산업공학석사 학위논문

Word embedding for sentiment analysis considering emotional dimensions

감정 차원을 고려한 단어 벡터 모델과 감성 분석에 관한 연구

2016 년 2 월

서울대학교 대학원
산업공학과
구 본 효

Abstract

Word embedding for sentiment analysis considering emotional dimensions

Koo Bonhyo
Industrial Engineering
The Graduate School
Seoul National University

Recent studies have shown that word embeddings based on the word-context co-occurrence statistics are suited to measure semantic similarities. However, word embeddings are deficient in emotional information. This thesis reviews current word embedding models and presents word embeddings enriched with emotional information. Word embeddings are learned based on the previous word embeddings using a semi-supervised autoencoder model to incorporate affective norms data. Then, the thesis evaluates word embeddings enriched with emotional data on sentiment classification datasets.

Keywords: word embedding, distributional hypothesis, emotional dimension, sentiment analysis, semi-supervised learning, autoencoder

Student Number: 2014-21803

Table of Contents

Index of Tables

Index of Figures

Chapter 1 Introduction

Word embeddings represent semantics of words as vectors of real numbers. Unlike the traditional language models which directly map words into indices, word embedding models encode semantic similarities among words using the co-occurrence statistics of the corpus. A popular approach to capture semantics of words is called the continuous vector space model. The vector space model measures attributional similarities among words on the basis of their contexts (Turney, P. D., & Pantel, P., 2010). Another popular approach is called the neural network language model. The neural network language model constructs a neural network which predict probability distribution of words given contexts.

Recently, Mikolov, T. et al. (2013) proposed an unsupervised shallow network model for estimating word embeddings called the skip-gram with negative sampling; the state-of-the-art model architecture across various tasks (Mikolov, T. et al., 2013). Remarkably, it has been shown that the skip-gram captures not only attributional similarities, but also relational similarities. Namely, word embeddings derived by the skip-gram better represent semantic similarities among words. For example, the model allows one to predict Paris is to France as Rome is to Italy. Baroni, M., Dinu, G., & Kruszewski, G. (2014) claimed that the skip-gram model is highly superior to vector space models.

However, recent studies suggest the skip-gram is closely related to other traditional vector space models (Levy, O., & Goldberg, Y. ,2014; Arora, S. et al, 2015). It has been known that vector space models base on the co-occurrence statistics are suited to measure attributional similarities. According to Levy, O., & Goldberg, Y. (2014), vector space models may perform as well as the skip-gram under additional hyperparameters and re-weightings (Levy, O., & Goldberg, Y., 2014).

Furthermore, Arora, S. et al. (2015) provided theoretical explanation for word embeddings using co-occurrence statistics (Arora, S. et al., 2015).

Training word embedding models may be categorized as unsupervised learning. Models observe sequence of words from corpus, and estimate vectors of real numbers which reflect contextual structure of words. Vector space models such as pointwise mutual information or positive pointwise mutual information can be directly computed from co-occurrence statistics. Predictive models such as neural network language models or the skip-gram estimate word embeddings by maximizing the probability of co-occurrence statistics. Thus, training word embedding models does not require labeled, or task-specific data.

Word embeddings are found to be useful for several natural language processing tasks such as chunking or name entity recognition (Turian, J. et al., 2010). Word embeddings are generally even useful for sentence- or document-level tasks (Maas, A. L. et al., 2011; Socher R. et al., 2013). However, estimating sentence- or document-level embeddings out of word embeddings is another major task in natural language processing.

1 Contribution

This thesis aims to estimate word embeddings suitable to analyze sentiment polarity and subjectivity of document. In general, sentence- or document-level sentiment polarity classification, namely, sentiment analysis use bag-of-words representations. Instead, the thesis propose semi-supervised learning algorithm to estimate word embeddings for sentiment analysis incorporating dimensional models of emotion.

Firstly, the thesis reviews the concept of current word embedding models. The thesis mainly focuses on the skip-gram with negative sampling and pointwise mutual information. Word embeddings derived

by the skip-gram with negative sampling or pointwise mutual information are evaluated on the basis of relational similarities.

Secondly, the thesis presents a semi-supervised autoencoder with regularization on similarities among word embeddings. Rather than capturing semantic similarities among words only via unsupervised learning, the thesis incorporates sentiment information of words via semi-supervised learning. Recent researches suggest that affective states arise in the early stage of processing emotional words (Recio, G. et al., 2014).

The proposed method is compared with term frequency document inverse frequency representations and other well-known word embedding models for sentiment analysis. The thesis evaluates the method on four different datasets; sentence polarity dataset and subjectivity dataset from Cornell Movie Review Data (Pang, B., & Lee, L., 2005); Stanford Sentiment Treebank (Socher R. et al., 2013); Large Movie Review Dataset (Maas, A. L. et al., 2011). Since estimating sentence- or document-level embedding out of word embeddings is beyond the scope, the thesis generates sentence- or document-level embedding by simply averaging word embeddings.

2 Related Work

The idea of the estimating word embeddings based on the word-context co-occurrence statistics was already introduced in 1950s by Firth, J.R. (1957). However, it was not until the 1990s that the latent semantic analysis, one of the earliest vector space models was introduced. Latent semantic analysis, or latent semantic indexing is a method which applies a dimensionality reduction method, singular value decomposition to term-document co-occurrence statistics.

In 2003, Bengio, Y. et al. (2003) proposed the neural probabilistic language model which models the probability distribution of word given context (Bengio, T. et al., 2013). The neural probabilistic language model proved a great success. However, the model was computationally expensive to train. Morin, F., & Bengio, Y. (2005) proposed the hierarchical probabilistic neural network language model, which introduces hierarchical decomposition into probability estimation to efficiently train the neural network language model (Morin, F., & Bengio, Y., 2005).

Mikolov, T. et al. (2013) proposed an unsupervised shallow network model called the skip-gram with negative sampling. The model outperformed other word embedding models including the neural network language model and the vector space model (Mikolov, T. et al., 2013). Remarkably, the model captured both attributional and relational similarities with shallow architecture and efficient algorithm. Several attempts have been made to explain the skip-gram model and discovered that predictive models including the skip-gram are closely related to traditional count-based models (Levy, O., & Goldberg, Y. ,2014; Pennington, J. et al., 2014; Arora, S. et al., 2015; Osterlund, A. et al., 2015; Schnabel, T. et al., 2015).

Sentiment analysis using word embeddings has been studied in recent years. In 2011, Maas, A. L. et al (2011) proposed a probabilistic model which captures both semantic similarities and sentiment on sentimental documents. The model learns sentiment information of the word from the label of the document (Maas, A. L. et al, 2013). Socher, R. et al. (2011) proposed a recursive autoencoder with a semi-supervised node. The model estimates phrase and document-level embeddings considering compositionality of words. Tang, D. et al. (2014) proposed a neural probabilistic model which learns sentiment-specific word embeddings from predicting label of the document with n-gram word embeddings (Tang, D. et al., 2014).

Emotions are underlying components of sentimental state. It is commonly accepted that emotions are described by a number of dimensions, including valence and arousal (Russell, J. A., 1980; Bradley, M. M. et al., 1992; Kensinger, E. A., 2004; Posner, J. et al., 2005). Unlike sentimental states which expose themselves situations, emotions are more like neurophysiological responses. Thus, Warriner, A. B. et al. (2013) collected human responses over dimensions of emotions of 13,915 English lemma (Warriner, A. B. et al., 2013). Affective responses precede language processing and influence the whole process. In this regard, the affective norms database published by Warriner, A. B. et al. (2013) is a key to estimate word embeddings enriched with emotional information.

Chapter 2 Word Embedding

The word embedding has long been of interest in natural language processing. The underlying idea of the word embedding is to understand semantics of words from statistical patterns of human word usage. The distributional hypothesis is the most widely accepted hypothesis, which postulates that the meaning of word is characterized by the context. Namely, words used in similar contexts may have similar meanings. In this regard, the word embedding stand on the basis of the word-context co-occurrence statistics.

Mikolov, T. et al. (2013) introduced an efficient algorithm to learn word embeddings called the skip-gram with negative sampling, or the Word2Vec (Mikolov, T. et al., 2013). The skip-gram with negative sampling model gained much attraction over the past years in that the model not only provides the state-of-the-art results but also the model captures both attributional and relational similarities. The attributional similarity is a semantic similarity between two words, whereas the relational similarity is a semantic similarity between two pairs of words. For example, Paris and Rome have a high attributional similarity. Meanwhile, Paris and France have a high relational similarity to Rome and Italy because Paris is to France as Rome is to Italy. Remarkably, the skip-gram with negative sampling produces word embeddings with linear structure as below:

$$\mathbf{v}_{\text{Paris}} - \mathbf{v}_{\text{France}} = \mathbf{v}_{\text{Rome}} - \mathbf{v}_{\text{Italy}}$$

This chapter reviews the skip-gram with negative sampling model and the pointwise mutual information. It has recently been suggested by several studies that the skip-gram with negative sampling is in close relationship with the pointwise mutual information model (Levy, O., &

Goldberg, Y. ,2014; Arora, S. et al., 2015). The chapter explores the existing research in the area and identifies the relationship between these models. This chapter also evaluates word embeddings derived by the skip-gram with negative sampling and the pointwise mutual information model.

1 Skip-Gram with Negative Sampling

The skip-gram with negative sampling model is a variant of log-linear models. A sentence or document is a sequence of words, $w_1, w_2, w_3, \ldots, w_n$. The model learns word embeddings which maximize the sum of the probability of the word-context co-occurrence statistics. First, the model defines the conditional probability $p(w_{context}|w_{word})$ using the softmax function:

$$\underset{w}{argmax} \prod_{w \in document} \left(\prod_{w_{context} \in context} p\left(w_{context}|w_{word}\right) \right)$$

$$p(w_{context}|w_{word}) = \frac{exp(v_{context}^T v_{word})}{\sum_{v_{context} \in vocabulary} exp(v_{context}^T v_{word})}$$

Then, the model maximizes the logarithm of the average log probability of co-occurrences:

$$\underset{w}{argmax} \sum_{w_{word} \in document} \left(\sum_{w_{context} \in context} p\left(w_{context}|w_{word}\right) \right)$$

$$= \underset{w}{argmax} \sum_{(w_{word}, w_{context}) \in D} log\left(p(w_{context}|w_{word})\right)$$

The objective function of the skip-gram model is as below:

$$\underset{w}{argmax} \sum_{(w_{word},w_{conte}\ )\in D} \left(log\left(exp(v_{context}^{T} v_{word})\right) - log\left(\sum_{v_{context}\in vocabulary} exp\,(v_{context}^{T} v_{word}) \right) \right)$$

To efficiently estimate word embeddings from the objective function, Mikolov, T. et al. (2013) proposed a sampling algorithm called the negative sampling (Mikolov, T. et al., 2013). Traditional neural probabilistic language models are computationally intensive because of the model complexity. Mnih, A.,& Teh, Y. W. (2012) introduced an efficient and stable algorithm for training neural probabilistic language models called the noise contrastive estimation, which treats a density estimation problem as a binary classification problem (Mnih, A., & Teh, Y. W., 2012). The algorithm maximizes a simple logistic regression accuracy function which differentiates meaning samples from noise. The negative sampling is a simplified version of the noise contrastive estimation in that the algorithm maximizes log probabilities of observed word–context co–occurrence statistics over hypothetically possible word–context co–occurrence statistics. Considering a single word–context co–occurrence, the skip–gram with negative sampling maximizes the objective function as below:

$$log\sigma(v_{context}^{T} v_{word}) + \sum_{i\in(1,2,\ldots,k)} \underset{v_i\sim P_n(v)}{E} \left(log\sigma(-v_i^{T} v_{word})\right)$$

$$\sigma(v_{context}^{T} v_{word}) = p\left(observed|(word, context)\right) = \frac{1}{1+exp(-v_{context}^{T} v_{word})}$$

Thus, the objective function of the skip–gram with negative sampling model is as below:

$$\sum_{v_{word}} \sum_{v_{context}} \left(count_{(word,context)} \left(log\sigma(v_{context}^T v_{word}) + \sum_{i \in (1,2,\dots,k)} \underset{v_i \sim P_n(v)}{E} \left(log\sigma(-v_i^T v_{word}) \right) \right) \right)$$

There are at least three hyperparameters to be specified: the size of the context window, the dimension of word embeddings, and the number of negative samples. Also, the probability distribution of word embeddings needs to be specified. According to Mikolov, T. et al. (2013), the unigram distribution raised to power of 0.75 rather than the original unigram distribution performs better than other distributions. The size of the context window and the dimension of word embeddings determines the richness of word embeddings. The thesis investigated a number of possible hyperparameters.

2 Pointwise Mutual information

The pointwise mutual information measures a mutual dependence between a pair of instances from independent random variables, defined as:

$$PMI(x,y) = log \frac{p(x,y)}{p(x)p(y)} = log \frac{p(x|y)}{p(x)} = log \frac{p(y|x)}{p(y)}$$

The probability distribution of word and context can be estimated from the word-context co-occurrence statistics. Thus, the pointwise mutual information in this case is as blow:

$$PMI(w_{word}, w_{context}) = log \frac{p(w_{word}, w_{context})}{p(w_{word})p(w_{context})}$$

$$p(w_{word}) = \frac{count_{(w_{word})}}{\sum count_{(w_{word},w_{context})}}, p(w_{context}) = \frac{count_{(w_{context})}}{\sum count_{(w_{word},w_{context})}}$$

$$PMI(w_{word}, w_{context}) = log \frac{count_{(w_{word}, w_{context})} \sum count_{(w_{word}, w_{context})}}{count_{(w_{word})} count_{(w_{context})}}$$

The pointwise mutual information is defined by the logarithm. The word–context co–occurrence statistics becomes extremely sparse as the size of vocabulary extends. Thereby, the pointwise mutual information is unstable to measure the word–context co–occurrence statistics. Here as elsewhere in the natural language processing, smoothing processes have been introduced. The most common approach called positive pointwise mutual information is to replace all negative values to zeros (Niwa, Y., & Nitta, Y., 1994) defined as below:

$$PPMI(w_{word}, w_{context})$$
$$= max\left(0, log \frac{count_{(w_{word}, w_{context})} \sum count_{(w_{word}, w_{context})}}{count_{(w_{word})} count_{(w_{context})}}\right)$$

The positive pointwise mutual information is efficient to estimate and, furthermore, produces the sparse matrix. The dimension of the word–context co–occurrence statistics is generally large because it is the number of words times the number of contexts. Thus, the sparsity of the word positive pointwise mutual information is useful in practical terms. In addition, Levy, O., & Goldberg, Y. (2014) proposed the shifted positive pointwise mutual information, which shifts the matrix towards greater sparsity. The shift is an analogy to the negative sampling.

$$SPPMI(w_{word}, w_{context})$$
$$= max\left(0, log \frac{count_{(w_{word}, w_{context})} \sum count_{(w_{word}, w_{context})}}{count_{(w_{word})} count_{(w_{context})}} - logk\right)$$

3 Word Embedding and Distributed Semantics

Levy, O., & Goldberg, Y. (2014) claimed that the skip–gram with negative sampling model learns word embeddings from implicit

factorization of the shifted pointwise mutual information. Namely, under certain conditions, word embeddings derived by the skip-gram with negative sampling are based on the pointwise mutual information. For each word-context co-occurrence, the skip-gram with negative sampling model maximizes the following:

$$count_{(word,context)} log\sigma(v_{context}^T v_{word}) +$$

$$k \frac{count_{(word)} count_{(context)}}{\sum count_{(word,context)}} log\sigma(-v_{context}^T v_{word})$$

The gradient of the above is as follows:

$$count_{(word,context)} \sigma(-v_{context}^T v_{word}) -$$

$$k \frac{count_{(word)} count_{(context)}}{\sum count_{(word,context)}} \sigma(v_{context}^T v_{word})$$

By solving the equation, one may find that:

$$v_{context}^T v_{word} = log \frac{count_{(w_{word},w_{context})} \sum count_{(w_{word},w_{context})}}{count_{(w_{word})} count_{(w_{context})}} - logk$$

$$v_{context}^t v_{word} = PMI(w_{word}, w_{context}) - logk$$

Somewhat surprisingly, the skip-gram with negative sampling seems to factorize the shifted pointwise mutual information. Levy, O., & Goldberg, Y. (2014) also pointed out that, since the dimension of word embeddings generally is much smaller than the size of the vocabulary, the factorization deviates from the optimal case.

Arora, S. et al. (2014) has recently proved the following theorem,

Theorem. There is a constant $Z > 0$ and some $\epsilon = \epsilon(n, d)$ that goes to 0 as $d \to \infty$ such that with high probability over the choice of word vectors, for any two different word vectors w and w'

$$logp(w, w') = \frac{\|v_w + v_{w'}\|_2^2}{2d} - 2logZ \pm \epsilon$$

$$logp(w) = \frac{\|v_w\|_2^2}{2d} - logZ \pm \epsilon$$

Jointly these imply

$$PMI(w, w') = \frac{v_w^T v_{w'}}{d} \pm O(\epsilon)$$

where $p(w|w') \propto exp(v_w^T v_{w'})$. The last equation is equivalent to what Levy, O., & Goldberg, Y. (2014) has previously proved,

$$v_{context}^T v_{word} = PMI(w_{word}, w_{context}) - logk$$

except for the consideration of the dimension of word embeddings. The theorem implies that log-linear models under the distributional hypothesis are based on the pointwise mutual information. In addition, the linear structure of word embeddings seems to lie in the nature of log-linear models based on the pointwise mutual information. Pennington, J. et al. (2014) suggests the model which captures the relational similarities,

$$F(v_i, v_j, v_k) = \frac{P_{ij}}{P_{jk}}$$

where P_{ij} is the probability of $w_i$appearance in the context of w_i. To ensure the linear structure of word embeddings, the model should satisfy the following,

$$v_i^T v_k + b_i + b_k = log(count_{(w_i, w_k)})$$

which leads to the following trivial expression,

$$PMI(w_i, w_k) \approx log(count_{(w_i, w_k)}) - b_i - b_k$$

The skip-gram with negative sampling model received much attention in natural language processing literature. Although recent

studies showed that the skip-gram with negative sampling model is actually an implicit factorization of the conventional word embeddings, the word-context co-occurrence statistics, the model still is considered as the state-of-the-art learning algorithm. The skip-gram with negative sampling is known to be the most computationally efficient and cheap model up to date. This chapter evaluates word embeddings derived by two models -the skip-gram with negative sampling and the shifted positive pointwise mutual information- to verify the claim.

4 Experiment

Word embeddings are derived with different hyperparameters respectively on Wikipedia database. Wikipedia database is used because articles on Wikipedia are written in objective manner over abundant topic. The database contains 3.7 million documents with 1.9 billion tokens. Articles with less than 50 words are ignored and the vocabulary size is limited to 100,000. Conventional text pre-processing methods including stop-word removal, lemmatization and stemming are not used except for tokenization.

Word embeddings are derived by the skip-gram with negative sampling (SGNS) and the shifted positive pointwise mutual information (SPPMI). Hyperparameters are chosen based on the previous experiments. The size of the context window (C) and the number of negative samples (K) are considered. The size of the context window is chosen from(2,5,10)and the number of negative samples is chosen from (0,2,5,20). For the shifted positive pointwise mutual information, the logarithm of the number of negative samples is used instead. Thus, the number of negative samples is chosen from (1,2,5,20).

Google's analogy dataset is used for evaluation of word embeddings derived by models. The dataset includes total 19,258 instances, of which 8,869 are semantic questions and 10675 are syntactic questions. For example, (Athens, Greece, Oslo, Norway) or (brother, sister, grandson, granddaughter) belongs to semantic questions, and (great, greater, tough, tougher) or (walking, walked, swimming, swam) belongs to syntactic questions. The questions are answered by adding and subtracting word embeddings of given words. Only the most similar word embedding is considered as an answer.

	SGNS			SPPMI		
	C = 2	C = 5	C = 10	C = 2	C = 5	C = 10
K = 0 (K = 1)	52.49	53.98	52.02	52.35	46.70	48.36
K = 2	66.27	66.15	63.34	49.85	48.14	47.01
K = 5	68.19	67.07	64.47	43.45	43.26	40.82
K = 20	66.85	67.11	65.13	43.45	31.96	29.09

Table 1 Results of word embeddings with different hyperparameters

Table 1 shows the accuracy of models with each hyperparameter. The skip-gram with negative sampling outperforms the other model, the shifted positive pointwise mutual information. The best performance is achieved with a small context window (C = 2) and large negative samples (K = 5).

For the skip-gram with negative sampling, the performance improves as the number of negative samples grows. The average accuracy of the skip-gram with zero negative samples is 52.83%, whereas the average accuracy with nonzero negative samples is 66.40%. However, for the shifted positive pointwise mutual information, the performance gets worse as the number of negative sample grows. The average

accuracy of the shifted positive pointwise mutual information with zero negative samples is 49.14%, whereas the average accuracy with nonzero negative samples is only 41.89%.

Also, the shifted positive pointwise prefers smaller context windows. The average accuracy of the model with single context window is 47.28%, whereas the average with ten context window is only 41.32%. The skip-gram with negative sampling seems to prefer smaller context windows likewise, but the effect is less obvious. The average accuracy of the skip-gram with single context window is 63.45%, whereas the average with five context window is 61.24%.

Predictive models such as neural network language models or the skip-gram are known to outperform counting models such as the pointwise mutual information. Baroni, M., Dinu, G., & Kruszewski, G. (2014) systematically compared predicting models to counting models and concluded that the former is highly superior to the latter. The result from this chapter support the claim on the one hand.

However, on the other hand, hyperparameter settings seem to play a significant role in the performance of word embeddings. Arora, S. et al. (2014) explains that the models with smaller dimension of word embeddings work better. Since the noise which deteriorates linear structure of word embeddings diminishes as the dimension of word embeddings reduces, low-dimensional word embeddings is necessary (Arora, S. et al., 2014). The dimension of word embeddings derived by the skip-gram model is 500, whereas the dimension of word embeddings derived by the shifted positive pointwise mutual information is 100,000. In this regard, although the skip-gram model outperforms the pointwise mutual information in the experiment, it is hard to claim that the former is superior to the latter.

Chapter 3 Semi-Supervised Autoencoder

The unsupervised learning models learn structure of input data without any explicit target data. The objective of unsupervised learning is understanding data rather than solving tasks. Thus, one should establish what is, and how to measure meaningful understanding. The semi-supervised learning combines unlabeled data with labeled data to improve results on supervised learning tasks. Since labeled data is much expensive and scarce to obtain than unlabeled data, semi-supervised learning models utilize unlabeled data. Thus, semi-supervised learning models simultaneously learn structure of input data and inferred information from supervised tasks.

This chapter introduces a semi-supervised autoencoder which encapsulates input structure and target information within compressed representations. Word embeddings derived by unsupervised learning algorithms may not sufficiently reflect emotional similarities among words, since the co-occurrence statistics of the corpus only represents for contextual semantics. To incorporate emotional similarities within word embeddings, the thesis proposes semi-supervised learning of word representations with a semi-supervised autoencoder.

1 Autoencoder

The autoencoder is an unsupervised shallow feed-forward network model for learning efficient representations of input data. Given input data, the autoencoder learns a nonlinear mapping function between input space and feature space. To learn meaningful nonlinear mapping function, the model minimizes cost for reconstructing original data from

representations. The model of single layer with sigmoid activation function is closely related to principal component analysis. Several variants such as the denoising autoencoder or the contractive autoencoder were proposed (Vincent, P. et al., 2008; Rifai, S. et al., 2011).

For instance, the squared Euclidean distance cost for reconstruction of a simple autoencoder is as below, where f and g denotes nonlinearity:

$$J(W,b;x) = \frac{1}{2}\|x_{reconstructed} - x\|^2$$
$$J(W,b;x) = \frac{1}{2}\left\|g\left(W^{(2)}h + b^{(2)}\right) - x\right\|^2$$
$$J(W,b;x) = \frac{1}{2}\left\|g\left(W^{(2)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(2)}\right) - x\right\|^2$$

Then, efficient representations of input data can be obtained by computing $h = f\left(W^{(1)}x + b^{(1)}\right)$ from learned parameters $W^{(1)}$ and $b^{(1)}$. One may also use the cross-entropy cost for reconstruction as below:

$$J(W,b;x) = \sum x\, log(x_{reconstruct}) + (1-x) log(1 - x_{reconstruct})$$

Stochastic gradient descent is most common approach for training neural network models. Let $\theta = \left(W^{(1)}, W^{(2)}, b^{(1)}, b^{(2)}\right)$, then for each training iteration, parameters are updated as follows:

$$\theta_{i+1}^{(k)} \leftarrow \theta_i^{(k)} - \alpha \frac{\partial J}{\partial \theta_i^{(k)}}$$

The partial derivative above is computed from the standard backward propagation of errors.

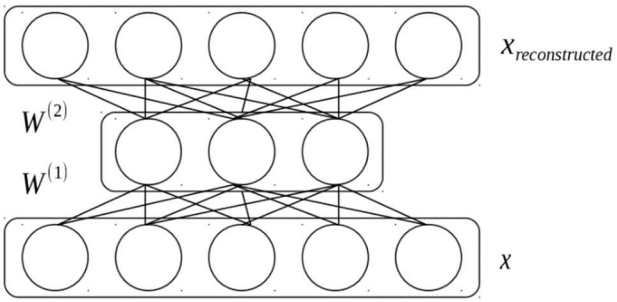

Figure 1 Illustration of shallow autoencoder

2 Semi-Supervised Autoencoder

The semi-supervised autoencoder is an shallow feed-forward network model for learning efficient and task-specific representation. Given input and target data, a semi-supervised autoencoder learns a nonlinear mapping function, which minimizes both cost for reconstructing input data and for solving supervised learning tasks. Thus, the cost function of a simple semi-supervised autoencoder is as below, where f, g and h denotes nonlinearity and L denotes prediction loss:

$$J(W,b;x,y) = J_{reconstruction}(W,b;x) + J_{prediction}(W,b;x,y)$$

$$J(W,b;x,y) = \frac{1}{2}\|x_{reconstructed} - x\|^2 + L\left(h\left(W^{(3)}h + b^{(3)}\right),y\right)$$

$$J(W,b;x,y) = \frac{1}{2}\left\|g\left(W^{(2)}h + b^{(2)}\right) - x\right\|^2 + L\left(h\left(W^{(3)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(3)}\right),y\right)$$

$$J(W,b;x,y) = \frac{1}{2}\left\|g\left(W^{(2)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(2)}\right) - x\right\|^2 + L\left(h\left(W^{(3)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(3)}\right),y\right)$$

One may also use the cross-entropy cost for reconstruction as below:

$$J(W,b;x,y) = \sum\left(xl \quad (x_{reconstruct}) + (1-x)log(x_{reconstruct})\right)$$

$$J(w,b;x,y) + L\left(h\left(W^{(3)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(3)}\right),y\right)$$

It is common to use the cross-entropy or the mean-squared error to measure prediction loss. In classification settings, it is natural to use the cross-entropy to measure prediction loss, since it postulates probability distribution over categories. In regression settings, it is more natural to use the mean-squared error to measure prediction loss. As the model learns a nonlinear mapping function, both reconstruction and prediction loss decrease. Therefore, representations of input data may encapsulate both input structure and target information.

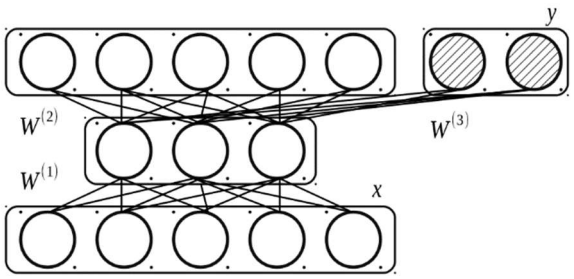

Figure 2 Illustration of semi-supervised autoencoder

3 Regularization and Sparsity

A principle called Occam's razor restrains models from being excessively complex. Regularization is most common approach to prevent excessive complexity, namely, overfitting. It has been shown that regularization evidently impact the outcome of the autoencoder (Vincent, P. et al., 2010). Variants of the autoencoder regularize in their separate ways; the denoising autoencoder stochastically corrupts input data; the contractive autoencoder adds an additional regularization term to the overall cost; sparse autoencoder also adds an additional sparsity term to the overall cost. Weight decay is generally applicable.

Semantic similarities among words are measured by cosine distance between word embeddings. This paper imposes another regularization term on which constrains overall cosine distances between word embeddings on the overall cost function. The regularization term would force word representations to be sparser in semantic space. Combined with semi-supervised learning, word representations may capture more interesting similarities among words. In practice, the regularization term slightly improved results.

Thus, the cost function of a regularized semi-supervised autoencoder is as below, where α, λ and γ denote hyperparameters and $\rho_{ij} = \|h_i\|\|h_j\|$:

$$J(W, b; x, y) = \alpha * J_{reconstruction}(W, b; x) + (1 - \alpha) J_{prediction}(W, b; x, y) + \frac{\lambda}{2}\|\theta\|^2 + \gamma \left\| \frac{hh^T}{\rho} \right\|^2$$

4 Training Semi-Supervised Autoencoder

Stochastic gradient descent approach may also be applied to a semi-supervised autoencoder. Let $\theta = (W^{(1)}, W^{(2)}, b^{(1)}, b^{(2)})$, then for each training iteration, parameters are updated from the gradients. To compute the gradient, standard backward propagation used. However, the cost function is not necessarily continuous and even non-convex, it might not be able to search for global optimal solution. In practice, mini-batch stochastic gradient descent algorithm works well.

Since the regularization term increases computational burden, the semi-supervised autoencoder requires efficient multidimensional computation. Also, since length of input data increases relative to vocabulary size, the model requires a scalable algorithm. Theano provides symbolic differentiation along with generic graphic processing

units computing. General purpose computing on graphic processing units provides massive processing power to data intensive computation. The thesis used Theano 0.7, a python library to optimize mathematical expressions, and other scientific computation libraries such as Numeric Python to construct and train a semi-supervised autoencoder.

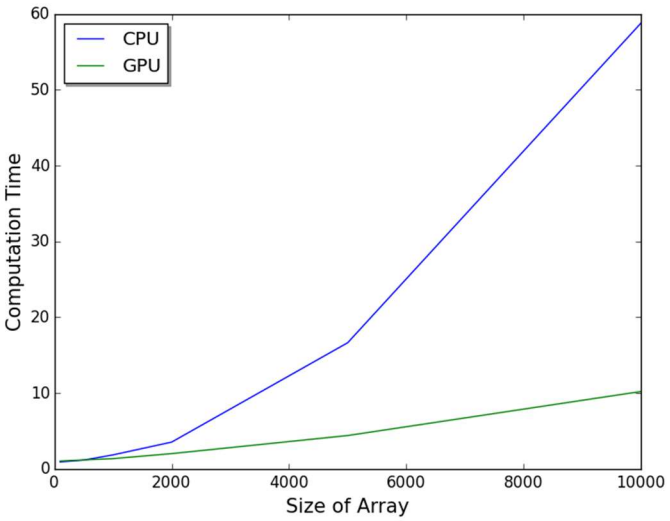

Figure 3 Comparison of computational time on CPU and GPU

Chapter 4 Sentiment Analysis

Sentiment analysis is the process to predict a sentimental state of the source. Mostly, sentimental states are binarized to two classes: objective/subjective or positive/negative. Only recently have sentiment analysis tasks with finely grained sentimental states emerged. A source is generally given as text corpus. In terms of machine learning, sentiment analysis is a supervised learning approach which an input is a representation of a sentence or document and output is its sentimental state. Bag-of-words or term frequency-inverse document frequency are commonly used to represent sentences or documents.

To utilize semantic similarities among words, sentence- or document-level embeddings should be estimated based on word embeddings. Several attempts have been made to explicitly compose word embeddings into phrase-, sentence-, and document-level embeddings. The recursive autoencoder is one of the most successful models up to date. The model greedily construct autoencoders over word embeddings. Thus, the representation assimilates semantic information of word embeddings as it grows. With the autoencoder of binary tree structure, the model learns sentence- or document-level embeddings (Socher R. et al., 2013). The convolutional neural network is another successful model to compose word embeddings. The model utilizes convolution operations and pooling to extract representations of documents (Kim, Y., 2014). However, learning composition model of word embeddings is beyond the scope of the thesis.

This chapter introduces word embeddings suitable to sentiment analysis. To embrace emotional aspects of meaning, the thesis tunes word embeddings derived by the skip-gram with negative sampling model based on emotional dimensions of words. Semi-supervised learning architecture is used, which extracts information from both unlabeled and

labeled data. A semi-supervised autoencoder is used to incorporate emotional similarities among words while preserving most prominent features of semantic similarities among words. A simple procedure is conducted to estimate sentence- or document-level embeddings.

1 Dimensional Model of Emotion

It is widely accepted that emotions could be described as multidimensional features. Emotions are hard to discretely differentiate because they are highly inter-correlated. The most plausible theoretical description up to date is that two independent dimensions compose every emotional states. Moreover, these two independent factors arise from independent neurophysiological systems (Russell, J. A., 1980; Bradley, M. M. et al., 1992; Kensinger, E. A., 2004; Posner, J. et al., 2005).

Mostly, two independent dimensions which compose emotional states are called valence and arousal. Valence measures pleasure or displeasure, and arousal measures activeness. For example, "excited" and "relaxed" have positive valence whereas "nervous" and "bored" have negative valence. "Excited" and "nervous" have heightened arousal whereas "relaxed" and "bored" have diminished arousal.

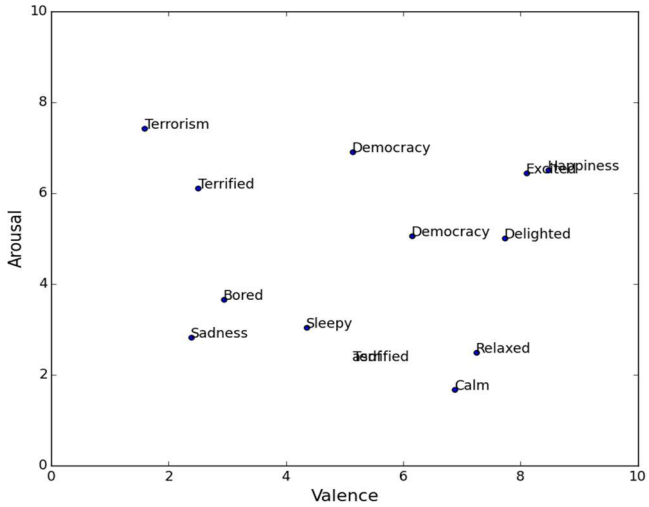

Figure 4 Valence-arousal space of English words

Valence and arousal are immediate neurophysiological responses to stimuli. When neurophysical systems process natural language, these physiological responses may arise and bring about certain changes in affective states. Recent researches suggest while reading emotional words, affective responses arise in advance of language processing. Furthermore, affective responses arise regardless of language processing (Egidi, G., & Nusbaum, H. C., 2012; Citron, F. M. et al., 2014; Kuperman, V. et al., 2014; Recio, G. et al., 2014). One may suppose then, that sentiment analysis within neurophysical system might embrace affective states into language processing.

The distributional hypothesis claims that words with similar contexts tend to have similar meanings. Namely, semantically similar words are distributionally similar. Since word embedding models are based on the hypothesis, their outcomes follow the hypothesis as well. However, are semantically similar words expected to have similar emotional effect? According to Potts, C. (2007), expressive content is independent from descriptive content. For example, a phrase "That bastard Kresge is

famous." means "Kresge is famous" in descriptive sense, whereas it means "Kresge is bastard" in expressive sense. "Positive" and "negative" have similar contexts whereas their valence are completely opposite. Therefore, One may argue that word embeddings based on the distributional hypothesis are deficient in emotional information.

2 Emotional Similarity

Semantic similarities between words are estimated from the word-context co-occurrence statistics. Emotional similarities between words cannot be estimated from the word-context co-occurrence statistics because emotional responses are rather immediate neurophysiological responses than sophisticated articulation. A few studies have measured emotional features directly from human participants. Recently, Warriner, A. B. et al. (2013) published a database with valence, arousal, and dominance of 13,915 English lemmas (Warriner, A. B. et al., 2013). The data were collected from participants who reside in the United States and include average, standard deviation and the number of responses.

	Valence		Arousal	
	Mean	Deviation	Mean	Deviation
Insanity	2.7	1.81	7.79	1.44
Dull	3.4	0.94	1.67	1.03
...	...			
Fantastic	8.36	0.79	6.4	2.6
Soothing	7.05	1.66	1.91	1.31

Table 2 Examples of valence and arousal of words

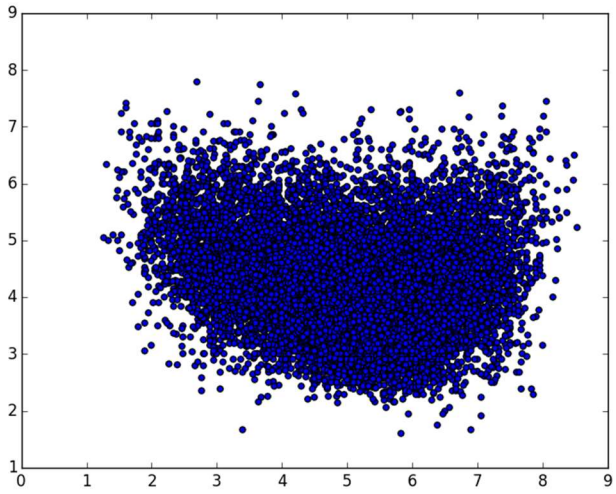

Figure 5 Distribution of valence and arousal of words

2.1 KL-Divergence

To measure emotional similarities among words, the thesis utilizes Kullback_Leibler divergence. The Kullback-Leibler divergence measures the difference between probability distributions. For continuous probability distributions, the Kullback-Leibler divergence is defined as below:

$$D_{KL}(P \parallel Q) = \int p\,(x) log \frac{p(x)}{q(x)} dx$$

The affective norms database provide average and standard deviation of emotional dimensions of each word. Let each dimension of a single word be a random variable with Gaussian distribution. Then, the Kullback-Leibler divergence becomes:

$$D_{KL}(P \parallel Q) = log \frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_2^2} - \frac{1}{2}$$

However, the Kullback–Leibler divergence does not satisfy the symmetry condition of metric. To symmetrize the Kullback–Leiber divergence without losing its theoretical background, the thesis follows the method called the resistor–average distance, which is defined as below (Johnson, D., & Sinanovic, S., 2001):

$$\frac{1}{d(P,Q)} = \frac{1}{D_{KL}(P \parallel Q)} + \frac{1}{D_{KL}(Q \parallel P)}$$

Emotional similarity matrix is built based on the affective norms database and the resistor–average distance metric. Given the similarity matrix of each emotional dimension, the multidimensional scaling algorithm is used to reconstruct latent geometry of the dimension. To make the problem simple, the thesis applied linear support vector machine algorithms to classify each emotional dimension.

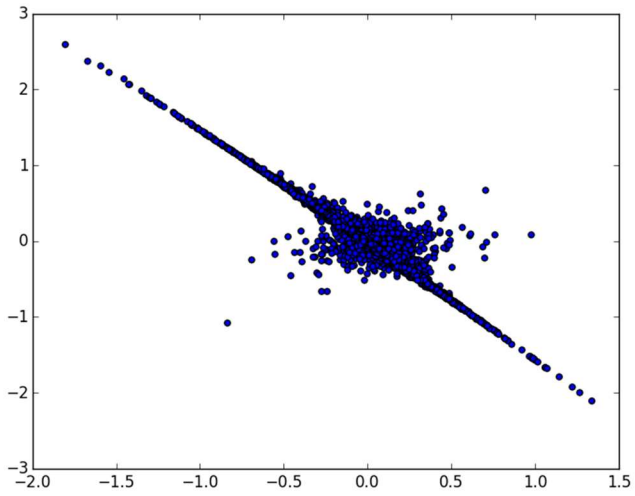

Figure 6 Distribution of valence from multidimensional scaling

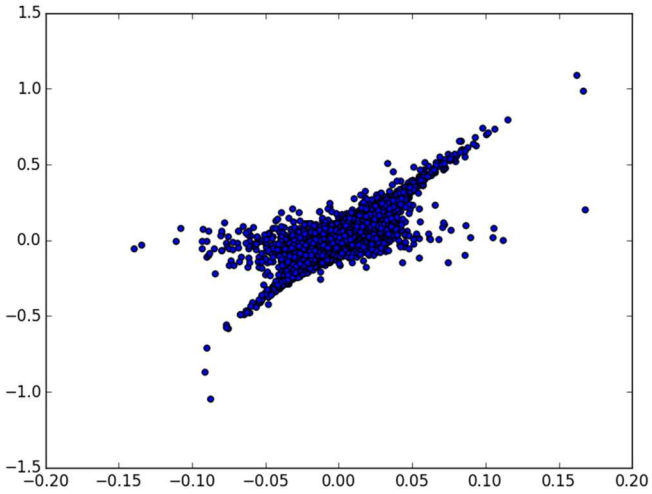

Figure 7 Distribution of arousal from multidimensional scaling

	Valence	**Arousal**
Addiction	negative	low
Compassion	positive	high
Antique	positive	neutral
...	...	
Oblivion	negative	neutral
Supervisor	neutral	neutral
Disrespectful	negative	low

Table 3 Examples of classes of valence and arousal of words

3 Experiment

The experiment involves two steps: (1) Estimating word embeddings which capture both semantic and emotional similarities with affective norms database via semi-supervised autoencoder and (2) analyze polarity and subjectivity of sentiment analysis dataset. Word embeddings

are first estimated using the skip-gram with negative sampling or positive pointwise mutual information model. Then, a representation learning is conducted using a semi-supervised autoencoder with emotional class derived by the affective norms database.

The thesis adopts both qualitative and quantitative assessment. Following Maas, A. L. et al (2011), the thesis compares word similarities of chosen words (Maas, A. L. et al., 2011). Then, by simply averaging word embeddings seen from texts, sentence- or document-level embeddings are estimated. A linear support vector machine with default hyperparameters is used for classifications. Since the compositionality of meaning is not being considered, quantitative assessment would be an indirect indicator of the richness of word embeddings.

3.1 Word Embedding

Word embeddings learned on Wikipedia corpus using the skip-gram with negative sampling or the positive pointwise mutual information are used to train a semi-supervised autoencoder. The best combination of hyperparameters is the skip-gram with negative sampling model with single window (window = 2) and five negative samples(negative = 5). The size of the dictionary is fixed at 100,000 words and the dimension of word embeddings is fixed at 500. Among 13,915 English lemmas which are labeled from the affective norms database, 13,011 lemmas are found in word embeddings. As previous experiments, Google's analogy dataset is used for evaluation of word embeddings.

A semi-supervised autoencoder learns a nonlinear mapping function which incorporates semantic and sentiment information. With 13,011 labeled word embeddings and 86,989 unlabeled word embeddings, the model learns five parameters:$W^{(1)}$,$W^{(2)}$,$W^{(3)}$,$b^{(1)}$,$b^{(1)}$. The model takes

three regularization methods: an explicit regularization term on similarities among word embeddings in the overall cost function, the weight decay, and the tied-weight. The cross-entropy cost between predicted classes of valence and target data is used for the prediction cost, and the squared Euclidean distance cost is used for the reconstruction cost. The overall cost of the model is as below:

$$J(W,b;x,y)=\frac{\alpha}{2}\left\|g\left(\left(W^{(1)}\right)^{(T)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(2)}\right)-x\right\|^2$$

$$J(w,b;x,y)+(1-\alpha)L\left(h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right),y\right)+\frac{\lambda}{2}\|\theta\|^2+\gamma\left\|\frac{hh^T}{\rho}\right\|^2$$

$$L\left(h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right),y\right)=$$

$$\sum y\,log\left(h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right)\right)L\left(h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right),y\right)+$$

$$(1-y)log\left(1-h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right)\right)$$

Table 4 shows similarities among induced word embeddings using the model. The thesis compares the method to Maas, A. L. et al. (2011). In order to allow fair comparison between models, word embeddings learned on Large Movie Review Dataset are also used to train a semi-supervised autoencoder. Word embeddings learned on Large Movie Review Dataset seem comparable to or worse the previous model. However, emotionally enhanced word embeddings seem better than the original word embeddings. Word embeddings learned on Wikipedia corpus seem better than the previous model.

Table 5 shows other examples. Emotional words such as "happiness" are semantically similar to their anti-sentimental words such as "sadness". However, emotionally enhanced word embeddings seem to have improved. For instance, the dissimilarity between "positive" and "unfavorable" has relatively decrease, while similarity between "joy" and "happiness" has relatively increased. Also, the dissimilarity between "happiness" and "sadness" has relatively decrease, while similarity between "happiness" and "wholeness" has relatively increased. Although the accuracy of Google's analogy test decreases in comparison with

semantic-only word embeddings, semantic similarities among word embeddings seem undiminished.

	This thesis				Maas, A. L. et al. (2011)		
	Word2Vec	Proposed	Word2Vec (Wikipedia)	Proposed (Wikipedia)	Full	Semantic	LSA
Melancholy	lyrical	evocative	melancholic	somber	poetic	thoughtful	bittersweet
	evocative	hauntingly	dreamy	melancholic	lyrical	warmth	heartbreaking
	deft	lyrical	wistful	dreamy	poetry	layer	happiness
	wry	accompanies	languid	elegiac	profound	gentle	tenderness
	hauntingly	aplomb	sadness	languid	vivid	loneliness	compassionate
Ghastly	threadbare	atrocious	loathsome	terrifying	hideous	predators	embarrassingly
	absurdly	absurdly	terrifying	loathsome	inept	hideous	trite
	ponderous	putrid	horrifying	horrifying	severely	tube	laughably
	drenched	unimpressive	nightmarish	nightmarish	grotesque	baffled	atrocious
	overwrought	threadbare	maddening	unsettling	unsuspecting	smack	appalling
Lackluster	lacklustre	uninspired	underwhelming	mediocre	uninspired	passable	lame
	leaden	leaden	lacklustre	underwhelming	flat	unconvincing	laughable
	threadbare	unimaginative	mediocre	subpar	bland	amateurish	unimaginative
	uninspired	plodding	subpar	lacklustre	forgettable	cliched	uninspired
	unimaginative	pedestrian	unimpressive	unimpressive	mediocre	insipid	awful
Romantic	romance	romance	romance	romance	romance	romance	romance
	screwball	drama	erotic	erotic	screwball	charming	love
	drama	comedy	melodrama	unrequited	grant	delightful	sweet
	comedy	screwball	unrequited	homoerotic	comedies	sweet	beautiful
	charming	comedies	amorous	amorous	comedy	chemistry	relationship

Table 4 Similarity of learned word embeddings

	Word2Vec (Wikipedia)	Proposed (Wikipedia)
Positive	negative	negative
	favorable	favorable
	positively	favourable
	favourable	positively
	unfavorable	review
Happiness	contentment	prosperity
	sadness	contentment
	prosperity	wholeness
	blissful	blissful
	transience	joy
Insensitive	condescending	condescending
	hurtful	sexist
	judgmental	uncouth
	sexist	demeaning
	sensitive	boorish

Table 5 Emotional similarity of learned word embeddings

	Word2Vec (Wikipedia)	Proposed (Wikipedia)
Accuracy	68.19	59.49

Table 6 Evaluation of learned word embeddings on Google's analogy data

3.2 Sentiment Analysis

One may doubt whether emotionally-enriched word embeddings would improve sentiment analysis or not. It seems obvious that better word embeddings would guarantee better results in sentiment analysis

under ideal conditions, since sentences or documents are compositions of words. Several researches have studied compositionality in sentiment as well (Moilanen, K., & Pulman, S., 2007; Choi, Y., & Cardie, C., 2008; Yessenalina, A., & Cardie, C., 2011;). The recursive deep model is the most prominent model in that it works well with complex compositional structures of words with multiple negations (Socher R. et al., 2013).

To evaluate word embeddings obtained from the semi-supervised learning, the thesis conducted sentence- and document-level sentiment polarity classification, subjectivity classification and sentimental state classification. Since valence is the most relevant emotional dimension to sentiment analysis, the thesis used valence dimension to conduct semi-supervised learning. Based on four different sentiment datasets, sentence- and document-level embeddings are estimated with word embeddings learned on Wikipedia corpus and, in addition, word embeddings learned on each sentiment dataset. Sentence- or document-level embeddings are average of their word embeddings. Namely, no other weighting method such as term frequency-inverse document frequency is applied. Since the compositionality of word embeddings is somewhat excessively simple and naive, sentiment classification results may fall short of the state-of-the-art results. Maas, A. L. et al. (2011) provides results over binary-weighted sentence- or document-level embeddings (Maas, A. L. et al., 2011).

3.2.1 Dataset

The sentence polarity dataset 1.0 from Cornell Movie Review Dataset (PL05) consists of 5,331 positive reviews and 5,331 negative reviews with one sentence collected from Rotten Tomatoes. The dataset

is not split into training and testing samples, and results are obtained by the 10-fold cross validation (Pang, B., & Lee, L., 2005).

The Large Movie Review Dataset (IMDB) consists of 50,000 labeled reviews and 50,000 unlabeled reviews with varying length collected from IMDB. The dataset is split into 25,000 labeled training samples and 25,000 labeled testing samples. Labeled samples are polarized, while unlabeled samples are not. Word embeddings can be learned on 25,000 labeled training samples and 50,000 unlabeled samples (Maas, A. L. et al., 2011).

The subjectivity dataset 1.0 from Cornell Movie Review Dataset (SUBJ) consists of 5,000 subjective reviews and 5,000 negative reviews with one sentence collected from Rotten Tomatoes. The dataset is not split into training and testing samples, and results are obtained by the 10-fold cross validation (Pang, B., & Lee, L., 2004).

The Stanford sentiment treebank (SST) is based on the sentence polarity dataset from Cornell Movie Review Dataset, and consists of 11,855 sentences. After having parsed each sentence, 215,514 phrases are labeled as one of five categories by human annotators: negative, somewhat negative, neutral, somewhat positive, positive. These "fine-grained sentiment" dataset is split into 8,544 training samples, 1,101 developing samples, and 2,210 testing samples. Word embeddings can be learned on 8,544 training samples and 1,101 developing samples (Socher R. et al., 2013).

3.2.2 Result

Table 7 shows the performance of the model compared to other models reported from the literature (Pang, B., & Lee, L., 2004; Maas, A.

L. et al., 2011; Socher R. et al., 2011; Socher R. et al., 2013; Kim, Y., 2014; Le, Q. V., & Mikolov, T., 2014).

Model	PL05	IMDB	SUBJ	SST
Word2Vec	67.9	83.6	87.2	32.9
Proposed Word Embeddings	70.2	83.9	88.6	33.9
Word2Vec (Wikipedia)	75.9	83.6	90.4	41.3
Proposed Word Embeddings (Wikipedia)	76.1	83.4	90.7	43.3
Proposed Word Embeddings (Wikipedia) + TF-IDF	78.1	87.8	91.4	39.2
TF-IDF (binary)	76.9	87.5	90.5	38.5
TF-IDF (sublinear)	77.1	87.6	90.6	38.4
Bag of Words (Pang, B., & Lee, L., 2004)	-	-	90.0	-
(Maas, A. L. et al., 2011)	-	88.0	-	-
Recursive Autoencoder (Socher R. et al., 2011)	77.7	-	-	43.2
Recursive Neural Tensor Network (Socher R. et al., 2013)	77.9	-	-	45.7
Convolutional Neural Network (Random) (Kim, Y., 2014)	76.1	-	-	45.0
Paragraph Vector (Le, Q. V., & Mikolov, T., 2014)	-	92.6	-	48.7

Table 7 Sentiment classification results on datasets

Word embeddings enriched with emotional information mostly improve classification results. Except for the large movie review dataset, emotional word embeddings performed better than the original. Also, word embeddings become more sentimental with emotional information mostly when they are learned on the dataset.

For the subjectivity dataset and the Stanford Sentiment Treebank, word embeddings with emotional information outperformed the term frequency–inverse document frequency. For the sentence polarity dataset, word embeddings with emotional information matches recursive and convolutional models. Also, for the Stanford Sentiment Treebank, word embeddings with emotional information matches the recursive

autoencoder. The term frequency–inverse document frequency estimate sentence- or document-level representations by weighting terms, and other models estimate sentence- or document-level representations considering compositionality of semantics whereas the thesis uses simple vector averaging method.

3.3 Discussion

The semantic representation of word as embeddings has proven to be useful. However, there is also a room for development of more emotionally elaborate model. Although the word embedding model well captures semantic similarities in words, it has limits on capturing emotional similarities. The gap between expressive content and descriptive context seems irreducible within the distributed semantics hypothesis. What makes the problem worse is that emotions are neurophysiological responses.

Learning word embeddings enriched with emotional information using the semi-supervised autoencoder thus has independent objectives. Word embeddings should capture emotional similarities in words while preserving semantic similarities. Hyperparameters of the model are chosen to satisfy both objectives. Still, the affective norms include only a small portion of the whole vocabulary, and is only an indicator of actual responses.

Several attempts have also been made to improve word embeddings for sentiment analysis, most of which utilize target data of sentimental sentences or documents (Maas, A. L. et al., 2011; Socher R. et al., 2011; Socher R. et al., 2013; Tang, D. et al., 2013). These methods reflect emotional information in word embeddings rather indirectly.

Nevertheless, these indirect models outperform other semantic-only models for most cases.

Chapter 5 Conclusion

Although the word embedding model has long been of interest, an examination on its representability of emotional responses has not yet been held. Recent papers have attempted to enrich word embeddings with emotional information (Maas, A. L. et al., 2011; Socher R. et al., 2011; Tang, D. et al., 2013). However, these methods are rather sentiment analysis task-specific than general. In order to examine word embeddings on its representability of emotional responses, a bidirectional approach has been made.

Firstly, the thesis addresses that current word embedding models including the skip-gram with negative sampling are based on the distributional hypothesis of semantics. According to the hypothesis, semantics of words are held firmly on their co-occurrence statistics. Chapter 2 explains how current word embedding models are derived by the word-context co-occurrence statistics. Chapter 4 explains how the distributional hypothesis limits word embedding models' representability of emotional responses.

Secondly, the thesis explains that beneath the notable success of predictive models such as the skip-gram with negative sampling lies the conventional pointwise mutual information approach. It has widely been accepted that count-based models only capture the attributional similarities. However, recent studies have shown that the log-linear model on the probability distribution of word occurrence is based on the word-context co-occurrence statistics. Chapter 2 reviews current literature on word embeddings and conduct evaluation on word embedding models.

Lastly, the thesis proposes a semi-supervised learning algorithm to enrich word embeddings with emotional information. Since expressive

and descriptive content are somewhat independent, and since emotions are neurophysiological responses, word embedding models based on the distributional hypothesis of semantics are insufficient. In order to produce more elaborate word embeddings which capture both semantic and emotional similarities in words, a semi-supervised autoencoder is used. Chapter 3 addresses a semi-supervised autoencoder architecture and its regularization. Chapter 4 addresses representability of word embeddings enriched with emotional information and performs evaluation on sentiment analysis.

Throughout chapters, the thesis addresses that the representability of current word embeddings has limits in that the model from which they are derived concerns only contextual information. Since emotional information lies outside contextual information, additional sources including sentimental documents or emotional norms of words themselves are required to enrich word embeddings.

It is evident that more emotional word embedding models would improve results in sentiment analysis, since semantic and emotional meaning of sentences or documents are based on the composition of words. Thus, the proposed word embedding model can be utilized over applications of sentiment analysis. For example, the model could be used in analyzing sentiment polarity of pre-news data.

The proposed word embeddings with emotional information in this thesis better capture emotional similarities among words. Nevertheless, sentiment analysis still requires more sophisticated compositionality of words. Sentences or documents have underlying semantic or syntactic structures, which differentiate the whole meaning. In particular, it is observed that word embeddings with emotional information are less effective with a longer sentence or document. Thus, more efforts should focus on the compositionality issue in future.

Bibliography

Arora, S., Li, Y., Liang, Y., Ma, T., & Risteski, A. (2015). Random Walks on Context Spaces: Towards an Explanation of the Mysteries of Semantic Word Embeddings. arXiv preprint arXiv:1502.03520.

Baroni, M., Dinu, G., &Mnih, A., & Teh, Y. W. (2012). A fast and simple algorithm for training neural probabilistic language models. arXiv preprint arXiv:1206.6426.

Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A neural probabilistic language model. The Journal of Machine Learning Research, 3, 1137-1155.

Bradley, M. M., Greenwald, M. K., Petry, M. C., & Lang, P. J. (1992). Remembering pictures: pleasure and arousal in memory. Journal of experimental psychology: Learning, Memory, and Cognition, 18(2), 379.

Choi, Y., & Cardie, C. (2008, October). Learning with compositional semantics as structural inference for subsentential sentiment analysis. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 793-801). Association for Computational Linguistics.

Citron, F. M., Gray, M. A., Critchley, H. D., Weekes, B. S., & Ferstl, E. C. (2014). Emotional valence and arousal affect reading in an interactive way: neuroimaging evidence for an approach-withdrawal framework. Neuropsychologia, 56, 79-89.

Egidi, G., & Nusbaum, H. C. (2012). Emotional language processing: how mood affects integration processes during discourse comprehension. Brain and language, 122(3), 199-210.

Firth, J.R. (1957). "A synopsis of linguistic theory 1930-1955". Studies in Linguistic Analysis (Oxford: Philological Society): 1－32. Reprinted in F.R. Palmer, ed. (1968). Selected Papers of J.R. Firth 1952-1959. London: Longman.

Johnson, D., & Sinanovic, S. (2001). Symmetrizing the kullback-leibler distance.

Kensinger, E. A. (2004). Remembering emotional experiences: The contribution of valence and arousal. Reviews in the Neurosciences, 15(4), 241–252.

Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Vol. 1, pp. 238–247).

Kuperman, V., Estes, Z., Brysbaert, M., & Warriner, A. B. (2014). Emotion and language: valence and arousal affect word recognition. Journal of Experimental Psychology: General, 143(3), 1065.

Levy, O., & Goldberg, Y. (2014). Dependency based word embeddings. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Vol. 2, pp. 302–308).

Le, Q. V., & Mikolov, T. (2014). Distributed representations of sentences and documents. arXiv preprint arXiv:1405.4053.

Levy, O., & Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In Advances in Neural Information Processing Systems (pp. 2177–2185).

Levy, O., Goldberg, Y., & Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. Transactions of the Association for Computational Linguistics, 3, 211–225.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1 (pp. 142–150). Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

Mnih, A., & Teh, Y. W. (2012). A fast and simple algorithm for training neural probabilistic language models. arXiv preprint arXiv:1206.6426.

Moilanen, K., & Pulman, S. (2007, September). Sentiment composition. In Proceedings of RANLP (Vol. 7, pp. 378-382).

Morin, F., & Bengio, Y. (2005, January). Hierarchical probabilistic neural network language model. In Proceedings of the international workshop on artificial intelligence and statistics (pp. 246-252).

Niwa, Y., & Nitta, Y. (1994, August). Co-occurrence vectors from corpora vs. distance vectors from dictionaries. In Proceedings of the 15th conference on Computational linguistics-Volume 1 (pp. 304-309). Association for Computational Linguistics.

Osterlund, A., Odling, D., & Sahlgren, M. Factorization of Latent Variables in Distributional Semantic Models.

Pang, B., & Lee, L. (2004, July). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the 42nd annual meeting on Association for Computational Linguistics (p. 271). Association for Computational Linguistics.

Pang, B., & Lee, L. (2005, June). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (pp. 115-124). Association for Computational Linguistics.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014), 12, 1532-1543.

Posner, J., Russell, J. A., & Peterson, B. S. (2005). The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. Development and psychopathology, 17(03), 715-734.

Potts, C. (2007). The expressive dimension. Theoretical linguistics, 33(2), 165–198.

Recio, G., Conrad, M., Hansen, L. B., & Jacobs, A. M. (2014). On pleasure and thrill: The interplay between arousal and valence during visual word recognition. Brain and language, 134, 34–43.

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). Contractive auto-encoders: Explicit invariance during feature extraction. In Proceedings of the 28th International Conference on Machine Learning (ICML-11) (pp. 833–840).

Russell, J. A. (1980). A circumplex model of affect. Journal of personality and social psychology, 39(6), 1161.

Schnabel, T., Labutov, I., Mimno, D., & Joachims, T. (2015). Evaluation methods for unsupervised word embeddings. In Proc. of EMNLP.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., & Manning, C. D. (2011, July). Semi-supervised recursive autoencoders for predicting sentiment distributions. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 151–161). Association for Computational Linguistics.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013, October). Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the conference on empirical methods in natural language processing (EMNLP) (Vol. 1631, p. 1642).

Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning sentiment-specific word embedding for twitter sentiment classification. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Vol. 1, pp. 1555–1565).

Turian, J., Ratinov, L., & Bengio, Y. (2010, July). Word representations: a simple and general method for semi-supervised learning. In Proceedings of the 48th annual meeting of the association for computational linguistics (pp. 384–394). Association for Computational Linguistics.

Turney, P. D., & Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. Journal of artificial intelligence research, 37(1), 141–188.

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (pp. 1096-1103). ACM.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P. A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. The Journal of Machine Learning Research, 11, 3371-3408.

Warriner, A. B., Kuperman, V., & Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. Behavior research methods, 45(4), 1191-1207.

Yessenalina, A., & Cardie, C. (2011, July). Compositional matrix-space models for sentiment analysis. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 172-182). Association for Computational Linguistics.

Appendix

1 Pseudocode of Semi-Supervised Autoencoder

1 Initialize *Theano shared variables* $\theta = \left(W^{(1)}, W^{(2)}, W^{(3)}, b^{(1)}, b^{(2)}, b^{(3)}\right)$

2 Build training function for semi-supervised autoencoder

2.1 Compute hidden representation matrix $f\left(W^{(1)}x + b^{(1)}\right)$ and reconstruction matrix $g\left(W^{(2)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(2)}\right)$

2.2 Compute prediction matrix $h\left(W^{(3)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(3)}\right)$

2.3 Compute average loss of input matrix with reconstruction loss, prediction loss and regularization term with hyperparameters, α, γ, λ as $\alpha * J_{reconstruction}(W, b; x) + (1 - \alpha)J_{prediction}(W, b; x, y) + \frac{\lambda}{2}\|\theta\|^2 + \gamma \left\|\frac{hh^T}{\rho}\right\|^2$

2.4 Get gradient of average loss and update parameters using *Theano tensor gradient*

3 While stopping criterion is not met,

3.1 For each mini-batch, a *Theano matrix*, run training function

3.2 Compute average loss of current iteration and check if stopping criterion is met

요약 (국문초록)

본 연구는 감정 차원을 고려한 단어 벡터 모델을 제시하고 감성 분석에 적용하였다. 특히, 단어의 감정 정보를 기존의 단어 벡터 모델에 종합하기 위하여 준지도학습을 수행하는 오토인코더를 활용하였다. 감성 분석은 문장이나 문서로부터 작성자의 감성 상태를 추론하는 것이다. 감정 차원은 감성 상태의 요소이다. 즉, 단어 벡터를 활용한 감성 분석이 효과적으로 이루어지기 위하여는 단어 벡터가 감정 차원의 정보를 포함하고 있어야 할 것이다. 그러나 분포 가설에 기반한 기존의 단어 벡터는 감정 정보를 온전히 포함하지 못한다. 단어 벡터를 활용한 감성 분석에서 이를 극복하기 위하여 본 연구는 문장이나 문서가 아닌 단어 자체의 감정 차원을 고려하였다. 준지도학습을 바탕으로 기존의 단어 벡터 모델이 감정 차원의 정보를 담을 수 있도록 모델을 제시하였다. 이를 바탕으로 감성 분석을 수행한 결과 감정 정보가 미비한 단어 벡터에 비하여 향상된 결과를 얻을 수 있었다.

주요어: 단어 벡터, 분포 가설, 감정 차원, 감성 분석, 준지도학습, 오토인코더

학번: 2014-21803

산업공학석사 학위논문

Word embedding for sentiment analysis considering emotional dimensions

감정 차원을 고려한 단어 벡터 모델과 감성 분석에 관한 연구

2016 년 2 월

서울대학교 대학원
산업공학과
구 본 효

Abstract

Word embedding for sentiment analysis considering emotional dimensions

Koo Bonhyo
Industrial Engineering
The Graduate School
Seoul National University

Recent studies have shown that word embeddings based on the word-context co-occurrence statistics are suited to measure semantic similarities. However, word embeddings are deficient in emotional information. This thesis reviews current word embedding models and presents word embeddings enriched with emotional information. Word embeddings are learned based on the previous word embeddings using a semi-supervised autoencoder model to incorporate affective norms data. Then, the thesis evaluates word embeddings enriched with emotional data on sentiment classification datasets.

Keywords: word embedding, distributional hypothesis, emotional dimension, sentiment analysis, semi-supervised learning, autoencoder

Student Number: 2014-21803

Table of Contents

Index of Tables

Index of Figures

Chapter 1 Introduction

Word embeddings represent semantics of words as vectors of real numbers. Unlike the traditional language models which directly map words into indices, word embedding models encode semantic similarities among words using the co-occurrence statistics of the corpus. A popular approach to capture semantics of words is called the continuous vector space model. The vector space model measures attributional similarities among words on the basis of their contexts (Turney, P. D., & Pantel, P., 2010). Another popular approach is called the neural network language model. The neural network language model constructs a neural network which predict probability distribution of words given contexts.

Recently, Mikolov, T. et al. (2013) proposed an unsupervised shallow network model for estimating word embeddings called the skip-gram with negative sampling; the state-of-the-art model architecture across various tasks (Mikolov, T. et al., 2013). Remarkably, it has been shown that the skip-gram captures not only attributional similarities, but also relational similarities. Namely, word embeddings derived by the skip-gram better represent semantic similarities among words. For example, the model allows one to predict Paris is to France as Rome is to Italy. Baroni, M., Dinu, G., & Kruszewski, G. (2014) claimed that the skip-gram model is highly superior to vector space models.

However, recent studies suggest the skip-gram is closely related to other traditional vector space models (Levy, O., & Goldberg, Y. ,2014; Arora, S. et al, 2015). It has been known that vector space models base on the co-occurrence statistics are suited to measure attributional similarities. According to Levy, O., & Goldberg, Y. (2014), vector space models may perform as well as the skip-gram under additional hyperparameters and re-weightings (Levy, O., & Goldberg, Y., 2014).

Furthermore, Arora, S. et al. (2015) provided theoretical explanation for word embeddings using co-occurrence statistics (Arora, S. et al., 2015).

Training word embedding models may be categorized as unsupervised learning. Models observe sequence of words from corpus, and estimate vectors of real numbers which reflect contextual structure of words. Vector space models such as pointwise mutual information or positive pointwise mutual information can be directly computed from co-occurrence statistics. Predictive models such as neural network language models or the skip-gram estimate word embeddings by maximizing the probability of co-occurrence statistics. Thus, training word embedding models does not require labeled, or task-specific data.

Word embeddings are found to be useful for several natural language processing tasks such as chunking or name entity recognition (Turian, J. et al., 2010). Word embeddings are generally even useful for sentence- or document-level tasks (Maas, A. L. et al., 2011; Socher R. et al., 2013). However, estimating sentence- or document-level embeddings out of word embeddings is another major task in natural language processing.

1 Contribution

This thesis aims to estimate word embeddings suitable to analyze sentiment polarity and subjectivity of document. In general, sentence- or document-level sentiment polarity classification, namely, sentiment analysis use bag-of-words representations. Instead, the thesis propose semi-supervised learning algorithm to estimate word embeddings for sentiment analysis incorporating dimensional models of emotion.

Firstly, the thesis reviews the concept of current word embedding models. The thesis mainly focuses on the skip-gram with negative sampling and pointwise mutual information. Word embeddings derived

by the skip-gram with negative sampling or pointwise mutual information are evaluated on the basis of relational similarities.

Secondly, the thesis presents a semi-supervised autoencoder with regularization on similarities among word embeddings. Rather than capturing semantic similarities among words only via unsupervised learning, the thesis incorporates sentiment information of words via semi-supervised learning. Recent researches suggest that affective states arise in the early stage of processing emotional words (Recio, G. et al., 2014).

The proposed method is compared with term frequency document inverse frequency representations and other well-known word embedding models for sentiment analysis. The thesis evaluates the method on four different datasets; sentence polarity dataset and subjectivity dataset from Cornell Movie Review Data (Pang, B., & Lee, L., 2005); Stanford Sentiment Treebank (Socher R. et al., 2013); Large Movie Review Dataset (Maas, A. L. et al., 2011). Since estimating sentence- or document-level embedding out of word embeddings is beyond the scope, the thesis generates sentence- or document-level embedding by simply averaging word embeddings.

2 Related Work

The idea of the estimating word embeddings based on the word-context co-occurrence statistics was already introduced in 1950s by Firth, J.R. (1957). However, it was not until the 1990s that the latent semantic analysis, one of the earliest vector space models was introduced. Latent semantic analysis, or latent semantic indexing is a method which applies a dimensionality reduction method, singular value decomposition to term-document co-occurrence statistics.

In 2003, Bengio, Y. et al. (2003) proposed the neural probabilistic language model which models the probability distribution of word given context (Bengio, T. et al., 2013). The neural probabilistic language model proved a great success. However, the model was computationally expensive to train. Morin, F., & Bengio, Y. (2005) proposed the hierarchical probabilistic neural network language model, which introduces hierarchical decomposition into probability estimation to efficiently train the neural network language model (Morin, F., & Bengio, Y., 2005).

Mikolov, T. et al. (2013) proposed an unsupervised shallow network model called the skip-gram with negative sampling. The model outperformed other word embedding models including the neural network language model and the vector space model (Mikolov, T. et al., 2013). Remarkably, the model captured both attributional and relational similarities with shallow architecture and efficient algorithm. Several attempts have been made to explain the skip-gram model and discovered that predictive models including the skip-gram are closely related to traditional count-based models (Levy, O., & Goldberg, Y. ,2014; Pennington, J. et al., 2014; Arora, S. et al., 2015; Osterlund, A. et al., 2015; Schnabel, T. et al., 2015).

Sentiment analysis using word embeddings has been studied in recent years. In 2011, Maas, A. L. et al (2011) proposed a probabilistic model which captures both semantic similarities and sentiment on sentimental documents. The model learns sentiment information of the word from the label of the document (Maas, A. L. et al, 2013). Socher, R. et al. (2011) proposed a recursive autoencoder with a semi-supervised node. The model estimates phrase and document-level embeddings considering compositionality of words. Tang, D. et al. (2014) proposed a neural probabilistic model which learns sentiment-specific word embeddings from predicting label of the document with n-gram word embeddings (Tang, D. et al., 2014).

Emotions are underlying components of sentimental state. It is commonly accepted that emotions are described by a number of dimensions, including valence and arousal (Russell, J. A., 1980; Bradley, M. M. et al., 1992; Kensinger, E. A., 2004; Posner, J. et al., 2005). Unlike sentimental states which expose themselves situations, emotions are more like neurophysiological responses. Thus, Warriner, A. B. et al. (2013) collected human responses over dimensions of emotions of 13,915 English lemma (Warriner, A. B. et al., 2013). Affective responses precede language processing and influence the whole process. In this regard, the affective norms database published by Warriner, A. B. et al. (2013) is a key to estimate word embeddings enriched with emotional information.

Chapter 2 Word Embedding

The word embedding has long been of interest in natural language processing. The underlying idea of the word embedding is to understand semantics of words from statistical patterns of human word usage. The distributional hypothesis is the most widely accepted hypothesis, which postulates that the meaning of word is characterized by the context. Namely, words used in similar contexts may have similar meanings. In this regard, the word embedding stand on the basis of the word-context co-occurrence statistics.

Mikolov, T. et al. (2013) introduced an efficient algorithm to learn word embeddings called the skip-gram with negative sampling, or the Word2Vec (Mikolov, T. et al., 2013). The skip-gram with negative sampling model gained much attraction over the past years in that the model not only provides the state-of-the-art results but also the model captures both attributional and relational similarities. The attributional similarity is a semantic similarity between two words, whereas the relational similarity is a semantic similarity between two pairs of words. For example, Paris and Rome have a high attributional similarity. Meanwhile, Paris and France have a high relational similarity to Rome and Italy because Paris is to France as Rome is to Italy. Remarkably, the skip-gram with negative sampling produces word embeddings with linear structure as below:

$$\mathbf{v}_{\text{Paris}} - \mathbf{v}_{\text{France}} = \mathbf{v}_{\text{Rome}} - \mathbf{v}_{\text{Italy}}$$

This chapter reviews the skip-gram with negative sampling model and the pointwise mutual information. It has recently been suggested by several studies that the skip-gram with negative sampling is in close relationship with the pointwise mutual information model (Levy, O., &

Goldberg, Y. ,2014; Arora, S. et al., 2015). The chapter explores the existing research in the area and identifies the relationship between these models. This chapter also evaluates word embeddings derived by the skip-gram with negative sampling and the pointwise mutual information model.

1 Skip-Gram with Negative Sampling

The skip-gram with negative sampling model is a variant of log-linear models. A sentence or document is a sequence of words, $w_1, w_2, w_3, \ldots, w_n$. The model learns word embeddings which maximize the sum of the probability of the word-context co-occurrence statistics. First, the model defines the conditional probability $p(w_{context}|w_{word})$ using the softmax function:

$$\underset{w}{argmax} \prod_{w \in document} \left(\prod_{w_{context} \in context} p\left(w_{context}|w_{word}\right) \right)$$

$$p(w_{context}|w_{word}) = \frac{exp(v_{context}^T v_{word})}{\sum_{v_{context} \in vocabulary} exp(v_{context}^T v_{word})}$$

Then, the model maximizes the logarithm of the average log probability of co-occurrences:

$$\underset{w}{argmax} \sum_{w_{word} \in document} \left(\sum_{w_{context} \in context} p\left(w_{context}|w_{word}\right) \right)$$

$$= \underset{w}{argmax} \sum_{(w_{word}, w_{context}) \in D} log\left(p(w_{context}|w_{word})\right)$$

The objective function of the skip-gram model is as below:

$$\underset{w}{argmax} \sum_{(w_{word},w_{conte}\ )\in D} \left(log\left(exp(v_{context}^{T} v_{word})\right) - log\left(\sum_{v_{context}\in vocabulary} exp\,(v_{context}^{T} v_{word}) \right) \right)$$

To efficiently estimate word embeddings from the objective function, Mikolov, T. et al. (2013) proposed a sampling algorithm called the negative sampling (Mikolov, T. et al., 2013). Traditional neural probabilistic language models are computationally intensive because of the model complexity. Mnih, A.,& Teh, Y. W. (2012) introduced an efficient and stable algorithm for training neural probabilistic language models called the noise contrastive estimation, which treats a density estimation problem as a binary classification problem (Mnih, A., & Teh, Y. W., 2012). The algorithm maximizes a simple logistic regression accuracy function which differentiates meaning samples from noise. The negative sampling is a simplified version of the noise contrastive estimation in that the algorithm maximizes log probabilities of observed word–context co–occurrence statistics over hypothetically possible word–context co–occurrence statistics. Considering a single word–context co–occurrence, the skip–gram with negative sampling maximizes the objective function as below:

$$log\sigma(v_{context}^{T} v_{word}) + \sum_{i\in(1,2,\ldots,k)} \underset{v_i\sim P_n(v)}{E} \left(log\sigma(-v_i^{T} v_{word})\right)$$

$$\sigma(v_{context}^{T} v_{word}) = p\left(observed|(word, context)\right) = \frac{1}{1+exp(-v_{context}^{T} v_{word})}$$

Thus, the objective function of the skip–gram with negative sampling model is as below:

$$\sum_{v_{word}} \sum_{v_{context}} \left(count_{(word,context)} \left(log\sigma(v_{context}^T v_{word}) + \sum_{i \in (1,2,\dots,k)} \underset{v_i \sim P_n(v)}{E} \left(log\sigma(-v_i^T v_{word}) \right) \right) \right)$$

There are at least three hyperparameters to be specified: the size of the context window, the dimension of word embeddings, and the number of negative samples. Also, the probability distribution of word embeddings needs to be specified. According to Mikolov, T. et al. (2013), the unigram distribution raised to power of 0.75 rather than the original unigram distribution performs better than other distributions. The size of the context window and the dimension of word embeddings determines the richness of word embeddings. The thesis investigated a number of possible hyperparameters.

2 Pointwise Mutual information

The pointwise mutual information measures a mutual dependence between a pair of instances from independent random variables, defined as:

$$PMI(x,y) = log \frac{p(x,y)}{p(x)p(y)} = log \frac{p(x|y)}{p(x)} = log \frac{p(y|x)}{p(y)}$$

The probability distribution of word and context can be estimated from the word-context co-occurrence statistics. Thus, the pointwise mutual information in this case is as blow:

$$PMI(w_{word}, w_{context}) = log \frac{p(w_{word}, w_{context})}{p(w_{word})p(w_{context})}$$

$$p(w_{word}) = \frac{count_{(w_{word})}}{\sum count_{(w_{word},w_{context})}}, p(w_{context}) = \frac{count_{(w_{context})}}{\sum count_{(w_{word},w_{context})}}$$

$$PMI(w_{word}, w_{context}) = log \frac{count_{(w_{word}, w_{context})} \sum count_{(w_{word}, w_{context})}}{count_{(w_{word})} count_{(w_{context})}}$$

The pointwise mutual information is defined by the logarithm. The word–context co–occurrence statistics becomes extremely sparse as the size of vocabulary extends. Thereby, the pointwise mutual information is unstable to measure the word–context co–occurrence statistics. Here as elsewhere in the natural language processing, smoothing processes have been introduced. The most common approach called positive pointwise mutual information is to replace all negative values to zeros (Niwa, Y., & Nitta, Y., 1994) defined as below:

$$PPMI(w_{word}, w_{context})$$
$$= max\left(0, log \frac{count_{(w_{word}, w_{context})} \sum count_{(w_{word}, w_{context})}}{count_{(w_{word})} count_{(w_{context})}}\right)$$

The positive pointwise mutual information is efficient to estimate and, furthermore, produces the sparse matrix. The dimension of the word–context co–occurrence statistics is generally large because it is the number of words times the number of contexts. Thus, the sparsity of the word positive pointwise mutual information is useful in practical terms. In addition, Levy, O., & Goldberg, Y. (2014) proposed the shifted positive pointwise mutual information, which shifts the matrix towards greater sparsity. The shift is an analogy to the negative sampling.

$$SPPMI(w_{word}, w_{context})$$
$$= max\left(0, log \frac{count_{(w_{word}, w_{context})} \sum count_{(w_{word}, w_{context})}}{count_{(w_{word})} count_{(w_{context})}} - logk\right)$$

3 Word Embedding and Distributed Semantics

Levy, O., & Goldberg, Y. (2014) claimed that the skip–gram with negative sampling model learns word embeddings from implicit

factorization of the shifted pointwise mutual information. Namely, under certain conditions, word embeddings derived by the skip-gram with negative sampling are based on the pointwise mutual information. For each word-context co-occurrence, the skip-gram with negative sampling model maximizes the following:

$$count_{(word,context)} log\sigma(v_{context}^T v_{word}) +$$

$$k \frac{count_{(word)} count_{(context)}}{\sum count_{(word,context)}} log\sigma(-v_{context}^T v_{word})$$

The gradient of the above is as follows:

$$count_{(word,context)} \sigma(-v_{context}^T v_{word}) -$$

$$k \frac{count_{(word)} count_{(context)}}{\sum count_{(word,context)}} \sigma(v_{context}^T v_{word})$$

By solving the equation, one may find that:

$$v_{context}^T v_{word} = log \frac{count_{(w_{word},w_{context})} \sum count_{(w_{word},w_{context})}}{count_{(w_{word})} count_{(w_{context})}} - logk$$

$$v_{context}^t v_{word} = PMI(w_{word}, w_{context}) - logk$$

Somewhat surprisingly, the skip-gram with negative sampling seems to factorize the shifted pointwise mutual information. Levy, O., & Goldberg, Y. (2014) also pointed out that, since the dimension of word embeddings generally is much smaller than the size of the vocabulary, the factorization deviates from the optimal case.

Arora, S. et al. (2014) has recently proved the following theorem,

Theorem. There is a constant $Z > 0$ and some $\epsilon = \epsilon(n, d)$ that goes to 0 as $d \to \infty$ such that with high probability over the choice of word vectors, for any two different word vectors w and w'

$$logp(w, w') = \frac{\|v_w + v_{w'}\|_2^2}{2d} - 2logZ \pm \epsilon$$

$$logp(w) = \frac{\|v_w\|_2^2}{2d} - logZ \pm \epsilon$$

Jointly these imply

$$PMI(w, w') = \frac{v_w^T v_{w'}}{d} \pm O(\epsilon)$$

where $p(w|w') \propto exp(v_w^T v_{w'})$. The last equation is equivalent to what Levy, O., & Goldberg, Y. (2014) has previously proved,

$$v_{context}^T v_{word} = PMI(w_{word}, w_{context}) - logk$$

except for the consideration of the dimension of word embeddings. The theorem implies that log-linear models under the distributional hypothesis are based on the pointwise mutual information. In addition, the linear structure of word embeddings seems to lie in the nature of log-linear models based on the pointwise mutual information. Pennington, J. et al. (2014) suggests the model which captures the relational similarities,

$$F(v_i, v_j, v_k) = \frac{P_{ij}}{P_{jk}}$$

where P_{ij} is the probability of $w_i$appearance in the context of w_i. To ensure the linear structure of word embeddings, the model should satisfy the following,

$$v_i^T v_k + b_i + b_k = log(count_{(w_i, w_k)})$$

which leads to the following trivial expression,

$$PMI(w_i, w_k) \approx log(count_{(w_i, w_k)}) - b_i - b_k$$

The skip-gram with negative sampling model received much attention in natural language processing literature. Although recent

studies showed that the skip-gram with negative sampling model is actually an implicit factorization of the conventional word embeddings, the word-context co-occurrence statistics, the model still is considered as the state-of-the-art learning algorithm. The skip-gram with negative sampling is known to be the most computationally efficient and cheap model up to date. This chapter evaluates word embeddings derived by two models -the skip-gram with negative sampling and the shifted positive pointwise mutual information- to verify the claim.

4 Experiment

Word embeddings are derived with different hyperparameters respectively on Wikipedia database. Wikipedia database is used because articles on Wikipedia are written in objective manner over abundant topic. The database contains 3.7 million documents with 1.9 billion tokens. Articles with less than 50 words are ignored and the vocabulary size is limited to 100,000. Conventional text pre-processing methods including stop-word removal, lemmatization and stemming are not used except for tokenization.

Word embeddings are derived by the skip-gram with negative sampling (SGNS) and the shifted positive pointwise mutual information (SPPMI). Hyperparameters are chosen based on the previous experiments. The size of the context window (C) and the number of negative samples (K) are considered. The size of the context window is chosen from(2,5,10)and the number of negative samples is chosen from (0,2,5,20). For the shifted positive pointwise mutual information, the logarithm of the number of negative samples is used instead. Thus, the number of negative samples is chosen from (1,2,5,20).

Google's analogy dataset is used for evaluation of word embeddings derived by models. The dataset includes total 19,258 instances, of which 8,869 are semantic questions and 10675 are syntactic questions. For example, (Athens, Greece, Oslo, Norway) or (brother, sister, grandson, granddaughter) belongs to semantic questions, and (great, greater, tough, tougher) or (walking, walked, swimming, swam) belongs to syntactic questions. The questions are answered by adding and subtracting word embeddings of given words. Only the most similar word embedding is considered as an answer.

	SGNS			SPPMI		
	C = 2	C = 5	C = 10	C = 2	C = 5	C = 10
K = 0 (K = 1)	52.49	53.98	52.02	52.35	46.70	48.36
K = 2	66.27	66.15	63.34	49.85	48.14	47.01
K = 5	68.19	67.07	64.47	43.45	43.26	40.82
K = 20	66.85	67.11	65.13	43.45	31.96	29.09

Table 1 Results of word embeddings with different hyperparameters

Table 1 shows the accuracy of models with each hyperparameter. The skip-gram with negative sampling outperforms the other model, the shifted positive pointwise mutual information. The best performance is achieved with a small context window (C = 2) and large negative samples (K = 5).

For the skip-gram with negative sampling, the performance improves as the number of negative samples grows. The average accuracy of the skip-gram with zero negative samples is 52.83%, whereas the average accuracy with nonzero negative samples is 66.40%. However, for the shifted positive pointwise mutual information, the performance gets worse as the number of negative sample grows. The average

accuracy of the shifted positive pointwise mutual information with zero negative samples is 49.14%, whereas the average accuracy with nonzero negative samples is only 41.89%.

Also, the shifted positive pointwise prefers smaller context windows. The average accuracy of the model with single context window is 47.28%, whereas the average with ten context window is only 41.32%. The skip-gram with negative sampling seems to prefer smaller context windows likewise, but the effect is less obvious. The average accuracy of the skip-gram with single context window is 63.45%, whereas the average with five context window is 61.24%.

Predictive models such as neural network language models or the skip-gram are known to outperform counting models such as the pointwise mutual information. Baroni, M., Dinu, G., & Kruszewski, G. (2014) systematically compared predicting models to counting models and concluded that the former is highly superior to the latter. The result from this chapter support the claim on the one hand.

However, on the other hand, hyperparameter settings seem to play a significant role in the performance of word embeddings. Arora, S. et al. (2014) explains that the models with smaller dimension of word embeddings work better. Since the noise which deteriorates linear structure of word embeddings diminishes as the dimension of word embeddings reduces, low-dimensional word embeddings is necessary (Arora, S. et al., 2014). The dimension of word embeddings derived by the skip-gram model is 500, whereas the dimension of word embeddings derived by the shifted positive pointwise mutual information is 100,000. In this regard, although the skip-gram model outperforms the pointwise mutual information in the experiment, it is hard to claim that the former is superior to the latter.

Chapter 3 Semi-Supervised Autoencoder

The unsupervised learning models learn structure of input data without any explicit target data. The objective of unsupervised learning is understanding data rather than solving tasks. Thus, one should establish what is, and how to measure meaningful understanding. The semi-supervised learning combines unlabeled data with labeled data to improve results on supervised learning tasks. Since labeled data is much expensive and scarce to obtain than unlabeled data, semi-supervised learning models utilize unlabeled data. Thus, semi-supervised learning models simultaneously learn structure of input data and inferred information from supervised tasks.

This chapter introduces a semi-supervised autoencoder which encapsulates input structure and target information within compressed representations. Word embeddings derived by unsupervised learning algorithms may not sufficiently reflect emotional similarities among words, since the co-occurrence statistics of the corpus only represents for contextual semantics. To incorporate emotional similarities within word embeddings, the thesis proposes semi-supervised learning of word representations with a semi-supervised autoencoder.

1 Autoencoder

The autoencoder is an unsupervised shallow feed-forward network model for learning efficient representations of input data. Given input data, the autoencoder learns a nonlinear mapping function between input space and feature space. To learn meaningful nonlinear mapping function, the model minimizes cost for reconstructing original data from

representations. The model of single layer with sigmoid activation function is closely related to principal component analysis. Several variants such as the denoising autoencoder or the contractive autoencoder were proposed (Vincent, P. et al., 2008; Rifai, S. et al., 2011).

For instance, the squared Euclidean distance cost for reconstruction of a simple autoencoder is as below, where f and g denotes nonlinearity:

$$J(W,b;x) = \frac{1}{2}\|x_{reconstructed} - x\|^2$$
$$J(W,b;x) = \frac{1}{2}\left\|g\left(W^{(2)}h + b^{(2)}\right) - x\right\|^2$$
$$J(W,b;x) = \frac{1}{2}\left\|g\left(W^{(2)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(2)}\right) - x\right\|^2$$

Then, efficient representations of input data can be obtained by computing $h = f\left(W^{(1)}x + b^{(1)}\right)$ from learned parameters $W^{(1)}$ and $b^{(1)}$. One may also use the cross-entropy cost for reconstruction as below:

$$J(W,b;x) = \sum x\, log(x_{reconstruct}) + (1-x) log(1 - x_{reconstruct})$$

Stochastic gradient descent is most common approach for training neural network models. Let $\theta = \left(W^{(1)}, W^{(2)}, b^{(1)}, b^{(2)}\right)$, then for each training iteration, parameters are updated as follows:

$$\theta_{i+1}^{(k)} \leftarrow \theta_i^{(k)} - \alpha \frac{\partial J}{\partial \theta_i^{(k)}}$$

The partial derivative above is computed from the standard backward propagation of errors.

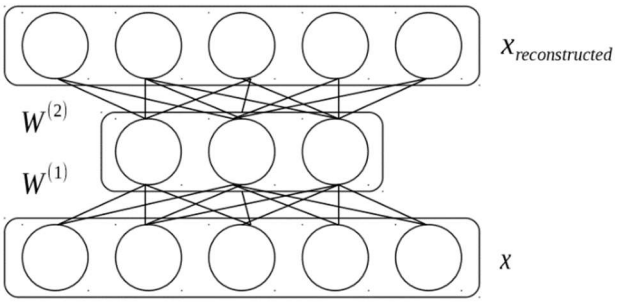

Figure 1 Illustration of shallow autoencoder

2 Semi-Supervised Autoencoder

The semi-supervised autoencoder is an shallow feed-forward network model for learning efficient and task-specific representation. Given input and target data, a semi-supervised autoencoder learns a nonlinear mapping function, which minimizes both cost for reconstructing input data and for solving supervised learning tasks. Thus, the cost function of a simple semi-supervised autoencoder is as below, where f, g and h denotes nonlinearity and L denotes prediction loss:

$$J(W,b;x,y) = J_{reconstruction}(W,b;x) + J_{prediction}(W,b;x,y)$$

$$J(W,b;x,y) = \frac{1}{2}\|x_{reconstructed} - x\|^2 + L\left(h\left(W^{(3)}h + b^{(3)}\right),y\right)$$

$$J(W,b;x,y) = \frac{1}{2}\left\|g\left(W^{(2)}h + b^{(2)}\right) - x\right\|^2 + L\left(h\left(W^{(3)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(3)}\right),y\right)$$

$$J(W,b;x,y) = \frac{1}{2}\left\|g\left(W^{(2)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(2)}\right) - x\right\|^2 + L\left(h\left(W^{(3)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(3)}\right),y\right)$$

One may also use the cross-entropy cost for reconstruction as below:

$$J(W,b;x,y) = \sum\left(xl \quad (x_{reconstruct}) + (1-x)log(x_{reconstruct})\right)$$

$$J(w,b;x,y) + L\left(h\left(W^{(3)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(3)}\right),y\right)$$

It is common to use the cross-entropy or the mean-squared error to measure prediction loss. In classification settings, it is natural to use the cross-entropy to measure prediction loss, since it postulates probability distribution over categories. In regression settings, it is more natural to use the mean-squared error to measure prediction loss. As the model learns a nonlinear mapping function, both reconstruction and prediction loss decrease. Therefore, representations of input data may encapsulate both input structure and target information.

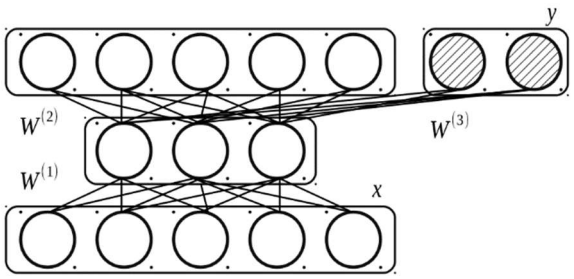

Figure 2 Illustration of semi-supervised autoencoder

3 Regularization and Sparsity

A principle called Occam's razor restrains models from being excessively complex. Regularization is most common approach to prevent excessive complexity, namely, overfitting. It has been shown that regularization evidently impact the outcome of the autoencoder (Vincent, P. et al., 2010). Variants of the autoencoder regularize in their separate ways; the denoising autoencoder stochastically corrupts input data; the contractive autoencoder adds an additional regularization term to the overall cost; sparse autoencoder also adds an additional sparsity term to the overall cost. Weight decay is generally applicable.

Semantic similarities among words are measured by cosine distance between word embeddings. This paper imposes another regularization term on which constrains overall cosine distances between word embeddings on the overall cost function. The regularization term would force word representations to be sparser in semantic space. Combined with semi-supervised learning, word representations may capture more interesting similarities among words. In practice, the regularization term slightly improved results.

Thus, the cost function of a regularized semi-supervised autoencoder is as below, where α, λ and γ denote hyperparameters and $\rho_{ij} = \|h_i\|\|h_j\|$:

$$J(W, b; x, y) = \alpha * J_{reconstruction}(W, b; x) + (1 - \alpha) J_{prediction}(W, b; x, y) + \frac{\lambda}{2}\|\theta\|^2 + \gamma \left\| \frac{hh^T}{\rho} \right\|^2$$

4 Training Semi-Supervised Autoencoder

Stochastic gradient descent approach may also be applied to a semi-supervised autoencoder. Let $\theta = (W^{(1)}, W^{(2)}, b^{(1)}, b^{(2)})$, then for each training iteration, parameters are updated from the gradients. To compute the gradient, standard backward propagation used. However, the cost function is not necessarily continuous and even non-convex, it might not be able to search for global optimal solution. In practice, mini-batch stochastic gradient descent algorithm works well.

Since the regularization term increases computational burden, the semi-supervised autoencoder requires efficient multidimensional computation. Also, since length of input data increases relative to vocabulary size, the model requires a scalable algorithm. Theano provides symbolic differentiation along with generic graphic processing

units computing. General purpose computing on graphic processing units provides massive processing power to data intensive computation. The thesis used Theano 0.7, a python library to optimize mathematical expressions, and other scientific computation libraries such as Numeric Python to construct and train a semi-supervised autoencoder.

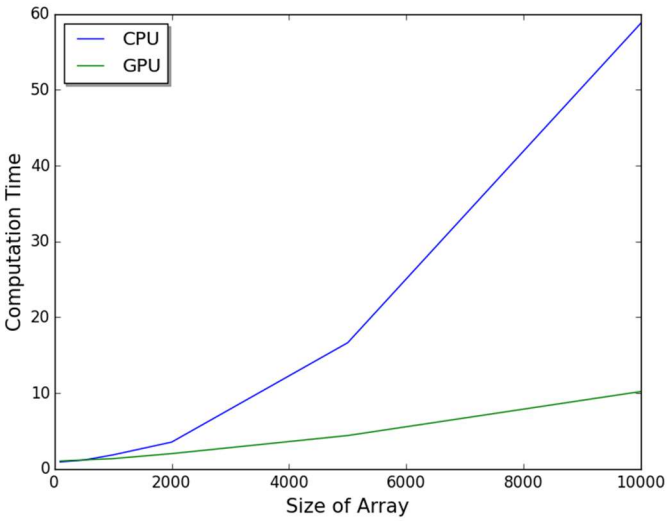

Figure 3 Comparison of computational time on CPU and GPU

Chapter 4 Sentiment Analysis

Sentiment analysis is the process to predict a sentimental state of the source. Mostly, sentimental states are binarized to two classes: objective/subjective or positive/negative. Only recently have sentiment analysis tasks with finely grained sentimental states emerged. A source is generally given as text corpus. In terms of machine learning, sentiment analysis is a supervised learning approach which an input is a representation of a sentence or document and output is its sentimental state. Bag-of-words or term frequency-inverse document frequency are commonly used to represent sentences or documents.

To utilize semantic similarities among words, sentence- or document-level embeddings should be estimated based on word embeddings. Several attempts have been made to explicitly compose word embeddings into phrase-, sentence-, and document-level embeddings. The recursive autoencoder is one of the most successful models up to date. The model greedily construct autoencoders over word embeddings. Thus, the representation assimilates semantic information of word embeddings as it grows. With the autoencoder of binary tree structure, the model learns sentence- or document-level embeddings (Socher R. et al., 2013). The convolutional neural network is another successful model to compose word embeddings. The model utilizes convolution operations and pooling to extract representations of documents (Kim, Y., 2014). However, learning composition model of word embeddings is beyond the scope of the thesis.

This chapter introduces word embeddings suitable to sentiment analysis. To embrace emotional aspects of meaning, the thesis tunes word embeddings derived by the skip-gram with negative sampling model based on emotional dimensions of words. Semi-supervised learning architecture is used, which extracts information from both unlabeled and

labeled data. A semi-supervised autoencoder is used to incorporate emotional similarities among words while preserving most prominent features of semantic similarities among words. A simple procedure is conducted to estimate sentence- or document-level embeddings.

1 Dimensional Model of Emotion

It is widely accepted that emotions could be described as multidimensional features. Emotions are hard to discretely differentiate because they are highly inter-correlated. The most plausible theoretical description up to date is that two independent dimensions compose every emotional states. Moreover, these two independent factors arise from independent neurophysiological systems (Russell, J. A., 1980; Bradley, M. M. et al., 1992; Kensinger, E. A., 2004; Posner, J. et al., 2005).

Mostly, two independent dimensions which compose emotional states are called valence and arousal. Valence measures pleasure or displeasure, and arousal measures activeness. For example, "excited" and "relaxed" have positive valence whereas "nervous" and "bored" have negative valence. "Excited" and "nervous" have heightened arousal whereas "relaxed" and "bored" have diminished arousal.

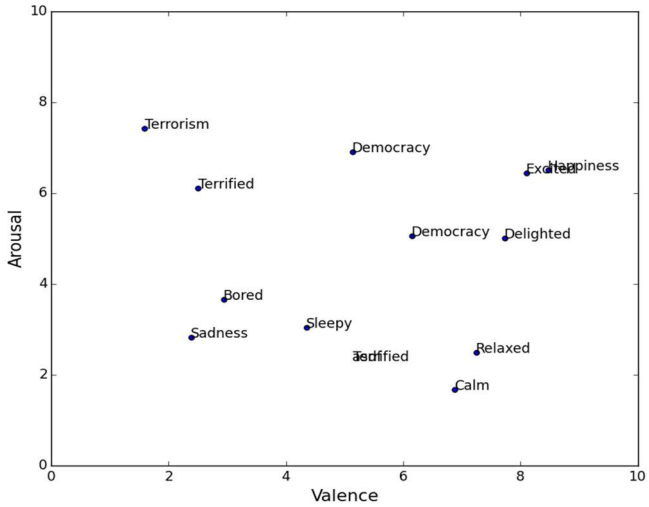

Figure 4 Valence-arousal space of English words

Valence and arousal are immediate neurophysiological responses to stimuli. When neurophysical systems process natural language, these physiological responses may arise and bring about certain changes in affective states. Recent researches suggest while reading emotional words, affective responses arise in advance of language processing. Furthermore, affective responses arise regardless of language processing (Egidi, G., & Nusbaum, H. C., 2012; Citron, F. M. et al., 2014; Kuperman, V. et al., 2014; Recio, G. et al., 2014). One may suppose then, that sentiment analysis within neurophysical system might embrace affective states into language processing.

The distributional hypothesis claims that words with similar contexts tend to have similar meanings. Namely, semantically similar words are distributionally similar. Since word embedding models are based on the hypothesis, their outcomes follow the hypothesis as well. However, are semantically similar words expected to have similar emotional effect? According to Potts, C. (2007), expressive content is independent from descriptive content. For example, a phrase "That bastard Kresge is

famous." means "Kresge is famous" in descriptive sense, whereas it means "Kresge is bastard" in expressive sense. "Positive" and "negative" have similar contexts whereas their valence are completely opposite. Therefore, One may argue that word embeddings based on the distributional hypothesis are deficient in emotional information.

2 Emotional Similarity

Semantic similarities between words are estimated from the word-context co-occurrence statistics. Emotional similarities between words cannot be estimated from the word-context co-occurrence statistics because emotional responses are rather immediate neurophysiological responses than sophisticated articulation. A few studies have measured emotional features directly from human participants. Recently, Warriner, A. B. et al. (2013) published a database with valence, arousal, and dominance of 13,915 English lemmas (Warriner, A. B. et al., 2013). The data were collected from participants who reside in the United States and include average, standard deviation and the number of responses.

	Valence		Arousal	
	Mean	Deviation	Mean	Deviation
Insanity	2.7	1.81	7.79	1.44
Dull	3.4	0.94	1.67	1.03
...	...			
Fantastic	8.36	0.79	6.4	2.6
Soothing	7.05	1.66	1.91	1.31

Table 2 Examples of valence and arousal of words

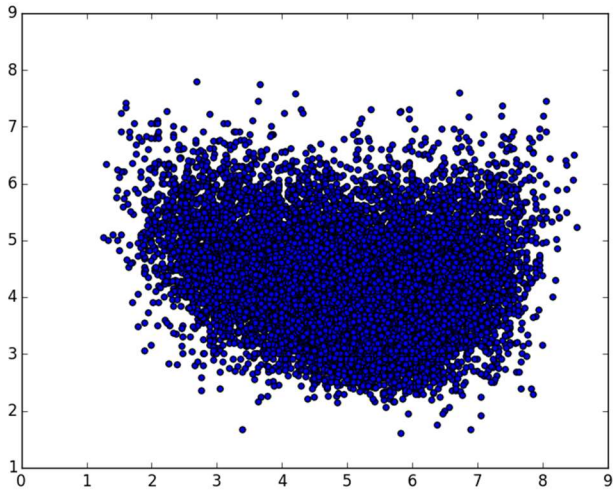

Figure 5 Distribution of valence and arousal of words

2.1 KL-Divergence

To measure emotional similarities among words, the thesis utilizes Kullback_Leibler divergence. The Kullback-Leibler divergence measures the difference between probability distributions. For continuous probability distributions, the Kullback-Leibler divergence is defined as below:

$$D_{KL}(P \parallel Q) = \int p\,(x) log \frac{p(x)}{q(x)} dx$$

The affective norms database provide average and standard deviation of emotional dimensions of each word. Let each dimension of a single word be a random variable with Gaussian distribution. Then, the Kullback-Leibler divergence becomes:

$$D_{KL}(P \parallel Q) = log \frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_2^2} - \frac{1}{2}$$

However, the Kullback–Leibler divergence does not satisfy the symmetry condition of metric. To symmetrize the Kullback–Leiber divergence without losing its theoretical background, the thesis follows the method called the resistor–average distance, which is defined as below (Johnson, D., & Sinanovic, S., 2001):

$$\frac{1}{d(P,Q)} = \frac{1}{D_{KL}(P \parallel Q)} + \frac{1}{D_{KL}(Q \parallel P)}$$

Emotional similarity matrix is built based on the affective norms database and the resistor–average distance metric. Given the similarity matrix of each emotional dimension, the multidimensional scaling algorithm is used to reconstruct latent geometry of the dimension. To make the problem simple, the thesis applied linear support vector machine algorithms to classify each emotional dimension.

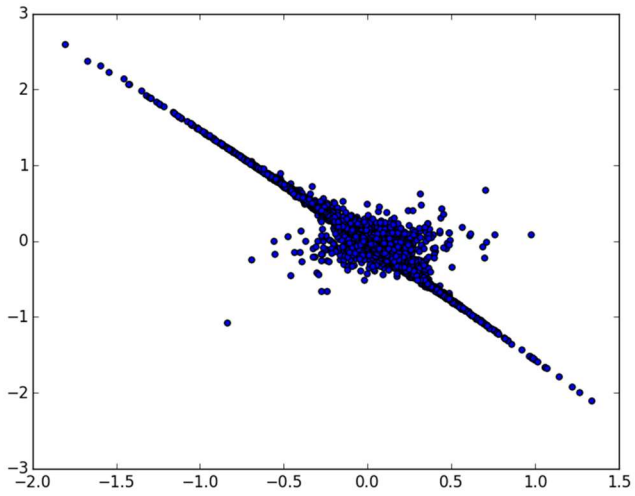

Figure 6 Distribution of valence from multidimensional scaling

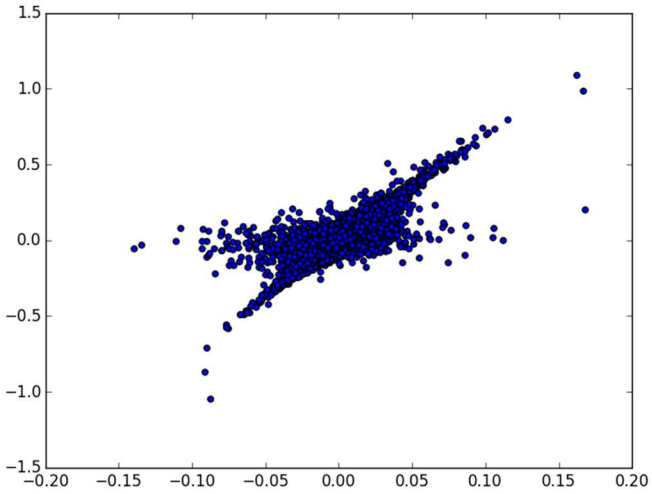

Figure 7 Distribution of arousal from multidimensional scaling

	Valence	**Arousal**
Addiction	negative	low
Compassion	positive	high
Antique	positive	neutral
...	...	
Oblivion	negative	neutral
Supervisor	neutral	neutral
Disrespectful	negative	low

Table 3 Examples of classes of valence and arousal of words

3 Experiment

The experiment involves two steps: (1) Estimating word embeddings which capture both semantic and emotional similarities with affective norms database via semi-supervised autoencoder and (2) analyze polarity and subjectivity of sentiment analysis dataset. Word embeddings

are first estimated using the skip-gram with negative sampling or positive pointwise mutual information model. Then, a representation learning is conducted using a semi-supervised autoencoder with emotional class derived by the affective norms database.

The thesis adopts both qualitative and quantitative assessment. Following Maas, A. L. et al (2011), the thesis compares word similarities of chosen words (Maas, A. L. et al., 2011). Then, by simply averaging word embeddings seen from texts, sentence- or document-level embeddings are estimated. A linear support vector machine with default hyperparameters is used for classifications. Since the compositionality of meaning is not being considered, quantitative assessment would be an indirect indicator of the richness of word embeddings.

3.1 Word Embedding

Word embeddings learned on Wikipedia corpus using the skip-gram with negative sampling or the positive pointwise mutual information are used to train a semi-supervised autoencoder. The best combination of hyperparameters is the skip-gram with negative sampling model with single window (window = 2) and five negative samples(negative = 5). The size of the dictionary is fixed at 100,000 words and the dimension of word embeddings is fixed at 500. Among 13,915 English lemmas which are labeled from the affective norms database, 13,011 lemmas are found in word embeddings. As previous experiments, Google's analogy dataset is used for evaluation of word embeddings.

A semi-supervised autoencoder learns a nonlinear mapping function which incorporates semantic and sentiment information. With 13,011 labeled word embeddings and 86,989 unlabeled word embeddings, the model learns five parameters:$W^{(1)}$,$W^{(2)}$,$W^{(3)}$,$b^{(1)}$,$b^{(1)}$. The model takes

three regularization methods: an explicit regularization term on similarities among word embeddings in the overall cost function, the weight decay, and the tied-weight. The cross-entropy cost between predicted classes of valence and target data is used for the prediction cost, and the squared Euclidean distance cost is used for the reconstruction cost. The overall cost of the model is as below:

$$J(W,b;x,y)=\frac{\alpha}{2}\left\|g\left(\left(W^{(1)}\right)^{(T)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(2)}\right)-x\right\|^2$$

$$J(w,b;x,y)+(1-\alpha)L\left(h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right),y\right)+\frac{\lambda}{2}\|\theta\|^2+\gamma\left\|\frac{hh^T}{\rho}\right\|^2$$

$$L\left(h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right),y\right)=$$

$$\sum y\,log\left(h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right)\right)L\left(h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right),y\right)+$$

$$(1-y)log\left(1-h\left(W^{(3)}f\left(W^{(1)}x+b^{(1)}\right)+b^{(3)}\right)\right)$$

Table 4 shows similarities among induced word embeddings using the model. The thesis compares the method to Maas, A. L. et al. (2011). In order to allow fair comparison between models, word embeddings learned on Large Movie Review Dataset are also used to train a semi-supervised autoencoder. Word embeddings learned on Large Movie Review Dataset seem comparable to or worse the previous model. However, emotionally enhanced word embeddings seem better than the original word embeddings. Word embeddings learned on Wikipedia corpus seem better than the previous model.

Table 5 shows other examples. Emotional words such as "happiness" are semantically similar to their anti-sentimental words such as "sadness". However, emotionally enhanced word embeddings seem to have improved. For instance, the dissimilarity between "positive" and "unfavorable" has relatively decrease, while similarity between "joy" and "happiness" has relatively increased. Also, the dissimilarity between "happiness" and "sadness" has relatively decrease, while similarity between "happiness" and "wholeness" has relatively increased. Although the accuracy of Google's analogy test decreases in comparison with

semantic-only word embeddings, semantic similarities among word embeddings seem undiminished.

	This thesis				Maas, A. L. et al. (2011)		
	Word2Vec	Proposed	Word2Vec (Wikipedia)	Proposed (Wikipedia)	Full	Semantic	LSA
Melancholy	lyrical	evocative	melancholic	somber	poetic	thoughtful	bittersweet
	evocative	hauntingly	dreamy	melancholic	lyrical	warmth	heartbreaking
	deft	lyrical	wistful	dreamy	poetry	layer	happiness
	wry	accompanies	languid	elegiac	profound	gentle	tenderness
	hauntingly	aplomb	sadness	languid	vivid	loneliness	compassionate
Ghastly	threadbare	atrocious	loathsome	terrifying	hideous	predators	embarrassingly
	absurdly	absurdly	terrifying	loathsome	inept	hideous	trite
	ponderous	putrid	horrifying	horrifying	severely	tube	laughably
	drenched	unimpressive	nightmarish	nightmarish	grotesque	baffled	atrocious
	overwrought	threadbare	maddening	unsettling	unsuspecting	smack	appalling
Lackluster	lacklustre	uninspired	underwhelming	mediocre	uninspired	passable	lame
	leaden	leaden	lacklustre	underwhelming	flat	unconvincing	laughable
	threadbare	unimaginative	mediocre	subpar	bland	amateurish	unimaginative
	uninspired	plodding	subpar	lacklustre	forgettable	cliched	uninspired
	unimaginative	pedestrian	unimpressive	unimpressive	mediocre	insipid	awful
Romantic	romance	romance	romance	romance	romance	romance	romance
	screwball	drama	erotic	erotic	screwball	charming	love
	drama	comedy	melodrama	unrequited	grant	delightful	sweet
	comedy	screwball	unrequited	homoerotic	comedies	sweet	beautiful
	charming	comedies	amorous	amorous	comedy	chemistry	relationship

Table 4 Similarity of learned word embeddings

	Word2Vec (Wikipedia)	Proposed (Wikipedia)
Positive	negative	negative
	favorable	favorable
	positively	favourable
	favourable	positively
	unfavorable	review
Happiness	contentment	prosperity
	sadness	contentment
	prosperity	wholeness
	blissful	blissful
	transience	joy
Insensitive	condescending	condescending
	hurtful	sexist
	judgmental	uncouth
	sexist	demeaning
	sensitive	boorish

Table 5 Emotional similarity of learned word embeddings

	Word2Vec (Wikipedia)	Proposed (Wikipedia)
Accuracy	68.19	59.49

Table 6 Evaluation of learned word embeddings on Google's analogy data

3.2 Sentiment Analysis

One may doubt whether emotionally-enriched word embeddings would improve sentiment analysis or not. It seems obvious that better word embeddings would guarantee better results in sentiment analysis

under ideal conditions, since sentences or documents are compositions of words. Several researches have studied compositionality in sentiment as well (Moilanen, K., & Pulman, S., 2007; Choi, Y., & Cardie, C., 2008; Yessenalina, A., & Cardie, C., 2011;). The recursive deep model is the most prominent model in that it works well with complex compositional structures of words with multiple negations (Socher R. et al., 2013).

To evaluate word embeddings obtained from the semi-supervised learning, the thesis conducted sentence- and document-level sentiment polarity classification, subjectivity classification and sentimental state classification. Since valence is the most relevant emotional dimension to sentiment analysis, the thesis used valence dimension to conduct semi-supervised learning. Based on four different sentiment datasets, sentence- and document-level embeddings are estimated with word embeddings learned on Wikipedia corpus and, in addition, word embeddings learned on each sentiment dataset. Sentence- or document-level embeddings are average of their word embeddings. Namely, no other weighting method such as term frequency-inverse document frequency is applied. Since the compositionality of word embeddings is somewhat excessively simple and naive, sentiment classification results may fall short of the state-of-the-art results. Maas, A. L. et al. (2011) provides results over binary-weighted sentence- or document-level embeddings (Maas, A. L. et al., 2011).

3.2.1 Dataset

The sentence polarity dataset 1.0 from Cornell Movie Review Dataset (PL05) consists of 5,331 positive reviews and 5,331 negative reviews with one sentence collected from Rotten Tomatoes. The dataset

is not split into training and testing samples, and results are obtained by the 10-fold cross validation (Pang, B., & Lee, L., 2005).

The Large Movie Review Dataset (IMDB) consists of 50,000 labeled reviews and 50,000 unlabeled reviews with varying length collected from IMDB. The dataset is split into 25,000 labeled training samples and 25,000 labeled testing samples. Labeled samples are polarized, while unlabeled samples are not. Word embeddings can be learned on 25,000 labeled training samples and 50,000 unlabeled samples (Maas, A. L. et al., 2011).

The subjectivity dataset 1.0 from Cornell Movie Review Dataset (SUBJ) consists of 5,000 subjective reviews and 5,000 negative reviews with one sentence collected from Rotten Tomatoes. The dataset is not split into training and testing samples, and results are obtained by the 10-fold cross validation (Pang, B., & Lee, L., 2004).

The Stanford sentiment treebank (SST) is based on the sentence polarity dataset from Cornell Movie Review Dataset, and consists of 11,855 sentences. After having parsed each sentence, 215,514 phrases are labeled as one of five categories by human annotators: negative, somewhat negative, neutral, somewhat positive, positive. These "fine-grained sentiment" dataset is split into 8,544 training samples, 1,101 developing samples, and 2,210 testing samples. Word embeddings can be learned on 8,544 training samples and 1,101 developing samples (Socher R. et al., 2013).

3.2.2 Result

Table 7 shows the performance of the model compared to other models reported from the literature (Pang, B., & Lee, L., 2004; Maas, A.

L. et al., 2011; Socher R. et al., 2011; Socher R. et al., 2013; Kim, Y., 2014; Le, Q. V., & Mikolov, T., 2014).

Model	PL05	IMDB	SUBJ	SST
Word2Vec	67.9	83.6	87.2	32.9
Proposed Word Embeddings	70.2	83.9	88.6	33.9
Word2Vec (Wikipedia)	75.9	83.6	90.4	41.3
Proposed Word Embeddings (Wikipedia)	76.1	83.4	90.7	43.3
Proposed Word Embeddings (Wikipedia) + TF-IDF	78.1	87.8	91.4	39.2
TF-IDF (binary)	76.9	87.5	90.5	38.5
TF-IDF (sublinear)	77.1	87.6	90.6	38.4
Bag of Words (Pang, B., & Lee, L., 2004)	-	-	90.0	-
(Maas, A. L. et al., 2011)	-	88.0	-	-
Recursive Autoencoder (Socher R. et al., 2011)	77.7	-	-	43.2
Recursive Neural Tensor Network (Socher R. et al., 2013)	77.9	-	-	45.7
Convolutional Neural Network (Random) (Kim, Y., 2014)	76.1	-	-	45.0
Paragraph Vector (Le, Q. V., & Mikolov, T., 2014)	-	92.6	-	48.7

Table 7 Sentiment classification results on datasets

Word embeddings enriched with emotional information mostly improve classification results. Except for the large movie review dataset, emotional word embeddings performed better than the original. Also, word embeddings become more sentimental with emotional information mostly when they are learned on the dataset.

For the subjectivity dataset and the Stanford Sentiment Treebank, word embeddings with emotional information outperformed the term frequency–inverse document frequency. For the sentence polarity dataset, word embeddings with emotional information matches recursive and convolutional models. Also, for the Stanford Sentiment Treebank, word embeddings with emotional information matches the recursive

autoencoder. The term frequency–inverse document frequency estimate sentence- or document-level representations by weighting terms, and other models estimate sentence- or document-level representations considering compositionality of semantics whereas the thesis uses simple vector averaging method.

3.3 Discussion

The semantic representation of word as embeddings has proven to be useful. However, there is also a room for development of more emotionally elaborate model. Although the word embedding model well captures semantic similarities in words, it has limits on capturing emotional similarities. The gap between expressive content and descriptive context seems irreducible within the distributed semantics hypothesis. What makes the problem worse is that emotions are neurophysiological responses.

Learning word embeddings enriched with emotional information using the semi-supervised autoencoder thus has independent objectives. Word embeddings should capture emotional similarities in words while preserving semantic similarities. Hyperparameters of the model are chosen to satisfy both objectives. Still, the affective norms include only a small portion of the whole vocabulary, and is only an indicator of actual responses.

Several attempts have also been made to improve word embeddings for sentiment analysis, most of which utilize target data of sentimental sentences or documents (Maas, A. L. et al., 2011; Socher R. et al., 2011; Socher R. et al., 2013; Tang, D. et al., 2013). These methods reflect emotional information in word embeddings rather indirectly.

Nevertheless, these indirect models outperform other semantic-only models for most cases.

Chapter 5 Conclusion

Although the word embedding model has long been of interest, an examination on its representability of emotional responses has not yet been held. Recent papers have attempted to enrich word embeddings with emotional information (Maas, A. L. et al., 2011; Socher R. et al., 2011; Tang, D. et al., 2013). However, these methods are rather sentiment analysis task-specific than general. In order to examine word embeddings on its representability of emotional responses, a bidirectional approach has been made.

Firstly, the thesis addresses that current word embedding models including the skip-gram with negative sampling are based on the distributional hypothesis of semantics. According to the hypothesis, semantics of words are held firmly on their co-occurrence statistics. Chapter 2 explains how current word embedding models are derived by the word-context co-occurrence statistics. Chapter 4 explains how the distributional hypothesis limits word embedding models' representability of emotional responses.

Secondly, the thesis explains that beneath the notable success of predictive models such as the skip-gram with negative sampling lies the conventional pointwise mutual information approach. It has widely been accepted that count-based models only capture the attributional similarities. However, recent studies have shown that the log-linear model on the probability distribution of word occurrence is based on the word-context co-occurrence statistics. Chapter 2 reviews current literature on word embeddings and conduct evaluation on word embedding models.

Lastly, the thesis proposes a semi-supervised learning algorithm to enrich word embeddings with emotional information. Since expressive

and descriptive content are somewhat independent, and since emotions are neurophysiological responses, word embedding models based on the distributional hypothesis of semantics are insufficient. In order to produce more elaborate word embeddings which capture both semantic and emotional similarities in words, a semi-supervised autoencoder is used. Chapter 3 addresses a semi-supervised autoencoder architecture and its regularization. Chapter 4 addresses representability of word embeddings enriched with emotional information and performs evaluation on sentiment analysis.

Throughout chapters, the thesis addresses that the representability of current word embeddings has limits in that the model from which they are derived concerns only contextual information. Since emotional information lies outside contextual information, additional sources including sentimental documents or emotional norms of words themselves are required to enrich word embeddings.

It is evident that more emotional word embedding models would improve results in sentiment analysis, since semantic and emotional meaning of sentences or documents are based on the composition of words. Thus, the proposed word embedding model can be utilized over applications of sentiment analysis. For example, the model could be used in analyzing sentiment polarity of pre-news data.

The proposed word embeddings with emotional information in this thesis better capture emotional similarities among words. Nevertheless, sentiment analysis still requires more sophisticated compositionality of words. Sentences or documents have underlying semantic or syntactic structures, which differentiate the whole meaning. In particular, it is observed that word embeddings with emotional information are less effective with a longer sentence or document. Thus, more efforts should focus on the compositionality issue in future.

Bibliography

Arora, S., Li, Y., Liang, Y., Ma, T., & Risteski, A. (2015). Random Walks on Context Spaces: Towards an Explanation of the Mysteries of Semantic Word Embeddings. arXiv preprint arXiv:1502.03520.

Baroni, M., Dinu, G., &Mnih, A., & Teh, Y. W. (2012). A fast and simple algorithm for training neural probabilistic language models. arXiv preprint arXiv:1206.6426.

Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A neural probabilistic language model. The Journal of Machine Learning Research, 3, 1137-1155.

Bradley, M. M., Greenwald, M. K., Petry, M. C., & Lang, P. J. (1992). Remembering pictures: pleasure and arousal in memory. Journal of experimental psychology: Learning, Memory, and Cognition, 18(2), 379.

Choi, Y., & Cardie, C. (2008, October). Learning with compositional semantics as structural inference for subsentential sentiment analysis. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 793-801). Association for Computational Linguistics.

Citron, F. M., Gray, M. A., Critchley, H. D., Weekes, B. S., & Ferstl, E. C. (2014). Emotional valence and arousal affect reading in an interactive way: neuroimaging evidence for an approach-withdrawal framework. Neuropsychologia, 56, 79-89.

Egidi, G., & Nusbaum, H. C. (2012). Emotional language processing: how mood affects integration processes during discourse comprehension. Brain and language, 122(3), 199-210.

Firth, J.R. (1957). "A synopsis of linguistic theory 1930-1955". Studies in Linguistic Analysis (Oxford: Philological Society): 1－32. Reprinted in F.R. Palmer, ed. (1968). Selected Papers of J.R. Firth 1952-1959. London: Longman.

Johnson, D., & Sinanovic, S. (2001). Symmetrizing the kullback-leibler distance.

Kensinger, E. A. (2004). Remembering emotional experiences: The contribution of valence and arousal. Reviews in the Neurosciences, 15(4), 241–252.

Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Vol. 1, pp. 238–247).

Kuperman, V., Estes, Z., Brysbaert, M., & Warriner, A. B. (2014). Emotion and language: valence and arousal affect word recognition. Journal of Experimental Psychology: General, 143(3), 1065.

Levy, O., & Goldberg, Y. (2014). Dependency based word embeddings. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Vol. 2, pp. 302–308).

Le, Q. V., & Mikolov, T. (2014). Distributed representations of sentences and documents. arXiv preprint arXiv:1405.4053.

Levy, O., & Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In Advances in Neural Information Processing Systems (pp. 2177–2185).

Levy, O., Goldberg, Y., & Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. Transactions of the Association for Computational Linguistics, 3, 211–225.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1 (pp. 142–150). Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

Mnih, A., & Teh, Y. W. (2012). A fast and simple algorithm for training neural probabilistic language models. arXiv preprint arXiv:1206.6426.

Moilanen, K., & Pulman, S. (2007, September). Sentiment composition. In Proceedings of RANLP (Vol. 7, pp. 378-382).

Morin, F., & Bengio, Y. (2005, January). Hierarchical probabilistic neural network language model. In Proceedings of the international workshop on artificial intelligence and statistics (pp. 246-252).

Niwa, Y., & Nitta, Y. (1994, August). Co-occurrence vectors from corpora vs. distance vectors from dictionaries. In Proceedings of the 15th conference on Computational linguistics-Volume 1 (pp. 304-309). Association for Computational Linguistics.

Osterlund, A., Odling, D., & Sahlgren, M. Factorization of Latent Variables in Distributional Semantic Models.

Pang, B., & Lee, L. (2004, July). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the 42nd annual meeting on Association for Computational Linguistics (p. 271). Association for Computational Linguistics.

Pang, B., & Lee, L. (2005, June). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (pp. 115-124). Association for Computational Linguistics.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014), 12, 1532-1543.

Posner, J., Russell, J. A., & Peterson, B. S. (2005). The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. Development and psychopathology, 17(03), 715-734.

Potts, C. (2007). The expressive dimension. Theoretical linguistics, 33(2), 165–198.

Recio, G., Conrad, M., Hansen, L. B., & Jacobs, A. M. (2014). On pleasure and thrill: The interplay between arousal and valence during visual word recognition. Brain and language, 134, 34–43.

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). Contractive auto-encoders: Explicit invariance during feature extraction. In Proceedings of the 28th International Conference on Machine Learning (ICML-11) (pp. 833–840).

Russell, J. A. (1980). A circumplex model of affect. Journal of personality and social psychology, 39(6), 1161.

Schnabel, T., Labutov, I., Mimno, D., & Joachims, T. (2015). Evaluation methods for unsupervised word embeddings. In Proc. of EMNLP.

Socher, R., Pennington, J., Huang, E. H., Ng, A. Y., & Manning, C. D. (2011, July). Semi-supervised recursive autoencoders for predicting sentiment distributions. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 151–161). Association for Computational Linguistics.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013, October). Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the conference on empirical methods in natural language processing (EMNLP) (Vol. 1631, p. 1642).

Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning sentiment-specific word embedding for twitter sentiment classification. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Vol. 1, pp. 1555–1565).

Turian, J., Ratinov, L., & Bengio, Y. (2010, July). Word representations: a simple and general method for semi-supervised learning. In Proceedings of the 48th annual meeting of the association for computational linguistics (pp. 384–394). Association for Computational Linguistics.

Turney, P. D., & Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. Journal of artificial intelligence research, 37(1), 141–188.

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (pp. 1096-1103). ACM.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P. A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. The Journal of Machine Learning Research, 11, 3371-3408.

Warriner, A. B., Kuperman, V., & Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. Behavior research methods, 45(4), 1191-1207.

Yessenalina, A., & Cardie, C. (2011, July). Compositional matrix-space models for sentiment analysis. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 172-182). Association for Computational Linguistics.

Appendix

1 Pseudocode of Semi-Supervised Autoencoder

1 Initialize *Theano shared variables* $\theta = \left(W^{(1)}, W^{(2)}, W^{(3)}, b^{(1)}, b^{(2)}, b^{(3)}\right)$

2 Build training function for semi-supervised autoencoder

2.1 Compute hidden representation matrix $f\left(W^{(1)}x + b^{(1)}\right)$ and reconstruction matrix $g\left(W^{(2)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(2)}\right)$

2.2 Compute prediction matrix $h\left(W^{(3)}f\left(W^{(1)}x + b^{(1)}\right) + b^{(3)}\right)$

2.3 Compute average loss of input matrix with reconstruction loss, prediction loss and regularization term with hyperparameters, α, γ, λ as $\alpha * J_{reconstruction}(W, b; x) + (1 - \alpha)J_{prediction}(W, b; x, y) + \frac{\lambda}{2}\|\theta\|^2 + \gamma \left\|\frac{hh^T}{\rho}\right\|^2$

2.4 Get gradient of average loss and update parameters using *Theano tensor gradient*

3 While stopping criterion is not met,

3.1 For each mini-batch, a *Theano matrix*, run training function

3.2 Compute average loss of current iteration and check if stopping criterion is met

요약 (국문초록)

본 연구는 감정 차원을 고려한 단어 벡터 모델을 제시하고 감성 분석에 적용하였다. 특히, 단어의 감정 정보를 기존의 단어 벡터 모델에 종합하기 위하여 준지도학습을 수행하는 오토인코더를 활용하였다. 감성 분석은 문장이나 문서로부터 작성자의 감성 상태를 추론하는 것이다. 감정 차원은 감성 상태의 요소이다. 즉, 단어 벡터를 활용한 감성 분석이 효과적으로 이루어지기 위하여는 단어 벡터가 감정 차원의 정보를 포함하고 있어야 할 것이다. 그러나 분포 가설에 기반한 기존의 단어 벡터는 감정 정보를 온전히 포함하지 못한다. 단어 벡터를 활용한 감성 분석에서 이를 극복하기 위하여 본 연구는 문장이나 문서가 아닌 단어 자체의 감정 차원을 고려하였다. 준지도학습을 바탕으로 기존의 단어 벡터 모델이 감정 차원의 정보를 담을 수 있도록 모델을 제시하였다. 이를 바탕으로 감성 분석을 수행한 결과 감정 정보가 미비한 단어 벡터에 비하여 향상된 결과를 얻을 수 있었다.

주요어: 단어 벡터, 분포 가설, 감정 차원, 감성 분석, 준지도학습, 오토인코더

학번: 2014-21803