Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Text Classification with Heterogeneous Information Network Kernels

Chenguang Wang[a], **Yangqiu Song**[b], **Haoran Li**[a], **Ming Zhang**[a], **Jiawei Han**[c]
[a]School of EECS, Peking University
[b]Lane Department of Computer Science and Electrical Engineering, West Virginia University
[c]Department of Computer Science, University of Illinois at Urbana-Champaign
{wangchenguang, lihaoran_2012, mzhang_cs}@pku.edu.cn, yangqiu.song@mail.wvu.edu, hanj@illinois.edu

Abstract

Text classification is an important problem with many applications. Traditional approaches represent text as a bag-of-words and build classifiers based on this representation. Rather than words, entity phrases, the relations between the entities, as well as the types of the entities and relations carry much more information to represent the texts. This paper presents a novel text as network classification framework, which introduces 1) a structured and typed heterogeneous information networks (HINs) representation of texts, and 2) a meta-path based approach to link texts. We show that with the new representation and links of texts, the structured and typed information of entities and relations can be incorporated into kernels. Particularly, we develop both simple linear kernel and indefinite kernel based on meta-paths in the HIN representation of texts, where we call them HIN-kernels. Using Freebase, a well-known world knowledge base, to construct HIN for texts, our experiments on two benchmark datasets show that the indefinite HIN-kernel based on weighted meta-paths outperforms the state-of-the-art methods and other HIN-kernels.

Introduction

Text classification has been widely used for many applications, such as news article categorization, social media analysis, and online advertisement. Most text classification techniques represent the text as bag-of-words (BOW) features, and build classifiers based on the features to learn the prediction functions between texts and labels. Although representing the text as BOW is simple and commonly used, the structural nature of semantic relationships among words and entities inside is less explored but informative. Entities can be less ambiguous and have more information about the categories compared to single words. For example, "Nobel Son" and "Nobel Prize" represent different meanings. They provide more useful information for the *Film* and *Award* categories respectively, compared to the words "Nobel," "Son," etc.. If we can recognize the entity names and types (coarse-grained types such as person, location and organization; fine-grained types such as politician, musician, country, and city), these will help better determine the categories of the texts. Moreover, the link information between entities and words are also important. For example, if there are two documents talking about CEOs of Google and Microsoft respectively, and if we build a link between "Larry Page" of sub-type *Entrepreneur* in one text and "Bill Gates" of sub-type *Entrepreneur* in another, then they become in the same category in the sense that they both talk about entrepreneur and connect to the "United States" where Google and Microsoft locate in. Therefore, the structural information in the unstructured text can be utilized to further improve the performance of text classification.

There have been some existing studies on using external knowledge bases such as WordNet (Hotho, Staab, and Stumme 2003), Wikipedia (Gabrilovich and Markovitch 2007), or Probase (Song, Wang, and Wang 2015), to automatically enrich the text representation, thus improve the prediction capabilities of the classifiers. However, these methods ignore the structural information in the knowledge bases, and only treat the knowledge as "flat" features. Others try to build a network from the words in a document (Wang, Do, and Lin 2005; Rousseau, Kiagias, and Vazirgiannis 2015) and compare the text using the graph similarities or graph kernels (Vishwanathan et al. 2010). However, they do not consider the entities and relations inside texts, as well as the types of the entities and relations.

In this paper, we propose to represent a text as a heterogeneous information network (HIN), and classify the texts considering the structured and typed information connecting different documents. The structured and typed information is generated by grounding text to world knowledge bases (Wang et al. 2015a). Then we use meta-paths (Sun and Han 2012) to link different documents. We develop both simple link based augmentation of traditional feature representation of text and a more complicated weighted meta-path based similarity to compare documents. Both methods can be unified as HIN based kernels (HIN-kernels). Since the complicated similarity is not positive semi-definite, to define a legitimate kernel based on the similarity, we propose to use an indefinite kernel SVM to solve the problem. Experimental results on two benchmark datasets show that the indefinite HIN-kernel with weighted meta-path based similarities outperforms the state-of-the-art representations as well as other HIN-kernels. The main contributions of this paper are highlighted as follows:

- We study the problem of converting text classification to a

structured and typed network classification problem to incorporate the rich semantic information from knowledge bases.

- We propose a general classification framework by incorporating typed link information using different types of HIN-kernels.

Link Based Linear Kernel

Our text as network classification framework aims to turn the text classification to network classification, in order to leverage the structured and typed information of the network to improve the predication capabilities of learning models. In this section, we present a straightforward method to simply incorporate the links shown via meta-paths into the features.

HIN-links Based Text Classification

We use the world knowledge specification framework proposed by Wang et al. (Wang et al. 2015a), to ground the text data to world knowledge bases. It comprises two major steps, semantic parsing (generating partial logic forms from texts) and semantic filtering (disambiguating the entities and relations detected from texts). The output of semantic parsing and semantic filtering is the text associated with not only the entities but also the types and relations. We then use an HIN (Sun and Han 2012) to represent the data. *Definition 1* shows the important concepts we use of HIN.

Definition 1 *Given an HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with the entity type mapping ϕ: $\mathcal{V} \rightarrow \mathcal{A}$ and the relation type mapping ψ: $\mathcal{E} \rightarrow \mathcal{R}$, the **network schema** for network $\mathcal{G}$, denoted as $\mathcal{T}_{\mathcal{G}} = (\mathcal{A}, \mathcal{R})$, is a graph with nodes as entity types from $\mathcal{A}$ and edges as relation types from $\mathcal{R}$. A **meta-path** $\mathcal{P}$ is a path defined on the graph of network schema $\mathcal{T}_G = (\mathcal{A}, \mathcal{R})$, and is denoted in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \dots \xrightarrow{R_L} A_{L+1}$, which defines a composite relation $R = R_1 \cdot R_2 \cdot \dots \cdot R_L$ between types A_1 and A_{L+1}, where $\cdot$ denotes relation composition operator, and L is the length of $\mathcal{P}$.*

Then the text network contains multiple entity types: *text* $\mathcal{D}$, *word* $\mathcal{W}$, *named entities* $\{\mathcal{E}^{\mathrm{I}}\}_{I=1}^{T}$, and *relation types* connecting the *entity types*. Our HIN based text classification aims to integrate the structured and typed link information inside the text, as well as the text representation (e.g., BOW) for the classification task.

An intuitive way to formulate the classification problem is to use the link based classification framework (Lu and Getoor 2003). We introduce the features based on document (an entity in HIN) and its relations to other entities. Formally, we represent the relations between documents as $\{(d_i, d_j), \forall d_i, d_j \in \mathcal{D} \wedge d_i \neq d_j\} \in \mathcal{E}$. We denote the entity and relation features of a document $d \in \mathcal{D}$ as $\boldsymbol{x}^{\mathcal{V}}$ and $\boldsymbol{x}^{\mathcal{E}}$, respectively. For entity features $\boldsymbol{x}^{\mathcal{V}}$, we just use bag-of-words with the term frequency (tf) weighting mechanism. We introduce the relation feature construction based on the HIN as follows. Inspired by *count-link* method proposed in (Lu and Getoor 2003), for each meta-path connecting two documents d_i and d_j, we use the number of meta-path instances of the meta-path as one corresponding feature for both d_i and d_j. Different from Lu and Getoor's setting, we do not distinguish in-, out-, and co-links, since HIN is undirected graph.

Now we can incorporate the HIN-links into commonly used models, e.g., Naive Bayes and SVM, and propose NB^{HIN} and SVM^{HIN} respectively. We denote a set of training examples as $\mathcal{X} = \{\boldsymbol{x}_i : i \in \{1, 2, \cdots, n\}\}$, and the corresponding labels as $\boldsymbol{y} = \{y_i \in \mathcal{Y} : i \in 1, 2, \cdots, n\}$.

NB^{HIN}. Traditional Naive Bayes classifier for text classification is formulated as:

$$P(y|\boldsymbol{x}^{\mathcal{V}}) = \frac{P(y)\prod P(x^{\mathcal{V}}|y)}{\sum P(y)\prod P(x^{\mathcal{V}}|y)}. \tag{1}$$

where $x^{\mathcal{V}}$ represent a feature in entity[1] feature vector $\boldsymbol{x}^{\mathcal{V}}$ of document d.

We also incorporate the links into Naive Bayes model:

$$P(y|\boldsymbol{x}^{\mathcal{E}}) = \frac{P(y)\prod P(x^{\mathcal{E}}|y)}{\sum P(y)\prod P(x^{\mathcal{E}}|y)}. \tag{2}$$

Then the combined estimation function is:

$$\hat{y} = arg\max_{y\in\mathcal{Y}} P(y)\prod P(x^{\mathcal{V}}|y)\prod P(x^{\mathcal{E}}|y). \tag{3}$$

SVM^{HIN}. Let matrix $\mathbf{X}$ be the matrix where $\mathbf{X}_{\cdot i} = \boldsymbol{x}_i^T$, matrix $\mathbf{Y} = diag(\boldsymbol{y})$, vector $\mathbf{1}$ be an n-dimensional vector of all ones and C be a positive trade-off parameter. Then, the dual formulation of 1-norm soft margin SVM is given by

$$\begin{aligned} &\max_{\alpha} \mathbf{1}^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{Y}(\mathbf{X}^T\mathbf{X})\mathbf{Y}\boldsymbol{\alpha} \\ &s.t. \quad \boldsymbol{y}^T\boldsymbol{\alpha} = 0, 0 \leq \boldsymbol{\alpha} \leq C\mathbf{1}. \end{aligned} \tag{4}$$

Here we have $\mathbf{X}_{\cdot i}$ equals to $[{\boldsymbol{x}_i}^{\mathcal{V}^T}, {\boldsymbol{x}_i}^{\mathcal{E}^T}]^T$ in SVM^{HIN}. By doing so, SVM^{HIN} provides a simple way to combine the structured information with traditional features. To learn the SVM^{HIN}, we use a convex quadratic programming to solve the dual problem in Eq. (4).

HIN-links Based Kernel

In general, the dominant family of models used in text classification task are linear-like models, which can be represented as a parameter vector $\boldsymbol{\theta}$, corresponding to the features. For an input instance $\boldsymbol{x} \in \mathbb{R}^z$ and an output assignment y, the score of the instance can be expressed as $\boldsymbol{\theta}^T\boldsymbol{x}$. Our framework introduces both entity and relation features, thus $\boldsymbol{x} = [{\boldsymbol{x}^{\mathcal{V}}}^T, {\boldsymbol{x}^{\mathcal{E}}}^T]^T$. The aim of our text as network classification framework is to infer the best label assignment to the output variable,

$$\hat{y} = \arg\max_{y} f(\boldsymbol{\theta}^T\boldsymbol{x}), \tag{5}$$

where $f(\boldsymbol{\theta}^T\boldsymbol{x})$ is a mapping function from features to labels. Especially for discriminative models such as SVM, it is easy to verify that $\mathbf{K} = \mathbf{X}^T\mathbf{X}$ is a linear kernel that can incorporate the structured link information from HIN.

[1]Note that in the HIN for text, entity features include both tf of words and named entities.

Indefinite HIN-Kernel SVM

Although the HIN-links based classification can successfully use the structured and typed information in HIN, it still loses a lot of information, e.g., the importance of different meta-paths. Some of the meta-paths are more important than the others. For example, for the document talking about sports, the following meta-path is more important:
Document→Baseball→Sports→Baseball→Document,
than this one about religion:
Document→Religion→Government→Religion→Document.
Therefore, we should take the meta-paths as a whole into consideration instead of just treating them as links to the documents.

In this section, we introduce a new similarity for text classification. Instead of using meta-paths as links, we introduce a weighted meta-path similarity based kernel for SVM. To incorporate all the interested (or important) meta-paths connecting two documents, we develop the following similarity based on the meta-paths from the text HIN we develop in the previous section.

Definition 2 KnowSim: a knowledge-driven document similarity measure. *Given a collection of symmetric meta-paths, denoted as* $\mathbf{P} = \{\mathcal{P}_m\}_{m=1}^{M'}$*, KnowSim between two documents* d_i *and* d_j *is defined as:*

$$KS(d_i, d_j) = \frac{2 \times \sum_m^{M'} \omega_m |\{p_{i \rightsquigarrow j} \in \mathcal{P}_m\}|}{\sum_m^{M'} \omega_m |\{p_{i \rightsquigarrow i} \in \mathcal{P}_m\}| + \sum_m^{M'} \omega_m |\{p_{j \rightsquigarrow j} \in \mathcal{P}_m\}|}. \tag{6}$$

where $p_{i \rightsquigarrow j} \in \mathcal{P}_m$ *is a path instance between* d_i *and* d_j *following meta-path* $\mathcal{P}_m$*,* $p_{i \rightsquigarrow i} \in \mathcal{P}_m$ *is that between* d_i *and* d_i*, and* $p_{j \rightsquigarrow j} \in \mathcal{P}_m$ *is that between* d_j *and* d_j*. We have* $|\{p_{i \rightsquigarrow j} \in \mathcal{P}_m\}| = \mathbf{M}_{\mathcal{P}_m}(i,j)$*,* $|\{p_{i \rightsquigarrow i} \in \mathcal{P}_m\}| = \mathbf{M}_{\mathcal{P}_m}(i,i)$*, and* $|\{p_{j \rightsquigarrow j} \in \mathcal{P}_m\}| = \mathbf{M}_{\mathcal{P}_m}(j,j)$*.*

Here we use the the definition of commuting matrix for HIN as follow.

Definition 3 Commuting matrix. *Given a network* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ *and its network schema* $\mathcal{T}_\mathcal{G}$*, a commuting matrix* $\mathbf{M}_\mathcal{P}$ *for a meta-path* $\mathcal{P} = (A_1 - A_2 - \ldots - A_{L+1})$ *is defined as* $\mathbf{M}_\mathcal{P} = \mathbf{W}_{A_1A_2}\mathbf{W}_{A_2A_3}\ldots\mathbf{W}_{A_LA_{L+1}}$*, where* $\mathbf{W}_{A_iA_j}$ *is the adjacency matrix between types* A_i *and* A_j*.* $\mathbf{M}_\mathcal{P}(i,j)$ *represents the number of path instances between objects* x_i *and* y_j*, where* $\phi(x_i) = A_1$ *and* $\phi(y_j) = A_{L+1}$*, under meta-path* $\mathcal{P}$*.*

We use a meta-path dependent PageRank-Nibble algorithm to accelerate the computing process of all commuting matrices (Andersen, Chung, and Lang 2006), and use Laplacian score (He, Cai, and Niyogi 2006) to score the importance of different meta-paths based on document-document similarities which are corresponding to the weights ω_m (Wang et al. 2015b).

SVM with Indefinite HIN-Kernel

We use $\mathbf{K}$ to present the kernel matrix. Suppose that $\mathbf{K}$ is positive semi-definite (PSD). Similar to Eq. (4), let matrix $\mathbf{Y} = diag(\boldsymbol{y})$, vector $\mathbf{1}$ be an n-dimensional vector of all ones and C be a positive trade-off parameter. Then the dual formulation of 1-norm soft margin SVM is given by

$$\begin{aligned} &\max_{\alpha} \mathbf{1}^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{Y}\mathbf{K}\mathbf{Y}\boldsymbol{\alpha} \\ &s.t. \quad \boldsymbol{y}^T\boldsymbol{\alpha} = 0, 0 \le \boldsymbol{\alpha} \le C\mathbf{1}. \end{aligned} \tag{7}$$

When $\mathbf{K}$ is PSD, the above problem is a convex quadratic program and solved effectively.

However, the KnowSim matrix $\mathbf{K}$, where $\mathbf{K}_{ij} = KS(d_i, d_j)$, is non-PSD (Berg, Christensen, and Ressel 1984). We use $\mathbf{K}_0$ ($\mathbf{K}_{0ij} = KS(d_i, d_j)$) to present the indefinite kernel matrix generated by KnowSim. Luss and d'Aspremont (Luss and d'Aspremont 2008) proposed a saddle (min-max) approach to simultaneously learn a proxy PSD kernel matrix $\mathbf{K}$ for the indefinite matrix $\mathbf{K}_0$ and the SVM classification as follow:

$$\begin{aligned} &\min_{\mathbf{K}} \max_{\boldsymbol{\alpha}} \mathbf{1}^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{Y}\mathbf{K}\mathbf{Y}\boldsymbol{\alpha} + \rho||\mathbf{K} - \mathbf{K}_0||_F^2 \\ &s.t. \quad \boldsymbol{y}^T\boldsymbol{\alpha} = 0, 0 \le \boldsymbol{\alpha} \le C\mathbf{1}, \mathbf{K} \succeq 0. \end{aligned} \tag{8}$$

Let $\mathcal{Q} = \{\boldsymbol{\alpha} \in \mathbb{R}^n : \boldsymbol{y}^T\boldsymbol{\alpha} = 0, 0 \le \boldsymbol{\alpha} \le C\mathbf{1}\}$, $F(\boldsymbol{\alpha}, \mathbf{K}) = \mathbf{1}^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{Y}\mathbf{K}\mathbf{Y}\boldsymbol{\alpha} + \rho||\mathbf{K} - \mathbf{K}_0||_F^2$. The parameter $\rho > 0$ controls the magnitude of the penalty on the distance between $\mathbf{K}$ and $\mathbf{K}_0$. If any matrix $\mathbf{A}$ is PSD, we write it as $\mathbf{A} \succeq 0$. Based on the min-max theorem (Boyd and Vandenberghe 2004), Eq. (8) equals to $\max_{\boldsymbol{\alpha}\in\mathcal{Q}} \min_{\mathbf{K}\succeq 0} F(\boldsymbol{\alpha}, \mathbf{K})$. Thus the objective function is represented as

$$J(\boldsymbol{\alpha}) = \min_{\mathbf{K}\succeq 0} F(\boldsymbol{\alpha}, \mathbf{K}). \tag{9}$$

We then follow Theorem 1 in (Ying, Campbell, and Girolami 2009) to establish the differentiability of the objective function as

$$\nabla J(\boldsymbol{\alpha}) = \mathbf{1} - \mathbf{Y}(\mathbf{K}_0 + \mathbf{Y}\boldsymbol{\alpha}\boldsymbol{\alpha}^T\mathbf{Y}/(4\rho))_+\mathbf{Y}\boldsymbol{\alpha}. \tag{10}$$

Based on (Luss and d'Aspremont 2008), for fixed $\boldsymbol{\alpha}$, the optimal solution of $\mathbf{K}(\boldsymbol{\alpha}) = \arg\min_{\mathbf{K}\succeq 0} F(\boldsymbol{\alpha}, \mathbf{K})$ is given by $\mathbf{K}(\boldsymbol{\alpha}) = (\mathbf{K}_0 + \mathbf{Y}\boldsymbol{\alpha}\boldsymbol{\alpha}^T\mathbf{Y}/(4\rho))_+$. For any matrix $\mathbf{A}$, the notion $\mathbf{A}_+$ denotes the positive part of $\mathbf{A}$ by setting its negative eigenvalues to zero. Then we follow Theorem 2 in (Ying, Campbell, and Girolami 2009), and show that the gradient of objective function in Eq. (10) is Lipschitz continuous gradient. The Lipschitz constant equals to $\lambda_{max}(\mathbf{K}_0) + \frac{nC^2}{\rho}$. Thus for any $\boldsymbol{\alpha}_0, \boldsymbol{\alpha}_1 \in \mathcal{Q}$, $||\nabla J(\boldsymbol{\alpha}_0) - \nabla J(\boldsymbol{\alpha}_1)|| \le [\lambda_{max}(\mathbf{K}_0) + \frac{nC^2}{\rho}]||\boldsymbol{\alpha}_0 - \boldsymbol{\alpha}_1||$.

This suggests that there is no need to smooth the objective function which will greatly facilitate the gradient family algorithms. We use the Nesterov's efficient smooth optimization method (Nesterov 2005) for solving our convex programming problem. Because this scheme has the optimal convergence rate $\mathcal{O}(1/k^2)$ compared to that of commonly used projected gradient method proposed in (Luss and d'Aspremont 2008) ($\mathcal{O}(1/k)$). k is the number of iterations. We particularly apply the specific first-order smooth optimization scheme introduced in (Nesterov 2005) to our objective function (9). Then we get the smooth optimization algorithm for indefinite SVM.

Experiments

In this section, we show empirically how to incorporate external knowledge into the HIN-kernels.

Datasets

We derive four classification problems from the two benchmark datasets as follow.

20Newsgroups (20NG): In the spirit of (Basu, Bilenko, and Mooney 2004), two datasets are created by selecting three categories from 20NG. *20NG-SIM* consists of three newsgroups on similar topics (comp.graphics, comp.sys.mac.hardware, and comp.os.ms-windows.misc) with significant overlap among the groups. *20NG-DIF* consists of articles in three newsgroups that cover different topics (rec.autos, comp.os.mswindows.misc, and sci.space) with well separated categories.

RCV1: We derive two subsets of RCV1 (Lewis et al. 2004) from the top category GCAT (Government/Social). Similar to 20NG, each of them contains three leaf categories. *GCAT-SIM* consists of articles from three leaf categories of similar topics (GWEA (Weather), GDIS (Disasters), and GENV (Environment and Natural World)) with significant overlap among the categories. We have 1014, 2083 and 499 documents for the three categories respectively. *GCAT-DIF* consists of three leaf categories that cover different topics (GENT (Arts, Culture, and Entertainment), GODD (Human Interest), and GDEF (Defense)) with well separated categories. We have 1062, 1096 and 542 documents for the three categories respectively.

Grounding Text to Freebase

We use Freebase as our world knowledge source in our experiment. Freebase contains over 2 billions relation expressions between 40 millions entities. Moreover, there are 1,500+ entity types and 3,500+ relation types in Freebase (Dong et al. 2014). We convert a logical form generated by our semantic parser into a SPARQL query and execute it on our copy of Freebase using the Virtuoso engine.

After performing semantic parsing and filtering, the numbers of entities in different document datasets with Freebase are summarized in Table 1. The numbers of relations for the datasets are (logical forms parsed by semantic parsing and filtering) 20NG-SIM $(1,834,399)$, 20NG-DIF $(1,587,864)$, GCAT-SIM $(962,084)$ and GCAT-DIF $(1,486,961)$. We keep 20 and 43 top level entity types for 20NG and GCAT, then 325 and $1,682$ symmetric meta-paths are generated based on the meta-path dependent PageRank-Nibble (Andersen, Chung, and Lang 2006) algorithm to compute the commuting matrices.

Classification Results

In this experiment, we analyze the performance of our classification methods.

HIN-links Based Text Classification We first evaluate the effectiveness of the HIN-links based classification by comparing NB^{HIN} and SVM^{HIN} with traditional Naive Bayes and SVM. The feature settings regarding to the NB and SVM are defined as follows.

Table 1: Statistics of entities in different datasets with semantic parsing and filtering using Freebase: #(Document) is the number of all documents; similar for #(Word) (# of words), #(FBEntity) (# of Freebase entities), #(Total) (the total # of entities), and #Types (the total # of entity subtypes).

	#(Document)	#(Word)	#(FBEntity)	#(Total)	#(Types)
20NG-SIM	3,000	22,686	5,549	31,235	1,514
20NG-DIF	3,000	25,910	6,344	35,254	1,601
GCAT-SIM	3,596	22,577	8,118	34,227	1,678
GCAT-DIF	2,700	33,345	12,707	48,752	1,523

- BOW. Traditional bag-of-words model with tf weighting mechanism.
- BOW+ENTITY. BOW integrated with additional features from entities in grounded world knowledge from Freebase. This setting incorporates world knowledge as flat features.
- WE_{Avg}. We use Word2Vec (Mikolov et al. 2013) to train the word embedding based on the 20NG and GCAT respectively. We then use the average word vectors as features to feed them to the classifiers. We set the window size as 5, and the learned word representation is of 400 dimensions using CBOW model and hierarchical softmax for fast training.

NB^{HIN} and SVM^{HIN}, are the HIN-links based text classification algorithms. The entity features and relation features are constructed accordingly. We experiment on the four datasets above. Each data split has three binary classification tasks. For each task, the corresponding data is randomly divided into 80% training and 20% testing data. We apply 5-fold cross validation on the training set to determine the optimal hyperparameter C for SVM and SVM^{HIN}. Then all the classification models are trained based on the full training set (SVM based methods with C), and tested on the test set. We employ classification accuracy as the evaluation measure.

In Table 2, we show the performance of all the classification models with different settings on all the four datasets. We report the average classification accuracy of the three binary classification results in each dataset of the four. Notice that here we focus on NB^{HIN} vs. NB, and SVM^{HIN} vs. SVM, to directly test our general classification framework. From the results, we find that NB^{HIN} and SVM^{HIN} are competitive with NB and SVM with WE_{Avg}, and outperform NB and SVM with other settings. This means that by using link information in HIN extracted from the world knowledge (specifically refer to relation features), we can improve the text classification, especially comparing with the ones only using entity as additional features (BOW+ENTITY). The results are even competitive with the state-of-the-art word embedding approach trained based on 20NG and GCAT data respectively. Also, we find the improvement of SVM^{HIN} and NB^{HIN} on GCAT-SIM and GCAT-DIF are more than that on 20NG-SIM and 20NG-DIF. As Table 1 shows, GCAT-SIM and GCAT-DIF have more entities and associated types grounded from Freebase.

Table 2: Performance of different classification algorithms on 20NG-SIM, 20NG-DIF, GCAT-SIM and GCAT-DIF datasets. BOW, ENTITY represent bag-of-words feature and the entities generated by the world knowledge specification framework based on Freebase, respectively. NB^{HIN} and SVM^{HIN} are the variant of traditional Naive Bayes and SVM under our HIN-links based text classification framework. SVM^{HIN}+KnowSim represents the 1-norm soft margin SVM defined in Eq. (7) with indefinite KnowSim based kernel. IndefSVM^{HIN}+KnowSim represents the SVM with a proxy PSD kernel for the indefinite KnowSim matrix as shown in Eq. (8). DWD and DWD+MP represent the kernel matrix that is constructed based on KnowSim with a single $\mathcal{P}_{DWD}$ meta-path and all kinds of meta-paths generated based on the text HIN, respectively.

	NB			NB^{HIN}	SVM			SVM^{HIN}	SVM^{HIN}+KnowSim		IndefSVM^{HIN}+KnowSim	
Settings / Datasets	BOW	BOW +ENTITY	$\mathrm{WE}_{\mathrm{Avg}}$		BOW	BOW +ENTITY	$\mathrm{WE}_{\mathrm{Avg}}$		DWD	DWD +MP	DWD	DWD +MP
20NG-SIM	86.95%	89.76%	90.82%	90.83%	90.81%	91.11%	91.67%	91.60%	92.32%	92.68%	92.65%	**93.38%**
20NG-DIF	96.37%	96.94%	97.16%	97.37%	96.66%	96.90%	98.27%	97.20%	97.83%	98.01%	98.13%	**98.45%**
GCAT-SIM	88.49%	89.12%	91.87%	90.02%	94.15%	94.29%	96.81%	94.82%	95.29%	96.04%	95.63%	**98.10%**
GCAT-DIF	86.73%	88.08%	91.56%	88.65%	88.98%	90.18%	90.64%	91.19%	90.70%	91.88%	91.63%	**93.51%**

Indefinite HIN-Kernel Based SVM We next test the performance of the KnowSim kernel methods by comparing them with the other classification methods under the framework (i.e., SVM^{HIN} and NB^{HIN}). We derive two SVM with KnowSim kernel methods.

- One is denoted as "SVM^{HIN}+KnowSim" using the 1-norm soft margin SVM defined in Eq. (7) by setting the negative eigenvalues of the KnowSim matrix being zeros.
- The other is denoted as "IndefSVM^{HIN}+KnowSim." It learns a proxy PSD kernel for the indefinite KnowSim matrix as shown in Eq. (8). The parameters C and ρ for indefinite SVM are tuned based on the 5-fold cross validation and the Nesterov's efficient smooth optimization method (Nesterov 2005) is terminated if the value of the object function changes less than 10^{-6} following (Ying, Campbell, and Girolami 2009).

We also explore what should be the best way to use KnowSim (Definition 2) as kernel matrix for the text classification. We particularly explore two different KnowSim computation settings.

- DWD. Kernel matrix is constructed based on KnowSim using only meta-path instances belonging to $\mathcal{P}_{DWD}$ = *Document* $\xrightarrow{\text{contain}}$ *Word* $\xrightarrow{\text{contain}^{-1}}$ *Document* meta-path (i.e., $M' = 1$ in Eq. (6)). This setting aims to test whether kernel methods themselves are still effective, even with the simplest structural information in the HIN, when we have almost the same amount of information compared to bag-of-words features.
- DWD+MP. Kernel matrix is constructed based on KnowSim using meta-path instances belong to all kinds of meta-paths in the text HIN. This setting aims to test that how good can kernel based SVM leverage the specified world knowledge for text classification.

As shown in Table 2, IndefSVM^{HIN}+KnowSim with DWD+MP consistently performs the best on all datasets. With t-test, we find the improvements are significant at 0.05 significance level. Especially, we can draw the following observations and conclusions.

(1) The performance of SVM^{HIN}+KnowSim with DWD is better than SVM with BOW. This is because in Eq. (6), there is a normalization term for the values in commuting matrix which is $\mathbf{W}_{DW}\mathbf{W}_{DW}^T$ and $\mathbf{W}_{DW}$ is the matrix between documents and words. The normalization terms in Eq. (6), $|\{p_{i\rightsquigarrow i} \in \mathcal{P}_m\}|$ and $|\{p_{j\rightsquigarrow j} \in \mathcal{P}_m\}|$, correspond to the degree for the document node in the information network. Compare to Eq. (4) where no normalization is performed, it shows normalization indeed helps to formulate a better similarity. Note that, cosine similarity is another widely used approach for normalizing document length, but it cannot be applied to information network.

(2) Both kernel methods with DWD+MP outperform NB^{HIN} and SVM^{HIN}. The reason is by considering the meta-path information as a whole, and use some weighting mechanisms to select the more important meta-paths do help encode more informative information for text classification.

(3) In both SVM^{HIN}+KnowSim and IndefSVM^{HIN}+KnowSim, DWD+MP is better than DWD. This indicates that meta-paths in HIN with knowledge (e.g., entities and relations), capture more similarity information for documents than just the links between documents via words.

(4) IndefSVM^{HIN}+KnowSim always works better than SVM^{HIN}+KnowSim. The reason can be denoising the non-PSD kernel by removing the negative eigenvalues can lose some useful information about the similarity.

(5) IndefSVM^{HIN}+KnowSim with DWD+MP consistently outperforms classifiers with $\mathrm{WE}_{\mathrm{Avg}}$. This means that KnowSim kernel with world knowledge carries more semantics about the similarities between texts compared to that the implicit embedding implies.

Moreover, we test the effectiveness of world knowledge for improving classification performance. We choose one dataset (GCAT-SIM) and vary the size of training data (20%, 40%, 60%, 80%, 100%) for each algorithm. The results are summarized in Fig. 1. It seems that with less training data, the external knowledge can help more on improving the classification accuracy.

Besides using KnowSim, we also use the knowledge-based graph semantic similarity (GSim) proposed in (Schuhmacher and Ponzetto 2014) to measure the document similarity. We use the indefinite SVM to encode the GSim similarity in the kernel. Due to the high time complexity of GSim, we implement GSim based on the text HIN, which is a subgraph of Freebase. We then achieve the accuracy of 50.44% on 20NG-SIM dataset. This indicates that GSim

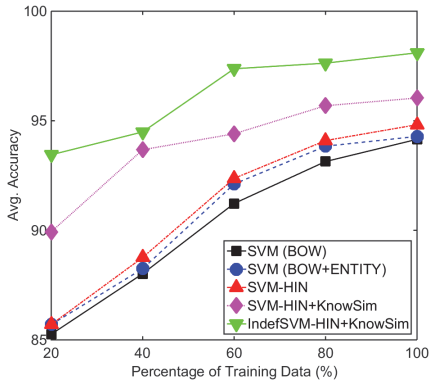

Figure 1: Effects of the size of training data on GCAT-SIM. SVM-HIN+KnowSim and IndefSVM-HIN+KnowSim denote SVM^{HIN}+KnowSim and IndefSVM^{HIN}+KnowSim with DWD+MP.

may be not very suitable to be used in indefinite SVM in large-scale datasets.

Related Work

Some in-depth reviews of the early studies in text classification can be found in (Sebastiani 2002; Aggarwal and Zhai 2012). Several milestone studies include using support vector machine (SVM) (Joachims 1998) and Naive Bayes (McCallum, Nigam, and others 1998) with BOW features for text classification. One direction of recent work is on leveraging structural information for better classification. Link based classification (Lu and Getoor 2003; Kong et al. 2012) use relationship between text (e.g., number of links) as additional features to original BOWs feature. Graph-of-words (Wang, Do, and Lin 2005; Hassan, Mihalcea, and Banea 2007; Rousseau, Kiagias, and Vazirgiannis 2015) representation is recently proposed and show better results compared to BOW. However, these approaches focus on data statistics without considering the semantics of the link. For example, in graph-of-words, if two words occur near in one document, the words will be linked. Our method aims to leverage the semantics of links for classification, i.e., the entities and links are with types.

Another direction is on enriching the text representation with semantics from world knowledge. Linguistic knowledge bases such as WordNet (Hotho, Staab, and Stumme 2003) or general purpose knowledge bases such as Open Directory (Gabrilovich and Markovitch 2005), Wikipedia (Gabrilovich and Markovitch 2007; Hu et al. 2008; 2009), or knowledge extracted from open domain data such as Web pages (Wang et al. 2013; 2015c) and Probase (Song et al. 2011; Song, Wang, and Wang 2015), have been used to extend the features of documents to improve text categorization. Yet we do not use such knowledge as flat features, and instead encode link (meta-path) based similarities among documents in kernels, in the networks generated from knowledge base, Freebase.

Building semantic kernel using world knowledge for text categorization has been proposed in (Siolas and Buc 2000; Wang et al. 2007; Wang and Domeniconi 2008). The semantic kernel is constructed in a supervised way and only considers the direct (one-hop) links. However, we do not need an extra proximity matrix to construct the kernel. Besides, KnowSim kernel takes multi-hop links (i.e., meta-paths) via a totally unsupervised way. Besides KnowSim, knowledge-based graph semantic similarity (GSim) is proposed in (Schuhmacher and Ponzetto 2014) to measure the document similarity based on DBpedia. However, the time complexity of computing GSim is high. So it is not feasible on our large-scale datasets (in original paper they experiment on a document set with 50 documents). KnowSim however can be computed in nearly linear time. Recently, Kim et al. (Kim, Rousseau, and Vazirgiannis 2015) introduce sentence kernel generated by word distances from a given word vector space based on word embedding. Yet our proposed KnowSim based kernel is built on the HIN constructed by explicit world knowledge from the knowledge base. It is also interesting to integrate the word embedding results and explicit world knowledge information (Song and Roth 2015). In this way, the KnowSim can be more robust when facing the scarcity of knowledge for some specific domains.

Conclusion

In this paper, we study the problem of converting text classification to structured heterogeneous information network classification. We first propose an HIN-links based text classification framework, and show it is equivalent to introducing a linear kernel combining entities and relations in HIN. We further develop an SVM classifier using indefinite kernel matrices based on KnowSim, a knowledge driven text similarity measure that could naturally encode the structural information in the text HIN. Improved classification results have been shown on various benchmark datasets.

Acknowledgments

Chenguang Wang gratefully acknowledges the support by the National Natural Science Foundation of China (NSFC Grant No. 61472006 and 61272343), the National Basic Research Program (973 Program No. 2014CB340405), and Doctoral Fund of Ministry of Education of China (MOEC RFDP Grant No. 20130001110032). The research is also partially supported by the Army Research Laboratory (ARL) under agreement W911NF-09-2-0053, and by DARPA under agreement number FA8750-13-2-0008. Research is also partially sponsored by National Science Foundation IIS-1017362, IIS-1320617, and IIS-1354329, HDTRA1-10-1-0120, and grant 1U54GM114838 awarded by NIGMS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative (www.bd2k.nih.gov), and MIAS, a DHS-IDS Center for Multimodal Information Access and Synthesis at UIUC.

References

Aggarwal, C. C., and Zhai, C. 2012. A survey of text classification algorithms. In *Mining text data*. Springer. 163–222.

Andersen, R.; Chung, F.; and Lang, K. 2006. Local graph partitioning using pagerank vectors. In *FOCS*, 475–486.

Basu, S.; Bilenko, M.; and Mooney, R. J. 2004. A probabilistic framework for semi-supervised clustering. In *KDD*, 59–68.

Berg, C.; Christensen, J. P. R.; and Ressel, P. 1984. Harmonic analysis on semigroups. *Graduate Texts in Mathematics*.

Boyd, S., and Vandenberghe, L. 2004. *Convex optimization*. Cambridge university press.

Dong, X.; Gabrilovich, E.; Heitz, G.; Horn, W.; Lao, N.; Murphy, K.; Strohmann, T.; Sun, S.; and Zhang, W. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *KDD*, 601–610.

Gabrilovich, E., and Markovitch, S. 2005. Feature generation for text categorization using world knowledge. In *IJCAI*, 1048–1053.

Gabrilovich, E., and Markovitch, S. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, 1606–1611.

Hassan, S.; Mihalcea, R.; and Banea, C. 2007. Random walk term weighting for improved text classification. In *ICSC*, 242C–249.

He, X.; Cai, D.; and Niyogi, P. 2006. Laplacian score for feature selection. In *NIPS*. 507–514.

Hotho, A.; Staab, S.; and Stumme, G. 2003. Ontologies improve text document clustering. In *ICDM*, 541–544.

Hu, J.; Fang, L.; Cao, Y.; Zeng, H.-J.; Li, H.; Yang, Q.; and Chen, Z. 2008. Enhancing text clustering by leveraging Wikipedia semantics. In *SIGIR*, 179–186.

Hu, X.; Zhang, X.; Lu, C.; Park, E. K.; and Zhou, X. 2009. Exploiting wikipedia as external knowledge for document clustering. In *KDD*, 389–396.

Joachims, T. 1998. *Text categorization with support vector machines: Learning with many relevant features*. Springer.

Kim, J.; Rousseau, F.; and Vazirgiannis, M. 2015. Convolutional sentence kernel from word embeddings for short text categorization. In *EMNLP*, 775–780.

Kong, X.; Yu, P. S.; Ding, Y.; and Wild, D. J. 2012. Meta path-based collective classification in heterogeneous information networks. In *CIKM*, 1567–1571.

Lewis, D. D.; Yang, Y.; Rose, T. G.; and Li, F. 2004. RCV1: A new benchmark collection for text categorization research. *JMLR* 5:361–397.

Lu, Q., and Getoor, L. 2003. Link-based classification. In *ICML*, 496–503.

Luss, R., and d'Aspremont, A. 2008. Support vector machine classification with indefinite kernels. In *NIPS*, 953–960.

McCallum, A.; Nigam, K.; et al. 1998. A comparison of event models for naive bayes text classification. In *AAAI Workshop*, volume 752, 41–48.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, 3111–3119.

Nesterov, Y. 2005. Smooth minimization of non-smooth functions. *Mathematical programming* 103(1):127–152.

Rousseau, F.; Kiagias, E.; and Vazirgiannis, M. 2015. Text categorization as a graph classification problem. In *ACL*, 1702–1712.

Schuhmacher, M., and Ponzetto, S. P. 2014. Knowledge-based graph document modeling. In *WSDM*, 543–552.

Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM CSUR* 34(1):1–47.

Siolas, G., and Buc, F. D. A. 2000. Support vector machines based on a semantic kernel for text categorization. In *IJCNN*, 205–209.

Song, Y., and Roth, D. 2015. Unsupervised sparse vector densification for short text similarity. In *NAACL HLT*, 1275–1280.

Song, Y.; Wang, H.; Wang, Z.; Li, H.; and Chen, W. 2011. Short text conceptualization using a probabilistic knowledgebase. In *IJCAI*, 2330–2336.

Song, Y.; Wang, S.; and Wang, H. 2015. Open domain short text conceptualization: A generative + descriptive modeling approach. In *IJCAI*, 3820–3826.

Sun, Y., and Han, J. 2012. Mining heterogeneous information networks: principles and methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery* 3(2):1–159.

Vishwanathan, S. V. N.; Schraudolph, N. N.; Kondor, R.; and Borgwardt, K. M. 2010. Graph kernels. *Journal of Machine Learning Research* 11:1201–1242.

Wang, P., and Domeniconi, C. 2008. Building semantic kernels for text classification using wikipedia. In *KDD*, 713–721.

Wang, P.; Hu, J.; Zeng, H.-J.; Chen, L.; and Chen, Z. 2007. Improving text classification by using encyclopedia knowledge. In *ICDM*, 332–341.

Wang, C.; Duan, N.; Zhou, M.; and Zhang, M. 2013. Paraphrasing adaptation for web search ranking. In *ACL*, 41–46.

Wang, C.; Song, Y.; El-Kishky, A.; Roth, D.; Zhang, M.; and Han, J. 2015a. Incorporating world knowledge to document clustering via heterogeneous information networks. In *KDD*, 1215–1224.

Wang, C.; Song, Y.; Li, H.; Zhang, M.; and Han, J. 2015b. Knowsim: A document similarity measure on structured heterogeneous information networks. In *ICDM*, 506–513.

Wang, C.; Song, Y.; Roth, D.; Wang, C.; Han, J.; Ji, H.; and Zhang, M. 2015c. Constrained information-theoretic tripartite graph clustering to identify semantically similar relations. In *IJCAI*, 3882–3889.

Wang, W.; Do, D. B.; and Lin, X. 2005. Term graph model for text classification. In *ADMA*, 19–30.

Ying, Y.; Campbell, C.; and Girolami, M. 2009. Analysis of svm with indefinite kernels. In *NIPS*, 2205–2213.