Automatic Prediction of Text Aesthetics and Interestingness

Debasis Ganguly
CNGL,
School of Computing,
Dublin City University,
Dublin 9, Ireland
dganguly@computing.dcu.ie

Johannes Leveling
CNGL,
School of Computing,
Dublin City University,
Dublin 9, Ireland
jleveling@computing.dcu.ie

Gareth J.F. Jones
CNGL,
School of Computing,
Dublin City University,
Dublin 9, Ireland
gjones@computing.dcu.ie

Abstract

This paper investigates the problem of automated text aesthetics prediction. The availability of user generated content and ratings, e.g. Flickr, has induced research in aesthetics prediction for non-text domains, particularly for photographic images. This problem, however, has yet not been explored for the text domain. Due to the very subjective nature of text aesthetics, it is difficult to compile human annotated data by methods such as crowd sourcing with a fair degree of inter-annotator agreement. The availability of the Kindle "popular highlights" data has motivated us to compile a dataset comprised of human annotated aesthetically pleasing and interesting text passages. We then undertake a supervised classification approach to predict text aesthetics by constructing real-valued feature vectors from each text passage. In particular, the features that we use for this classification task are word length, repetitions, polarity, part-of-speech, semantic distances; and topic generality and diversity. A traditional binary classification approach is not effective in this case because non-highlighted passages surrounding the highlighted ones do not necessarily represent the other extreme of unpleasant quality text. Due to the absence of real negative class samples, we employ the MC algorithm, in which training can be initiated with instances only from the positive class. On each successive iteration the algorithm selects new *strong negative* samples from the unlabeled class and retrains itself. The results show that the mapping convergence (MC) algorithm with a Gaussian and a linear kernel used for the mapping and convergence phases, respectively, yields the best results, achieving satisfactory accuracy, precision and recall values of about 74%, 42% and 54% respectively.

1 Introduction

Since their inception, Amazon Kindle device[1] and Apps for other general purpose hand-held devices, have led to a massive increase in the trend of reading e-books over paper printed ones. The Amazon Kindle and the Kindle Apps provide a very simple mechanism for highlighting a piece of text and sharing it on social media. The most popular highlighted pieces of text are shown in the Kindle device with an intention to help readers focus on passages that are pleasing or interesting to the greatest number of people. Every month, Kindle customers highlight millions of book passages that are meaningful to them[2]. The general trend among Kindle readers, while reading the classic English literary works, is to highlight text passages that are associated with a high aesthetic quality. An example highlighted passage is shown in Figure 1.

With the availability of such highlighted text, which may be considered as text passages which most readers find pleasing to read, an interesting research problem is to attempt automatic prediction of highlighted pieces of text. In other words, given a text passage, the objective is to

[1] https://kindle.amazon.com/

[2] https://kindle.amazon.com/most_popular

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 905–916, Dublin, Ireland, August 23-29 2014.

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness, it was the spring of hope, it was the winter of despair.

Figure 1: Passage from *A tale of two cities* (Charles Dickens), highlighted by 6843 Kindle readers.

determine the likelihood of it being aesthetically pleasing and interesting. Such an automated approach of identifying aesthetically pleasing text passages may potentially be used to endorse a newly released book on e-commerce websites with an aim to increase its sales. Moreover, such an approach may also, in principle, be used as a tool by an author to determine how likely it is for readers to appreciate a newly written text passage.

The key challenge in solving this problem is to determine the characteristic attributes of a popular highlighted text passage. An intuitive assumption is that the popularity of a highlighted passage depends on its aesthetic quality. Generally speaking, passages inclined towards expressing an author's view on a subject, which may often be philosophical in nature, with considerable application of atypical figures of speech, e.g. anaphora, alliteration, antithesis, metaphor, simile, personification etc., are more likely to be highlighted than a straight-forward story narrative passage. For example, the highlighted passage in Figure 1 is rich in anaphora (repetition of the same word or group of words in a paragraph, e.g. "times", "age", "epoch" etc.) and antithesis (juxtaposition of opposing or contrasting ideas, e.g. "best of times", "worst of times"; "wisdom", "foolishness" etc). An automated approach of aesthetic quality prediction thus has to take into account these different features of a text passage. The idea of using these features for text aesthetics prediction, in fact, forms a core part of our work.

It is particularly interesting to see that this problem of automatically predicting text aesthetics is largely different from the standard well researched problem of document text classification (Sebastiani, 2002). The reason is as follows. The problem of text categorization can effectively be solved by the application of discrete categorical features, such as character n-gram frequencies and word frequencies. In other words, the presence of characteristic words from a particular domain is a good indicator of the class of a document, e.g. the presence of the words "soccer", "goal" etc. in a document is a good indicator that the document is of the sports genre, whereas the presence of words such as "money", "bank" etc. would indicate that the genre is finance. Consequently, the generative framework of a multinomial Naive Bayes (NB) model with character n-gram and word n-grams based features works effectively for this class of problems (McCallum and Nigam, 1998).

In the case of aesthetic quality prediction, however, the mere presence of a particular word or character n-gram can hardly be a good indicator of the inherent literary quality of the text. The output classes of this classification problem, namely *aesthetic* or *not aesthetic*, do not comprise a small vocabulary of domain-specific representative terms such as in the case of the sports or finance domains. The vocabularies of the respective classes in this classification problem are largely unrestricted and mutually indistinguishable.

The rest of the paper is organized as follows. Section 2 presents related research. In Section 3, we present our proposed approach to solve the text aesthetics problem. Section 4 describes our experimental settings, following which Section 5 presents the results. In Section 6, we investigate the contribution from individual features and then the relative importance of the features when used in combination. Finally, Section 7 concludes the paper.

2 Related Work

A computational viewpoint of aesthetic quality, in general, takes into account the subjectivity of an observer and postulates that among several observations, the aesthetically most pleasing one

is the one with the shortest description, given the observer's previous knowledge (Schmidhuber, 2010). An agent driven reinforcement based learning algorithm can then be used in principle to produce creative (novel and interesting) outputs (Schmidhuber, 2010). Our work in this paper is largely different from the general reinforcement learning paradigm, because we focus on the particular problem of text aesthetics viewing the problem as a supervised classification task. Moreover, the proposition of minimum description length as an attribute of aesthetic quality (Schmidhuber, 2010) is counter-intuitive for literary works.

There has been considerable research interest in automatically predicting visual aesthetic quality of images (Dhar et al., 2011) and layout of web pages (Reinecke et al., 2013). Most empirically successful approaches to image aesthetics prediction first transform an image into a feature vector of characteristic attributes that play a pivotal role in differentiating an *interesting* image from a *non-interesting* one. Generally speaking, some of these attributes which determine whether an image is aesthetically pleasing are the presence of salient objects (indicated by a low depth of field), compositional attributes (e.g. the rule of thirds), the effect of light in natural landscapes, etc. The next step is to apply a supervised learning algorithm, e.g. support vector machine (SVM), to learn a two-class prediction model. Useful features, extracted from images for this classification task include: i) colourfulness, contrast, symmetry, vanishing point and facial features (Jiang et al., 2010); ii) face poses, between-face distances, and the consistency of expressions on multiple faces (Li et al., 2010); iii) high level describable attributes, such as compositional attributes (e.g. rule of thirds image layout), content attributes related to the presence of people, animals, sky illumination attributes etc. (Dhar et al., 2011).

Our proposed method of text aesthetics prediction is similarly based on extracting characteristic features from the text passages. However, in the case of literature, it is worth mentioning that in contrast to image aesthetics it is more difficult to describe the subtle attributes which differentiate an aesthetically pleasing text from its counterpart.

Although the authors are not aware of any reported research on text aesthetics, there has been a considerable amount of research in the somewhat closely related problem of detecting metaphors in text. Automated approaches to metaphor detection involve both supervised and unsupervised approaches, some of which include: i) supervised classification on extracted verbal target feature vectors of sentences (Gedigian et al., 2006); ii) expectation maximization (EM) based unsupervised approach to non-literal word sense detection (Birke and Sarkar, 2006); iii) unsupervised approach using hierarchical graph factorization clustering (Shutova and Sun, 2013).

In general, it is intuitive to assume that metaphorical or figurative parts of text are aesthetically pleasing and interesting, which makes the problem of text aesthetics prediction somewhat similar to that of metaphor detection. Unfortunately, this assumption is not often true, and this is particularly the case for literary works due to the availability of a large number of figures of speech at an author's disposal (metaphor just being one of them). For example, the sample Kindle highlighted passage shown in Section 1 has an obvious aesthetic appeal to a large number of readers, in spite of it being not metaphorical.

3 Our Approach to the Text Aesthetics Prediction Problem

In this section, we describe the details of our approach to text aesthetics prediction. We hypothesize that a NB classifier with word or character n-gram based features is not suitable for this particular problem due to the mutual overlap and lack of domain specific restriction in the vocabulary of the output classes (i.e. aesthetic and non-aesthetic). One thus needs to extract a set of characteristic features from the text passages which may be useful to solve the classification problem. We describe the features used in our approach in Section 3.1. In Section 3.2, we propose to use the mapping convergence (MC) algorithm for the text aesthetics problem, where the intention is to learn a classifier only from positive samples.

The truth is rarely pure and never simple. Modern life would be very tedious if it were either, and modern literature a complete impossibility!

Figure 2: Passage from *The Importance of Being Earnest* (Oscar Wilde).

3.1 Feature Vector Encoding of Text Passages

In this section, we introduce the various features used for the text aesthetics classification task. Each feature is a function which maps a passage of text $P = \{w_1 \dots w_N\}$ comprising N words into a real number.

3.1.1 Word-based Features

In Section 1, we illustrated that that an anaphora is a rheoteric device used by authors to emphasize a text passage, which in turn indicates that such a passage is likely to attract the attention of readers and hence are likely to be highlighted by them. Moreover, the closer the repetitions are, the stronger is the emphasis.

On the basis of this reasoning, we employ an average positional difference weighted count of word repetitions in a passage. To be more precise, for each word in a passage we compute the number of times a word w_i is repeated, divide this count by the difference between the repeating position (say at position j), and average the sum of counts for all repeating words over the passage length, as shown in Equation 1. In Equation 1, $\mathbb{1}(w_i = w_j)$ is the indicator function which is 1 if and only if $w_i = w_j$ and 0 otherwise.

The second word level feature which we use, is the average length of words in a passage. The reasoning behind using this feature is that authors tend to use relatively longer words (e.g. superlatives) to emphasize a passage. Equation 2 shows how this is computed.

$$W_1(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{\mathbb{1}(w_i = w_j)}{j - i} \tag{1}$$

$$W_2(P) = \frac{1}{N} \sum_{i=1}^{N} len(w_i) \tag{2}$$

3.1.2 Topic-based Features

An attribute which can be considered responsible for the aesthetic quality of a text passage is the diversity of topics it expresses. It is reasonable to assume that a text passage expressing a broad idea or opinion of an author, often philosophical in nature, is likely to be appealing to readers. Such general themed text passages typically cover a broad range of topics, as a result of which the constituent words of such text passages involve collocation of seemingly unrelated terms. For example, in the text passage shown in Figure 2, the word pairs (*truth*, *tedious*), and (*literature*, *impossibility*) would typically appear in different topic classes, where by a topic we mean a set of words with high co-ocurrence likelihood estimated from a collection of documents by standard topic modelling techniques such as the Latent Dirichlet allocation (LDA) (Blei et al., 2003). To encode this diversity of topics as a real valued feature function, we use Equation 3.

$$T_1(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{\mathbb{1}[z(w_i) \neq z(w_j)]}{(j - i)} \tag{3}$$

In Equation 3, $z(w)$ denotes the topic class of the word w obtained with the help of LDA. A mismatch in the topic class is divided by the distance between the mismatches to assign more weight to the close mismatches. As an example, the mismatch between (*literature*, *impossibility*) bears more importance than the mismatch between (*modern*, *impossibility*).

The second topic-based feature which we use pertains to predicting the abstractness of the content of a passage. It has been reported that words highly representative of topics are generally

not metaphorical. We apply a similar reasoning to hypothesize that since an interesting piece of text is more likely to be philosophical or abstract in nature in comparison to a story narrative, the constituent words are less likely to be the representatives of their topic classes. Formally speaking in terms of LDA, these words are expected to have smaller values of $\max_k \phi_k(w)$. Recall that a topic representative word in LDA exhibits a skewed distribution with a peak for one topic class (with a high value of $\max_k \phi_k(w)$), whereas a less representative word exhibits a more uniform distribution of $\phi_k(w)$ values over the topic classes (thus a low value of $\max_k \phi_k(w)$). We use Equation 4 to compute the average topic concreteness of a text passage.

$$T_2(P) = \frac{1}{N} \sum_{i=1}^{N} \max_k \phi_k(w_i) \tag{4}$$

3.1.3 Part of Speech Feature

We hypothesize that another attribute of an aesthetic passage is that it is likely to contain a rich usage of adjectives (mostly of superlative type for the sake of emphasis) and adverbs. We therefore employ the part of speech tag (POS) information of the constituent words of a text passage as one of our features. To be more specific, we use the average number of adjectives and adverbs of a text passage as the feature value. This is shown in Equation 5.

$$POS(P) = \frac{1}{N} \sum_{i=1}^{N} (\#adjectives + \#adverbs) \tag{5}$$

3.1.4 Sentiment Feature

We pointed out in Section 1 that authors often use the *antithesis* figure of speech to express contrasting concepts. Thus, another feature which we can use is the aggregated absolute difference values between the sentiment polarities of words in a text paragraph. This again is weighted by the difference in position between a positive sentiment word and its negative counterpart to assign more importance to closely occurring opposite sentiment concepts.

To obtain the sentiment values of the constituent words, we used the SentiWordNet[3]. To illustrate with an example, consider the closely occurring opposite sentiment word pairs (*best* (0.75), *worst* (-0.75)), (*wisdom* (0.375), *foolishness* (-0.375)) etc. of Figure 1 and the word pairs (*complete* (0.625), *impossibility* (-0.25)) of Figure 2, where the numbers in the parentheses show the positive or the negative sentiment value (a normalized number between 0 and 1). Equation 6 shows the real-valued function derived from the sentiment information of word pairs, where the function $s(w)$ denotes the sentiment value associated with the word w.

$$SENT(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{|s(w_i) - s(w_j)|}{(j-i)} \tag{6}$$

3.1.5 Inter-word Semantic Distance Feature

An alternative way to represent the topic diversity is to capture the likelihood of the event of occurrence of two words in close vicinity. The higher this likelihood is, the better is the semantic relation or coherence between the words. We make use of the DISCO[4] tool to compute the semantic relation between two words in a word pair. In DISCO, these semantic relations between the words are precomputed on the basis of co-occurrence likelihoods from a large corpus, e.g. the Wikipedia (Kolb, 2008). DISCO provides two similarity measurements (named the first order and the second order similarities) between two input words. While the first order similarity between two input words is computed based on their collocation sets, the second order similarity is computed based on their sets of distributionally similar words (Kolb, 2008). We denote the

[3] `http://sentiwordnet.isti.cnr.it/`

[4] `http://www.linguatools.de/disco/disco_en.html`

first order and the second order similarities between words w_i and w_j respectively as $ds_1(w_i, w_j)$ and $ds_2(w_i, w_j)$ respectively.

In relation to text aesthetics, we expect a small value of average first order and second order similarity values between word pairs in a highlighted piece of text in comparison to a non-highlighted one. Similar to our earlier features, we divide these similarity values by the positional difference between the words in order to put more emphasis on semantic diversity between closely occurring words. Equation 7 shows the two features extracted making use of these similarity values.

$$SD_k(P) = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \frac{|ds_k(w_i) - ds_k(w_j)|}{(j-i)},\ k = \{1, 2\} \tag{7}$$

3.2 Learning from Positive Examples: The MC Algorithm

Binary classifiers, such as SVMs, work particularly well with a sufficient number of both positive and negative class instances for training. In the case of text aesthetics prediction problem, the passages highlighted by Kindle readers serve as the positive class samples. Although it might be intuitive to use the non-highlighted passages as instances of the negative type, there can be problems associated with this approach.

Firstly, the non-highlighted passages are not essentially instances of the negative class because the non-highlighted passages are not necessarily aesthetically unpleasing. Secondly, there is an element of cognitive bias associated with the highlighting process because a reader, who can already see popular highlights while reading a page, may be biased to highlight the same passage himself, and may not in fact highlight some other passage which he himself found interesting.

Note that this observation in fact makes our problem more challenging to solve in comparison to aesthetics prediction in other domains, such as images, where information such as Flickr[5] photo ratings can be used as strong positive or negative indicators of an image interestingness or aesthetic quality, leading to effective classification results using a standard binary classification approach (Dhar et al., 2011).

Due to the presence of incompletely labeled examples, we apply the *mapping convergence* (MC) algorithm (Yu et al., 2003) for this task. The objective of the MC algorithm is to predict the positive samples from a test data, given a mixture of positive and unlabeled samples. These unlabeled samples in the MC algorithm can be treated as instances of either the positive or the negative class in order to obtain maximum classification effectiveness.

The two stages of the MC algorithm are summarized as follows.

1. The *mapping* stage identifies from the unlabeled samples the strong negative ones, i.e. the points distinctly different from the positive samples.
2. The *convergence* stage is an iterative step to learn a binary classification model, e.g. SVM, using the positive and the strong negative samples. Each iterative step of convergence classifies the remaining unlabeled samples to collect more strong negative samples. The convergence step is repeated until no more strong negative samples are found.

The objective of the convergence step of the MC algorithm is to maximize margin to make progressively better approximation of the negative data. At the end of the iteration, the class boundary eventually converges to the boundary around the positive data set in the feature space (Yu et al., 2003).

In our approach to the text aesthetics prediction task, we implement the mapping stage of the MC algorithm with the help of standard one-class classifiers, namely the one class SVM (OSVM) (Schölkopf et al., 1999) and the support vector data descriptor (SVDD) (Tax and Duin, 2004). The OSVM separates all the data points in the feature space from the origin, with the help of a separating hyperplane with maximum distance from the origin. The OSVM is thus

[5] `https://www.flickr.com/`

able to separate out regions in the input space with high probability densities (Schölkopf et al., 1999). SVDD, on the other hand, instead of a planar, takes a spherical approach to the one class problem. The algorithm obtains a spherical boundary in feature space around the data. The volume of this hypersphere is minimized to minimize the effect of incorporating outliers in the solution (Tax and Duin, 2004).

It is worth mentioning here that although the OSVM and the SVDD can be trained with positive samples only, these models are prone to over-fitting or under-fitting due to a small number of support vectors modeled from a small number of positive samples (Yu et al., 2003). In contrast, a binary SVM can model data more robustly due to the presence of the additional negative samples. Hence, OSVM and the SVDD are typically used as a weak classifier to obtain a set of initial strong negative samples in order to initiate the convergence step of the MC algorithm.

4 Experiment Settings

In this section, we describe the dataset and the tools used for our experiments.

4.1 Dataset Construction

The standard practice to evaluate the metaphor detection problem, which is somewhat similar to the text aesthetics prediction, is to make extensive use of manually annotated data typically obtained under controlled user-based studies, where the users or the participants are instructed to perform some given objectives, such as manually label metaphors in a collection of documents, e.g. (Hovy et al., 2013). The main difficulties with this approach are that: i) it takes a considerable amount of time to collect data; ii) the quality of the data depends largely on controlled experimental settings, e.g. the data quality may be susceptible to errors caused by targeted, malicious work efforts, since there is often a financial incentive to complete tasks quickly rather than effectively (Ipeirotis et al., 2010); and iii) it is very difficult to compare the effectiveness of two methods on two different datasets obtained under different controlled user study settings.

The availability of fairly large amounts of highlighted text on the Amazon website has ensured a reliable and fast way to construct the dataset for carrying out the text aesthetics experiments. The advantages are as follows. Firstly, it is not necessary to conduct crowd sourcing experiments for data collection. Secondly, since the data is not generated by controlled crowd sourcing, the quality of the data is more reliable because there is no financial incentive to complete tasks quickly. Thirdly, since the data is publicly available, it is possible to achieve a fair comparison between different problem solving approaches.

The Amazon "Popular Highlights"[6] web page presents a ranked list of the most highlighted passages, sorted in descending order by the number of highlights. However, at the time of writing this paper, Amazon has neither made the data publicly downloadable nor provided an API to access it. For conducting our experiments with this data, we therefore had to automatically crawl data from the Popular Highlights web page.

In addition to the highlighted passages (serving as the positive class samples in our dataset), we also need the non-highlighted ones (meant to serve as the unlabeled samples). The text from the non-highlighted passages, however, are not available in the Popular Highlights web page. This data was thus extracted from those books, the passages of which are popularly highlighted. In order to ensure free access to book content, we had to restrict our dataset to the 50 most popular highlighted classic English fictions.

More precisely speaking, for every highlighted passage found while crawling the Amazon Popular Highlights page, our crawler checks if the book is available on project Gutenberg[7]. If not, then we examine the next highlighted passage, otherwise we craw the full text of the book,

[6] `https://kindle.amazon.com/most_popular/highlights_all_time/`
[7] `http://www.gutenberg.org/`

in which the current highlighted passages belongs, from project Gutenberg website. The crawler continued to run until we had collected highlighted passages from 50 different literature classics.

The dataset for the prediction task is then constructed as follows. First, we add the text of all highlighted passages as instances of the positive class. Next, for each highlighted passage, we add the paragraph preceding and succeeding it into the dataset as the unlabeled samples. Note that selecting the unlabeled samples this way is better than random selection of non-highlighted passages from full text, because this way of choosing negative samples ensures a meaningful representation of reader judgments to highlight a particular passage of text from within a surrounding context.

We then partition the dataset comprised of the positive and unlabeled samples into equal sized training and test sets. In Table 1, we outline the characteristics of the dataset.

Dataset	# Books	Vocab.	# Passages		
		Size	Highlighted	Unhighlighted	Total
Train	25	9560	168	305	473
Test	25	7883	169	319	488
Total	50	13496	337	624	961

Table 1: Dataset characteristics

4.2 Implementation Details

For each passage in the dataset, we extract the features described in Section 3.1. To compute the topic modeling based features we used Mallet[8]. The number of topics (K) in LDA was set to 100. The POS tag feature was extracted with the help of the Stanford POS tagger[9]. For extracting the sentiment feature, we made use of the Java API of the SentiWordNet[10]. For the semantic word distance feature, we used the DISCO Java API[11].

For the naive Bayes experiment, we used the Stanford classifier[12]. The SVM experiments (binary SVM, one-class SVM, SVDD) were conducted with the libSVM software[13].

4.3 Evaluation Metrics

For all the experiments reported in this paper, the classification effectiveness mainly focuses on precision and recall with respect to the positive class. Consequently, precision, recall and the F-score measures, shown in Tables 2 and 3, are measured with respect to the positive class only.

Ideally, for this problem one would want to obtain a high recall, i.e. identify as many highlighted passages correctly as possible. In this situation, recall is thus more important than precision. Achieving a good precision is desirable, nonetheless, to minimize the false positives. Although we report accuracy, we emphasize that accuracy alone is not a good measure of classification effectiveness in this case, because correct identification of negative instances is not important for this problem.

5 Results

Before conducting experiments with the MC algorithm, we obtained baseline results by classifying the dataset using NB and SVMs. In the case of NB, instead of using the real valued features from the text passages (as proposed in Section 3.1), we simply used the character n-gram and word n-gram features (maximum value of n was set to 5) from the text, automatically extracted

[8] http://mallet.cs.umass.edu/
[9] http://nlp.stanford.edu/software/tagger.shtml
[10] http://sentiwordnet.isti.cnr.it/code/SentiWordNetDemoCode.java
[11] http://www.linguatools.de/disco/disco_en.html
[12] http://nlp.stanford.edu/software/classifier.shtml
[13] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/

Classifier	Kernel	Accuracy	Precision	Recall	F-score
NB	N/A	67.40	54.40	36.70	43.80
BSVM	Linear	66.19	35.71	5.92	10.15
BSVM	Gaussian	67.00	39.39	15.38	22.13
OSVM	Linear	38.32	32.46	51.48	39.82
OSVM	Gaussian	53.68	41.87	50.29	45.70
SVDD	Linear	35.04	34.77	100.00	51.60
SVDD	Gaussian	37.91	35.56	97.63	52.13

Table 2: Text aesthetics prediction results with Naive Bayes and SVM.

Classifier		Kernel		Accuracy	Precision	Recall	F-score
Mapping	Convergence	Mapping	Convergence				
OSVM	BSVM	Linear	Linear	66.18	35.71	5.92	10.15
OSVM	BSVM	Linear	Gaussian	64.96	40.26	36.69	38.39
OSVM	BSVM	Gaussian	Linear	66.80	44.44	11.83	18.69
OSVM	BSVM	Gaussian	Gaussian	64.34	36.87	39.05	37.93
SVDD	BSVM	Linear	Linear	40.98	35.76	92.90	51.64
SVDD	BSVM	Linear	Gaussian	43.44	36.17	90.53	51.69
SVDD	BSVM	Gaussian	Linear	56.76	42.90	74.64	**54.42**
SVDD	BSVM	Gaussian	Gaussian	47.34	38.60	88.17	53.69

Table 3: Text aesthetics prediction results by the MC algorithm with different settings.

by the Stanford classifier. The result of this experiment (see Table 2) shows that the recall value is very low, which in turn indicates that word vocabulary based features, typically used for text categorization, are not effective for this task.

The next classification method that we employ is standard binary class SVM (denoted as BSVM). The training phase of the BSVM used the non-highlighted passages as negative class instances. We experimented with both linear and Gaussian kernels. For all reported results which use the Gaussian kernel, the parameter γ was set to the default value of 1/(#features) as per the libSVM implementation. Although the accuracy achieved is comparable to NB, the recall achieved is worse, which shows that treating non-highlighted passages as negative class instances is not reasonable for this problem (see Section 6.2 for an illustration).

The recall value is significantly increased with the help of one-class SVM (OSVM). SVDD performs even better in terms of recall. However, SVDD significantly underfits the data because it classifies almost every test data point as an instance of the positive class, thus achieving low accuracy and precision due to the presence of too many false positives.

Our next set of experiments involves the MC algorithm for classification. Since, the mapping phase makes use of only the positive data, we employed both the one-class classifiers used in the experiments of Table 2, i.e. OSVM and SVDD, for this purpose. Mapping with OSVM results in an improvement in the accuracy at the cost of sacrificing recall, which is not desirable for this problem. However, note that the negative samples obtained with the OSVM mapping (with Gaussian kernel) improves the classification effectiveness of the BSVM (compare the fourth row of Table 3 with the second row of Table 2), which indicates that the MC algorithm does improve the classification effectiveness, confirming our hypothesis that it is reasonable not to consider every non-highlighted passage as negative samples.

The problem of SVDD underfitting (as evident from the SVDD results of Table 2) is alleviated by the MC approach. The most effective MC approach uses Gaussian/linear kernels for mapping/convergence (see the seventh row of Table 3). Accuracy is increased to around 56% with a satisfactory recall of around 74%. The use of Gaussian kernel during both the mapping and convergence steps yields a higher recall but at the cost of more false positives (lower accuracy, precision and F-score).

Feature combination vector				Evaluation Metrics			
Word	Topics	POS/Polarity	Semantic	Accuracy	Precision	Recall	F-score
1	0	0	0	36.06	34.88	97.63	51.40
0	1	0	0	37.91	35.74	99.40	52.58
0	0	1	0	36.05	35.01	98.81	51.70
0	0	0	1	42.41	37.03	94.67	53.24
1	1	1	1	56.76	42.90	74.64	**54.42**

Table 4: Individual feature contributions for identifying text aesthetics.

Feature	igain
Topic diversity (T_1)	0.3684
Sentiment ($SENT$)	0.2685
Word repetition (W_1)	0.2509
First-order semantic distance (SD_1)	0.1543
Part-of-speech (POS)	0.1448
Second-order semantic distance (SD_2)	0.1141
Word length (W_2)	0.0732
Topic abstractness (T_2)	0.0526

Table 5: Ranking features by their igain values.

6 Posthoc Analysis

In this section, we comment on the importance of the features used for classification, and also illustrate how the MC algorithm helps in increasing the separability between the classes.

6.1 Feature Importance

First, we investigate the importance of the different features by a selective choice of only one group of features at at time for the classification. The classifier we use for this experiment is MC with a Gaussian SVDD kernel for mapping and a linear SVM kernel for convergence (as per the best settings of Table 3). The results are shown in Table 4 from which it can be seen that the best accuracy is obtained with the use of the semantic distance features.

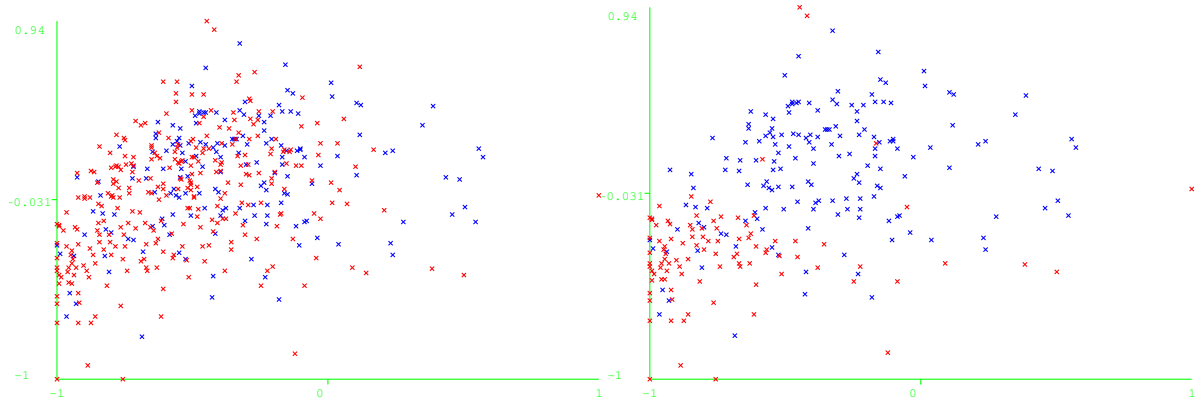
It can be observed that the accuracy values obtained with a single category of features, such as word-based (length and repetition), topic-based (generality and diversity) and so on, are considerably lower than the accuracy value obtained with a combination of all the features (the last row of Table 4. The precision values achieved with these individual feature groups are also considerably lower than the precision of 42.90% of the overall combination.

Next, we find out the relative importance of each feature in their overall combination by ranking the features with the help of a standard feature quality estimator, called *information gain* (*igain*) (Quinlan, 1986). The results are presented in Table 5. It can be seen that the topic diversity is the most discriminative feature having an igain value significantly higher than the second most important one in the list. This observation verifies our hypothesis that aesthetically appealing passages are those constituting terms from diverse topics.

The sentiment and the word repetition features, having close igain values, are second and third respectively in the list. The usefulness of the sentiment feature suggests that contrasting concepts packed in close vicinity of a sentence are likely to be aesthetically pleasing to read. The word repetition feature, on the other hand, suggests that the anaphora figure of speech is likely to be be associated with aesthetically pleasing text.

6.2 Illustration of the usefulness of the MC Algorithm

This section investigates the usefulness of the MC algorithm for the text aesthetics classification. In particular, we show that for this one class classification problem, the MC algorithm can selectively refine the set of unlabeled samples and retrain the model for better separability



(a) Before MC. (b) After MC convergence (5 iterations).

Figure 3: Visualization of the training set in the two most discriminating dimensions, i.e. topic diversity (Y-axis) and sentiment (X-axis).

between the positive and the unlabeled classes.

To illustrate our claim, we first plot the initial training set in two dimensional subspace before the application of MC, i.e. when all the unlabelled instances are treated as negative class samples; this is shown in Figure 3a. The two dimensions that we use for plotting this figure, are the two features having the highest igain values, i.e. the topic diversity (T_1) and sentiment ($SENT$) features. Figure 3a shows that the highlighted text passages (shown in blue) are not well separated from the non-highlighted ones (shown in red).

Next, in Figure 3b, we plot the training set with a reduced number of samples from the negative (non-aesthetic) class obtained after running the MC algorithm. Figure 3b clearly shows that after convergence the MC algorithm has retained only the strong negative samples for training, as is evident from a better visual separation between the classes. A binary classifier, trained on the dataset of Figure 3b, is thus likely to be more effective than that trained with Figure 3a.

7 Conclusions

This paper investigated the problem of automated text aesthetics prediction. As distinguishing features for text aesthetics identification, we applied different statistical features such as word repetitions, topic diversity, part-of-speech, word polarity etc. We collected aesthetically pleasing text passages from the Kindle "popular highlights" website for conducting our experiments. Due to the presence of only positive class samples, i.e. the highlighted passages, in this dataset, we apply the MC algorithm to iteratively train a binary classifier with the strongly negative samples.

The results of our experiments show that the MC algorithm with a Gaussian and a linear kernel applied for the mapping and convergence phases respectively, yields the best results achieving satisfactory recall, precision and F-score values of about 74%, 42% and 54% respectively. Moreover, the results also demonstrate that the topic diversity, word polarity and word repetition are the three most distinguishing features for text aesthetics identification. Furthermore, our results are comparable to those of a somewhat similar problem of figurative text detection where the best reported F-score values achieved are about 54% (Birke and Sarkar, 2006) and 64% (Shutova and Sun, 2013).

Acknowledgments

This research is supported by Science Foundation Ireland (SFI) as a part of the CNGL Centre for Global Intelligent Content at DCU (Grant No: 12/CE/I2267).

References

Julia Birke and Anoop Sarkar. 2006. A clustering approach for nearly unsupervised recognition of nonliteral language. In *EACL 2006, 11st Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, April 3-7, 2006, Trento, Italy*. The Association for Computer Linguistics.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, March.

Sagnik Dhar, Vicente Ordonez, and Tamara L. Berg. 2011. High level describable attributes for predicting aesthetics and interestingness. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 1657–1664.

Matt Gedigian, John Bryant, Srini Narayanan, and Branimir Ciric. 2006. Catching metaphors. In *Proceedings of the Third Workshop on Scalable Natural Language Understanding*, ScaNaLU '06, pages 41–48, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dirk Hovy, Shashank Shrivastava, Sujay Jauhar, Mrinmaya Sachan, Kartik Goyal, Huying Li, Whitney Sanders, and Eduard Hovy. 2013. Identifying metaphorical expressions with tree kernels. In *Proceedings of NAACL-HLT Meta4NLP Workshop*.

Panagiotis G. Ipeirotis, Foster Provost, and Jing Wang. 2010. Quality management on amazon mechanical turk. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '10, pages 64–67, New York, NY, USA. ACM.

Wei Jiang, Alexander C. Loui, and Cathleen Daniels Cerosaletti. 2010. Automatic aesthetic value assessment in photographic images. In *Proceedings of the 2010 IEEE International Conference on Multimedia and Expo, ICME 2010, 19-23 July 2010, Singapore*, pages 920–925.

Peter Kolb. 2008. DISCO: A Multilingual Database of Distributionally Similar Words. In *KONVENS 2008 – Ergänzungsband: Textressourcen und lexikalisches Wissen*, pages 37–44.

Congcong Li, Alexander C. Loui, and Tsuhan Chen. 2010. Towards aesthetics: A photo quality assessment and photo selection system. In *Proceedings of the International Conference on Multimedia*, MM '10, pages 827–830, New York, NY, USA. ACM.

Andrew McCallum and Kamal Nigam. 1998. A comparison of event models for naive bayes text classification. In *AAAI/ICML Workshop on Learning for Text Categorization*, pages 41–48.

J. R. Quinlan. 1986. Induction of decision trees. *Mach. Learn.*, 1(1):81–106, March.

Katharina Reinecke, Tom Yeh, Luke Miratrix, Rahmatri Mardiko, Yuechen Zhao, Jenny Liu, and Krzysztof Z. Gajos. 2013. Predicting users' first impressions of website aesthetics with a quantification of perceived visual complexity and colorfulness. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2049–2058, New York, NY, USA. ACM.

Jürgen Schmidhuber. 2010. Formal theory of creativity, fun, and intrinsic motivation (1990-2010). *IEEE T. Autonomous Mental Development*, 2(3):230–247.

Bernhard Schölkopf, Robert C. Williamson, Alex J. Smola, John Shawe-Taylor, and John C. Platt. 1999. Support vector method for novelty detection. In *Advances in Neural Information Processing Systems 12, [NIPS Conference, Denver, Colorado, USA, November 29 - December 4, 1999*, pages 582–588. The MIT Press.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, March.

Ekaterina Shutova and Lin Sun. 2013. Unsupervised metaphor identification using hierarchical graph factorization clustering. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 978–988. The Association for Computational Linguistics.

David M. J. Tax and Robert P. W. Duin. 2004. Support vector data description. *Mach. Learn.*, 54(1):45–66, January.

Hwanjo Yu, ChengXiang Zhai, and Jiawei Han. 2003. Text classification from positive and unlabeled documents. In *Proceedings of the 2003 ACM CIKM International Conference on Information and Knowledge Management, New Orleans, Louisiana, USA, November 2-8, 2003*, pages 232–239. ACM.