown in Figure 4.5. The top 20 best results of the experiment are summarized in Table 4.4.

4.2.2 Experimental evaluation

Method

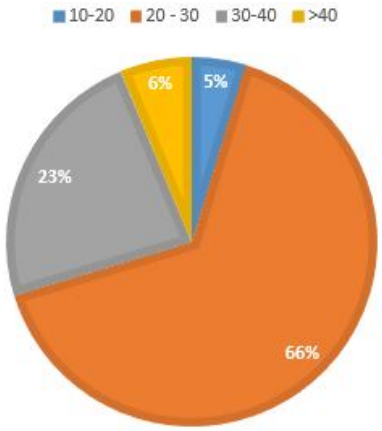
We evaluated our proposed method through the subjective experiment by 64 participants. They are from different nations and ages. Figure 4.6 show some information

Table 4.4: Some results from the experiments. There are five adjectives corresponding to each sentence, they appear in order of decreasing scores.

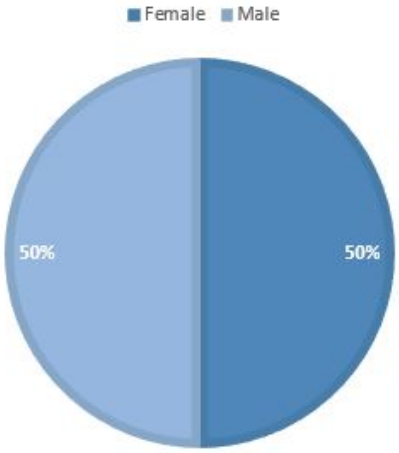
SENTENCES	ADJ_1	ADJ_2	ADJ_3	ADJ_4
She is driving a car	Enjoyable	Light	New	Great
I failed in an exam.	Possible	Better	Many	Failed
She succeeds as a group leader.	Competitive	Creative	Much	Easy
I am working at the company.	Hard	Extra	Corporate	Busy
She takes the exam.	Important	Necessary	Hard	Easy
I go swimming with my friend.	Hot	Great	Safe	Shallow
He forgot an appointment.	Important	Wrong	Free	Easy
People hunt the animals.	Environmental	Pervasive	Free	Common
She is smoking.	Harmful	Environmental	Toxic	Dust
I am listening to music.	Nice	Wonderful	Interesting	Enjoyable
He buys a car.	Used	New	Classic	Good
Flower is blooming.	Beautiful	Bright	Favorite	Wild
I walk home.	Short	Pleasant	Few	Able
I am drinking.	Alcoholic	Soft	Merry	Strong
Storm hits.	Vulnerable	Important	Magnetic	Bad
Summer comes.	Much	Hot	Moist	Temperate
She goes fishing.	Recreational	Agricultural	Small	Hot
We are cooking.	Free	Healthier	Temporary	Diabetic
She is making cake.	Cheese	Chocolate	Easy	Fresh
Winter comes.	Harsh	Beautiful	Gorgeous	Cold
Autumn comes.	Autumnal	Hot	Yellow	Comfortable
Spring comes.	Fresh	Natural	Beautiful	Colorful
Ice is melting.	Cold	Slippery	Antarctic	Cool

about them. Each invited user has to finish the survey set up with three parts: 1) general information, 2) relationship between adjectives and texts, and 3) relationship between adjectives and images and two level: 1) easy sentences, 2) difficult sentences.

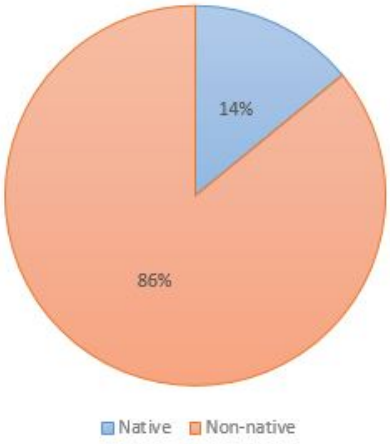
- Part 1: General information of participants.
- Part 2: Determine adjectives based on the provided texts. In this part, there are 80 texts (sentences) and a list of 5 adjectives for each sentence.



(a) Participants' age range.



(b) Participants' gender.



(c) Participants' Language.

Figure 4.6: Participants' information.

  * 5: All adjectives strongly relevant to his/her consideration.
  * 4: Almost relevant to his/her consideration.
  * 3: A half of the adjectives relevant to his/her consideration.
  * 2: Completely different from his her consideration.
  * 1: Difficult to imagine the adjectives.

- Part 3: For each text, the participants choose adjectives that may link to it (maybe more than one option). Then state what the relationship of the first word in the list is to the sentence in the scale from 1 to 10 (1: Not related at all and 10: Closely related). We created and distributed the survey to let

participants judge the output of the proposed method corresponding to the given inputs.

Moreover, the survey is also designed following 2 different levels of difficulties which are decided based on the number of the keywords of sentences. Sentences including less than 3 keywords are considered as easy ranking sentences, otherwise, they are difficult ones. The level of difficulties of sentences are not shown to the participants.

Evaluation result

1. **Adjective selection** The scores participants marked stretch from 1 to 5. Figure 4.7 shows the distribution of the evaluations. As can be seen from the table that the number of recommended adjectives chosen by participants in both difficult sentences and easy sentences are about 2.1. About 20.0% of the sentences have more than 4 or 5 appropriate adjective. Also, an independent t-test was also conducted to determine whether there is a significant difference between the evaluation of easy sentences and difficult sentences. Table 4.5 illustrates that there was no significant difference in the scores for easy sentences (M=2.06) and difficult sentences (M=2.13) conditions; t (773)=-0.97, p = 0.33. These results

Table 4.5: Descriptive statistics to show the difference between means of the output evaluations of two difficult levels of sentences.

	Easy	**Difficult**
Mean	2.06	2.13
Variance	1.42	1.56
Observation	681.00	387.00
Hypothesiszed Mean Difference	0.00	
df	773.00	
t Stat	-0.97	
P(T<=t) one-tail	0.17	
t Critical one-tail	1.65	
P(T<=t) two - tail	0.33	
t Critical two-tail	1.96	

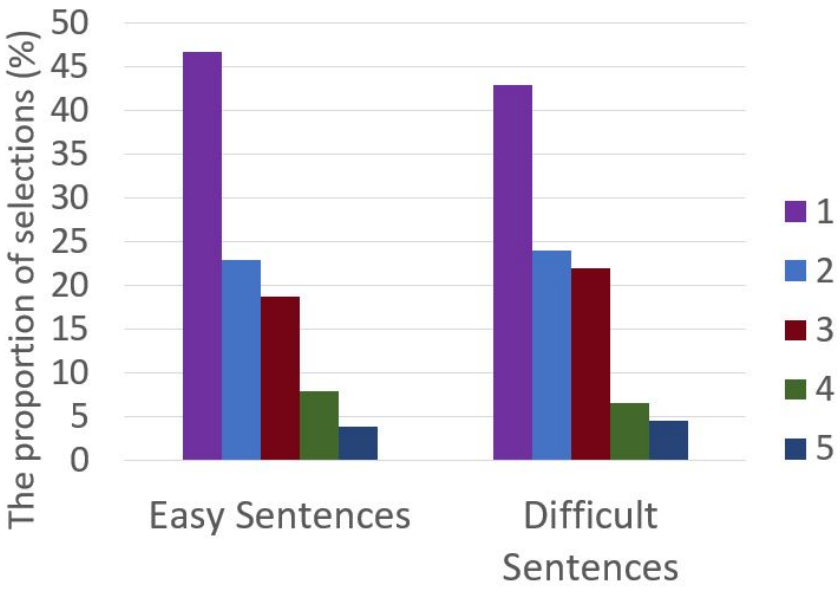


Figure 4.7: The level of agreement regarding how much the impression words matching with the sentences.

Table 4.6: Descriptive statistics to show the difference between means of the first word matching strength of two difficult levels of sentences.

	Easy	**Difficult**
Mean	6.57	7.14
Variance	9.29	7.70
Std	3.05	2.77
Low	3.52	4.37
High	10.09	11.51
Observation	681.00	387.00
Hypothesiszed Mean Difference	0.00	
df	866.00	
t Stat	-3.12	
P(T<=t) one-tail	0.00	
t Critical one-tail	1.65	
P(T<=t) two - tail	0.00	
t Critical two-tail	1.96	

suggest that level of difficulties of sentences really does not have an effect on the evaluation. Specifically, our results suggest that our proposed method always give about the same number of acceptable adjectives to users.

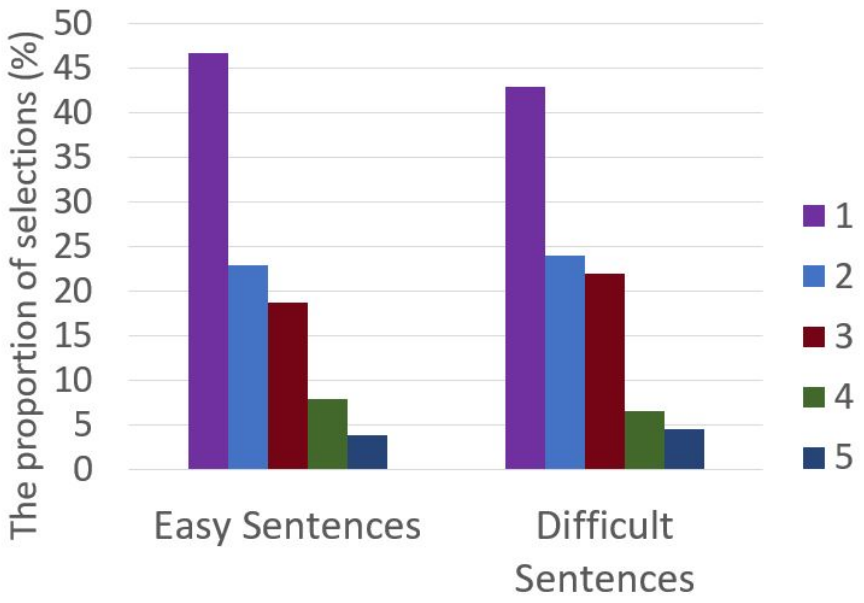


Figure 4.8: The level of agreement regarding how much strong the first word show the impression to the sentences.

2. **The association strength of the first word in the list to the sentence**

   Besides the judgements of the general output, we also asked the participants to evaluate the association strength of the first output adjective to the sentence. For a given sentence, the system generates a list of five adjectives, the first of which has the highest score. A survey was made in order to confirm the correlation between the high score and human perception of the context given sentence.

   Figure 4.8 shows that the first one the majority of participants chose has very strong relation with the sentence (about 7.0 score (10: perfect score)) in both difficult sentences and easy sentences. The result in the Table.4.6 provides a statistic to investigate the evaluation differences between 2 types of sentences. Since p-value $= 0.00 < .05 = \alpha$ we reject the null hypothesis. It is about 95% confident that there was a significant difference between the two groups. Specifically, the more complicated the sentences, and the more keywords we obtain, the better first adjective will be.

4.3 Summary

We have developed a method of impression estimation of a short sentence and proposed a combination of computational measurements with semantic relationship between words and sentences. The input is preprocessed to obtain keywords. Keywords are used to collect adjectives and the dataset preparing for the next steps. Then, their similarity score is calculated. After obtaining the association strength scores between adjectives and keywords, aggregation rank Borda's method is then adopted to sort the adjective candidates. Sorted adjectives sentiment are taken into consideration to compare their orientation with the input which aims at filtering adjectives opposite meaning with inputs. Finally, top n_a adjectives are considered as outputs of system.

After conducting the experiment, we distribute a survey to ask 64 participants to evaluate the results. The evaluation result shows that there are about two adjectives appropriated to the sentences and most of the first adjectives obtain the strong association with the sentence.

Chapter 5

Impression Estimation of an Image

5.1 Overview

In this chapter, we turn our attention to the issue of automatically achieving the impression of an image using our proposed method in Chapter 3. We propose a system that automatically shows "What comes into our mind after seeing a photo?". The challenge is to understand the topic of an image and show the impression of the image after the system viewing. Our focus is not only the possibility of the combination between the annotated tags of images like Table 5.1 on social webs and image classification approaches in discovering the keyword (topic) of the image, but also the challenge to apply the proposed method to estimate the impression of the image which becomes particularly important in this proposed system. Then, we continually apply our proposed method in Chapter 3 to obtain impression words of the images. The sentiment orientation of the given image and the impression words are then processed. Words having the same orientation are kept and ranked through the proposed ranking step in section 4.4.

The contributions of the proposed system in this chapter are:

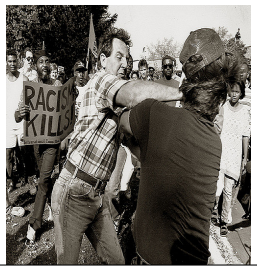
Image	Annotated tags	Image	Annotated tags
	• Flower • Flowers • Blackwhite • Texture ...		• Racism • Party • Monochrome ...
	• Cloud • Sky • Monochrome • Outdoor ...		• Racism • Fight • Hate • March ...

Table 5.1: Images with annotated tags.

- Combining an image processing method and image annotation to discover the topic of the image.
- Proposing a new approach using the measurement of the association between adjective and word to estimate the impression of the image.
- Modifying the method to measure their association combination between machine learning algorithm, SVM, for training and the similarity measurements for the proposed system.

5.2 Proposed system

Figure 5.1 shows the flow of the proposed system. Specifically, there are four main steps: image topic selection, adjective collection, semantic association measurement, and adjective selection. At first, the annotated tags of an image lead to ability to recognize the main context of an image. However, since there are a lot of ambiguous

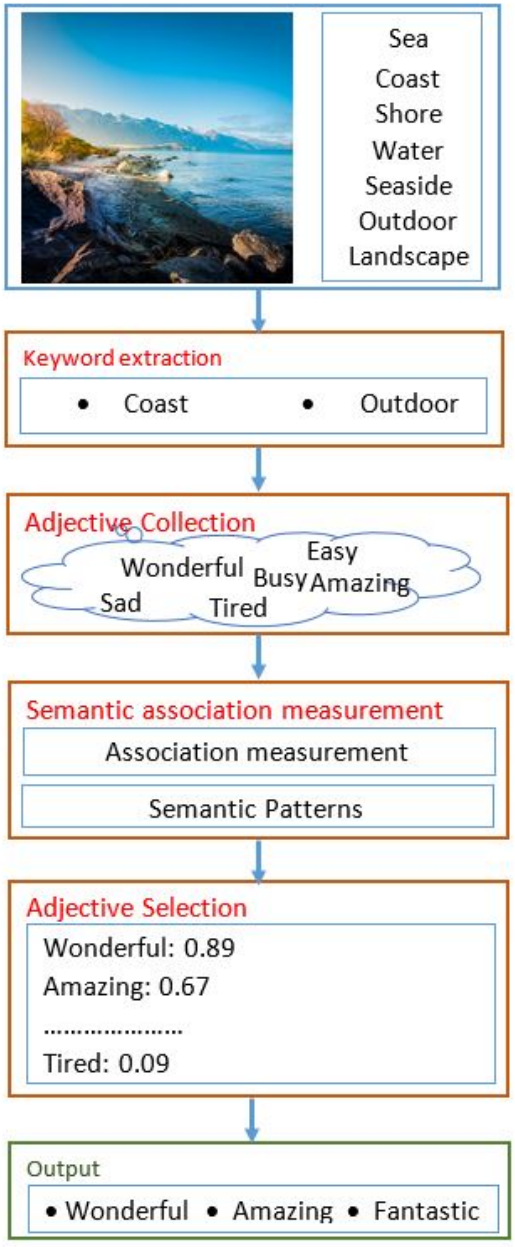


Figure 5.1: Flows of image impression system.

words, it is necessary to find the general idea of the image. To achieve this, we classify an image into categories which generalize the topic of the image using Bag of visual words (BOW) and SIFT. Then human annotated tags are considered as important information of the content of the image. In this step, the system selects n_k tags among the tags which have the highest relation with the category of the image as the

topic words. Selected topic candidates are used to query the N-gram utterances (N = 5 in this thesis) which are used in the next step - adjective candidates extraction. After obtaining keywords, we continue collection of adjectives which often appear with keywords using Google$N$-gram and use the proposed method to measure the associated scores between adjectives and keywords. Consequently, we sort adjectives for each keyword based on the scores, and then combine these orders together. In this step, we adopt the classical positional method, Borda's rank aggregation method to generate an acceptable ranking from computed rankings in the previous steps. The adjectives showing the impression of images can be considered as the output of our research. For example, when people see the picture shown in the Figure 5.2, they associate it with words like "wonderful", or "beautiful".

5.2.1 Keyword extraction

Figure 5.3 presents the flow of image keyword extraction which mainly aims at finding "looking at the photo, what do people stare at?". Given a collection of images, the proposed method automatically discovers the visual categories presented in the data and localize them in the image. To understand how the algorithm works, we analyze the flow specifically.

Figure 5.2: An example of an image.

Recently, social networking services such as Flickr [19] share photos and allow users to add tags freely. With pictures, a tag is an arbitrary word which associates with the image, and annotated by human aiming at describing a piece of image. In most images, there are tag words that are difficult to retrieve by image recognition. In fact, recent image annotation approaches have been proposed to narrow the semantic gap problem defined by [114] as the lack of coincidence between the low-level image descriptions using visual features and the richness of human semantics, i.e. "the interpretation that the image data has for a user in a given situation". According to

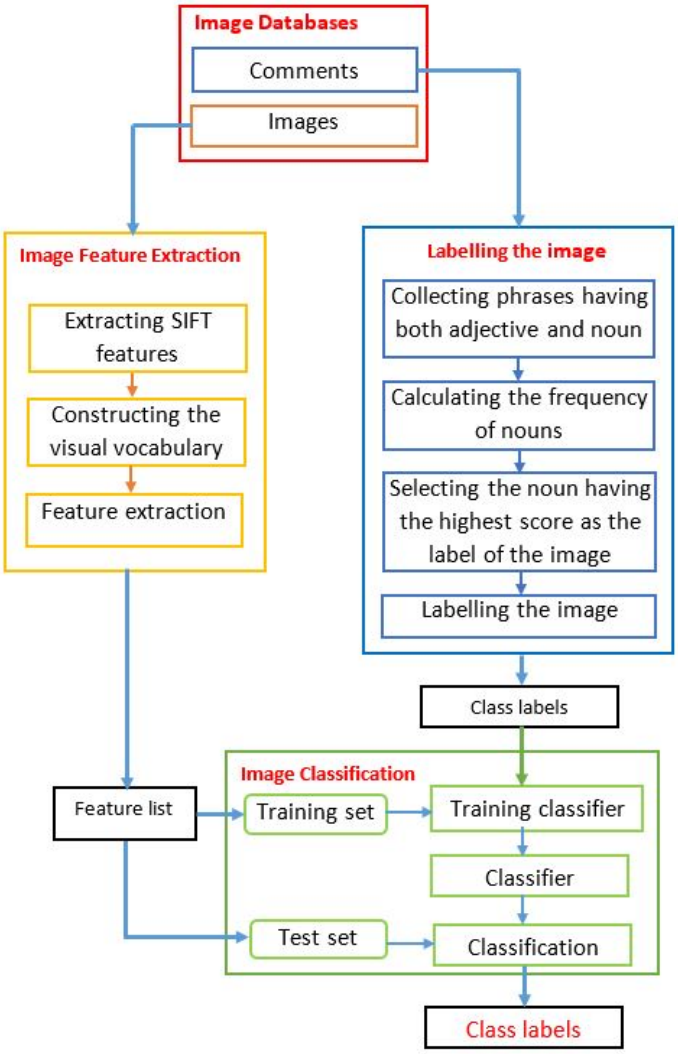


Figure 5.3: Overview of image topic extraction.

Figure 5.4: Vagueness of tags.

this definition, to understand the content of the images, in the scope of the research, the tags are used as a main tool to obtain the keyword.

Nevertheless, even if the notion of image semantics seems to be important for image retrieval related tasks, it is still vague and can vary according to the different approaches. For example, Figure 5.4 shows vagueness of tags. Indeed, to our knowledge, only few works have tried to provide a precise definition of the notion of image semantics. Therefore, we propose in this section a prestep to define the general idea of the image before taking a consideration to the tags.

General content of the image

We collected all of the phrases using N-gram from the comments dataset retrieved from Flickr. We believe that textual chunks involved both adjective and noun such as "dogs look lovely", "a sad face", "she is beautiful"are important and able to show the trend of the attention of the viewers for the objects of the images, so they are mainly analyzed and retrieved in the dataset. We collected all of nouns except for words happened frequently in every image such as "shot", "photo", "image", "composition", "portrait", and "capture". Then, we calculate the frequency of nouns. Finally, the one owning the highest frequency is chosen. We use Wordnet to find the highest class of the word and consider the class as the desired visual topic and used for labeling the images.

Image feature extraction

The steps involved in the training allow consideration of multiple possible vocabularies. Detection and description of image patches for a set of labeled training images using SIFT descriptors which depend on histograms of local orientation, gives some tolerance to illumination change. Vector quantization of these descriptors gives tolerance to morphology within an object category constructing a set of vocabularies (BOF): each is a set of cluster centers, with respect to which descriptors are vector quantized. The vector quantization is carried out here by k-means clustering computed from about 300 thousand regions. The regions are those extracted from random subsets. The number of clusters K helps to determine the words which give intra-class generalization.

1. **BOF**

   BOF [115, 116] is one of the popular visual descriptors used for visual data classification. BOF is inspired by a concept called Bag of Words that is used in document classification. A bag of words is a sparse vector of occurrence counts of words; that is, a sparse histogram over the vocabulary. BOF typically involves in the following main steps:

   - Select a large set of images.
   - Extract the SIFT feature points of all the images in the set and obtain the SIFT descriptor for each feature point that is extracted from each image.
   - Cluster the set of feature descriptors for the amount of bags we defined and train the bags with clustered feature descriptors (we can use the K-Means algorithm).
   - Obtain the visual vocabulary.

2. **Scale Invariant Feature Transform (SIFT)**

SIFT [117–119] has been shown to perform better than the other local descriptors. Given a feature point, the SIFT descriptor computes the gradient vector for each pixel in the feature point's neighborhood and builds a normalized histogram of gradient directions. The SIFT descriptor creates a 16x16 neighborhood that is partitioned into 16 subregions of 4x4 pixels each. For each pixel within a subregion, SIFT adds the pixel's gradient vector to a histogram of gradient directions by quantizing each orientation to one of 8 directions and weighting the contribution of each vector by its magnitude. Each gradient direction is further weighted by a Gaussian of scale $\sigma = \mathrm{n}/2$ where n is the neighborhood size and the values are distributed to neighboring bins using trilinear interpolation to reduce boundary effects as samples move between positions and orientations.

Image classification:

We employed the Support Vector Machine (SVM) framework with a Radial Basis Function kernel, in LIBSVM (a library for Support Vector Machines) [120] for the supervised learning of the different classes. The optimum scheme about the classifier can be obtained by through comparing the results. The SVM classifier provides a total accuracy which is the percentage of the correctly recognized classes.

Annotation based to obtain important information of the image

Usage of only labels attached to an image is not suitable to cover all of situations of impression expression of the image. Depending on the situation, or circumstance, the impressions of the images are different. Therefore, knowing more details about the objects will help the impression of the image be more accurate. For example, for the class of people, if we know what people are doing or what event they are joining such as "*bride*, *wedding*", showing the impression "*happy*" or "*beautiful*" is easier.

To get more visual content of the image, we consider that image annotations are the most helpful and the best solution. However, since manual tags are not always annotated into the images, to address the issue of the missing tags, we use Clarifai API [121] to do instead. It stores photos then uses machine learning to organize them with granular detail. It automatically generates tags, mining from a database of some 11,000 terms both concrete (tree, dogs, cake) and quite abstract (idyllic, togetherness, love).

Topic selection:

From the image tags, n_k words among them, that are likely related to the images, are selected. Words satisfying the following conditions are considered as important words to obtain: firstly, they are English words; secondly, they have a strong association with image's label. To address this task, we perform the following processes:

- Select n_k words among the input tag words and use Wordnet hierarchical structure to compare and cluster the concept of n_k words
- Calculate association score for two multivariate generalizations of pointwise mutual information between annotation clusters and the image label.
- Sort the results based on the scores.
- Select k concepts owning highest label and k concepts owning the lowest rank as the topic of the image. In this paper, k is 1.

5.2.2 Adjective collection

Impression words are collected using comment collection. All of the comments involved both adjective and noun such as "*cute baby*", "*amazing landscape* "are mainly analyzed and retrieved in the dataset. We collect all of adjectives except for those appearing with popular nouns such as "*shot*", "*photo*", "*image*", etc.

5.2.3 Semantic association measurements

Obtained keyword candidates are then combined with adjectives as pairs to measure. We utilize our proposed method in Chapter 3 to measure how much the adjective connecting to the keywords. The score for each candidate pair is outputted in this step.

5.2.4 Adjective selection

Depending on the scores attributed to the association of adjectives with the topic words, we continue to select the one which is the most connected to all of the words. First, we obtain t ordered list l_i of the adjective candidates corresponding with each keyword i. For each adjective (a_j) in (l_i), we assign a score as introduced in Chapter 4: $S_i(a_j) = |\{a_p : l_i(a_p) > l_i(a_j)\}|$.

Then, the candidates are sorted in a decreasing order of the total Borda score:

$$S(a_j) = \sum_{i=1}^{k} S_i(a_j) \tag{5.1}$$

Next, obtained sorted adjective list and keywords are one by one checked their sentiment orientations. Finally, top n_a adjectives having the same orientation with the keywords are outputted.

5.3 Experiment and evaluations

5.3.1 Experiment

Examples of output

Figure 5.5 shows an example of the processing of an image via our system. With a given input, the system classify the image into class "*people*"and tags are then one by

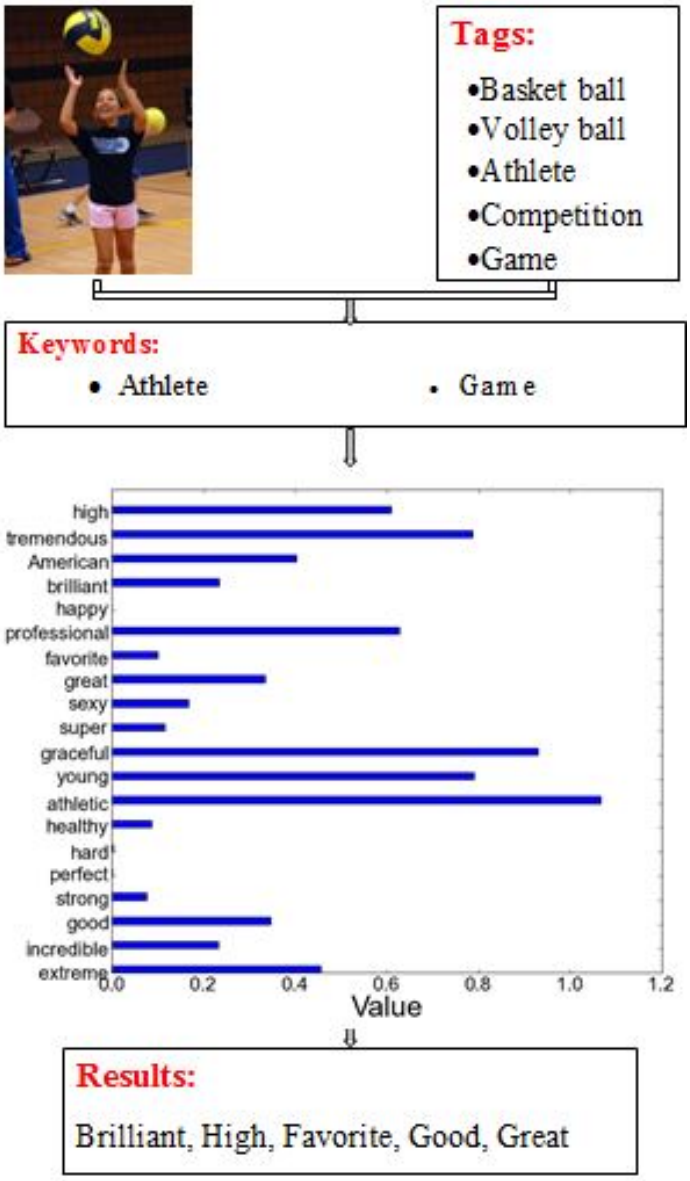


Figure 5.5: An example of a result.

one taken to check the strength of association with the input. Words that have the score higher than v_l are considered as the keyword candidates of the image. Adjectives are collected, combined with keywords, measure, sorted and outputted. In this case, the list of adjectives are: "Brilliant", "High", "Favorite", "Good", and "Great".

Tables 5.2 and 5.3 show some more examples of the images, the keyword we discovered, and an output of each image.

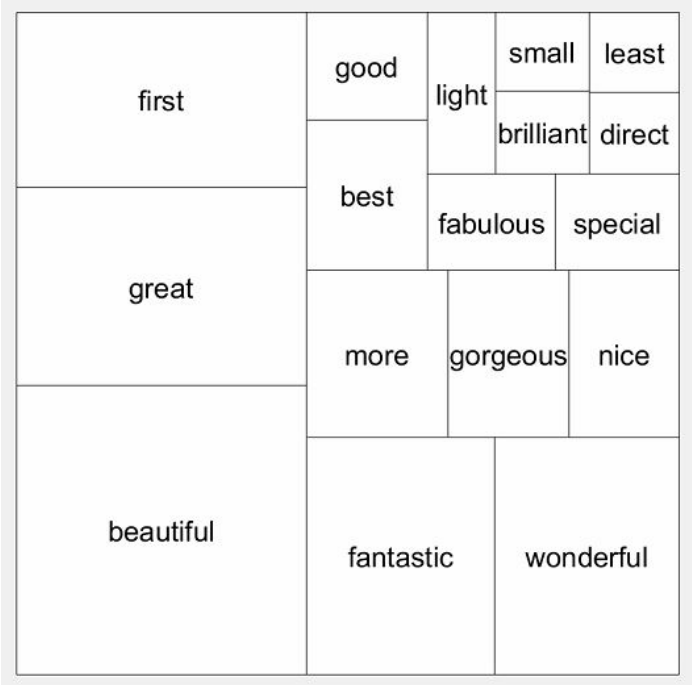


Figure 5.6: Frequency of the adjectives.

Considering Figure 5.6, we have summarized the popular adjectives outputted in the proposed system.

Table 5.2: Some example results of the image impression proposed system (1).

Image	Keywords	Annotated
	• Meat • Food • Barbecue ...	• Hot • Good • Fine • Sweet • Favorite ...
	• Dance • People ...	• Fun • Different • Happy • Hot • Fine ...
	• People • Cook ...	• Delicious • Hot • Good • Warm • Entertaining ...
	• Bride • Wedding ...	• Fun • Good • Ideal • Sweet • Big ...

Table 5.3: Some example results of the image impression proposed system (2).

Image	Keywords	Annotated
	• Sunrise • Landscape ...	• Busy • Cool • Hot • Romantic • Natural ...
	• Food • Fruit ...	• Hot • Fresh • Sweet • Soft • Good ...
	• Flower • Cheery • Blossom ...	• Beautiful • Green • Floral • Sweet • Favorite ...
	• Singer • Concert ...	• Great • Musical • Fun • Good • Happy ...

Dataset

We evaluate the proposed system on the images chosen randomly from an image hosting Website and online community platform, namely Flickr dataset [122]. The images are accompanied by rich surrounding textual descriptions including titles, tags, and categories. These photos were uploaded between 2004 and 2007. Flickr [122] offers tags of the images. The obtained collection consists of more than 188 million tags in total and about 3 million unique tags. The distribution of the number of tags in photos is shown in Figure 5.7. On average, each image has been associated with about 8 tags. However, based on the distribution of the number of tags in photos, it can be seen that there are many photos having no or only 1 annotation which seems not so informative for the proposed system mainly focusing on tags. Moreover, because of the requirement of the input: image and tags, it is better to use automatically annotating image tools to assist annotating images, and to help our evaluation work smoothly. Here, we use Clarifai [121] as an assistance example to annotate image and to input the system.

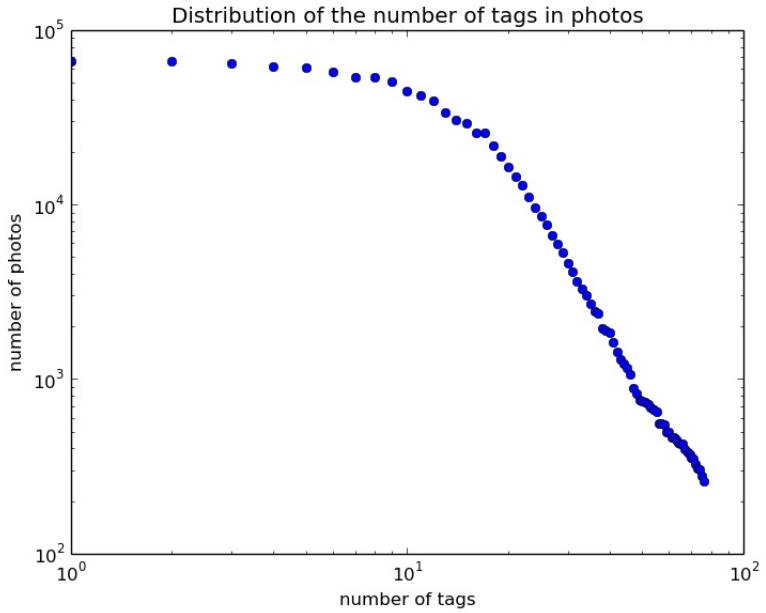
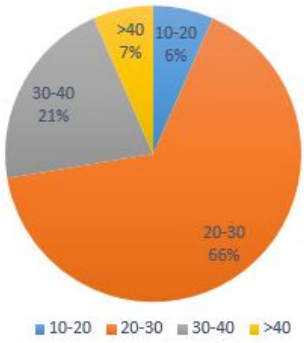


Figure 5.7: Distribution of annotated tags of Flickr database.

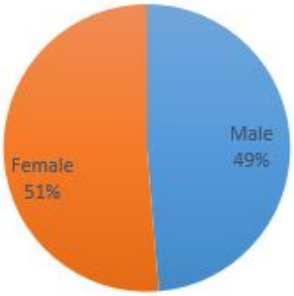
After collecting images from Flickr and labeling images in the datasets, 6 categories chosen to do the experiment are landscape, city, people, animal, flower, and food. The number of images in each category varies from 1500 to 2500.

We executed this process on 200 downloaded images. These datasets were used for the experiment. In order to evaluate the effectiveness of the proposed method, we conducted an experiment. The conditions of the experiment are as following:

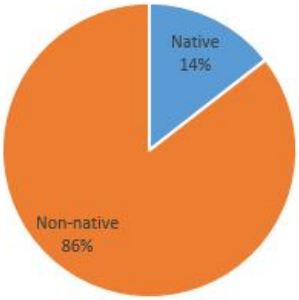
- Use the downloaded images obtained from Flickr.
- The maximum of the keywords are 5 words.
- Evaluated by 76 participants from different nations and ages. The information of the participants are shown in Figure 5.8.
- Evaluated 75 generated outputs of images randomly selected in 6 categories.



(a) Participants' age range.



(b) Participants' gender.



(c) Participants' Language.

Figure 5.8: Participants' information for image evaluation.

- Had to imagine the ordinary value range to indicated pair of image and keywords, and pair of image and adjectives based on his or her common sense. The evaluation criteria are as follows:

    5: All adjectives strongly relevant to his/her consideration.

    4: Almost relevant to his/her consideration.

    3: A half of the adjectives relevant to his/her consideration.

    2: Completely different from his her consideration.

    1: Difficult to imagine the adjectives.

- Stated what the relationship of the first word in the list is to the sentence in the scale from 1 to 10 (1-Not related at all and 10-Closely related).

5.3.2 Evaluation results

For the image classification task, we used the result to discover the keyword of the image. Table 5.4 shows the confusion table for the results of the task. In the confusion table, the rows represent the models for each class while the columns represent the ground truth categories. From the detailed result, it can be seen that the best classified classes are flower and street which share the same performance 86.0%. It means they had more outstanding scenes than the other images. The Bag of SIFT [117–119] can describe their difference decently and make them easier to be estimated. On the other hand, the classes Animal and People perform more poorly (66.0% and 70.0% respectively). We can see that the estimation cannot differentiate Animal and People very well (confused accuracy is 17.0%) and this causes the low accuracy of estimation of these two labels. Apparently, the obtainable accuracy is strongly affected by the consistency and accuracy of images belonging to the class, and sometimes the confusion of the image labels are quite popular. In this case, this percentage can be explained by the fact that in our image data to evaluate, among the Animal images,

Table 5.4: Confusion matrix for six categories in the dataset.

	Animal	Flower	Food	Landscape	People	Street
Animal	0.66	0.01	0.06	0.12	0.15	0.00
Flower	0.01	0.86	0.01	0.01	0.06	0.05
Food	0.09	0.02	0.75	0.06	0.08	0.00
Landscape	0.07	0.01	0.06	0.80	0.06	0.00
People	0.17	0.03	0.04	0.60	0.70	0.00
Street	0.01	0.10	0.00	0.01	0.02	0.86

there are many images containing a fair number of people whereas some People images also include some of animals. In addition, the scenes from Animal category were similar to People. Therefore, these reasons made the system difficult to differentiate.

1. **Adjective selection** The score participants marked stretches from 1 to 5. Figure 5.9 shows the distribution of the evaluations. In fact, images used for the evaluation task were clustered into 2 groups: easy and difficult. The easy one has less then 3 keywords and the difficult one has equal or greater than 3. As can be seen from the Table 5.5 that the number of recommended adjectives chosen by participants in both difficult and easy images are about two words. About 10% of the sentences have more than 4 or 5 appropriate adjectives. To determine whether there is a significant difference between the evaluation of two groups of images, we conducted an independent t-test. Table 5.5 shows that there is no significant difference in the scores for easy images (M=2.02) and difficult images (M=2.10); t (776)= -1.00, p = 0.32. These results suggest that level of difficulties of images really does not have an effect on the evaluation.

2. **The association strength of the first word in the list to the sentence** Besides the evaluation of the general outputs, we further evaluate the association strength of the first output adjective to the image. For a given image, the system generates a list of five adjectives, the first of which has the highest score. A

Table 5.5: Descriptive statistics to show the difference between means of the output evaluations of two difficult levels of images.

	Easy	**Difficult**
Mean	2.02	2.10
Variance	1.46	1.52
Observation	776.00	367.00
Hypothesiszed Mean Difference	0.00	
df	704.00	
t Stat	−1.00	
P(T<=t) one-tail	0.16	
t Critical one-tail	1.65	
P(T<=t) two - tail	0.32	
t Critical two-tail	1.96	

survey was made in order to confirm the correlation between the high score and human perception of the image.

Figure 5.10 shows that the word that the majority of the participants chose as the first are very strongly related with the images (about 7 score over 10) in both two cases. The result in Table 5.6 provides a statistic to investigate the evaluation differences between 2 types of images. Since p-value = $0.49 > .05 = \alpha$ we obtain the null hypothesis. We assume that there is no significant difference between the two groups.

We have done some preliminary experiments and the results show that our system provides promising results of impression estimation. Nevertheless, an obvious limitation of this system is that in the adjective selection part, adjectives are not organized well, so it leads to some drawbacks such as time consuming, or not well connection between output and input. Classification of adjectives should be taken into consideration as an important step and added in the future work.

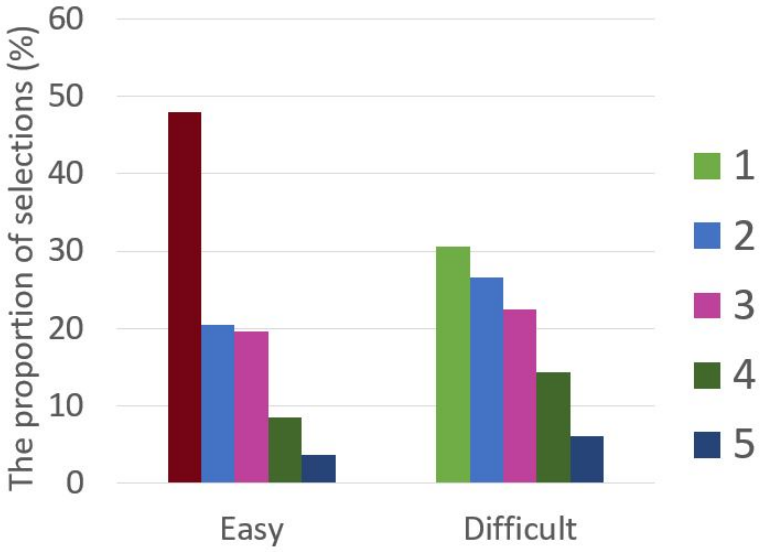


Figure 5.9: The level of agreement regarding how much the impression words matching with the images.

Table 5.6: Descriptive statistics to show the difference between means of the first word matching strength of two difficult levels of images.

	Easy	**Difficult**
Mean	6.56	6.68
Variance	8.43	7.32
Std	2.90	2.71
Low	3.66	3.97
High	10.22	10.65
Observation	777.00	387.00
Hypothesiszed Mean Difference	0.00	
df	821.00	
t Stat	−0.69	
P(T<=t) one-tail	0.24	
t Critical one-tail	1.65	
P(T<=t) two - tail	0.49	
t Critical two-tail	1.96	

5.4 Summary

In this chapter, we proposed a system to estimate impression of images. In order to obtain the image information, we used human annotated tags and image classification method. We combine the computational measurements with semantic relationship between words and adjective to select the one matching with the input the most.

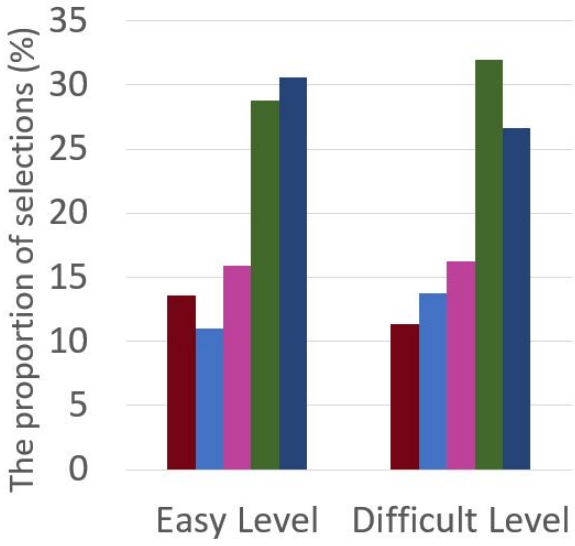


Figure 5.10: The level of agreement regarding how much strong the first word show the impression to the images.

After obtaining the association strength scores between adjectives and keywords, aggregation rank Bordas method is then adopted to sort the adjective candidates. Finally, top n_a adjectives are considered as an output of the system. Experimental results have shown the effectiveness of the proposed system in landscape and flower classes (precision range from 80.0% to 86.0%) in topic selection.

Chapter 6

Conclusion

In this chapter, we briefly summarize the main contributions of this thesis and discuss some further works.

6.1 Conclusion

This thesis has addressed the problem of semantic similarity between different forms of words. To this end, we have created a database of word-adjective pairs included association pairs and non-association pairs. The major contributions of this thesis are:

- Datasets: In chapter 3, we created two new datasets: noun-adjective pairs and verb-adjective association pairs. Both were collected from different sources and evaluated by 10 people.

- Semantic similarity: Also, we proposed a new approach of sematic similarity to estimate the impression of words. We adopted the semantic similarity measurement method [92] which takes lexical patterns into account as feature vectors together with corpus-based measurements. Because of different usages of forms of words with adjectives, depending on each case, they are treated differently to

collect patterns which are considered as features of vector together with corpus-based measurements. Collected patterns are then clustered into groups, aiming at improving the estimation performance and reducing noisy data. We used a SVM classifier and explored the performance of the proposed method towards different sizes of vector features. Finally, we obtained the best result for both noun-adjective and verb-adjective pairs when the vector feature length is about 30.0% of the number of patterns.

- Impression estimation for sentences: We approach our proposed method into a system which shows the impression of sentences. There are four main steps in this work: keyword extraction, adjective collection, semantic association measurement, and finally adjective selection to get the best result. A survey was conducted to measure how much the output matches with the common perspective. There are about two adjectives appropriated to the sentences (2.0 (5: perfect score)) and most of the first adjectives obtain the strong association with the sentence (7.0 (10: perfect score)).

- Impression estimation for images: We applied the our proposed method to visual impression estimation, aiming to learn from a set of training images and annotated-tags to output *"what viewers glance at"*. The topic of the image is then used to measure the semantic similarity with the list of adjectives collected from comments of a dataset of images. Finally, all of the results are aggregated and outputted. The top n_a outputs are shown as the most appealing impression of images. We evaluated the proposed method from two different perspectives: topic, and result. For the topic evaluation, we used the dataset containing six categories: flower, people, landscape, food and street. We take into account the classification method and user annotations to extract the keywords. The precision of class flower and street obtained the highest score (86.0%). Furthermore,

we distributed a survey to participants for the impression words obtained from the system. It is shown that participants agreed that most of the first impression words have strong association with the images, with the average score of 7.0 (10.0: perfect score).

6.2 Future Work

We will now present possible direction for future research. We have proposed a framework to estimate semantic similarity between adjectives and different forms of words. This framework can now be easily extended to many applications of Kansei or affective computing area. However, the next step to group adjectives having similar features and to learn the best combination should be considered more as further works. Since there are many adjectives locating at the same categories in the real life such as *"beautiful "* and *"pretty "*, taking clustering adjectives into account is also necessary and should be considered as further work.

Moreover, impressions are subjective experiences and as such, even if the object observed or story are the same, they can differ from person to person. However as a first step we address here, the problem needs to have more further consideration.

Bibliography

[1] Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. Generating typed dependency parses from phrase structure parses. In *Proceedings of International Conference on Language Resources and Evaluation (LREC)*, volume 6, pages 449–454, 2006.

[2] Martin Solli and Reiner Lenz. Color emotions for image classification and retrieval. In *Conference on Colour in Graphics, Imaging, and Vision*, volume 2008, pages 367–371. Society for Imaging Science and Technology, 2008.

[3] Damian Borth, Rongrong Ji, Tao Chen, Thomas Breuel, and Shih-Fu Chang. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 223–232. ACM, 2013.

[4] Jianbo Yuan, Sean Mcdonough, Quanzeng You, and Jiebo Luo. Sentribute: image sentiment analysis from a mid-level perspective. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, pages 1–8. ACM, 2013.

[5] Wei Peng and Dae Hoon Park. Generate adjective sentiment dictionary for social media sentiment analysis using constrained nonnegative matrix factorization. pages 273–280, 2011.

[6] Farah Benamara, Carmine Cesarano, Antonio Picariello, Diego Reforgiato Recupero, and Venkatramana S Subrahmanian. Sentiment analysis: Adjectives and adverbs are better than adjectives alone. In *International Conference on Weblogs and Social Media (ICWSM)*. Citeseer, 2007.

[7] Naho Ito and Masafumi Hagiwara. Image description generation without image processing using fuzzy inference. In *Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on*, pages 1–8. IEEE, 2012.

[8] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

[9] Alexander Budanitsky and Graeme Hirst. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47, 2006.

[10] Gaurav Kulkarni, Visruth Premraj, Vicente Ordonez, Sudipta Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara Berg. Babytalk: Understanding and generating simple image descriptions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):2891–2903, 2013.

[11] Jean Aitchison, Rodney Huddlestone, and Geoffrey K Pullum. *The Cambridge Grammar of the English Language*. JSTOR, 2003.

[12] Rodney Huddleston, Geoffrey K Pullum, et al. The cambridge grammar of english. *Language. Cambridge: Cambridge University Press*, pages 1–23, 2002.

[13] Randolph Quirk, David Crystal, and Pearson Education. *A comprehensive grammar of the English language*, volume 397. Cambridge Univ Press, 1985.

[14] Mark Baker. *Lexical categories: Verbs, nouns and adjectives*, volume 102. Cambridge University Press, 2003.

[15] Michael Gasser and Linda B Smith. Learning nouns and adjectives: A connectionist account. *Language and cognitive processes*, 13(2-3):269–306, 1998.

[16] Ningning Liu, Emmanuel Dellandrea, Bruno Tellez, and Liming Chen. Associating textual features with visual ones to improve affective image classification. In *Affective Computing and Intelligent Interaction*, pages 195–204. Springer, 2011.

[17] Shenghua Bao, Shengliang Xu, Li Zhang, Rong Yan, Zhong Su, Dingyi Han, and Yong Yu. Mining social emotions from affective text. *Knowledge and Data Engineering, IEEE Transactions on*, 24(9):1658–1670, 2012.

[18] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM, 2004.

[19] Minqing Hu and Bing Liu. Mining opinion features in customer reviews. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence*, volume 4, pages 755–760, 2004.

[20] Soo-Min Kim and Eduard Hovy. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, pages 1367–1373. Association for Computational Linguistics, 2004.

[21] Vasileios Hatzivassiloglou and Kathleen R McKeown. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th annual meeting of the association for computational linguistics and eighth conference of the european chapter of the association for computational linguistics*, pages 174–181. Association for Computational Linguistics, 1997.

[22] Peter Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 417–424. Association for Computational Linguistics, 2002.

[23] Hang Cui, Vibhu Mittal, and Mayur Datar. Comparative experiments on sentiment classification for online product reviews. In *Association for the Advancement of Artificial Intelligence (AAAI)*, volume 6, pages 1265–1270, 2006.

[24] Mikhail Bautin, Lohit Vijayarenu, and Steven Skiena. International sentiment analysis for news and blogs. In *International Conference on Weblogs and Social Media (ICWSM)*, 2008.

[25] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics, 2002.

[26] Bo Pang. *Automatic Analysis of Document Sentiment*. PhD thesis, Cornell University, 2006.

[27] Bing Liu. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167, 2012.

[28] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354. Association for Computational Linguistics, 2005.

[29] Andrea Esuli and Fabrizio Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of International Conference on Language Resources and Evaluation (LREC)*, volume 6, pages 417–422. Citeseer, 2006.

[30] Jaap Kamps, MJ Marx, Robert J Mokken, M de Rijke, et al. Using wordnet to measure semantic orientations of adjectives. In *Proceedings of International Conference on Language Resources and Evaluation (LREC)*, pages 1115–1118. European Language Resources Association (ELRA), 2004.

[31] Ioana Muresan, Andrei Stan, Mircea Giurgiu, and Rodica Potolea. Evaluation of sentiment polarity prediction using a dimensional and a categorical approach. In *Speech Technology and Human-Computer Dialogue (SpeD), 2013 7th Conference on*, pages 1–6. IEEE, 2013.

[32] Roni Rosenfeld and Philip Clarkson. Statistical language modeling using the cmu-cambridge toolkit. In *Eurospeech*, pages 2707–2710, 1997.

[33] Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. Online passive-aggressive algorithms. *The Journal of Machine Learning Research*, 7:551–585, 2006.

[34] Kamal Nigam and Matthew Hurst. Towards a robust metric of opinion. In *AAAI spring symposium on exploring attitude and affect in text*, pages 598–603, 2004.

[35] Lun-wei Ku, Yong-sheng Lo, and Hsin-hsi Chen. Using polarity scores of words for sentence-level opinion extraction. In *Proceedings of NTCIR-6 workshop meeting*, pages 316–322. Citeseer, 2007.

[36] Alena Neviarouskaya, Helmut Prendinger, and Mitsuru Ishizuka. Recognition of affect, judgment, and appreciation in text. In *Proceedings of the 23rd International Conference on Computational Linguistics*, International Conference on Computational Linguistics COLING '10, pages 806–814, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[37] Alena Neviarouskaya, Helmut Prendinger, and Mitsuru Ishizuka. Textual affect sensing for sociable and expressive online communication. In *Affective Computing and Intelligent Interaction*, pages 218–229. Springer, 2007.

[38] Paul Ekman. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200, 1992.

[39] Carlo Strapparava, Alessandro Valitutti, et al. Wordnet affect: an affective extension of wordnet. In *Proceedings of International Conference on Language Resources and Evaluation (LREC)*, volume 4, pages 1083–1086, 2004.

[40] Hugo Liu and Push Singh. Conceptneta practical commonsense reasoning toolkit. *BT technology journal*, 22(4):211–226, 2004.

[41] Carlo Strapparava and Rada Mihalcea. Semeval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 70–74. Association for Computational Linguistics, 2007.

[42] Guillermo Campitelli and Fernand Gobet. Herbert simon's decision-making approach: Investigation of cognitive processes in experts. *Review of General Psychology*, 14(4):354, 2010.

[43] Stefan Siersdorfer, Sergiu Chelaru, Wolfgang Nejdl, and Jose San Pedro. How useful are your comments?: analyzing and predicting youtube comments and comment ratings. In *Proceedings of the 19th international conference on World wide web*, pages 891–900. ACM, 2010.

[44] Martin Solli and Reiner Lenz. Color emotions for multi-colored images. *Color Research & Application*, 36(3):210–221, 2011.

[45] Shigenobu Kobayashi. The aim and method of the color image scale. *Color research & application*, 6(2):93–107, 1981.

[46] Osvaldo Da Pos and Paul Green-Armytage. Facial expressions, colours and basic emotions. *JAIC-Journal of the International Colour Association*, 1:1–20, 2012.

[47] John Xin, KM Cheng, Gale Taylor, Tetsuya Sato, and Aran Hansuebsai. Cross-regional comparison of colour emotions part ii: Qualitative analysis. *Color Research & Application*, 29(6):458–466, 2004.

[48] Jana Machajdik and Allan Hanbury. Affective image classification using features inspired by psychology and art theory. In *Proceedings of the international conference on Multimedia*, pages 83–92. ACM, 2010.

[49] Lyndon S Kennedy, Shih-Fu Chang, and Igor V Kozintsev. To search or to label?: predicting the performance of search-based automatic image classifiers. In *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, pages 249–258. ACM, 2006.

[50] Adrian Ulges, Christian Schulze, Markus Koch, and Thomas M Breuel. Learning automatic concept detectors from online video. *Computer vision and Image understanding*, 114(4):429–438, 2010.

[51] Ahmet Aker and Robert Gaizauskas. Generating image descriptions using dependency relational patterns. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 1250–1258. Association for Computational Linguistics, 2010.

[52] Yezhou Yang, Ching Lik Teo, Hal Daumé III, and Yiannis Aloimonos. Corpus-guided sentence generation of natural images. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 444–454. Association for Computational Linguistics, 2011.

[53] Polina Kuznetsova, Vicente Ordonez, Alexander C Berg, Tamara L Berg, and Yejin Choi. Collective generation of natural image descriptions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 359–368. Association for Computational Linguistics, 2012.

[54] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. Every picture tells a story: Generating sentences from images. In *Computer Vision–ECCV 2010*, pages 15–29. Springer, 2010.

[55] Yu Zhang, Stephane Bres, and Liming Chen. Semantic bag-of-words models for visual concept detection and annotation. In *Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference on*, pages 289–295. IEEE, 2012.

[56] Hao Ma, Jianke Zhu, Michael Rung-Tsong Lyu, and Irwin King. Bridging the semantic gap between image contents and tags. *Multimedia, IEEE Transactions on*, 12(5):462–473, 2010.

[57] Yahong Han, Xingxing Wei, Xiaochun Cao, Yi Yang, and Xiaofang Zhou. Augmenting image descriptions using structured prediction output. *Multimedia, IEEE Transactions on*, 16(6):1665–1676, 2014.

[58] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015.

[59] Ankush Gupta and Prashanth Mannem. From image annotation to image description. In *Neural information processing*, pages 196–204. Springer, 2012.

[60] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, volume 14, pages 1532–1543, 2014.

[61] Margaret Mitchell, Xufeng Han, Jesse Dodge, Alyssa Mensch, Amit Goyal, Alex Berg, Kota Yamaguchi, Tamara Berg, Karl Stratos, and Hal Daumé III. Midge: Generating image descriptions from computer vision detections. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 747–756. Association for Computational Linguistics, 2012.

[62] Ray Richardson, A Smeaton, and John Murphy. Using wordnet as a knowledge base for measuring semantic similarity between words. Technical Report Working Paper CA-1294, School of Computer Applications, Dublin City University, 1994.

[63] Courtney Corley and Rada Mihalcea. Measuring the semantic similarity of texts. In *Proceedings of the ACL workshop on empirical modeling of semantic equivalence and entailment*, pages 13–18. Association for Computational Linguistics, 2005.

[64] Claudia Leacock and Martin Chodorow. Combining local context and WordNet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283, 1998.

[65] Christiane Fellbaum. WordNet: An Electronic Lexical Database. *MIT Press*, 1998.

[66] Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. Wordnet:: Similarity: measuring the relatedness of concepts. In *Demonstration papers at HLT-NAACL 2004*, pages 38–41. Association for Computational Linguistics, 2004.

[67] Graeme Hirst and David St-Onge. Lexical chains as representations of context for the detection and correction of malapropisms. *WordNet: An electronic lexical database*, pages 305–332, 1998.

[68] Dongqiang Yang and David MW Powers. Verb similarity on the taxonomy of wordnet. In *In the 3rd International WordNet Conference (GWC-06), Jeju Island, Korea*, pages 121–128. Masaryk University, 2006.

[69] Philip Resnik and Mona Diab. Measuring verb similarity. In *Proceedings of the 22nd Annual Meeting of the Cognitive Science Society*, pages 399–404. Philadelphia, PA, 2000.

[70] Jiang Jay and Conrath David. Semantic similarity based on corpus statistics and lexical taxonomy. *Computing Research Repository (CoRR)*, cmp-lg/9709008, 1997.

[71] Dekang Lin. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics-Volume 2*, pages 768–774. Association for Computational Linguistics, 1998.

[72] Wael Gomaa and Aly Fahmy. A survey of text similarity approaches. *International Journal of Computer Applications*, 68(13), 2013.

[73] Siddharth Patwardhan, Satanjeev Banerjee, and Ted Pedersen. Using measures of semantic relatedness for word sense disambiguation. In *Computational linguistics and intelligent text processing*, pages 241–257. Springer, 2003.

[74] Sandeep Tata and Jignesh M Patel. Estimating the selectivity of tf-idf based cosine similarity predicates. *ACM Sigmod Record*, 36(2):7–12, 2007.

[75] Juan Ramos. Using tf-idf to determine word relevance in document queries. In *Technical report*, Department of Computer Science, Rutgers University, 2003.

[76] Ainura Madylova. A taxonomy based semantic similarity of documents using the cosine measure. In *Computer and Information Sciences, 2009. ISCIS 2009. 24th International Symposium on*, pages 129–134. IEEE, 2009.

[77] Peter Foltz, Walter Kintsch, and Thomas K Landauer. The measurement of textual coherence with latent semantic analysis. *Discourse processes*, 25(2-3):285–307, 1998.

[78] Akiko Aizawa. An information-theoretic perspective of tf–idf measures. *Information Processing & Management*, 39(1):45–65, 2003.

[79] Gabriel Murray, Steve Renals, and Jean Carletta. Extractive summarization of meeting recordings. In *Proceedings, Interspeech'2005 - Eurospeech, 9th European Conference on Speech Communication and Technology*, pages 1–3, 2005.

[80] Ani Nenkova, Sameer Maskey, and Yang Liu. Automatic summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts of ACL 2011*, page 3. Association for Computational Linguistics, 2011.

[81] Thomas Landauer and Susan T Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211, 1997.

[82] Simon Dennis, Tom Landauer, Walter Kintsch, and Jose Quesada. Introduction to latent semantic analysis. In *Slides from the tutorial given at the 25th Annual Meeting of the Cognitive Science Society*, 2003.

[83] Rudi Cilibrasi and Paul Vitanyi. The google similarity distance. *Knowledge and Data Engineering, IEEE Transactions on*, 19(3):370–383, 2007.

[84] Rudi Cilibrasi and Paul Vitanyi. Normalized web distance and word similarity. *Computing Research Repository (CoRR)*, abs/0905.4039, 2009.

[85] Rudi Cilibrasi and Paul Vitanyi. Automatic meaning discovery using google. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2006.

[86] Jorge Gracia and Eduardo Mena. Web-based measure of semantic relatedness. In *Web Information Systems Engineering-WISE 2008*, pages 136–150. Springer, 2008.

[87] Aminul Islam, Evangelos E. Milios, and Vlado Keselj. Comparing word relatedness measures based on google n-grams. In *International Conference on Computational Linguistics COLING (Posters)*, pages 495–506, 2012.

[88] Lou Burnard. The british national corpus users reference guide. MIT Press, Cambridge, MA, USA, 2000.

[89] Mehran Sahami and Timothy D Heilman. A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th international conference on World Wide Web*, pages 377–386. Association for Computing Machinery(ACM), 2006.

[90] Andrew McCallum, Xuerui Wang, and Andrés Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *Journal of artificial intelligence research*, pages 249–272, 2007.

[91] Yuhua Li, Zuhair A. Bandar, and David McLean. An approach for measuring semantic similarity between words using multiple information sources. *IEEE Trans. on Knowl. and Data Eng.*, 15(4):871–882, July 2003.

[92] Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka. A web search engine-based approach to measure semantic similarity between words. *Knowledge and Data Engineering, IEEE Transactions on*, 23(7):977–990, 2011.

[93] David Urbansky, Marius Feldmann, James A Thom, and Alexander Schill. Entity extraction from the web with webknox. In *Advances in Intelligent Web Mastering-2*, pages 209–218. Springer, 2010.

[94] Steffen Roth. Fashionable functions: A google ngram view of trends in functional differentiation (1800-2000). *International Journal of Technology and Human Interaction*, 10(2):34–58, 2014.

[95] Pankaj K Agarwal and Cecilia Magdalena Procopiuc. Exact and approximation algorithms for clustering. *Algorithmica*, 33(2):201–226, 2002.

[96] John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

[97] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. An efficient k-means clustering algorithm: Analysis and implementation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):881–892, 2002.

[98] Corinna Cortes and Vladimir Vapnik. Support vector machine. *Machine learning*, 20(3):273–297, 1995.

[99] Douglas Nelson, Cathy McEvoy, and Thomas Schreiber. The university of south florida free association, rhyme, and word fragment norms. *Behavior Research Methods, Instruments, & Computers*, 36(3):402–407, 2004.

[100] Yuriy Rotmistrov. Word Association Network. Available from CTAN, macros/latex/contrib/booktabs, `http://wordassociations.net/`, 2006-2016.

[101] Javed Aslam and Mark Montague. Models for metasearch. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 276–284. ACM, 2001.

[102] James Callan, Zhihong Lu, and W Bruce Croft. Searching distributed collections with inference networks. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 21–28. ACM, 1995.

[103] Edward Fox and Joseph A Shaw. Combination of multiple searches. *Nist special publication*, pages 243–243, 1994.

[104] Joon Ho Lee. Analyses of multiple evidence combination. In *ACM SIGIR Forum*, volume 31, pages 267–276. ACM, 1997.

[105] Raghavan Manmatha, T Rath, and Fangfang Feng. Modeling score distributions for combining the outputs of search engines. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 267–275. ACM, 2001.

[106] Christopher Vogt and Garrison Cottrell. Fusion via a linear combination of scores. *Information retrieval*, 1(3):151–173, 1999.

[107] Voorhees, Narendra Gupta, and Ben Johnson-Laird. The collection fusion problem. *Nist special publication (SP)*, pages 95–95, 1995.

[108] Ronald R Yager and Alexander Rybalov. On the fusion of documents from multiple collection information retrieval systems. *Journal of the American Society for Information Science*, 49(13):1177–1184, 1998.

[109] Peyton Young and Arthur Levenglick. A consistent extension of condorcet's election principle. *SIAM Journal on applied Mathematics*, 35(2):285–300, 1978.

[110] Peyton Young. An axiomatization of borda's rule. *Journal of Economic Theory*, 9(1):43–52, 1974.

[111] Cynthia Dwork, Ravi Kumar, Moni Naor, and D Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th International World Wide Web Conference*, pages 613–622, 2001.

[112] Fabrizio Sebastiani and Andrea Esuli. Determining term subjectivity and term orientation for opinion mining andrea esuli. In *In Proceedings of the 11th conference of the european chapter of the association for computational linguistics (EACL06*. Citeseer, 2006.

[113] Robert. Audi. The Cambridge dictionary. `http://dictionary.cambridge.org/`, 1999.

[114] Arnold WM Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12):1349–1380, 2000.

[115] Ana Lopes, Sandra EF de Avila, Anderson NA Peixoto, Rodrigo S Oliveira, and Arnaldo de A Araujo. A bag-of-features approach based on hue-sift descriptor for nude detection. In *Signal Processing Conference, 2009 17th European*, pages 1552–1556. IEEE, 2009.

[116] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Improving bag-of-features for large scale image search. *International Journal of Computer Vision*, 87(3):316–336, 2010.

[117] Yan Ke and Rahul Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–506. IEEE, 2004.

[118] Alaa Abdel-Hakim and Aly Farag. Csift: A sift descriptor with color invariant characteristics. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1978–1983. IEEE, 2006.

[119] David Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157, 1999.

[120] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[121] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, pages 818–833. Springer, 2014.

[122] Börkur Sigurbjörnsson and Roelof Van Zwol. Flickr tag recommendation based on collective knowledge. In *Proceedings of the 17th international conference on World Wide Web*, pages 327–336. ACM, 2008.