Radboud University

INSTITUTE FOR COMPUTING AND INFORMATION SCIENCES

Entropy-based Adaptation for URL Classification

Master's Thesis Report

Author:

Al Aminuddin

s4604822

Supervisor:

Prof.dr.ir Arjen P. de Vries

Second Assesor:

Dr. ir. E. Herder

Nijmegen, 03 July 2017

Abstract

To enhance user experience on focused browsing activities, as a mega website, university web sites need to provide topical related URLs. The purpose of this thesis was to see how domain adaptation method could be employed to classify URLs using labeled out-of-domain URLs as the training data. The classification was meant to support link prediction approach, which previously suggested "unlabeled "related URLs. The URL classification with ignoring the "difference"of both training and test data could possibly lead to poor performances. This thesis choosed data selection as a domain adaptation method to lower error rates of classification performances by minimazing disparity between training data and test data. To select the best data, entropy-based selection was the simple way of data selection to measure the closeness of data. This work used and compared four entropy-based selection methods: *cross entropy*, *entropy difference*, *cross entropy difference* and *average entropy gain*. This thesis results demonstrate that the four entropy-based methods need to be evaluated regarding their measurement issues on data sparseness. The finding indicates that prioritizing to the low entropy score data only as the closest data was problematic in URL classification. The miscalculated consideration could lead to data miss-selection. This thesis also revealed that *token* and *cross entropy* were the best pair of feature and method respectively to increase the classification performances.

Acknowledgements

Many thanks to my supervisor, Arjen P. de Vries, I really express my gratitude and appreciation for his patient, guide and generousity towards my limitations during finishing this thesis. I would not achieve this accomplishment without his useful advice and supports. Thanks also go to my teachers in Radboud University Nijmegen, especially Theo and Suzan who inspired my thesis by their teachings in the class. My friend Ricko, I thank for his help in solving many my thesis code problems. I also thank to Om Toni, Mbak Eva, Fitria, Fara, Jamila and my friends who supported me during my difficult time living in Nijmegen. My wife, my daughter and my big family, I thank them for their patient and motivation. They all are my energies in my academic life adventures. Finally I would like to thank LPDP (Indonesia Endowment Fund for Education) as my scholarship funding and UIN SUSKA Riau as my working institution. I could not be in this wonderful university without their supports.

Contents

List of Tables

List of Figures

List of Abbreviations and Symbols

URL	Uniform Resource Locator
S	Source
T	Target
F	Function
D	Domain
D_s	Domain Source
D_t	Domain Target
T_s	Task Source
T_t	Task Target
RDM	Random Data Selection
CE	Cross Entropy
ED	Entropy Difference
CED	Cross Entropy Difference
AEG	Average Entropy Gain
POS	Part-of-speech
NB	Naive Bayes classifier
SVM	Support Vector Machine classifier
tk	token feature
$ng2$	2-grams feature
$ng3$	3-grams feature
$ng4$	4-grams feature
X	Observed Variable
Y	Label/Class of x_i
$P(x\|y)$	Conditional Probability of x given y
$P(x, y)$	Joint Probability of x and y
$I()$	Information
$H(X)$	Entropy of X
$\sum$	Summation
s	A sentence
u	An URL
$*$	Multiplication
P_o	Actual (measured) aggreement
P_e	Expected (chance) aggrement
P	Probability
$H(s, p)$	Entropy of s w.r.t p
$\mathcal{D}$	Training data $\mathcal{D} = \{(\vec{x}_i, y_i) \| i = 1 : N\}$
$\mathcal{C}$	Number of classes

Chapter 1

Introduction

1.1 Background

As a mega-website which contains a huge amount of web pages, university web sites provide a lot of information for their users. Finding relevant information on this kind of web sites sometimes takes a lot of effort. To minimize this effort and optimize the user's experience on the site, the web sites normally offer links to pages related to the content of the pages where their users visit. The related pages are normally included in the visited page in the form of related or suggested links. By inserting these links (URLs) on the pages, the users are expected to easily explore relevant information on the web sites.

Focused browsing refers to a personalized web experience, where links are generated to macth hte user interest. Ideally the users would only view links to "on topic"pages that will be interest. It is similar to the concept of focused crawling [5] which is to selectively seek out web pages relevant to pre-defined topics. For instance, the users only want to see web pages related to *student* topic or *research* topic only. In such case, related URLs would be attractive and useful to those users if they are annotated or labeled with their category or topic.

Related URLs may be extracted from link prediction processes [6]. Link prediction is commonly described as the work of predicting relationships in a network. The network could be a collection of connected data. In an existing link prediction approach [2] which used Weblog access as their input data collection, the URLs as a part of the data were completely unlabeled. Those URLs were obtained from a web server for every single request it obtained from its users. Since the URLs were unlabeled, the related URLs proposed from the prediction approach do not include sufficient information (i.e. URL category) to the users who want to do focused browsing.

To provide the sufficient information for focused browsing users, URL labeling becomes an essential task to support link prediction. Labeling or categorizing is one of the tasks in machine learning called classification[7]. Classification in machine learning uses at least two part datasets: a training set and a test set. Normally they are derived from splitting a whole dataset into those two parts. The training set is assumed already having class/topic/category for each of its instance data and thereby from that training set, a classifier can learn to predict the testing dataset. Both the training and test sets are assumed generated from the same source of data in order to produce good performance accuracy.

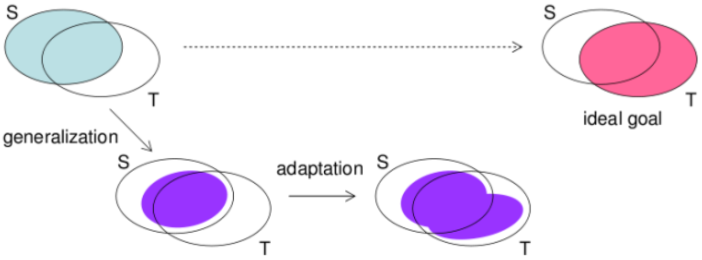


Figure 1.1: Abstract representation of domain adaptation. Here, S and T represents source and target domain respectively [1].

In case no labeled set of URLs exist that can be used as a training dataset, two solutions can be adopted to solve this issue. First, the dataset can be annotated manually, but this is a costly effort in terms of resources. The second solution is to use labeled out-of-domain data as training data to classify the whole URL dataset. However, with training data and test data coming from a different domain, learning a classifier with training data and classifying the test data may lead to degraded classification results [8].

Domain adaptation [9] is a method to tackle the issue of discrepancy between training data and testing data. Figure 1.1 shows an abstract representation of domain adaptation. The idea of domain adaptation is closely related to transfer learning. Given a source domain D_s and its learning task T_s, a target domain D_t and its learning task T_t, transfer learning aims to help improve the learning of the target predictive function F_t in D_t using the knowledge from D_s and D_t, where $D_s \neq D_t$ and $T_s \neq T_t$. Domain adaptation is a part of *transductive transfer learning* methods where T_s and T_t are the same, while D_s and D_t are different [10]. By designing algorithms to transfer knowledge from labeled data in D_s to D_t, we may succeed in T_t, for example, annotating the new data in D_t, while keeping high performance. As an example of domain adaptation, the approach to NE detection presented in [11] has trained a classifier on the ACE data to be evaluated in the CoNLL corpus. The approach demonstrated that allowing information to be shared between domains could significantly improve NE performances.

D_s data selection is a common approach to domain adaptation. It requires no labeled data in D_s and also independent from any classifiers [1]. Entropy-based adaptation [12, 13, 14] is a prevailing method for selecting data in D_s based on their entropy-measure scores. The entropy-measure can help to estimate the difference of probability distribution between D_s and D_t. By approximating which data in D_s are close to D_t using the entropy-measure, we can select the data that can be used to improve T_t performances [13].

Therefore, we can consider which approach that can be used to classify URLs to support link prediction with considering the cost of annotating the URLs manually and the cost of using external labeled URLs as the training data. Adopting labeled URLs from external or out-of-domain as the training data (D_s) to classify the URLs as the test data (D_t) using entropy-based adaptation approaches seems as a trade-off to deal with both cost issues.

1.2 Research Question

The challenge using entropy-based adaptation is to find a good measure for calculating the similarity between training URLs in D_s and test URLs in D_t to improve selection. From the proposed solution, a research question can be formulated as follows :

To what extent is it possible to classify URLs with labeled out-of-domain URLs as training set using entropy-based adaptation methods?

The research question can be divided into several sub-questions as follows:

- Can entropy-based adaptation methods help D_s data selection to increase URL classification performance in using labeled out-of-domain or D_s URLs?
- What is the most useful feature representative of URL that allows entropy-based adaptation methods to reduce error rates on URL classification in using labeled out-of-domain URLs in D_s?
- What is the most effective entropy-based adaptation method in selecting D_s for URL classification?

1.3 Scope of Study

In this thesis, Entropy-based adaptation methods are proposed to classify URLs using URL features only. Commonly URL classification is used to classify its web page [15]. To classify the URL, one can use its web page content as its metadata, but URL classification without its page's content is preferable for three reasons [16]. First, when the content of the URL is not available, for example, the content provider want to limit access to the corresponding content. The second reason is when the classification is needed before we can obtain the content of the URL, for example, when it is used in topic focused crawlers. If such a system can predict the topic of a hyperlink before downloading the page, it can limit the waste of bandwidth caused by irrelevant pages. The third reason is when the classification speed is crucial. This is particularly relevant for an on-the-fly classification of Web search results, where only limited content is available and speed is of utmost importance.

1.4 Outline

This thesis consists of five chapters as follows:

- Introduction: this chapter introduces the background of this research including problems, motivation selecting this topic, proposed solution, research question, research objective and limitation of this thesis.
- Related Work: this chapter describes the theoretical framework related to this thesis including Web Logs, Link Prediction concepts, Classification theories particularly URL Classification, and Domain Adaptation algorithms.

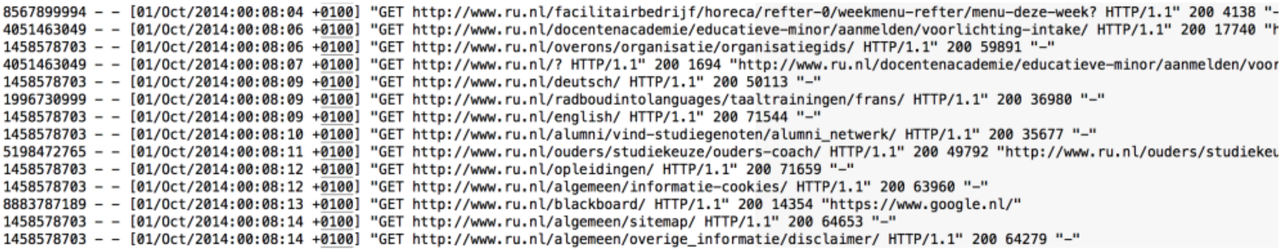
- Method: this chapter presents the research methodology of this research. It explains the procedures taken in this research from data selection until evaluation including which classification approaches used in this thesis.
- Evaluation: this chapter discusses the finding of the research and evaluation of this research conducted by human experts using standard evaluation criteria.
- Conclusion and Future Work: this chapter states the answer to the research questions posed as a final conclusion of this research and includes some discussion for further work.

Chapter 2

Related Work

2.1 Web Logs

Web servers record a web log access [17] for every single request they get from web user, including the URL requested, the IP address from which the requested originated, and a timestamps. A fragment of the Web Log we use in the study is shown as follows:



```
8567899994 - - [01/Oct/2014:00:08:04 +0100] "GET http://www.ru.nl/facilitairbedrijf/horeca/refter-0/weekmenu-refter/menu-deze-week? HTTP/1.1" 200 4138 "-
4051463049 - - [01/Oct/2014:00:08:06 +0100] "GET http://www.ru.nl/docentenacademie/educatieve-minor/aanmelden/voorlichting-intake/ HTTP/1.1" 200 17740 "h
1458578703 - - [01/Oct/2014:00:08:06 +0100] "GET http://www.ru.nl/overons/organisatie/organisatiegids/ HTTP/1.1" 200 59891 "-"
4051463049 - - [01/Oct/2014:00:08:07 +0100] "GET http://www.ru.nl/? HTTP/1.1" 200 1694 "http://www.ru.nl/docentenacademie/educatieve-minor/aanmelden/voor
1458578703 - - [01/Oct/2014:00:08:09 +0100] "GET http://www.ru.nl/deutsch/ HTTP/1.1" 200 50113 "-"
1996730999 - - [01/Oct/2014:00:08:09 +0100] "GET http://www.ru.nl/radboudintolanguages/taaltrainingen/frans/ HTTP/1.1" 200 36980 "-"
1458578703 - - [01/Oct/2014:00:08:09 +0100] "GET http://www.ru.nl/english/ HTTP/1.1" 200 71544 "-"
1458578703 - - [01/Oct/2014:00:08:10 +0100] "GET http://www.ru.nl/alumni/vind-studiegenoten/alumni_netwerk/ HTTP/1.1" 200 35677 "-"
5198472765 - - [01/Oct/2014:00:08:11 +0100] "GET http://www.ru.nl/ouders/studiekeuze/ouders-coach/ HTTP/1.1" 200 49792 "http://www.ru.nl/ouders/studiekeu
1458578703 - - [01/Oct/2014:00:08:12 +0100] "GET http://www.ru.nl/opleidingen/ HTTP/1.1" 200 71659 "-"
1458578703 - - [01/Oct/2014:00:08:12 +0100] "GET http://www.ru.nl/algemeen/informatie-cookies/ HTTP/1.1" 200 63960 "-"
8883787189 - - [01/Oct/2014:00:08:13 +0100] "GET http://www.ru.nl/blackboard/ HTTP/1.1" 200 14354 "https://www.google.nl/"
1458578703 - - [01/Oct/2014:00:08:14 +0100] "GET http://www.ru.nl/algemeen/sitemap/ HTTP/1.1" 200 64653 "-"
1458578703 - - [01/Oct/2014:00:08:14 +0100] "GET http://www.ru.nl/algemeen/overige_informatie/disclaimer/ HTTP/1.1" 200 64279 "-"
```

Figure 2.1: A fragment of web log file with masked IP addresses in the first column [2].

The first column in Figure 2.1 corresponds to IP addresses from which the URLs were requested. The IP addresses have been anonymized (for privacy concerns).

2.2 Link Prediction

Link prediction is a popular research area with important applications in a variety of disciplines, including biology, social science, security, and medicine. Link prediction refers to the problem of predicting relationships in a network [18, 19]. Link prediction can be used for recommending relevant web pages to the web user and thereby improving the user experience on a web domain [2]. Prior work employed a Markov model employed to estimate the probabilities of visiting other clusters and pages, given a weblog file and a current user access [20].

2.3 Focused Browsing

Focused Browsing fasilitates navigation between pages "on topic"that will most likely be interest to the user [20]. A focused Browsing approach is meant to provide the user with interactive feedback to support their browsing.

The idea of Focused Browsing is similar to the principle of Focused Crawling [5] which is to selectively seek out web pages relevant to pre-defined topics. Normally such users only predict and open the link offered to them to seek their preferences. They often can only know exactly what topic of the link they have opened after they see the web page content about. To not waste their time, one need to indicate the topic of the URLs to support their focused browsing behaviour.

2.4 Classification

Document classification can be useful at numerous stages of the Information Retrieval processes. The documents to be classified may be text, music; image, etc., each kind of document possesses its special classification problems. Text categorization is the task of assigning a Boolean value to each pair $\langle d_j, c_i \rangle \in \mathcal{D} \times \mathcal{C}$, where $\mathcal{D}$ is a domain of documents and $\mathcal{C} = \{c_1, ..., c_{|\mathcal{C}|}\}$ is a set of pre-defined *categories.* A value of T assigned to $\langle d_j, c_i \rangle$ indicates a decision to file d_j under c_i, while a value of F indicates a decision not to file d_j under c_i. More formally, the task is to approximate the unknown *target function* $\breve{\Phi} : \mathcal{D} \times \mathcal{C} \rightarrow \{T, F\}$, (that describes how documents ought to be classified) by means of a function $\Phi : \mathcal{D} \times \mathcal{C} \rightarrow \{T, F\}$ called the classifier (aka *rule*, or *hypothesis*, or *model*) such that $\breve{\Phi}$ and Φ “coincide as much as possible” [7].

2.4.1 Training Set and Test Set

A machine learning approach relies on the availability of an *initial corpus* $\Omega = \{d_1, ..., d_{|\Omega|}\} \subset \mathcal{D}$ of documents preclassified under $\mathcal{C} = \{c_1, ..., c_{|\mathcal{C}|}\}$. That is, the values of the total function $\breve{\Phi} : \mathcal{D} \times \mathcal{C} \rightarrow \{T, F\}$ are known for every pair $\langle d_j, c_i \rangle \in \Omega \times \mathcal{C}$. A document d_j is a *positive example* of c_i if $\breve{\Phi}(d_j, c_i) = T$, a *negative example* of c_i if $\breve{\Phi}(d_j, c_i) = F$.

Training Data

In machine learning, the decision criterion of the text classifier is learned automatically from training data [21]. The *training data* $Tr = \{d_1, ..., d_{|Tr|}\}$ is used to learn the best parameter values. The classifier Φ for categories $\mathcal{C} = \{c_1, ..., c_{|\mathcal{C}|}\}$ is inductively built by evaluating the characteristic of these data [7]. In the context of domain adaptation described later in the next sub-section, the term training data Tr will be changed by the term domain source D_s.

Test Data

A *test set* $Te = \{d_{|Tr|+1}, ..., d_{|\Omega|}\}$, used for testing the effectiveness of the classifiers. Each $d_j \in Te$ is fed to the classifier, and the classifier decisions $\Phi(d_j, c_i)$ are compared with the expert decisions $\breve{\Phi}(d_j, c_i)$. A measure of classification effectiveness is based on how often the $\Phi(d_j, c_i)$ values match the $\breve{\Phi}(d_j, c_i)$ values [7]. In the next sub-section and also the rest of this thesis, we will use domain target D_t as the term to refer to test set Te.

2.4.2 URL Classification

Commonly URL classification is used to classify its web page [15]. To classify the URL, one can use its web page content as its metadata, but URL classification without its page's content is preferable when the content of the URL is not available, when the classification is needed before we can obtain the content of the URL and when the classification speed is utmost importance [16]. Classification of wab pages based on URL-only is not new, see e.g. [22, 23]. Unlike URL classification using web page content as its metadata, URL classification using URL-only is more challenging since its features are derived only in limited numbers. The features are normally obtained either from tokenizing the URL into *tokens* or from partitioning the URLs into a subsequence of characters called *n-grams*. Features used in [22, 23] are *tokens* and *n-grams* characters of *tokens* respectively. The previous works used a standard classification setup i.e. where $T_s = T_t$ and $D_s = D_t$. The classifiers can be accurate but only when based on a large quantity and high quality of training set to predict new data in test set D_t.

2.5 Domain Adaptation

A survey conducted by [1] mentioned several definitions of a domain. One of the definitions mentions that adaptation happens between different corpora so that each corpus is considered as a unique domain. For example, we can consider the ACE corpus as the source domain and the CoNLL corpus as the target domain to perform named entity recognition[11].

The idea of domain adaptation is closely related to transfer learning. Transfer learning is a general term that refers to a class of machine learning problems that involve different tasks or domains[1]. A comprehensive survey of transfer learning techniques[10] provides a clear definition of transfer learning:

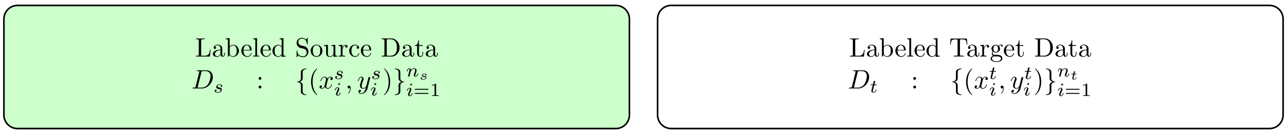
> Given a source domain D_s and its learning task T_s, a target domain D_t and its learning task T_t, transfer learning aims to help improve the learning of the target predictive function F_t in D_t using the knowledge from D_s and D_t, where $D_s \neq D_t$ and $T_s \neq T_t$.

It is assumed that training data is $D = \{(x_i, y_i)\}_{i=1}^{n}$ where $x_i \in X$ is an observed variable, and $y_i \in Y$ is the output label/class of x_i. The subscript S and T are used to distinguish source domain and target domain respectively. Therefore D_s means the training data in the source domain, and D_t stands for the training data in the target domain. The subscript l and u are also used to distinguish labeled and unlabeled data, for example $D_{t,l}$ refers to labeled data in the target domain. Domain adaptation is a part of *transductive transfer learning* methods where T_s and T_t are the same, while D_s and D_t are different [10]. In classification setting, by designing algorithms to transfer knowledge from labeled data in D_s to D_t, we may succeed in T_t for annotating the new data in D_t while keeping high performance.

As reviewed in [3], we can distinguish three Domain Adaptation settings: Supervised Domain Adaptation, Unsupervised Domain Adaptation and Semi-supervised Domain Adaptation.

Supervised Domain Adaptation

In the supervised domain adaptation setting, depicted in Figure 2.2, we are given a rather large amount of labeled source data $D_s : \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ and only a limited amount of labeled data from the target domain: $D_t : (x_i^t, y_i^t)_{i=1}^{n_t}$. That is, there is considerably more source data than target data, i.e $n_s \gg n_t$. The goal of this setting is to exploit the limited target data together with the source data in order to build a model that performs well on the new target domain.



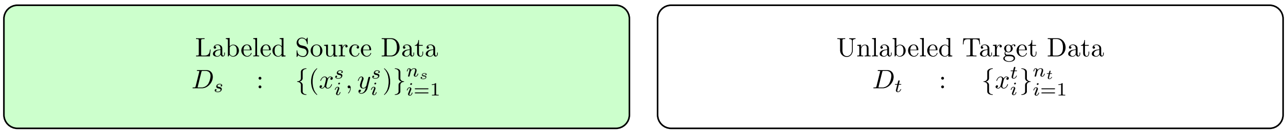
Labeled Source Data
$D_s \quad : \quad \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$

Labeled Target Data
$D_t \quad : \quad \{(x_i^t, y_i^t)\}_{i=1}^{n_t}$

Figure 2.2: Supervised domain adaptation scenario. Both D_s and D_t are labeled, however $n_s \gg n_t$ [3].

Unsupervised Domain Adaptation

In the *unsupervised domain adaptation*, illustrated in Figure 2.3, instead of having labeled target domain data, we only have *unlabeled data* from the target domain. The goal of this setting is to use the original, labeled source domain data together with the unlabeled target domain data to build a model that performs well on the new target domain.



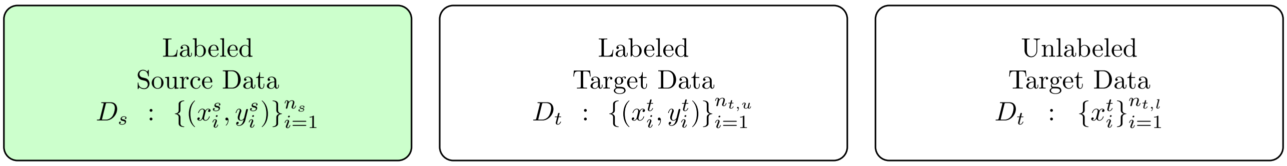
Labeled Source Data
$D_s \quad : \quad \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$

Unlabeled Target Data
$D_t \quad : \quad \{x_i^t\}_{i=1}^{n_t}$

Figure 2.3: Unsupervised domain adaptation scenario. Rather than having labeled data for the target domain, in this setting only unlabeled data od D_t is available. However, there might be lots of unlabeled data, i.e $n_t \gg n_s$ [3].

Semi-supervised Domain Adaptation

Recent studies have started to employ both labeled and unlabeled data from target domain, as illustrated in Figure 2.4. The goal of this setting is to use the labeled source data as well as a limited amount of labeled target data together with lots of unlabeled target data.



Labeled
Source Data
$D_s \; : \; \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$

Labeled
Target Data
$D_t \; : \; \{(x_i^t, y_i^t)\}_{i=1}^{n_{t,u}}$

Unlabeled
Target Data
$D_t \; : \; \{x_i^t\}_{i=1}^{n_{t,l}}$

Figure 2.4: Semi-supervised domain adaptation setting. There is both labeled and unlabeled data available for the target domain. However, there is only a small amount of labeled target data, i.e $n_{t,u} \gg n_{t,l}$ [3].

The scenario in which there are no annotated data available in D_t (*unsupervised domain adaptation*) is a much more realistic situation. That is a most prominently reason where the idea of domain adaptation come from. To deal with the situation, empirical studies [13, 12, 14] have used adapatation approaches based on entropy. The methods measure entropy-estimations in D_s with respect to D_t in order to select data in D_s which are close to D_t. These methods, discussed below, are preferable in most practical settings not only because they are independent of the availability of labeled data in D_t but also they are agnostic of the underlying machine learning algorithms since they can be considered as a preprocessing step before performing any implementations [1].

2.6 Entropy-based Data Selection

D_s data selection is a common method for domain adaptation. The goal of the method is to select a subset of D_s that can give better results for a given D_t. The method is independent of the choise of classifier, especially suitable to the situation in which there exist many examples x_i in D_s for which $p_s(y|x_i)$ is similar to $p_t(y|x_i)$ [1]. The main challenge of this method is how to evaluate the importance of the data that we want to select in D_s according to their relevance to D_t. The entropy-based measure discussed below is a simple way to tackle the challenge of adapting D_s to D_t.

Entropy, Information Theory and Probability

If two independent events x_1 and x_2 occur (whose $p(x_1, x_2) = p(x_1) * p(x_2)$), then the information we get from observing the events is the sum of the two informations as explained in [24]:

$$I(p(x_1) * p(x_2)) = I(p(x_1)) + I(p(x_2)) \tag{2.1}$$

Suppose we have n events $\{x_1, x_2, ..., x_n\}$ and some source is providing us with a stream of these events. Suppose further that the source produces probabilities of the events $\{p(x_1), p(x_2), .., p(x_n)\}$. For now, we also assume that the events are transmitted independently (successive events do not depend in any way on past events). What is the average amount of information we get from each event we see in the stream?

What we really want here is a weighted average. If we observe the event x_i, we will then obtain $\log(1/p(x_i))$ *information* from that particular observation. In a long run (say N) of observations, we will see (approximately) $N * p(x_i)$ occurrences of symbol x_i (in the frequentist sense, that's what it means to say that the probability of seeing x_i is $p(x_i)$). Thus, in the N (independent) observations, we will get total information I of

$$I = \sum_{i=1}^{n} (N * p(x_i)) \log_2(\frac{1}{p(x_i)}) \tag{2.2}$$

But then, the average information we get per event observed will be

$$\begin{aligned} \frac{I}{N} &= (\frac{1}{N}) \sum_{i=1}^{n} (N * p(x_i)) \log_2(\frac{1}{p(x_i)}) \\ &= \sum_{i=1}^{n} (p(x_i)) \log_2(\frac{1}{p(x_i)}) \\ &= -\sum_{i=1}^{n} (p(x_i)) \log_2(p(x_i)) \end{aligned} \tag{2.3}$$

This leads to a fundamental definition. This definition is essentially coming from Shannon's seminal paper [4]. As we have observed, we have defined information strictly in terms of the probabilities of events. Therefore, suppose we have a discrete probability distribution $P = \{p_1, p_2, ..., p_n\}$. We define the *entropy* of the distribution P by:

$$H(P) = -\sum_{i=1}^{n} p_i \log_2 p_i \tag{2.4}$$

In information theory, Eq. 2.5 shows the standard definition of entropy, where X is a discrete random variable with m possible outcomes x_i,..., x_m and p is a probability distribution of X.

$$H(X) = -\sum_{i=1}^{m} p(x_i) \log_2 p(x_i) \tag{2.5}$$

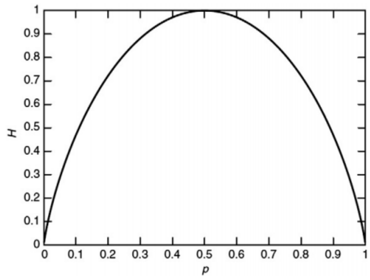


Figure 2.5: Entropy vs Probability [4]

Figure 2.5 shows the relation between entropy and probability of a fair coin. According to the entropy curve as depicted in Figure 2.5, entropy is a measure of uncertainty of unknown or random variables where the higher entropy scores are placed in between two extreme probabilities; low and high.

Cross Entropy

As described in [12], in the case of word segmentation and POS tagging, cross entropy becomes a measure by calculating the cross entropy for a sentence s over two discrete probability distributions p and q, where p and q are estimated from D_s and D_t, respectively and $x_1, x_2, ..x_n$ are feature representatives of s. This measure is meant to calculate similarity of sentences in p with respect to q as formulated in Eq.
refeqn:ce-related-work. The sentences with lowest cross entropy score are prioritized. The intuition of this criterion is to select sentences in D_s whose distribution is similar to D_t data.

$$CE(s,p,q) = -\sum_{i=1}^{n} p(x_i) \log_2 q(x_i) \tag{2.6}$$

Cross Entropy Difference

To estimate similarity between sentences, cross entropy difference measure can be used as described in [14]. To describe cross entropy difference measure formally, let p be an training set or D_s and q be a test set or D_t. Let *H(s,p,q)* be the cross-entropy, according to x feature representatives of s in q, of a sentence s drawn from p and let *H(s,p)* be the cross-entropy of s according to x feature representatives of s in p. For each sentence, we score it according to *H(s,p,q)* $-$ *H(s,p)* . It is assumed that if a sentence has a low cross entropy difference score then it is close or similar to sentences in q.

$$H(s,p,q) - H(s,p) = |(-\sum_{i=1}^{n} p(x_i) \log_2 q(x_i)) - (-\sum_{i=1}^{n} p(x_i) \log_2 p(x_i))| \tag{2.7}$$

Entropy Difference

As described in [12], in word segmentation and POS tagging, entropy difference can be used to estimate similarity of sentences. For formal definition, given a sentence s, s is represented as a set of information units x_1,...,x_n, where an information unit can be a word/n-gram tokens. Let p be the probability distribution over all the information units collected from a data set C. Instead of calculating the entropy of the random variable

X as in Eq 2.5 which uses all the possible x_i in C, it will be better if it focuses only on the x_i in s; therefore, a new function *H(s, p)* is defined as in Eq 2.8.

$$H(s,p) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \tag{2.8}$$

Let p and q be the probability distributions estimated from D_s and D_t, respectively. Let s be a sentence in the D_s. Eq. 2.9 defines the difference of sentence entropy, *ED(s,p,q)*. Intuitively, choosing sentences with low values of entropy difference means the sentences are preferable since their units x_i have similar values with respect to p and q.

$$ED(s,p,q) = |H(s,p) - H(s,q)| \tag{2.9}$$

Average Entropy Gain

The term *entropy gain* refers to how much entropy is accordingly changed when the data is changed a little bit. In word segmentation and POS tagging case [12], average entropy gain can be employed to give better similarity measures. To define average entropy gain formally, let C be the test corpus and s be a sentence, and entropy gain (EG) is defined as in Eq 2.10, where q is a probability distribution estimated from C and *q1* is one estimated from *C+s*, a new corpus formed by adding s to C. Intuitively, if s is similar to C, *q1* will be very similar to q and *EG(s,c)* will be small.

$$EG(s,C) = |H(C+s, q1) - H(C,q)| \tag{2.10}$$

The measures in Eq 2.10 can be normalized by sentence length. Eq 2.11 shows the normalized entropy gain and it is called average entropy gain.

$$AEG(s,C) = \frac{EG(s,C)}{length(s)} \tag{2.11}$$

Chapter 3

Method

We present a method of applying entropy-based adaptation to the problem of URL classification in focused browsing setting. Here we represent URLs by their constituent feature entropies. We will empirically evaluate the best choice of features in the evaluation chapter.

3.1 Independence feature assumption

Let $\{x_1, x_2, ..., x_n\}$ be feature representatives of URLs in a collection and $\{p_1, p_2, ..., p_n\}$ be the probability of each feature respectively. We assume $\{x_1, x_2, ..., x_n\}$ are independence each other so that the joint probability

$$p(x_1, x_2, ..., x_n) = p(x_1) * p(x_2) * ... * p(x_n) \tag{3.1}$$

We can estimate the information from a particular feature $I(x_i) = -\log_2 p(x_i)$. From the independence assumption, we can derive total information in collection $I(x_1 * x_2 * ... * x_n) = I(x_1) + I(x_2) + ... + I(x_n)$ and then the average information we get per feature observed as known also as *entropy* as follows:

$$I/N = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \tag{3.2}$$

Let $\{u_1, u_2, ..., u_n\}$ be URLs in a collection and $\{x_1, ..., x_m\}$ be the features of u_i, if we want to estimate the entropy of features $H(u_{xi})$ in a specific URL u_i, we can sum all the average of information of features in u_i as follows:

$$H(u_{xi}) = -\sum_{j=1}^{m} p(x_j) \log_2 p(x_j) \tag{3.3}$$

where x_j indicates features in u_i and $p(x_j)$ is probability of x_j in the collection.

3.2 URL entropy-based calculation

The following sections will use D_s and D_t as the input data in their algorithmic explanations. To make it clear to understand, both D_s and D_t consist of URLs and their representative features. The features could be words/tokens/n-grams derived from preprocessing step. In detail, we will discuss the preprocessing steps in chapter 4.

3.2.1 Cross entropy

Cross entropy of URL features is calculated as follows. Let u be an URL from D_s, $\{x_1, x_2, ..., x_n\}$ be feature representatives of u, p_s be discrete probability distribution in D_s and p_t be discrete probability distribution in D_t.

$$CE(u, p_s, p_t) = -\sum_{i=1}^{n} p_s(x_i) \log_2 p_t(x_i) \tag{3.4}$$

The cross entropy of features in u is estimated over two discrete probability distributions p_s and p_t. The following algo. details the calculation of cross entropy of URL features:

```
Data: $D_s$, $D_t$
Result: sorted URLs in $D_s$
$urltemp = [\ \ ]$;
for each URL $u$ in $D_s$ do
    $CE_{xu} = 0$;
    for each feature $x_i$ in $u$ do
        Count $fx_{si}$ frequency of $x_i$ in $D_s$;
        Count $fx_{ti}$ frequency of $x_i$ in $D_t$;
        Count $fx_s$ total features in $D_s$;
        Count $fx_t$ total features in $D_t$;
        Count $p_s(x_i)$ probability of $x_i$ in $D_s : fx_{si}/fx_s$;
        Count $p_t(x_i)$ probability of $x_i$ in $D_t : fx_{ti}/fx_t$;
        if $p_t(x_i) = 0$ then
            $CE_{xu} \ += \ 0$;
        else
            Count $CE_{xi}$ cross entropy score of $x_i : -(p_s(x_i) \log_2 p_t(x_i))$;
            $CE_{xu} \ += \ CE_{xi}$;
        end
    end
    Store $u$ along with its $CE_{xu}$ score in urltemp;
end
Sort all $u$ in urltemp based on their $CE_{xu}$ score in ascending order;
```

Algorithm 1: URL feature cross entropy calculation.

3.2.2 Entropy difference

To count entropy difference of URL features, let u be a URL and u is represented as a set of unit features $\{x_1,...,x_n\}$. Let p_s be the discrete probability distribution over all the unit features collected from D_s. If we focus only on the entropy of x_i in u, then *H(u, p_s)* is defined as in Eq 3.5.

$$H(u, p_s) = -\sum_{i=1}^{n} p_s(x_i) \log_2 p_s(x_i) \tag{3.5}$$

Let p_s and p_t be the discrete probability distributions estimated from D_s and D_t, respectively. We define the difference of URL feature entropy, *ED(u,p_s,p_t)*, as in Eq 3.6. The detailed calculation can be seen in algo.

2.

$$ED(u, p_s, p_t) = |H(u, p_s) - H(u, p_t)| \tag{3.6}$$

```
Data: $D_s$, $D_t$
Result: sorted URLs in $D_s$
$urltemp = [\quad]$ ;
for each url $u$ in $D_s$ do
    $E_{su} = 0$;
    $E_{tu} = 0$;
    for each feature $x_i$ in $u$ do
        Count $fx_{si}$ frequency of $x_i$ in $D_s$;
        Count $fx_{ti}$ frequency of $x_i$ in $D_t$;
        Count $fx_s$ total features in $D_s$;
        Count $fx_t$ total features in $D_t$;
        Count $p_s(x_i)$ probability of $x_i$ in $D_s : fx_{si}/fx_s$;
        Count $p_t(x_i)$ probability of $x_i$ in $D_t : fx_{ti}/fx_t$;
        if $p_s(x_i) = 0$ then
            $E_{su} \;\; += \;\; 0$;
        else
            Count $E_{si}$ entropy score of $x_i$ with respect to $D_s : -(p_s(x_i) \log_2 p_s(x_i))$ ;
            $E_{su} \;\; += \;\; E_{si}$;
        end
        if $p_t(x_i) = 0$ then
            $E_{tu} \;\; += \;\; 0$;
        else
            Count $E_{ti}$ entropy score of $x_i$ with respect to $D_t : -(p_t(x_i) \log_2 p_t(x_i))$ ;
            $E_{tu} \;\; += \;\; E_{ti}$;
        end
    end
    Count $ED_{xu}$ entropy difference of $x$ in $u : |E_{su} - E_{tu}|$ ;
    Store $u$ along with its $ED_{xu}$ score in urltemp;
end
Sort all $u$ in urltemp based on their $ED_u$ score in ascending order ;
```

Algorithm 2: URL feature entropy difference calculation.

3.2.3 Cross entropy difference

To calculate cross entropy of features in each URL in D_s, let $\{x_1, x_2, ..., x_n\}$ be feature representatives of URL u, p_s be a discrete probability distribution in D_s and p_t be discrete probability distribution in D_t. Let $H(u, p_s, p_t)$ be the cross-entropy, according to x_i in p_t, of u drawn from p_s and let $H(u, p_s)$ be the cross-entropy of features of u according to x_i in p_s only. For each u, we estimate it according to $|H(u, p_s, p_t) - H(u, p_s)|$. The detail calculation of cross entropy difference can be seen in algo. 3.

$$|H(u, p_s, p_t) - H(u, p_s)| = |(-\sum_{i=1}^{n} p_s(x_i) \log_2 p_t(x_i)) - (-\sum_{i=1}^{n} p_s(x_i) \log_2 p_s(x_i))| \tag{3.7}$$

```
Data: $D_s$, $D_t$
Result: sorted URLs in $D_s$
$urltemp = [\ \ ]$ ;
for each URL $u$ in $D_s$ do
    $CED_{su} = 0$;
    $CED_{tu} = 0$;
    for each feature $x_i$ in $u$ do
        Count $fx_{si}$ frequency of $x_i$ in $D_s$;
        Count $fx_{ti}$ frequency of $x_i$ in $D_t$;
        Count $fx_s$ total features in $D_s$;
        Count $fx_t$ total features in $D_t$;
        Count $p_s(x_i)$ probability of $x_i$ in $D_s : fx_{si}/fx_s$;
        Count $p_t(x_i)$ probability of $x_i$ in $D_t : fx_{ti}/fx_t$;
        if $p_t(x_i) = 0$ then
            $CE_{stu}\ \ += \ \ 0$;
        else
            Count $CE_{xsti}$ cross entropy score of $x_i$ with respect to $D_s$ and $D_t : -(p_s(x_i) \log_2 p_t(x_i))$ ;
            $CE_{stu}\ \ += \ \ CE_{xsti}$;
            Count $CE_{xsi}$ cross entropy score of $x_i$ with respect to $D_s$ only : $-(p_s(x_i) \log_2 p_s(x_i))$ ;
            $CE_{su}\ \ += \ \ CE_{xti}$;
        end
    end
    Count $CED_{xu}$ cross entropy difference of $x$ in $u : |CE_{stu} - CE_{su}|$ ;
    Store $u$ along with its $CED_{xu}$ score in urltemp;
end
Sort all $u$ in urltemp based on their $CED_{xu}$ score in ascending order;
```

Algorithm 3: URL feature cross entropy difference calculation.

3.2.4 Average entropy gain

Average entropy gain adopted from [12] can be employed for URL selection. Let u be a URL from D_s, and entropy gain (EG) of u features is defined as in Eq 3.8, where p_t is a probability distribution estimated from D_t and p_{t1} is one estimated from a new corpus formed by adding u to D_t. The detail calculation can be seen in algo. 4.

$$EG(u, X) = |H(X + u, p_{t1}) - H(X, p_t)| \tag{3.8}$$

$H(X, p)$ follows the standard definition of entropy in information theory, where X is a discrete random variable with n possible outcomes $\{x_1, x_2, ..., x_n\}$ and p is a probability distribution of X.

$$H(X, p) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \tag{3.9}$$

then URL feature average entropy gain calculation can be seen as follows:

$$AEG(u, X) = \frac{EG(u, X)}{length(u)} \tag{3.10}$$

```
Data: $D_s$, $D_t$
Result: sorted URLs in $D_s$
$urltemp = [\ \ ]$ ;
for each url $u$ in $D_s$ do
    Make a new corpus $D_q : D_t + u$ ;
    Count $fx_t$ total features in $D_t$;
    Count $fx_q$ total features in $D_q$;
    Count $fx_u$ total features in $u$;
    $E_{qu} = 0$;
    $E_{tu} = 0$;
    for each feature $x_{qi}$ in $D_q$ do
        Count $fx_{qi}$ frequency of $x_{qi}$ in $D_q$;
        Count $p_q(x_i)$ probability of $x_{qi}$ in $D_q : fx_{qi}/fx_q$;
        if $p_q(x_i) = 0$ then
            $E_{qu} \ += \ 0$ ;
        else
            Count $E_{xqi}$ entropy score of $x_{qi}$ with respect to $D_q : -(p_q(x_{qi}) \log_2 p_q(x_{qi}))$ ;
            $E_{qu} \ += \ E_{xqi}$ ;
        end
    end
    for each feature $x_{ti}$ in $D_t$ do
        Count $fx_{ti}$ frequency of $x_{ti}$ in $D_t$;
        Count $p_t(x_i)$ probability of $x_{ti}$ in $D_t : fx_{ti}/fx_t$;
        if $p_t(x_i) = 0$ then
            $E_{tu} \ += \ 0$ ;
        else
            Count $E_{xti}$ entropy score of $x_{ti}$ with respect to $D_t : -(p_t(x_{ti}) \log_2 p_t(x_{ti}))$;
            $E_{tu} \ += \ E_{xti}$ ;
        end
    end
    Count $EG_{xu}$ entropy gain of $x$ in $u : |E_{qu} - E_{tu}|$;
    Count $AEG_{xu}$ average entropy gain of $x$ in $u : EG_{xu}/fx_u$;
    Store $u$ along with its $AEG_{xu}$ score in $urltemp$;
end
Sort all $u$ in $urltemp$ based on their $AEG_{xu}$ score in ascending order;
```

Algorithm 4: URL feature average entropy gain calculation.

Chapter 4

Experimental Setup

In this chapter, we will introduce the experimental evaluation setup that was taken and considered in this thesis from data that were used, data preprocessing methods, approaches and algorithms that were used to classify URLs and evaluation methods.

4.1 Data

For this experiment, Web-KB [25] dataset was employed either as D_s or D_t data. It was choosen because it contains URLs from various universities. Web-KB URLs were collected and annotated from the computer science departments in four universities (Cornell, Texas, Washington, Wisconsin) and one from universities grouped as Misc. Each "university" group subset of Web-KB URLs was considered as an unique domain because it has its own URL characteristics.

In this thesis, we only selected the class of Web-KB dataset with sufficient number of URLs per each "university" group. As the result, from seven classes, three classes were removed i.e. *staff*, *department* and *other*. In total, there were five unique domains: Cornell, Texas, Washington, Wisconsin and Misc. where each domain contains four classes-labels : *course, faculty, student,* and *project*. In detail, we can see the number of Web-KB URLs for each domain and class in table 4.1.

We used leave-one-domain-out setup where each domain in D_t with four domains in D_s. To see the reliablity of the experimental result on Web-KB dataset, 100 random english-version URLs of Radboud University Nijmegen was also set as D_t while the five group universities in Web-KB (Cornell, Texas, Washington, Wisconsin and Misc.) was set as D_s.

University	(course)	(faculty)	(project)	(student)	(staff)*	(department)*	(other)*
Cornell	44	34	20	128	21	1	619
Texas	77	31	21	126	3	1	571
Washington	85	42	25	156	10	1	939
Wisconsin	38	46	20	148	12	1	942
Misc.	686	971	418	1083	91	178	693

Table 4.1: The number of URLs in WebKB dataset. * indicates removed classes.

4.2 Preprocessing and feature extraction

In this thesis, two data representations are used as features in the classifier: *tokens* as features and character *n-grams* of tokenized features as features as employed in [23]. For the *n-grams* features, only 2, 3 and 4-grams were used, following the empirical results of [26]. In the evaluation phase, these features were evaluated to see which one was useful to reduce error rates for URL classification using entropy-based adaptation methods.

The preprocessing relied on the pattern of features mentioned above. Therefore, two preprocessing processes were conducted. The first preprocessing was for obtaining *tokens* as features as described instructively in algo. 5. The second preprocessing was for extracting character *n-grams* as features from tokenized features as explained algorithmically in preprocess algo. 6.

```
Data: URLs
Result: URL token features
for each URL u do
    Remove the domain name and the protocol (e.g. http) of $u$;
    Obtain $r$ the rest part of $u$;
    Tokenize $r$ into tokens $t$ by removing any special character, symbol and number;
    $feattemp = [\quad]$;
    for each $t > 1$ do
        $t_i \leftarrow$ lowercase($t_i$) ;
        Store $t_i$ in $feattemp$ ;
    end
    Add all tokens in $feattemp$ as $feature$ values into URL dataset;
end
```

Algorithm 5: Preprocessing steps to obtain *tokens* as features of URLs

To see at a glance the input and the output of both type of preprocessing, consider the examples below:

An input (class, university, URL all seperated by a tab) from URL dataset is given as follows:

http://www.cs.cornell.edu/edu/courses/341/spring96/index.html

As the result of algo. 5, the URL in the input will be transformed into *token* features as follows:

edu courses spring index html

```
Data: URLs
Result: URL $n - gram$ features
for each URL $u$ do
    Set $n$ as the number of sub-sequence characters in $n - grams$;
    Remove the domain name and the protocol (e.g. http) of $u$;
    Obtain $r$ the rest part of $u$;
    Tokenize $r$ into tokens $t$ by removing any special character, symbol and number;
    $feattemp = [\ \ ]$ ;
    for each $t > 1$ do
        if $t_i \leq n$ then
            $t_i \leftarrow$ lowercase($t_i$) ;
            Store $t_i$ in $feattemp$ ;
        else
            Partition the sub-sequence $t_i$ into $n - gram$ characters $ng$;
            Store $ng$ in $feattemp$;
        end
    end
    Add all $ng$ in $feattemp$ as $feature$ values into URL dataset;
end
```

Algorithm 6: Preprocessing steps to obtain character $n - grams$ features of URLs

and for character *n-gram* features, given an input data:

http://www.cs.cornell.edu/edu/courses/341/spring96/index.html

As the result of algo. 6, the URL will be transformed into *4-grams* characters of tokenized features as follows:

edu cour ours urse rses spri prin ring inde ndex html

Since the domain name of URLs in all datasets are constant, it was removed in both feature representations, as well as the sub-domain parts. It means that only the right part of URLs were preprocessed. For Radboud University Nijmegen URL, since we only selected the english version of URLs, we extended the removed part until sub-path */english/* or */en/* part. That was because the URLs are all constant until to those parts.

4.3 Approach and algorithm

The objective of the study is to find out the effectiveness of the entropy-based selection. To answer that question we compared entropy-based selection methods to random selections as the baseline. We used standard text classification methods in both scenarios.

1. Random selection (baseline).

   In this setting, URLs in D_s were selected randomly and then used as the training data to classify URLs in D_t using standard text classification algorithms i.e. Multinomial Naive Bayes and Linear Support Vector Machine (LinearSVC) with default settings and TF-IDF weighting as the baseline using *scikit-learn* software [27].

2. Entropy-based selection

In this approach, the classification process was similar to random selection setting but before the classification was taken, the training data must be sorted and selected using entropy-based measures in *unsupervised domain adaptation* setting.

3. Selecting D_s as the training data

   Before both random selection and entropy-based selection were performed, to see variances of the classification performances over the increasing number of URLs in the selected part of D_s, the D_s were divided into 10 percentage parts of total URLs: 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 %.

4.4 Measuring effectiveness

There were two evaluations conducted in this thesis as described as follows:

1. Classification evaluation over Web-KB dataset

   In this phase, the test data will be partitioned into 10 parts with keeping all the four classes' data available in those parts. It was designed to see the stability of the classification performance over all partitions of data through a macro-averaged F1-score evaluation that would compensate the imbalanced classes. To determine whether the result from applying entropy-based selection leads to a statistically significant difference in classification effectiveness or not, the clasification result for each part of the test data and each percentage of the training data were evaluated using a ten-partition *two-tailed paired student* t-test as also used in [12], comparing each entropy-based selection with average of three random selection experiment results.

   To be more specific, the t-test was conducted in the following steps:

   (a) Split the test data into 10 parts.

   (b) For each percentage of training data, calculate the classification result on each part the test data when using random selection (the average of three random selection results) vs. a particular selection method (e.g. entropy difference). That gives 10 pairs of scores.

   (c) Compute t-test scores on the 10 pair of scores aboved to determine whether the difference between random selection and a particular entropy-based selection method is statistically significant.

2. Classification evaluation over Radboud University Nijmegen URLs

   In this evaluation phase, we performed URL classification over a random sample of 100 english-version URLs from Radboud University Nijmegen. This evaluation was meant to validate reliability and consistency of the best features and entropy-based adaptation mehtod based on the outcome of the first evaluation. We set all subset of Web-KB dataset as D_s and URLs from Radboud University Nijmegen as D_t.

   As the effectiveness measurement, we asked two human judges from Radboud University Nijmegen students [1] to choose a suitable category/class for each URL of the 100-random in selected URLs and then we compared the results between the rater categories and the prediction from both random and entropy-based selection methods using Cohen's kappa measurement. We also counted how many times the judges

[1] We choosed one student from Information Science programme and another from Lingustics programme.

agree and then we evaluated the URLs (which both judges agree) using cross-validation with leave-one-out approach to see how close or far the results between the prediction derived from entropy-methods proposed and human labels.

The Cohen's Kappa measurement proposed are described as follows:

$$k = \frac{p_o - p_e}{1 - p_e} \tag{4.1}$$

where p_o is the relative observed agreement among raters (identical to accuracy), and p_e is the hypothetical probability of chance agreement, using the observed data to calculate the probabilities of each observer randomly saying each category. If the raters are in complete agreement then $k = 1$. If there is no agreement among the raters other than what would be expected by chance (as given by p_e), $k \leq 0$.

Chapter 5

Evaluation

5.1 Classification performance on Web-KB URLs

In this chapter, we will see the effectiveness of each entropy-based selection method on selecting data in D_s to give better URL classification performance compared to random selection over Web-KB dataset. We expect that the entropy-based selection method could select the URLs in D_s which are close to URLs in D_t through calculating their features using entropy-measures. The small portion of selected data in D_s, but close to or similar to data distribution in D_t, could be equal or even better to give classification performance than a huge amount of noisy data in D_s.

All results presented below are the average of 10 result performances of 10 test sets in macro-averaged F1 scores. Abbreviation RDM, CE, ED, CED, AEG mean random, cross entropy, entropy difference, cross entropy difference, average entropy gain respectively. For feature abbreviations, tk, ng2, ng3 and ng4 mean tokens, 2-grams, 3-grams and 4-grams features respectively.

		RDM				CE				ED				CED				AEG			
Cl.	%Train	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4
NB	10%	32.34	57.00	54.94	**60.00**	18.09	24.78	18.09	18.09	24.25	51.54	44.83	18.00	21.55	41.98	30.53	26.31	27.37	34.88	46.77	46.31
	20%	34.48	*58.54*	55.07	53.99	18.09	15.46	18.09	18.09	23.52	56.22	50.48	30.35	20.31	49.50	42.54	27.79	29.91	52.70	51.25	53.14
	30%	33.85	57.48	53.88	54.26	18.09	15.36	18.11	18.09	23.04	57.34	**59.27**	39.16	33.85	50.60	52.81	36.97	28.20	58.42	53.84	52.97
	40%	34.37	57.81	54.48	54.57	18.09	37.96	15.63	18.09	32.97	53.84	**61.55**	45.58	44.58	53.58	54.82	51.33	32.73	57.57	54.20	53.54
	50%	37.78	57.72	54.47	54.44	43.99	42.87	38.97	20.72	38.08	56.29	**64.34***	55.22	42.91	54.88	54.54	55.74	39.10	60.46	55.59	55.24
	60%	38.86	59.20	54.17	55.99	53.87**	51.61	53.08	47.43	40.18	57.85	**64.21***	58.22	42.54	53.86	49.99	53.15	37.81	60.46	56.86	55.18
	70%	36.72	58.92	55.66	55.55	53.87**	**64.25**	56.07	54.88	42.59	58.67	62.24	56.22	45.79	54.21	52.50	53.20	37.98	59.75	56.56	55.18
	80%	36.06	58.63	55.70	55.47	64.76**	64.18	**68.22***	62.55	46.07	58.64	56.67	54.83	47.97*	54.88	56.19	55.91	37.13	59.66	56.15	55.71
	90%	37.67	58.38	56.07	55.88	51.64**	64.22	**67.05**	63.91	32.96	59.37	58.12	53.74	41.63	55.89	58.14	57.60	37.13	58.35	55.41	55.71
	100%	37.69	57.61	55.70	55.71	38.19	57.61	55.70	55.71	37.52	57.61	55.41	55.71	37.92	**57.61**	55.70	55.71	37.69	57.61	55.70	55.71
SVM	10%	47.60	**64.93**	62.16	61.43	18.09	23.58	18.09	18.09	15.91	60.34	43.30	27.51	29.90	56.40	53.82	39.24	42.70	45.86	53.31	49.09
	20%	53.50	60.77	**63.69**	57.39	18.09	23.32	18.09	18.09	15.91	58.39	54.84	34.44	29.90	57.64	59.01	44.32	53.56	53.91	57.50	55.64
	30%	54.63	61.19	61.13	57.74	18.09	34.93	17.85	18.09	20.65	60.17	59.31	46.65	46.27	61.33	**62.89**	52.06	52.93	56.39	57.60	56.09
	40%	54.86	60.96	58.52	57.35	18.09	58.46	17.88	18.09	32.23	59.62	62.87	54.69	54.70	**64.80**	63.24	52.56	54.35	57.59	57.22	55.67
	50%	54.87	59.63	59.28	56.05	48.68	**69.31**	55.36	36.21	38.21	64.17	66.53	55.01	58.27	60.77	62.32	53.48	55.53	58.38	57.51	55.41
	60%	55.25	60.17	59.40	56.66	53.15	**68.74**	60.11	50.96	40.03	61.26	63.05	58.67	54.38	59.87	53.84	54.99	55.94	57.89	58.67	57.04
	70%	55.23	59.07	59.40	56.44	52.87	62.00	58.42	54.40	41.99	**64.70**	63.49	57.88	54.68	58.85	55.91	54.17	55.94	57.99	58.06	55.05
	80%	55.37	59.22	58.45	55.38	54.31	61.48	56.13	55.02	49.04	62.50	**63.25**	56.47	55.48	59.32	58.00	53.53	55.48	57.52	57.78	54.77
	90%	55.50	59.26	59.63	55.51	55.48	59.29	58.81	57.36	54.30	62.48	**62.88**	51.64	55.48	57.33	58.19	54.91	55.48	59.49	57.84	55.20
	100%	55.48	**60.38**	57.14	55.56	55.48	60.22	57.27	55.97	55.48	59.84	57.27	55.56	55.48	59.30	57.27	55.56	55.48	60.22	57.27	55.56

Table 5.1: **URL classification performance**: Tested on Cornell and trained on the other three group universities. * and ** indicate significance at 0.05 and 0.01 respectively. The highest score in each row is in bold.

Imbalanced shared feature

For Cornell as the D_t setup, the classification performance can be seen in Table 5.1. To make it easy to observe, we plotted the data from the table into graphs as depicted in Figure 5.1. We clearly notice that the average

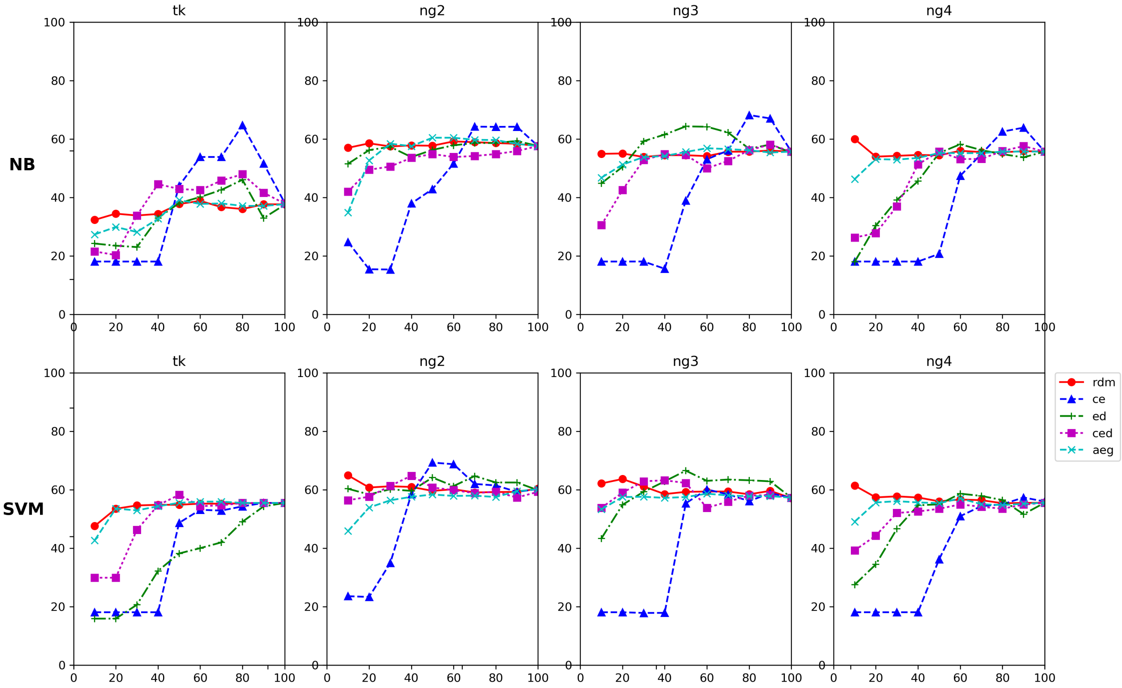


Figure 5.1: **The classification performance over the size of D_s**: The graph plots the performance score that is made up by the fraction of D_s with Cornell as the D_t. X-axis and Y-axis represents D_s percentage and macro-averaged F-1 score respectively.

of the classification performance in random selection was really poor in *tk* as feature i.e. only around 36. By contrast, the Cornell setup had relatively a higher number of *token* and *vocabulary* similarities between D_t and D_s over the size of D_s compared to the other three group university settings (see figure 5.3 and appendix F).

D_t	Class	Token D_t	Token D_s
Cornell	Student	**people:126**, **info:126**, html:126, home:26, index:12, welcome:3, kuen:2, jiawang:2, aswin:2, ychung:2	html:441, **users:235**, homes:127, home:115, students:96, grads:79, phd:65, index:63, **people:41**, www:28
Cornell	Faculty	**info:34**, html:31, **people:23**, faculty:11, department:9, annual:9, dean:2, sam:2, lnt:2, cardie:2	html:496, Faculty:284, cs:98, **users:87**, **people:86**, **info:82**, index:68, home:65, fac:51, dept:43

Table 5.2: **Top Ten Keywords in class *student* and *faculty* in Cornell setup:** The table shows top ten *token* features and shared tokens (bold text) in both class *student* and *faculty* along with their frequencies for each D_t and D_s setup.

However, many of those tokens and vocabularies were shared among classes, for example, the words *info* and *people*, as shown in Table D.2. Those two words are keywords to determine class *faculty* and *student*. Since majority of D_t URLs of class *student* contained those two words and the D_s URLs of class *faculty* had those word frequencies bigger than the D_s URLs of class *student*, as the result many D_t URLs of class *student* were miss-classified as *faculty* URLs. Those imbalanced shared features were the most likely causes for miss-classification using Multinomial Naive Bayes classifier in *tk* as feature as shown briefly in Table 5.3. Unlike the Multinomial Naive Bayes results, results using SVM classifier shows almost consistent for all features. That was because SVM measured the complexity of hypotheses based on the margin with which it separated the data, and not the number of features.

URL	true class	predicted class
http://www.cs.cornell.edu/Info/People/eva/eva.html	faculty	faculty
http://www.cs.cornell.edu/Info/People/kguo/home.html	student	faculty
http://www.cs.utexas.edu/users/wylee/	student	student
http://www.cs.utexas.edu/users/vin/	faculty	student
http://www.cs.washington.edu/homes/dougz/	student	student
http://www.cs.washington.edu/homes/levy/	faculty	student
http://www.cs.wisc.edu/∼carey/carey.html	faculty	faculty
http://www.cs.wisc.edu/∼zeiden/zeiden.html	student	faculty

Table 5.3: **A chunk of true and miss-classification of Web-KB URLs :** The table shows the similarity of URL patterns that can lead to miss-classifications.

Smoothing

We found that the classification performance using CE selection method in this Cornell setup suffered poor classification performances for all features at the small percentage of D_s (10% - 40%). Figure 5.3 shows that the average frequency of similar data among D_t and D_s partitions was zero at tk and ng4 as feature in those D_s fraction. It means that CE prioritized data whose $P_t(x) = 0$ as the closest data to D_t. We suspected that those data which contained many $P_t(x) = 0$ caused CE scores to be 0 since we set $-log(0) = 0$. The data ascending order which prioritized the low CE scores in the beginning of selection polluted the data partitions with many of these “unsimilar”data. These data could give poor performances since they could not help the classifiers to identify correctly data in D_t. To solve this issue, we tried to perform *add-one* smoothing to give the “unseen”data a little bit probability in order to increase their ”information” $(-log(P_t(x)))$ as follows:

$$P(x) = \frac{c(x_i) + 1}{N + \alpha} \tag{5.1}$$

where x is a feature representative of URLs, *P(x)* is probability of x, *c(x_i)* is total feature representative x in dataset, N is total all feature representatives in dataset and α is vocabulary sizes in dataset.

Normalization

All the same, the smoothing still produced poor performances and only selected a few similar data at those small percentages of D_s (see Figure 5.2 and Appendix F). We assumed that the lowest CE scores derived from summations of many pairs of $P_s(x)$ and $P_t(x)$ were still owned by abundant data with $P_t(x) = 0$; therefore they were still placed as the best data to classify D_t. It means that the summation results of many pairs of $P_s(x)$ and $P_t(x) = 0$ with *add-one* smoothing were still lower and dominant at those data partitions than the results from $P_s(x)$ and $P_t(x) > 0$ pairs. We wondered that the features, which had the expected probabilities: $P_t(x) > 0$ and low $P_s(x)$, most likely appeared along with many more features in an URL compared to other features with $P_t(x) = 0$. Therefore, the summation of their average individual “information”(-$P_s(x)log(P_t(x))$) could possibly produce a higher CE value. To fix the issue, after we performed the smoothing, we then tried to normalize the CE score by dividing it with the URL length.

Data sparseness

The performances from the normalization still did not really increase significantly at the whole problematic data partitions except some partitions between 30% and 60% as shown in Figure 5.2. Another possible explanation for the poor performances was that $P_t(x) > 0$ most likely existed along with $P_s(x)$ whose value was relatively

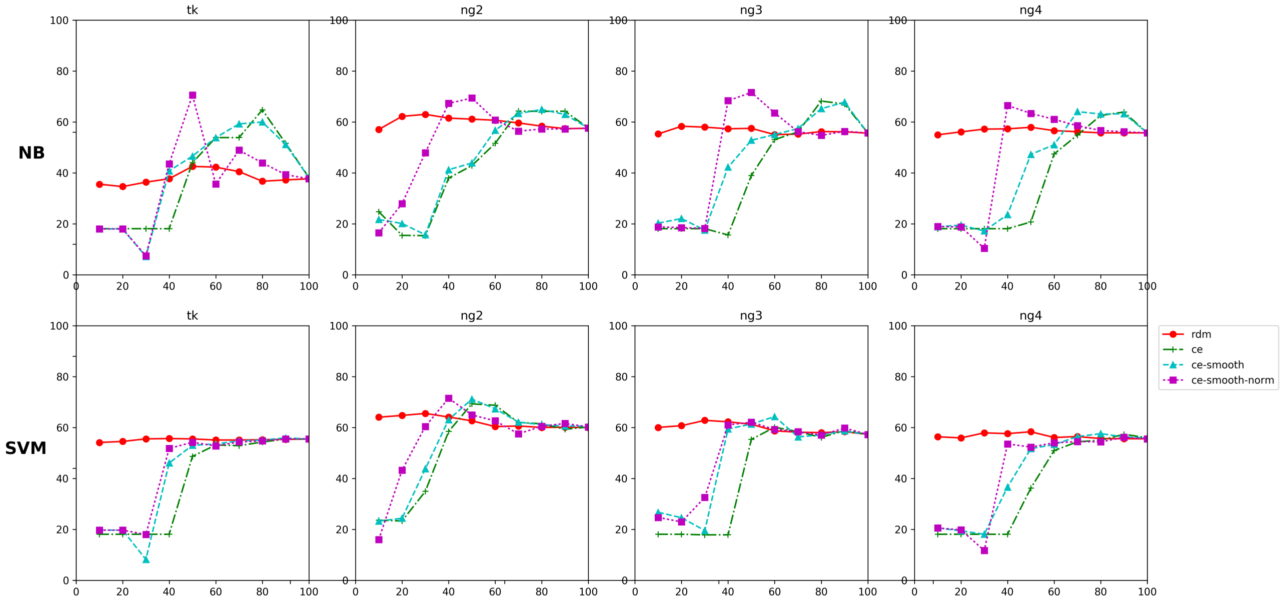


Figure 5.2: **The difference of classification performance using different type of CE over the size of D_s:** The graph shows the performance differences random selection, natural, smoothing and normalized CE with Cornell as the D_t. X-axis and Y-axis represents D_s percentage and macro-averaged F-1 score respectively.

higher than $P_s(x)$ where $P_t(x) = 0$ paired. Since the feature probability $P_s(x)$ was most likely smaller than $P_t(x)$ because $D_s \gg D_t$ in this setup, the smaller $P_s(x)$ values could possibly compensate the $P_t(x) = 0 +$ *add-one smoothing* to produce a low CE score. This last possibility was more realistic since the token similarity among D_s and D_t was lower compared to other features (see Appendix E). The CE scores derived from that possible cause could most likely exceed the CE score derived from a pair of low $P_s(x)$ and $P_t(x) = 0$ at those small percentage of D_s.

Class imbalance

The poor performances at the small percentages of D_s (10%-40%) selected using CE in Cornell setup were almost similar to its non-machine learning classification performances i.e. majority class classification performances (see Table 5.4). It indicates that the low performances were not only because of bad quality of D_s data produced from the selection, but also due to the imbalanced class data (see Table B.1 in Appendix B). The classifiers seemed to predict D_t classes solely based on the majority class data appeared in D_s.

University	Majority Class Classifier				
	$E[F1]$				$E[macro - ave.F1]$
	(co.)	(fa.)	(pr.)	(st.)	
Cornell	0.0	0.0	0.0	72.3	18.1
Texas	0.0	0.0	0.0	66.1	16.5
Washington	0.0	0.0	0.0	67.2	16.8
Wisconsin	0.0	0.0	0.0	73.9	18.5

Table 5.4: **The expected F1 and macro-average F1 score for Majority Class Classifier**. co., fa., pr. and st. indicate class *course, faculty, project* and *student* respectively.

On the other side, as we measured the performance using macro-averaged F1 estimation that gave each class equal weight, the significance results at the relatively big portion of D_s fractions (around 60% - 80%) especially in *tk* as feature as shown in Table 5.1 indicate that CE selection method helped the classifier to lower error rates of the classification performances in many classes in this Cornell setup. The results were derived mostly from true-classification of the other three class URLs such as class *course*, *project* and *course*. It indicates that the best data we wanted to select appeared at relatively big amount of D_s partitions. In other words, we assumed that the data which had low $P_s(x)$ and $P_t(x) > 0$ started to show up at these D_s partitions.

Data miss-selection

CED and ED selection methods seemed not really effective to find similar tokens at the small percentage of D_s in this Cornell setting (see Figure 5.3). Estimating the closeness of features by their entropy difference in D_s and D_t could lead to data miss-selection. If some data did not really exist in D_s and their probabilities in D_t were 0 then those data could possibly have a low entropy score because they could have two almost close low entropy scores; a low score from D_s estimation and a zero score from D_t. Those data in D_s could be selected and prioritized as the closest data to data in D_t simply because they had a small entropy difference scores. Though those data were not really useful for classifying data in D_t because they were not similar to data in D_t. However, the assumption above was partially true because the measurement of difference could be also applicable in determining the closeness of data in which the D_s data had similarity to D_t. The data could have also a lower entropy difference when their probabilities in both D_s and D_t compensated each other, so they had approximately the same entropy scores.

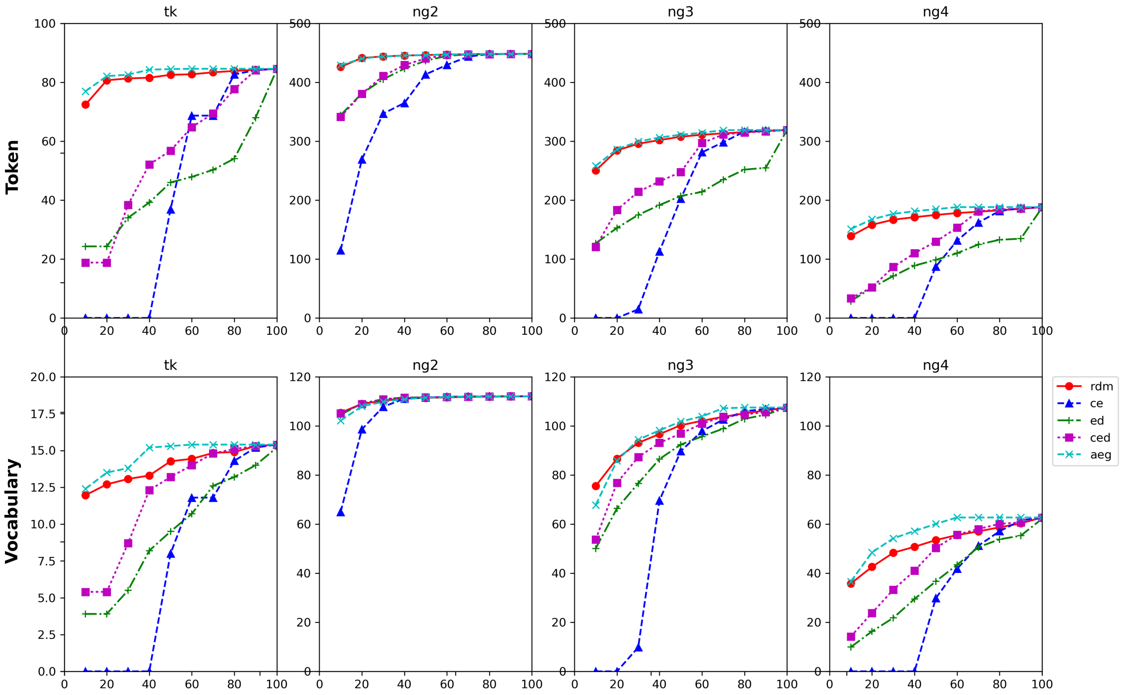


Figure 5.3: **The average number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Cornell as the D_t and the other four group universities as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

We also observed that AEG selection method in all features could not really help to increase the classification performances in this Cornell setup. The performances derived from that method estimation was frequently close to the performances produced from random selection. We found that selecting data by their *entropy gain* calculation could also lead to data miss-selection. Since *entropy gain* is the measurement of how much entropy is changed when the data is changed a little bit, then it could give the data which have low probabilities low entropy gain scores because their entropy changed could approach to 0. To give an illustration, let assume an URL in D_s consists of *tk* features such as words like *texas*, *washington* and *wisconsin.* Those tokens should be considered as the words which are not similar to data in *Cornell* dataset due to their existence probabilities in *Cornell* dataset possibly can be zero. However, according to AEG calculation (see Eq. 3.10), the URL which has those features can be considered as close to D_t only because it can have a lower AEG score. This is because its small feature probabilities in D_t can also produce lower entropy scores and adding those scores to D_t's feature entropy scores as the *entropy gain* estimation input can also derive a lower AEG score. Again, the assumption was also partially true because the entropy gain was also suitable to measure the closeness of data. The D_s data which had a significance frequency in D_t could also contribute to a lower entropy gain as their high probabilities could also produce low entropy scores.

Useful feature

In this setup, *tk* seemed helpful to give entropy-based methods an opportunity to select data in D_s in order to lower error rates of random selection performances. The entropy-based methods were applicable to specifically seek out and retrieve the *tokens* as important keywords for a specific class. *N-grams* on the other hand, could boost random selection performances (without entropy-based method assistances) by feeding the classifiers with a huge amount of data. The classifiers would then be able to classify correctly not only some of "student" URLs but also some of the other three class URLs i.e. *course, project* and *faculty* at the small percentages of D_s. *N-grams* reduced the vocabulary sizes that led to many similarities between D_s and D_t data as shown in Figure 5.3. 2-grams, for example, lowered the number of vocabularies in Web-KB dataset from ± 3000 to only ± 600 approaching l^n where l is the number of letters in the english alphabet (see Figure E.1 in Appendix E).

The other group university setups produced the same performance patterns for the above analysis (see appendix A for the complete proofs). In the next section we will present the result of classification performance using URL from Radboud University Nijmegen (RUN) as the D_t and URLs from Web-KB as the D_s with the same feature and selection method settings.

5.2 Classification performance on Radboud University Nijmegen (RUN) URLs

Since the 100-random RUN URLs were unlabeled (see Appendix G), we measured the prediction results from both random selection and entropy-based selection methods by estimating their prediction performance on the ground truth data. We used Cohen's kappa agreement to see whether both prediction results make sense or not.

Cross Validation	
Total URLs (both judges agree)	64
KFolds CV	8
Cohen Kappa Score*	0.62

Table 5.5: **The performances of Cross Validation on RUN URLs**. * indicates the average performances from all features (tk, ng2, ng3 and ng4) using Linear SVM classifier.

The empirical results are summarized in Table 5.6. All scores are in Cohen's kappa aggreement scores. To make it clear to observe, again we plotted the table into graphs as shown in Figure 5.4. We see that the average of the acceptance results are around 0.50 which is not really far from cross validation score (see Table 5.5). This 0.50 score are considered as a moderate aggreement score [28]. These all moderate scores were caused by the differences of aggreements of both judges as shown briefly in Table 5.7.

		RDM				*CE*				*ED*				*CED*				*AEG*			
Cl.	%Train	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4
NB	10%	0.52	0.53	0.52	0.52	0.40	0.45	0.40	0.40	0.40	0.40	0.44	0.44	0.40	0.58	0.42	0.42	0.53	0.52	0.45	0.51
	20%	0.52	0.49	0.47	0.52	0.40	0.47	0.50	0.40	0.40	0.44	0.47	0.44	0.40	0.52	0.42	0.39	0.53	0.56	0.43	0.50
	30%	0.52	0.51	0.47	0.52	0.40	0.44	0.46	0.40	0.39	0.44	0.46	0.42	0.39	0.51	0.46	0.42	0.53	0.51	0.47	0.50
	40%	0.53	0.52	0.46	0.52	0.40	0.47	0.49	0.40	0.40	0.41	0.49	0.52	0.40	0.51	0.42	0.40	0.53	0.51	0.46	0.52
	50%	0.53	0.49	0.50	0.50	0.40	0.50	0.57	0.40	0.50	0.41	0.53	0.52	0.50	0.46	0.45	0.49	0.50	0.51	0.48	0.49
	60%	0.53	0.51	0.50	0.49	0.40	0.51	0.59	0.40	0.53	0.42	0.54	0.50	0.50	0.49	0.49	0.46	0.53	0.50	0.49	0.51
	70%	0.53	0.50	0.48	0.51	0.40	0.46	0.54	0.44	0.53	0.45	0.47	0.47	0.52	0.46	0.45	0.50	0.53	0.52	0.48	0.51
	80%	0.53	0.50	0.46	0.51	0.40	0.48	0.55	0.52	0.53	0.47	0.42	0.49	0.53	0.47	0.41	0.50	0.53	0.49	0.47	0.51
	90%	0.53	0.50	0.46	0.49	0.45	0.51	0.46	0.54	0.53	0.51	0.46	0.49	0.53	0.50	0.43	0.51	0.53	0.49	0.48	0.51
	100%	0.53	0.50	0.46	0.49	0.53	0.50	0.48	0.49	0.53	0.50	0.48	0.49	0.53	0.50	0.48	0.49	0.53	0.50	0.48	0.49
SVM	10%	0.51	0.49	0.56	0.54	0.40	0.46	0.40	0.40	0.40	0.43	0.47	0.46	0.38	0.55	0.45	0.42	0.53	0.51	0.48	0.51
	20%	0.53	0.43	0.55	0.54	0.40	0.46	0.54	0.40	0.40	0.49	0.53	0.49	0.38	0.49	0.43	0.40	0.52	0.42	0.51	0.58
	30%	0.49	0.40	0.57	0.46	0.40	0.51	0.44	0.40	0.39	0.41	0.61	0.46	0.39	0.48	0.49	0.44	0.52	0.43	0.64	0.51
	40%	0.47	0.43	0.60	0.48	0.40	0.45	0.68	0.40	0.51	0.43	0.57	0.50	0.40	0.40	0.60	0.44	0.50	0.42	0.62	0.49
	50%	0.47	0.41	0.53	0.51	0.40	0.44	0.57	0.40	0.50	0.45	0.71	0.43	0.50	0.40	0.68	0.44	0.49	0.41	0.66	0.44
	60%	0.49	0.44	0.51	0.48	0.40	0.40	0.55	0.40	0.47	0.42	0.66	0.42	0.51	0.42	0.64	0.54	0.49	0.45	0.64	0.47
	70%	0.49	0.41	0.57	0.50	0.40	0.45	0.56	0.41	0.47	0.36	0.63	0.45	0.53	0.43	0.66	0.54	0.49	0.45	0.66	0.47
	80%	0.50	0.41	0.61	0.49	0.40	0.41	0.59	0.49	0.50	0.37	0.57	0.49	0.50	0.40	0.60	0.51	0.49	0.45	0.64	0.48
	90%	0.49	0.38	0.53	0.50	0.44	0.40	0.59	0.49	0.49	0.40	0.52	0.48	0.49	0.39	0.61	0.49	0.49	0.41	0.56	0.48
	100%	0.49	0.35	0.56	0.49	0.49	0.36	0.56	0.49	0.49	0.35	0.56	0.49	0.49	0.35	0.56	0.49	0.49	0.35	0.56	0.49

Table 5.6: **URL classification acceptance**: Tested on the Radboud 100-random-URLs and trained on Web-KB URLs. All values are in Cohen's kappa scores.

As shown in Figure 5.4, prediction results derived from CE selection estimation obtained almost constantly poor acceptances in all features especially in the beginning of D_s percentages. We believed that indicates our assumption in the first evaluation on Web-KB dataset also occured in this setup. We then performed smoothing and normalization for CE estimation. The prediction performances as shown in Table 5.5 seems close to our analysis in the first evaluation on Web-KB dataset.

We found that prediction results of ED and CED selection methods lead to inaccurate classification results for almost all features, especially at the small percentage of D_s. We interpreted this finding as an indication that ED and CED again performed miss-selection of similar data on D_s data as shown in figure 5.6. They only choosed correctly a small number of similar tokens and vocabularies in D_s at the small percentages of D_s.

URL	**Judge 1**	**Judge 2**
http://www.ru.nl/english/education/masters/historical-literary/	course	project
http://www.ru.nl/english/education/masters/pathobiology/our-approach-to-this/	course	project
http://www.ru.nl/english/@928631/grant-worth-22-9/	project	other
http://www.ru.nl/english/about-us/our-university/history/prime-minsters/	faculty	other
http://www.ru.nl/english/research/radboud/themes/astronomy/vm/alma-world-largest/	project	course
http://www.ru.nl/english/education/masters/philosophy-social/contact/	faculty	course
http://www.ru.nl/english/education/masters/computing-foundation	course	course
http://www.ru.nl/english/education/masters_student/financial_matters/student_budget_and/	student	student
http://www.ru.nl/english/research/radboud/themes/brain-cognition/vm/news-brain-cognition/?	project	project
http://www.ru.nl/english/@936463/prof-nico/?	faculty	faculty

Table 5.7: **A chunk of the similarity and difference of agreements on RUN URLs :** The table shows the similarity and difference of class aggreements on RUN URLs between two observers.

AEG in this setup seems to be similar to the first evaluation analysis even though its results received slighlty more acceptances in *tk* as feature than random selection predictions in small percentage of D_s (see Figure 5.4 or table 5.6). That was because it helped to collect more token and vocabulary similarities in D_s than random

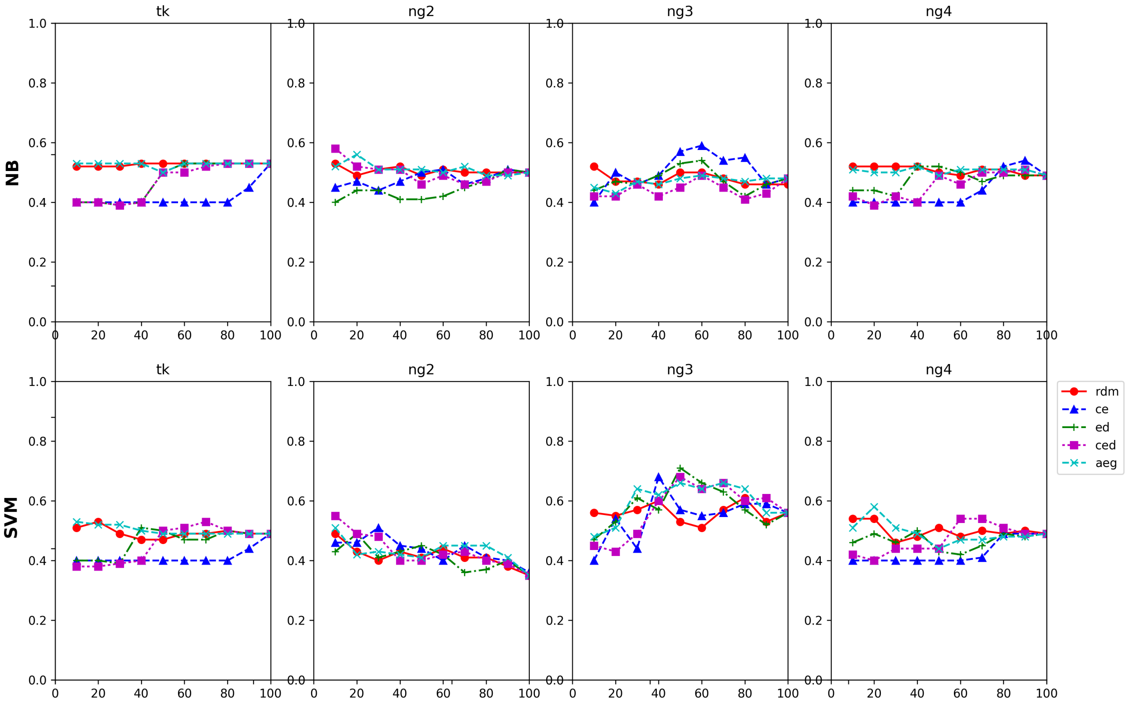


Figure 5.4: **The classification performance over the size of D_s with Radboud as the D_t:** The graph plots the performance score that is made up by the fraction of D_s. X-axis and Y-axis represents D_s percentage and Cohen's kappa score respectively.

selection at those fractions of D_s as shown in Figure 5.6. However, in general, its prediction results are about the same (low) level as these obtained with random selection.

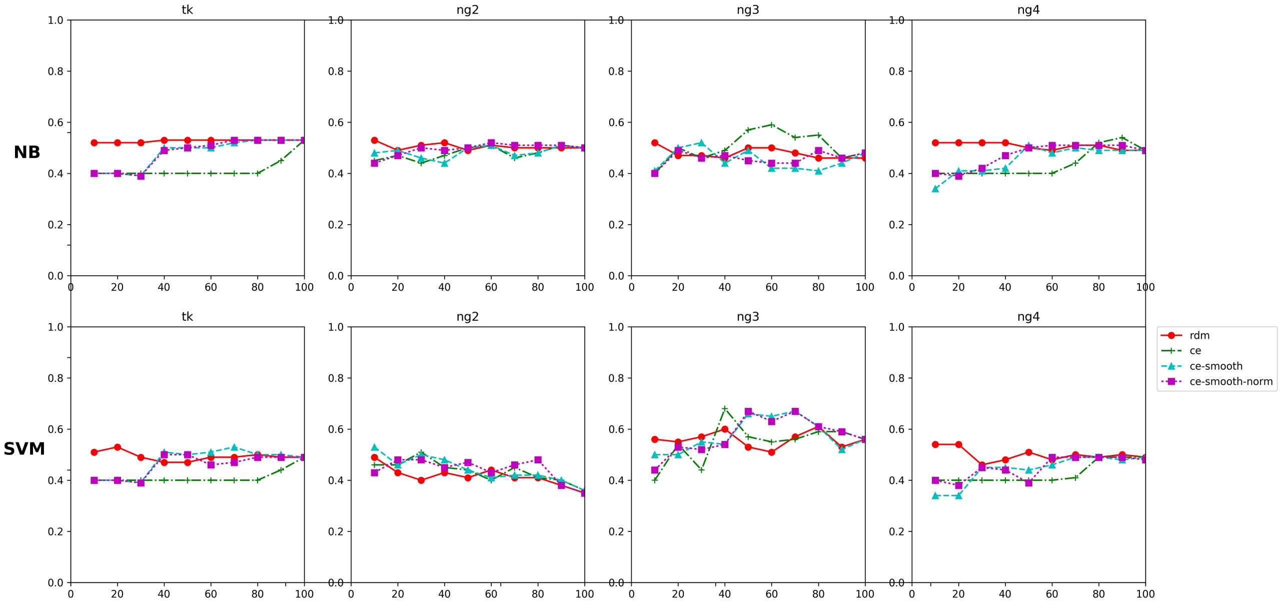


Figure 5.5: **The difference of classification performance using different type of CE over the size of D_s with RUN as the D_t:** The graph shows the performance differences over CE scores in **smoothing** and **normalized** CE. X-axis and Y-axis represents D_s percentage and Cohen's kappa score respectively.

In the first evaluation on Web-KB dataset, we achieved generally that *tk* as the most useful feature for entropy-based selections to lower the error rates of random selection. In this second evaluation, we found that random selection received relatively higher acceptances at *tk* as the feature (see Figure 5.4). As shown in

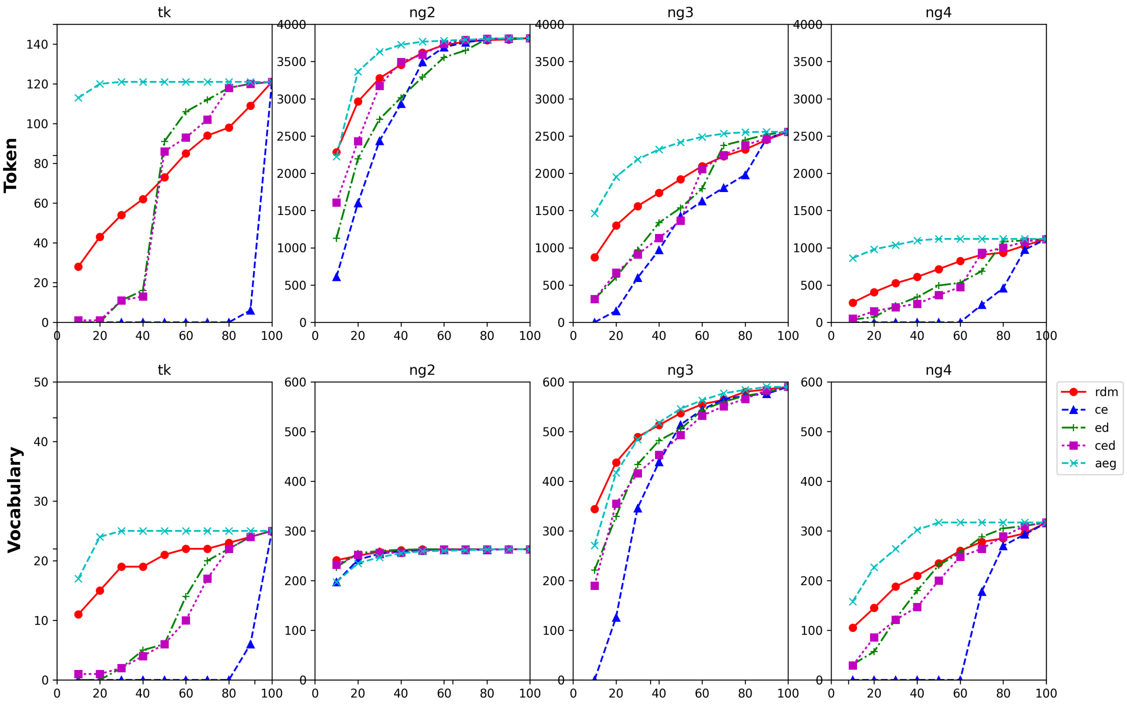


Figure 5.6: **The number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Radboud as the D_t and Web-KB as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

Table 5.8, we identified that the top ten keywords in RUN dataset such as *education* and *research* appeared also in top ten keywords Web-KB dataset (see appendix D). The top keyword intersection was the significant source that led random selection to receiving relatively good acceptances from both observers. For that reason, entropy-based methods were difficult to exceed those random selection acceptance scores using *tk* as feature.

tk	**ng2**	**ng3**	**ng4**
education:75, masters:75, programmes:19, radboud:18, master:16, vm:16, research:10, news:9, and:8, science:8	ti:127, on:155, io:108, at:105, er:101, st:97, ca:92, ma:90, te:83, ed:81	ion:98, tio:95, ati:92, cat:77, duc:76, edu:75, ter:75, uca:75, ste:69, ast:68	tion:95, atio:87, cati:76, duca:75, educ:75, ucat:75, ster:69, aste:66, mast:66, ters:51

Table 5.8: **Top Ten Keywords in RUN dataset:** The table shows top ten *tokens* and *n-grams* along with their frequencies.

Chapter 6

Conclusion and Discussion

6.1 Conclusion

In this section, the conclusion is constructed to answer the research questions. We will describe directly the answers supported by the analysis from previous chapters.

- **Can entropy-based adaptation methods help D_s data selection to increase URL classification performances using labeled out-of-domain?**

  Entropy-based selection methods in some circumstances could increase the URL classification performances of random D_s selection. In some cases, they were able to select labeled out-of-domain URLs in D_s by estimating their feature probabilities in D_t and hence the URLs could contribute to high performances. However, *cross entropy* method needs to be evaluated deeply in regard to their data prioritizing problems. *Entropy difference*, *cross entropy difference* and *average entropy gain* calculations also need to be evaluated regarding their similarity measurement issues.

- **What is the most useful feature representative of URL that allows entropy-based adaptation methods to reduce error rates of URL classification using labeled out-of-domain URLs?**

  Token seemed as the right feature to give entropy-based selection methods a chance to increase the performance of random selection when the D_s contained sparse relevant features to determine D_t's URL class. In other words, the entropy-based selection method were only useful when the D_s consisted of *token* features in which their existence were insignificant in small percentage of D_s selections. Entropy-based selection methods could seek and select specifically the labeled out-of-domain URLs in D_s by estimating their *token* features and discriminate the URLs either as the important D_s URLs or not important D_s URLs.

- **What is the most effective entropy-based adaptation method in selecting D_s for URL classification?**

  Cross entropy, entropy difference, cross entropy difference and *average entropy gain* all measured the importantness of labeled out-of-domain URLs with respect to in-domain URLs by estimating their feature probabilities. Considering D_s and D_t are assumed from different domains, a majority of data in D_s mostly would have probabilities around 0 - 0.5 with respect to D_t. Since the probabilites of D_s data in D_t were impossible to reach 1, *cross entropy* is considered as the most effective method to calculate data based on their probabilities both in D_s and D_t. It can simply select and prioritize linearly D_s data by putting low entropy scores on the data whose $P_s(x)$ are low and $P_t(x)$ are high and high entropy scores for the data whose $P_s(x)$ are high approaching 0.5 and $P_t(x)$ are low even though it should deal with $P_t(x) = 0$. The other three methods could incorrectly select the closest data between D_s and D_t because their *difference* and *gain* measurements could give the unlike data the same score as the similar data scores.

6.2 Discussion and future work

Entropy-based selections are supposed to be an easy and competitive approach to select closest data in domain adaptation area [1]. Previous reports [12, 13, 14] demonstrated that the data D_s which had lowest entropy-based scores indicate that the data were close enough to the target data D_t. The entropy-based studies reported that the task performances were improved significantly while adapting different domain data as the training data by estimating their entropy values. Such studies may have observed the large amount of data while missing the sparseness of small data.

Since the idea of data selection explains that the small high quality data is better than big noisy data, we expect that machine learning classifiers combined with entropy-based methods are smarter than a classifier built by assigning classes arbitrarily in using small dataset. In fact, the performances we got could not really fulfill the idea demanded. The performances from *cross entropy* which were almost similar to majority class classification performances and the performances from other entropy-based methods using *tk* as feature were also close to random classification performances at small partitions of D_s (see Table 6.1, Table 5.1 and Appendix A).

University	Random Classifier					Majority Class Classifier				
	$E[F1]$				$E[macro-ave.F1]$	$E[F1]$				$E[macro-ave.F1]$
	(co.)	(fa.)	(pr.)	(st.)		(co.)	(fa.)	(pr.)	(st.)	
Cornell	21.9	18.7	13.0	34.7	24.9	0.0	0.0	0.0	72.3	18.1
Texas	28.9	16.4	12.3	33.2	25.4	0.0	0.0	0.0	66.1	16.5
Washington	26.2	17.6	12.2	33.5	24.9	0.0	0.0	0.0	67.2	16.8
Wisconsin	18.8	21.1	12.0	35.0	25.0	0.0	0.0	0.0	73.9	18.5

Table 6.1: **The expected F1 and macro-average F1 score for Random and Majority Class Classifier**. The abbreviations co., fa., pr. and st. mean class *course, faculty, project* and *student* respectively.

Our finding shows that prioritizing only to the data with low entropy scores was considered problematic approach in data selection for URL classification using labeled out-of-domain training data. Since we used relatively small datasets and both D_s and D_t were not the same domains, disparity of features between both datasets could be high. This inequality of data may lead to many data in D_s having low or even zero entropy scores with respect to D_t. The problem could be more complicated since the words or *n-gram* tokens in a URL may vary. The words sometimes had different entropy scores or even their scores contrasted each others. Deciding to select or to not select a URL that has those words in that situation was not an easy work.

Although our analysis seems to fit the data presented in the evaluation, it solely represents one interpretation of our results. Different experimental procedures, including different scoring and weighting for each features in a URL, would be required to reveal the underlying problem on why entropy-based measurements only did not work as expected. In the future, we would like to see if other weighting score estimations could be implemented to reduce the importantness of “bad”features to improve the performances, potentially by using estimations that increase the importantness of other “good”features in a URL at the same time.

Bibliography

[1] Q. Li, "Literature survey: Domain adaptation algorithms for natural language processing," *Department of Computer Science The Graduate Center, The City University of New York*.

[2] S. Verberne, B. Arends, W. Kraaij, and A. de Vries, "Longitudinal navigation log data on a large web domain," in *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, (New York, NY, USA), pp. 697–700, ACM, 2016.

[3] B. Plank, *Domain Adaptation for parsing*. PhD thesis, University of Groningen, 2011.

[4] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[5] S. Chakrabarti, M. van den Berg, and B. Dom, "Focused crawling: A new approach to topic-specific web resource discovery," *Comput. Netw.*, vol. 31, pp. 1623–1640, May 1999.

[6] J. Zhu, J. Hong, and J. G. Hughes, "Using markov models for web site link prediction," in *Proceedings of the Thirteenth ACM Conference on Hypertext and Hypermedia*, HYPERTEXT '02, (New York, NY, USA), pp. 169–170, ACM, 2002.

[7] F. Sebastiani, "Machine learning in automated text categorization," *ACM Comput. Surv.*, vol. 34, pp. 1–47, Mar. 2002.

[8] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *Journal of Statistical Planning and Inference*, vol. 90, no. 2, pp. 227 – 244, 2000.

[9] J. Jiang, "Domain adaptation in natural language processing," *P.hD Dissertation*, 200.

[10] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, pp. 1345–1359, Oct 2010.

[11] J. R. Finkel and C. D. Manning, "Hierarchical bayesian domain adaptation," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, (Stroudsburg, PA, USA), pp. 602–610, Association for Computational Linguistics, 2009.

[12] Y. Song, P. Klassen, F. Xia, and C. Kit, "Entropy-based training data selection for domain adaptation," in *Proceedings of COLING 2012 : Posters*, (Mumbai, India), pp. 1191–1200, COLING 2012, 2012.

[13] A. Axelrod, X. He, and J. Gao, “Domain adaptation via pseudo in-domain data selection,” in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP ’11, (Stroudsburg, PA, USA), pp. 355–362, Association for Computational Linguistics, 2011.

[14] R. C. Moore and W. Lewis, “Intelligent selection of language model training data,” in *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort ’10, (Stroudsburg, PA, USA), pp. 220–224, Association for Computational Linguistics, 2010.

[15] X. Qi and B. D. Davison, “Web page classification: Features and algorithms,” *ACM Comput. Surv.*, vol. 41, pp. 12:1–12:31, Feb. 2009.

[16] E. Baykan, M. Henzinger, L. Marian, and I. Weber, “A comprehensive study of features and algorithms for url-based topic classification,” *ACM Trans. Web*, vol. 5, pp. 15:1–15:29, July 2011.

[17] T. A. S. Foundation, “Log files.” `https://httpd.apache.org/docs/1.3/logs.html`. [Online; accessed 19-July-2016].

[18] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki, “Link prediction using supervised learning,” in *In Proc. of SDM 06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

[19] Z. Huang, X. Li, and H. Chen, “Link prediction approach to collaborative filtering,” in *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL ’05, (New York, NY, USA), pp. 141–142, ACM, 2005.

[20] G. J. F. Jones and Q. Li, *Focused Browsing: Providing Topical Feedback for Link Selection in Hypertext Browsing*, pp. 700–704. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.

[21] C. D. Manning, P. Raghavan, and H. Schutze, *Introduction to Information Retrieval.* Cambridge University Press, 2008.

[22] M.-Y. Kan, “Web page classification without the web page,” in *Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers &Amp; Posters*, WWW Alt. ’04, (New York, NY, USA), pp. 262–263, ACM, 2004.

[23] E. Baykan, M. Henzinger, L. Marian, and I. Weber, “Purely url-based topic classification,” in *Proceedings of the 18th International Conference on World Wide Web*, WWW ’09, (New York, NY, USA), pp. 1109–1110, ACM, 2009.

[24] T. Carter, “An introduction to information theory and entropy,” 2000.

[25] C. T. L. Group, “The 4 universities datasets.” `http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/`. [Online; accessed 19-July-2016].

[26] A. Tomović and P. Janičić, *A Variant of N-Gram Based Language Classification*, pp. 410–421. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[28] A. J. Vierra and J. M.Garret, “Understanding interobserver agreement: the kappa statistic.,” *Family Medicine*, vol. 37, no. 5, pp. 360–363, 2005.

Appendices

Appendix A

Classification Performances

		RDM				CE				ED				CED				AEG			
Cl.	%Train	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4
NB	10%	36.91	37.93	27.42	28.98	18.51	22.02	18.51	18.51	18.51	26.86	25.93	19.30	22.55	30.80	24.41	26.27	**43.14**	31.11	33.74	35.73
	20%	30.76	**40.77**	27.73	28.71	18.51	23.89	18.51	18.51	18.51	36.91	32.89	18.43	21.33	34.82	28.49	26.78	35.47	34.84	30.58	32.72
	30%	28.62	**41.24**	27.79	28.58	18.51	25.07	21.87	18.51	30.46	35.79	29.89	30.41	36.19	36.10	31.83	30.75	31.44	32.98	28.41	30.39
	40%	28.36	39.69	27.72	28.77	18.51	25.98	27.30	18.51	**39.83***	37.37	32.21	37.25	28.76	39.00	34.38	32.06	30.20	37.80	27.72	29.55
	50%	30.17	41.37	29.90	28.98	26.39	22.05	26.41	31.48	**42.18***	41.10	33.75	41.47*	27.67	40.09	33.69	32.26	30.36	37.59	28.11	28.23
	60%	29.84	40.46	29.66	28.17	27.22	28.71	27.51	31.48	**41.96****	41.91	29.01	38.12	28.43	41.49	31.68	30.18	30.36	36.56	27.39	29.58
	70%	29.87	41.10	28.19	28.11	27.86	36.13	28.01	31.48	33.50	40.78	28.97	34.78	28.89	**41.33**	29.15	29.56	30.80	38.25	26.43	28.81
	80%	30.51	40.75	28.04	28.02	27.86	36.42	27.43	30.59	**43.64***	39.16	27.75	26.78	29.15	40.42	29.69	29.26	30.80	38.48	27.64	28.81
	90%	30.65	**41.80**	28.22	27.97	30.20	40.20	26.48	28.55	32.89	40.47	29.62	29.02	29.72	40.53	30.14	29.37	30.91	40.38	28.13	28.97
	100%	30.91	**40.91**	28.88	27.37	30.91	40.46	28.86	27.66	30.91	40.38	30.60	27.66	30.91	40.43	28.92	27.66	30.91	40.46	28.86	27.66
SVM	10%	43.29	38.11	33.81	41.65	18.51	20.16	18.51	18.51	21.85	39.48	31.74	24.76	26.46	41.09	32.76	34.37	38.69	34.34	39.49	**45.25**
	20%	46.01	46.16	36.81	43.94	18.51	18.48	18.51	18.51	21.85	40.21	37.20	28.78	26.46	42.45	36.26	41.24	44.13	43.33	**46.64***	39.44
	30%	46.08	47.07	38.30	45.81	18.51	31.79	21.29	18.51	34.84	47.48	37.16	38.68	46.80	45.86	46.79	**53.62**	46.00	43.79	50.22*	47.32
	40%	46.83	**49.72**	42.67	47.53	18.51	31.68	24.76	18.51	42.75	45.37	31.83	46.09	41.13	45.48	44.93	47.03	48.07	45.82	48.87	45.57
	50%	50.92	50.65	47.05	49.55	47.64	28.34	44.09	51.01	40.32	45.73	27.52	48.20	49.83	47.12	49.81	53.87	**54.23**	48.79	49.19	45.64
	60%	51.03	49.54	45.48	49.29	47.89	45.28	52.11	**54.43**	39.03	42.43	25.08	39.25	53.95	50.36	50.87	51.99	47.81	50.16	50.66	46.62
	70%	51.19	48.84	47.74	46.97	47.89	45.17	53.15	**54.19**	41.42	41.78	31.51	40.11	50.80	48.27	50.48	52.78	53.59	48.73	50.97	48.81
	80%	51.23	47.98	48.15	48.42	47.73	46.03	52.37	51.81	46.48	50.17	37.07	38.37	**55.58**	47.20	49.81	51.07	47.81	46.02	51.47	48.81
	90%	48.36	47.13	49.03	49.71	47.87	47.40	49.30	48.59	46.55	**52.76**	38.24	47.39	46.99	46.38	48.78	51.13	47.81	46.89	49.93	49.07
	100%	47.81	47.59	48.63	49.71	47.81	47.39	48.57	49.71	47.81	47.39	49.49	49.62	47.81	47.33	48.57	49.71	47.81	47.39	48.57	**49.71**

Table A.1: **URL classification performance** : Tested on Texas and trained on the other universities. * and ** indicate significance at 0.05 and 0.01 respectively. The highest score in each row is in bold.

		RDM				CE				ED				CED				AEG			
Cl.	%Train	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4	tk	ng2	ng3	ng4
NB	10%	77.94	71.08	81.25	**81.66**	16.54	17.91	16.54	16.54	20.73	37.14	27.54	19.64	16.40	34.66	52.46	30.61	54.14	67.92	63.03	58.11
	20%	68.96	64.74	79.83	**80.88**	16.54	20.87	16.54	16.54	18.63	44.80	54.11	24.41	16.40	44.18	59.67	38.68	71.67	73.24	71.40	66.84
	30%	71.18	72.96	79.36	**80.76**	16.54	20.18	17.95	16.54	34.56	39.96	63.80	37.70	24.90	53.82	74.85	44.61	73.64	70.20	76.33	75.35
	40%	60.73	73.17	79.94	81.32	16.54	36.99	39.94	16.37	78.70**	43.85	63.76	63.48	71.45	57.08	**82.39**	73.01	63.43	75.23	77.51	77.33
	50%	60.83	73.25	79.71	80.19	73.39*	71.46	81.57	82.16	82.16**	48.77	74.27	72.66	81.17**	59.74	83.13	**83.48**	53.78	74.37	79.54	77.61
	60%	60.94	73.35	79.02	79.97	82.56**	72.17	81.36	82.16	82.55**	44.97	78.46	76.37	**83.42****	67.83	83.26	82.79	71.79	75.37	79.49	77.98
	70%	71.81	73.38	79.21	80.34	**83.48***	72.05	82.16	82.16	81.47	57.97	80.53	78.60	81.40*	71.84	80.96	82.15	83.42*	74.54	79.27	80.20
	80%	71.81	74.07	77.81	80.12	**83.48***	75.98	79.14	82.26	81.57	51.09	78.43	77.76	82.32*	71.87	81.35	82.08	83.48*	72.26	79.27	81.09
	90%	83.48	74.78	78.35	81.65	**83.48**	78.43	81.65	81.91	82.44	61.00	79.73	78.15	83.48	72.76	80.69	81.33	83.48	74.04	79.27	81.09
	100%	83.48	74.80	79.59	81.17	**83.48**	74.62	79.86	81.33	83.48	74.26	79.59	81.09	83.48	75.38	79.59	81.09	83.48	74.62	79.59	81.09
SVM	10%	81.41	71.48	80.66	81.45	16.54	18.02	16.54	16.54	34.19	44.12	56.83	28.38	16.54	47.85	71.27	44.16	82.41	66.65	78.56	**82.83**
	20%	71.42	59.70	79.05	81.23	16.54	19.63	16.54	16.54	34.19	46.22	64.22	36.70	16.54	44.44	74.99	52.26	**81.98**	76.09**	80.79	81.43
	30%	71.33	73.66	78.72	80.50	16.54	26.21	18.35	16.54	68.31	36.27	65.79	51.20	34.44	49.24	77.34	57.01	**82.62***	74.78	80.87	80.07
	40%	71.13	74.60	79.45	79.76	16.54	62.72	40.76	16.76	81.45	43.92	67.82	59.63	76.09	56.00	81.83	77.89	**82.62***	75.66	79.06	81.10
	50%	60.78	73.48	77.20	78.51	74.90*	75.48	79.60	80.56	82.16**	58.32	73.02	65.95	82.69**	63.01	**83.11**	82.34	82.35**	75.53	78.66	80.61
	60%	71.56	74.59	76.61	78.30	82.42*	76.89	78.88	80.56	76.02	48.07	73.75	67.55	**82.69***	62.60	81.40	81.48	82.35*	76.08	79.03	80.31
	70%	**82.49**	73.49	76.40	79.03	82.42	75.30	80.94	80.93	72.55	42.66	64.31	72.56	82.42	57.68	81.62	79.75	82.35	75.06	77.03	80.24
	80%	82.14	72.55	75.57	78.40	81.86	76.05	74.48	76.32	69.68	44.66	71.62	76.79	**82.36**	61.83	79.88	80.32	82.35	74.16	75.23	79.94
	90%	**82.85**	73.83	77.14	79.10	81.37	76.79	75.10	77.77	82.35	54.46	74.60	76.56	82.15	73.17	77.48	79.49	82.35	77.46	75.81	79.94
	100%	82.35	77.05	76.65	78.67	82.35	76.82	76.19	78.45	82.35	76.82	76.19	78.45	**82.35**	77.17	76.54	78.75	82.35	76.82	76.19	78.45

Table A.2: **URL classification performance**: Tested on Washington and trained on the other universities. * and ** indicate significance at 0.05 and 0.01 respectively. The highest score in each row is in bold.

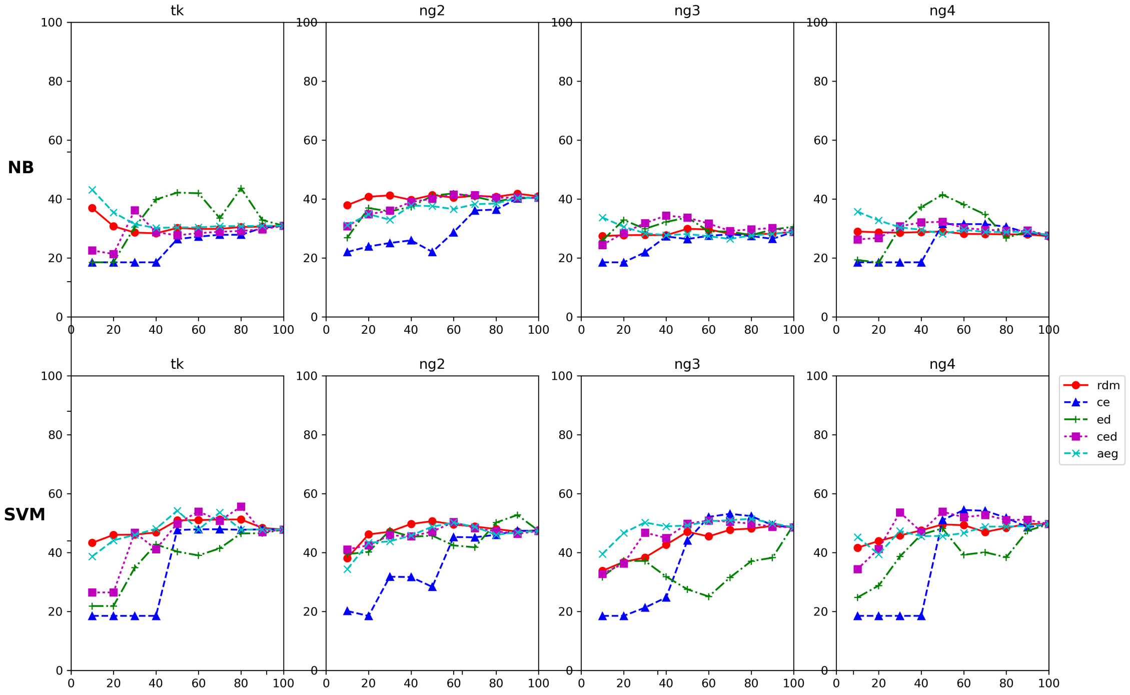


Figure A.1: **The classification performance over the size of D_s with Texas as D_t**: The graph plots the performance score that is made up by the fraction of D_s. X-axis and Y-axis represents D_s percentage and F-1 score domain respectively. The first and the second graph rows are the performances using Naive Bayes and SVM as the classifiers respectively.

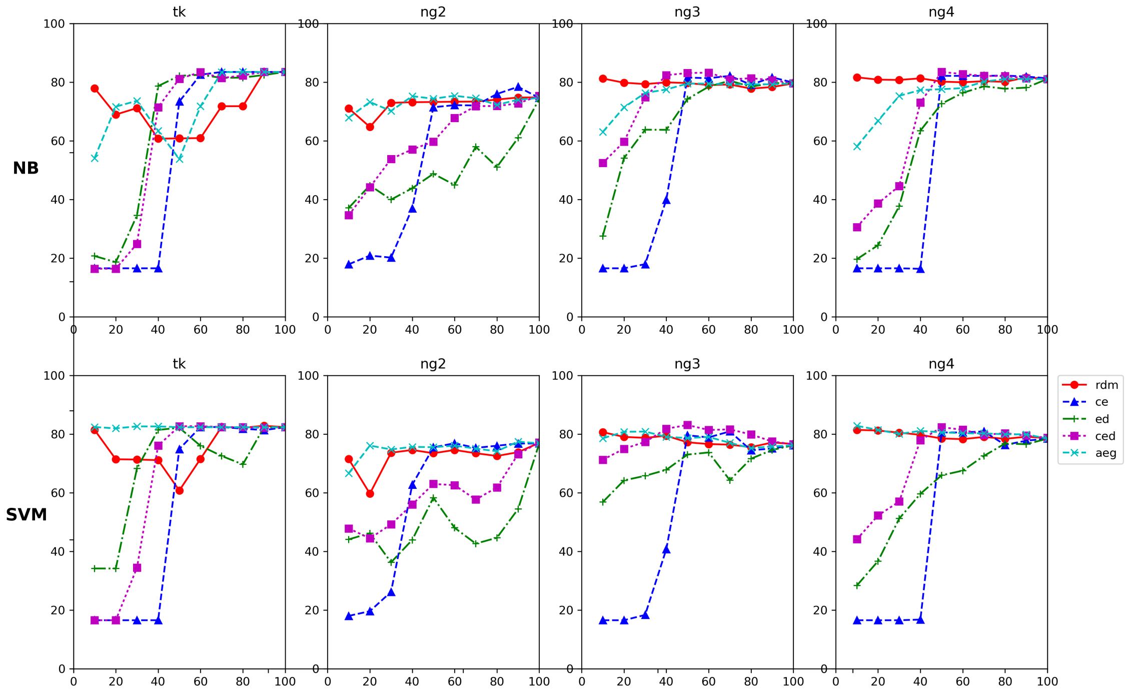


Figure A.2: **The classification performance over the size of D_s with Washington as D_t**: The graph plots the performance score that is made up by the fraction of D_s. X-axis and Y-axis represents D_s percentage and F-1 score domain respectively. The first and the second graph rows are the performances using Naive Bayes and SVM as the classifiers respectively.

Cl.	%Train	RDM tk	RDM ng2	RDM ng3	RDM ng4	CE tk	CE ng2	CE ng3	CE ng4	ED tk	ED ng2	ED ng3	ED ng4	CED tk	CED ng2	CED ng3	CED ng4	AEG tk	AEG ng2	AEG ng3	AEG ng4
NB	10%	45.05	45.40	**51.64**	49.84	16.82	21.13	16.82	16.82	14.69	44.81	29.73	12.69	17.72	34.17	28.65	21.11	18.04	36.19	30.39	27.32
	20%	44.81	48.41	54.06	**54.30**	16.82	33.77	16.82	16.82	14.77	44.67	37.36	10.85	17.72	35.26	31.30	26.62	23.15	39.34	33.33	33.50
	30%	44.95	46.11	52.98	**55.14**	16.82	40.16	16.82	16.82	17.58	48.35	45.15	26.47	31.27	36.40	49.75	24.56	24.51	41.51	36.83	32.92
	40%	**51.94**	46.39	50.23	51.36	16.82	43.66	20.01	16.82	20.66	50.17	49.51	34.52	45.82	41.57	42.71	40.95	27.04	42.70	41.36	36.58
	50%	45.36	46.36	50.45	51.50	20.36	47.34	32.51	28.48	36.95	48.50	**53.21**	44.20	32.41	42.71	48.80	45.26	26.78	44.85	45.77	42.25
	60%	34.86	46.06	49.85	47.19	**55.59****	47.14	50.72	47.82	38.49	48.44	51.74	48.57	40.70	42.44	47.03	46.29	26.64	45.83	48.28	44.88
	70%	35.00	46.27	49.28	48.07	**59.29****	47.24	50.70	51.38	39.02	47.70	50.47	47.71	35.51	45.69	47.82	46.61	28.04	46.62	49.18	45.83
	80%	31.11	45.98	49.74	48.78	**55.59****	46.88	54.13	50.21	39.08	46.32	50.96	49.58	39.82	45.68	50.58	48.38	27.86	46.25	50.72	46.13
	90%	35.16	46.55	49.87	45.70	44.22**	46.72	54.30	**54.89**	39.11	46.59	50.93	51.55	35.98	45.54	50.86	50.07	27.41	46.18	50.54	46.34
	100%	39.24	45.74	50.37	47.97	39.25	45.76	50.37	**50.83**	39.25	45.59	50.20	47.43	39.25	45.91	50.37	47.96	39.22	45.73	50.37	49.11
SVM	10%	34.18	49.39	**50.93**	41.89	16.82	21.63	16.82	16.82	20.96	48.16	42.64	14.12	18.80	41.77	27.26	21.93	22.80	47.53	45.38	40.49
	20%	29.43	49.28	**51.13**	40.82	16.82	36.03	16.82	16.82	24.13	49.33	48.44	20.11	18.80	44.70	34.11	26.27	23.79	47.90	47.60	43.05
	30%	29.69	49.21	52.41	42.49	16.82	41.43	16.82	16.82	22.78	49.19	51.16	31.14	37.96	51.86	**55.94**	33.21	25.18	48.80	48.73	41.92
	40%	26.94	50.14	**54.36**	43.09	16.82	46.09	22.10	16.82	37.97	52.14	52.57	40.74	50.95**	52.20	52.51	44.48	26.69	48.46	53.39	46.20
	50%	27.06	50.55	53.21	44.27	20.36	49.17	41.23	32.13	37.55*	52.31	53.56	50.28	31.61	53.23	52.19	42.00	26.97	49.09	**61.02**	46.56
	60%	26.79	50.39	53.82	44.47	27.67	52.95	55.58	40.13	29.73	50.54	56.15	50.23	27.87	51.52	50.40	42.95	27.51	50.15	**57.92**	48.78
	70%	27.11	51.04	53.89	45.36	27.67	52.50	53.24	41.21	28.50	50.38	54.05	43.93	27.73	53.10	51.63	43.85	27.51	52.48	**57.06**	48.03
	80%	27.44	52.40	56.18	46.83	27.67	53.11	55.90	43.99	27.74	51.56	55.38	47.48	27.96	52.13	53.14	46.71	27.51	52.22	**57.21**	46.41
	90%	27.60	51.99	56.23	46.69	27.52	53.88	56.07	48.36	27.96	52.36	53.97	46.62	27.79	53.10	54.65	45.64	28.21	53.47	**56.35**	48.99
	100%	27.79	53.77	55.72	46.55	27.79	53.50	55.72	46.55	27.79	53.65	55.72	46.55	27.79	53.89	55.73	46.55	27.79	53.65	**55.72**	48.39

Table A.3: **URL classification performance** : Tested on Wisconsin and trained on the other universities. * and ** indicate significance at 0.05 and 0.01 respectively. The highest score in each row is in bold.

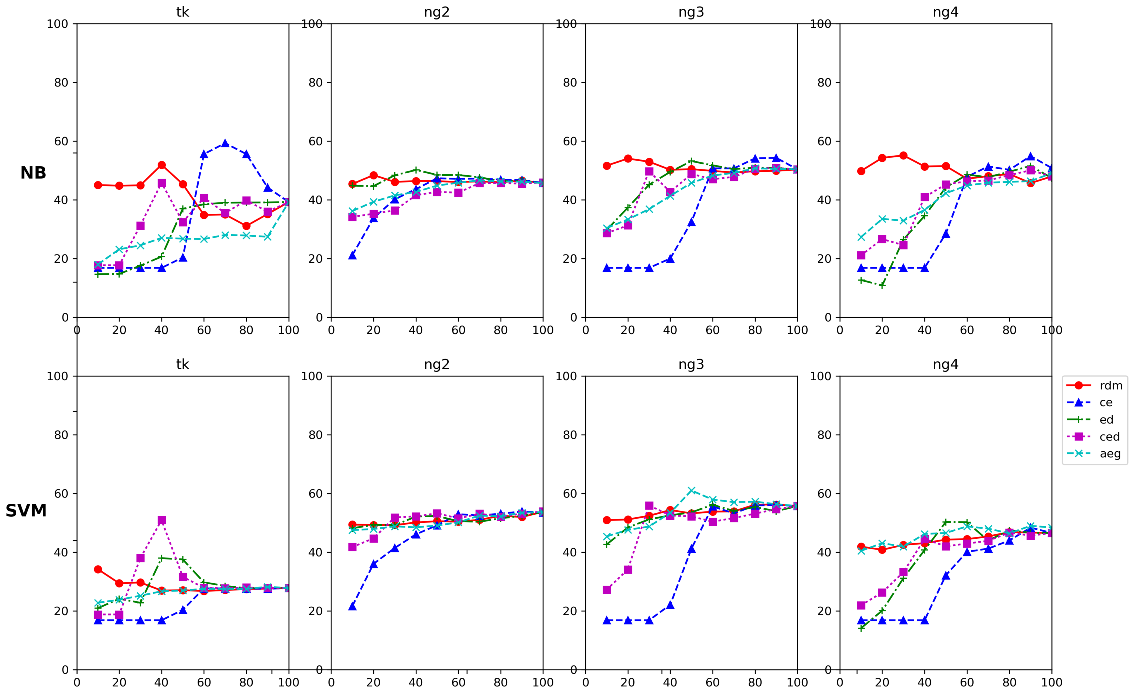


Figure A.3: **The classification performance over the size of D_s with Wisconsin as D_t**: The graph plots the performance score that is made up by the fraction of D_s. X-axis and Y-axis represents D_s percentage and F-1 score domain respectively. The first and the second graph rows are the performances using Naive Bayes and SVM as the classifiers respectively.

Appendix B

Training Size on Cross Entropy

	tk				ng2				ng3				ng4			
%Train	st.	fa.	pr.	co.	st.	fa.	pr.	co.	st.	fa.	pr.	co.	st.	fa.	.pr	co.
10%	209.9	123.9	42.7	19.5	167.6	119.1	73.7	35.6	206.1	130.0	44.8	15.1	208.8	124.7	42.3	20.2
20%	435.7	243.7	72.3	40.3	391.6	255.4	86.8	58.2	419.1	255.5	77.0	40.4	417.6	258.3	70.9	45.2
30%	676.0	340.7	100.4	70.9	593.4	394.4	107.3	92.9	626.1	376.7	124.2	61.0	643.9	359.1	105.2	79.8
40%	888.5	455.0	141.6	98.9	812.6	488.7	138.2	144.5	857.8	490.8	156.9	78.5	845.4	478.2	154.1	106.3
50%	1052.4	547.2	206.9	173.5	1031.9	570.1	173.6	204.4	1048.1	543.7	191.0	197.2	1082.5	550.8	168.9	177.8
60%	1135.1	636.7	252.2	352.0	1216.1	650.9	209.5	299.5	1164.6	626.1	260.8	324.5	1203.8	637.6	274.1	260.5
70%	1279.3	741.6	340.4	410.7	1309.6	734.3	275.8	452.3	1309.1	719.0	330.6	413.3	1341.4	726.1	339.1	365.4
80%	1416.9	847.6	414.7	488.8	1389.2	853.4	355.1	570.3	1390.6	820.9	383.9	572.6	1454.7	843.0	377.7	492.6
90%	1501.1	1021.2	467.1	574.6	1479.5	956.5	432.9	695.1	1496.4	922.9	459.0	685.7	1510.9	921.9	455.8	675.4
100%	1513.0	1089.0	482.0	876.0	1513.0	1088.7	482.0	876.3	1513.0	1089.0	482.0	876.0	1513.0	1089.0	482.0	876.0

Table B.1: **The average URL size selected per each class using CE**: Counted on Cornell. The abbreviation st., fa., pr. and co. mean class student, faculty, project and course respectively.

	tk				ng2				ng3				ng4			
%Train	st.	fa.	pr.	co.	st.	fa.	pr.	co.	st.	fa.	pr.	co.	st.	fa.	.pr	co.
10%	176.8	130.2	52.6	34.4	165.1	120.0	72.9	36.0	186.6	135.8	45.5	26.1	180.4	130.0	51.7	31.9
20%	373.9	258.0	89.7	66.4	378.8	265.5	91.5	52.2	383.9	262.4	78.9	62.8	376.5	260.4	88.0	63.1
30%	584.2	357.6	121.8	118.4	578.2	403.5	110.4	89.9	574.1	392.9	121.4	93.6	585.7	361.8	121.5	113.0
40%	767.0	485.7	167.3	156.0	771.8	519.3	134.4	150.5	776.8	511.2	159.8	128.2	771.9	489.8	166.9	147.4
50%	939.0	570.0	206.4	254.6	929.2	564.4	180.8	295.6	934.8	568.8	208.0	258.4	940.1	571.1	208.3	250.5
60%	1049.5	661.4	271.4	381.7	1065.7	654.6	247.2	396.5	1058.1	674.5	249.8	381.6	1060.7	669.1	273.1	361.1
70%	1188.7	798.0	345.5	425.8	1174.2	773.6	302.3	507.9	1219.8	803.2	298.6	436.4	1198.4	805.6	346.5	407.5
80%	1344.8	916.5	408.1	482.6	1299.7	891.4	374.4	586.5	1367.0	914.4	386.3	484.3	1352.9	924.1	408.3	466.7
90%	1481.3	1017.5	467.3	579.9	1419.5	981.9	433.8	710.8	1459.1	1007.0	450.6	629.3	1471.3	1017.9	459.7	597.1
100%	1493.0	1076.8	481.8	888.4	1493.0	1076.4	481.4	889.2	1492.9	1077.0	481.9	888.2	1492.7	1077.0	482.0	888.3

Table B.2: **The average URL size selected per each class using CE**: Counted on Texas. The abbreviation st., fa., pr. and co. mean class student, faculty, project and course respectively.

	tk				ng2				ng3				ng4			
%Train	st.	fa.	pr.	co.	st.	fa.	pr.	co.	st.	fa.	pr.	co.	st.	fa.	.pr	co.
10%	195.4	119.6	36.3	41.7	153.8	114.1	73.4	51.7	197.5	126.7	39.8	29.0	197.3	119.9	34.3	41.5
20%	404.6	241.2	61.3	78.9	378.1	253.0	90.4	64.5	400.7	249.4	68.6	67.3	400.9	244.9	57.8	82.4
30%	619.4	338.4	89.9	131.3	584.2	374.5	106.5	113.8	608.4	364.0	108.2	98.4	615.1	341.6	88.2	134.1
40%	819.0	448.7	127.6	176.7	771.9	478.8	137.9	183.4	823.3	469.9	130.4	148.4	810.7	456.1	128.4	176.8
50%	952.3	523.1	212.1	277.5	952.3	572.1	193.0	247.6	957.2	535.4	201.8	270.6	960.4	532.6	199.9	272.1
60%	1070.2	655.1	258.4	374.3	1093.2	650.7	248.8	365.3	1071.3	652.5	256.8	377.4	1077.0	653.8	256.4	370.8
70%	1204.7	744.9	310.7	490.7	1201.6	752.8	308.7	487.9	1214.9	731.2	308.9	496.0	1219.8	740.5	300.2	490.5
80%	1320.7	823.5	383.7	616.1	1317.4	852.3	384.4	589.9	1327.1	835.4	380.7	600.8	1347.0	839.0	361.2	596.8
90%	1448.3	897.1	441.6	750.0	1450.2	948.0	440.4	698.4	1460.8	906.6	444.7	724.9	1460.2	905.5	435.4	735.9
100%	1515.0	1086.6	480.8	847.6	1515.0	1086.4	480.9	847.7	1515.0	1085.8	480.8	848.4	1515.0	1085.9	480.3	848.8

Table B.3: **The average URL size selected per each class using CE**: Counted on Washington. The abbreviation st., fa., pr. and co. mean class student, faculty, project and course respectively.

	tk				ng2				ng3				ng4			
%Train	st.	fa.	pr.	co.	st.	fa.	pr.	co.	st.	fa.	pr.	co.	st.	fa.	.pr	co.
10%	195.3	115.9	43.5	33.3	164.9	118.0	69.4	35.7	202.0	119.3	40.6	26.1	198.2	119.9	40.3	29.6
20%	409.6	231.0	76.4	59.0	377.3	249.0	86.0	63.7	416.1	235.3	65.2	59.4	412.0	238.2	70.5	55.3
30%	634.4	332.0	99.3	98.3	586.5	380.6	102.9	94.0	629.5	338.0	100.5	96.0	640.7	335.6	93.6	94.1
40%	845.6	431.0	137.4	138.0	804.0	475.2	129.7	143.1	839.6	448.1	137.3	127.0	845.8	444.0	131.5	130.7
50%	1045.5	535.4	183.6	175.5	985.1	554.8	175.2	224.9	1023.6	534.7	183.3	198.4	1032.6	536.7	176.9	193.8
60%	1108.9	623.7	239.8	355.6	1113.7	654.2	238.2	321.9	1141.8	605.5	252.6	328.1	1145.7	607.3	249.0	326.0
70%	1230.8	732.1	332.0	421.1	1215.3	760.0	319.0	421.7	1249.7	715.1	338.9	412.3	1266.5	718.2	333.1	398.2
80%	1334.8	849.0	419.2	501.0	1342.8	846.9	382.2	532.1	1354.6	826.3	405.5	517.6	1384.9	808.6	414.1	496.4
90%	1400.2	970.8	458.0	663.0	1450.6	943.6	438.7	659.1	1457.2	908.0	460.0	666.8	1473.9	889.4	464.0	664.7
100%	1484.9	1077.2	477.0	840.9	1485.0	1077.8	477.0	840.2	1485.0	1077.6	476.9	840.5	1485.0	1078.2	477.0	839.8

Table B.4: **The average URL size selected per each class using CE**: Counted on Wisconsin. The abbreviation st., fa., pr. and co. mean class student, faculty, project and course respectively.

Appendix C

The Performances of Different Type of Cross Entropy

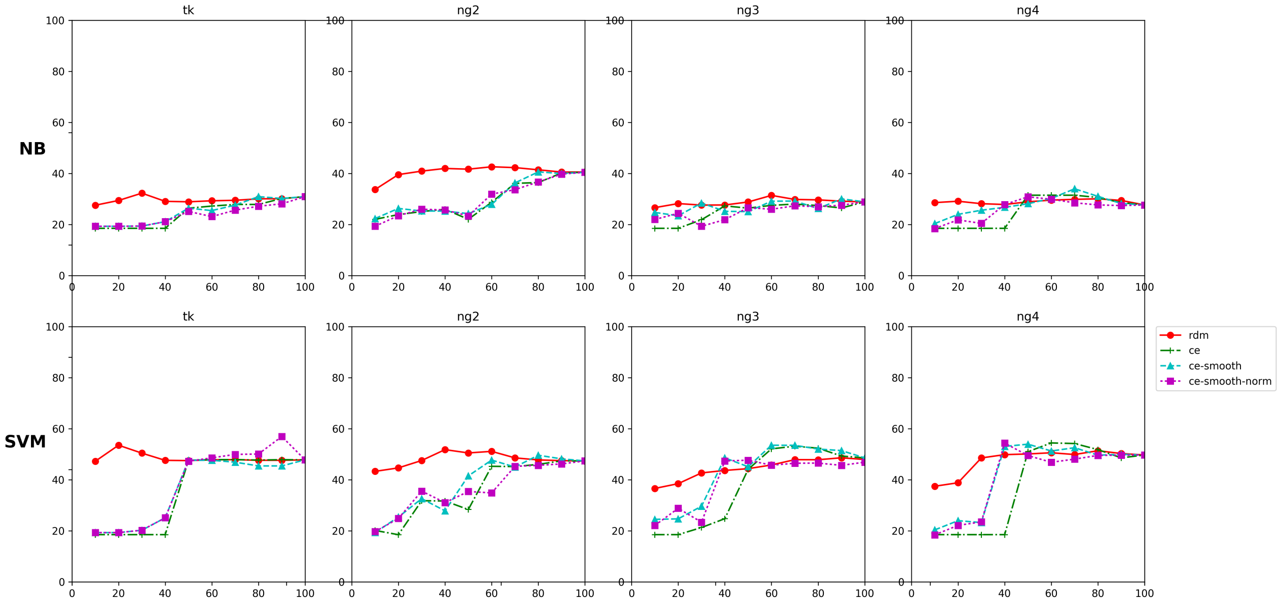


Figure C.1: **The difference of classification performance using different type of CE over the size of $\boldsymbol{D_s}$:** The graph shows the performance differences random selection, natural, smoothing and normalized CE with Texas as the D_t. X-axis and Y-axis represents D_s percentage and macro-averaged F-1 score respectively.

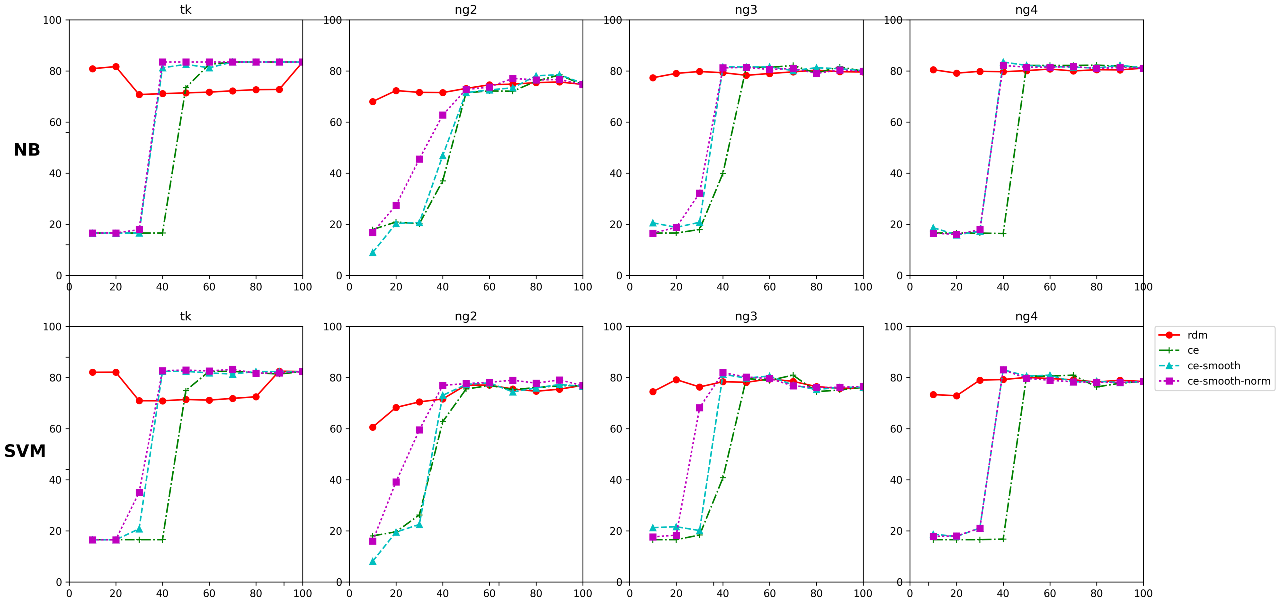


Figure C.2: **The difference of classification performance using different type of CE over the size of D_s:** The graph shows the performance differences random selection, natural, smoothing and normalized CE with Washington as the D_t. X-axis and Y-axis represents D_s percentage and macro-averaged F-1 score respectively.

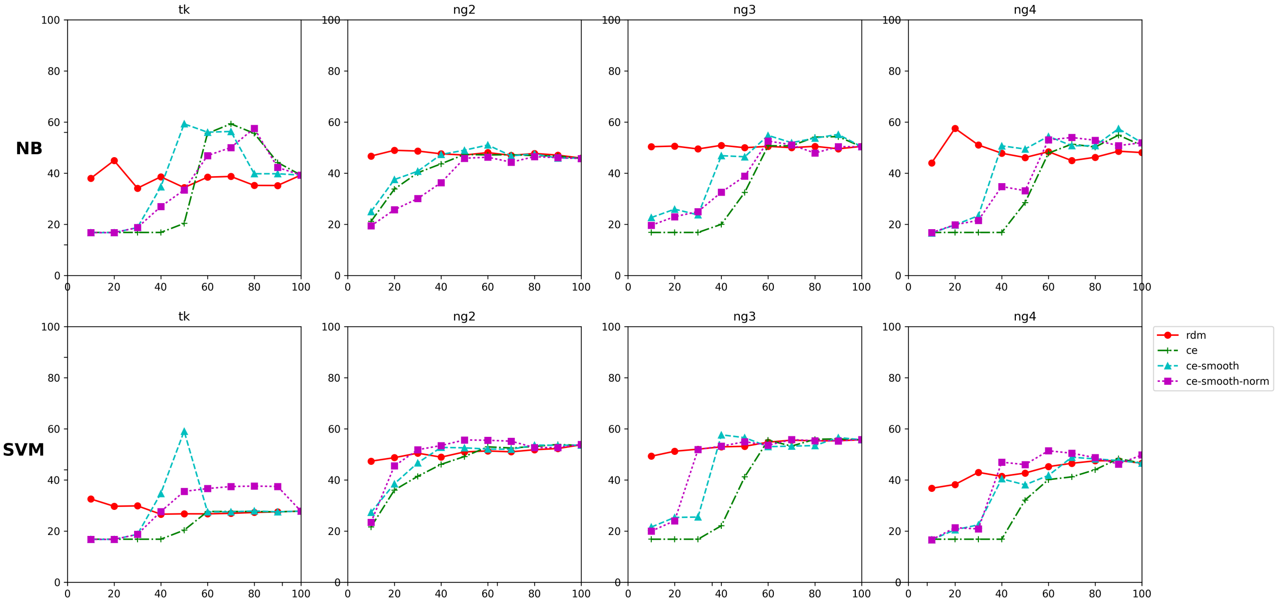


Figure C.3: **The difference of classification performance using different type of CE over the size of D_s:** The graph shows the performance differences between random selection, natural, smoothing and normalized CE with Wisconsin as the D_t. X-axis and Y-axis represents D_s percentage and macro-averaged F-1 score respectively.

Appendix D

Top Ten Keywords in Web-KB Dataset

Feature	Student	Faculty	Project	Course
tk	html:567, users:235, people:167, home:141, info:140, homes:127, students:96, grads:79, index:75, phd:65	html:527, faculty:295, info:116, people:109, cs:98, users:87, index:68, home:66, fac:51, dept:43	html:267, projects:118, research:113, cs:37, index:36, project:32, users:20, info:19, groups:17, brochure:17	cs:631, html:565, courses:278, index:110, info:110, classes:107, home:90, education:79, fall:70, cse:50
ng2	ht:584, tm:577, ml:575, er:413, in:412, ho:337, me:331, an:317, om:312, le:284	tm:572, ht:569, ml:560, ac:379, fa:357, er:322, ul:304, lt:302, ty:299, cu:297	ht:287, tm:276, ml:276, ro:232, pr:180, se:171, ec:169, ct:160, oj:160, ar:160	cs:761, tm:572, ht:570, ml:568, se:554, es:441, rs:399, ur:395, co:372, ou:355
ng3	htm:573, tml:569, ome:290, hom:289, ers:258, ser:239, use:237, eop:168, opl:167, ple:167	htm:561, tml:560, fac:346, ult:296, cul:296, lty:296, acu:295, ome:126, inf:126, ser:126	htm:273, tml:272, pro:166, roj:160, ect:153, jec:151, oje:151, cts:118, arc:118, sea:117	cs:631, htm:569, tml:565, ses:385, our:352, rse:346, urs:346, cou:346, cla:145, ass:137
ng4	html:569, home:285, user:236, sers:235, peop:167, eopl:167, ople:167, info:140, omes:127, uden:98	html:560, cult:295, ulty:295, acul:295, facu:295, info:126, home:122, peop:109, eopl:109, ople:109	htm:272, proj:160, ojec:151, ject:151, roje:151, ects:118, arch:114, earc:114, esea:114, sear:114	cs:631, html:565, urse:346, ours:346, cour:346, rses:278, lass:137, clas:137, inde:110, ndex:110

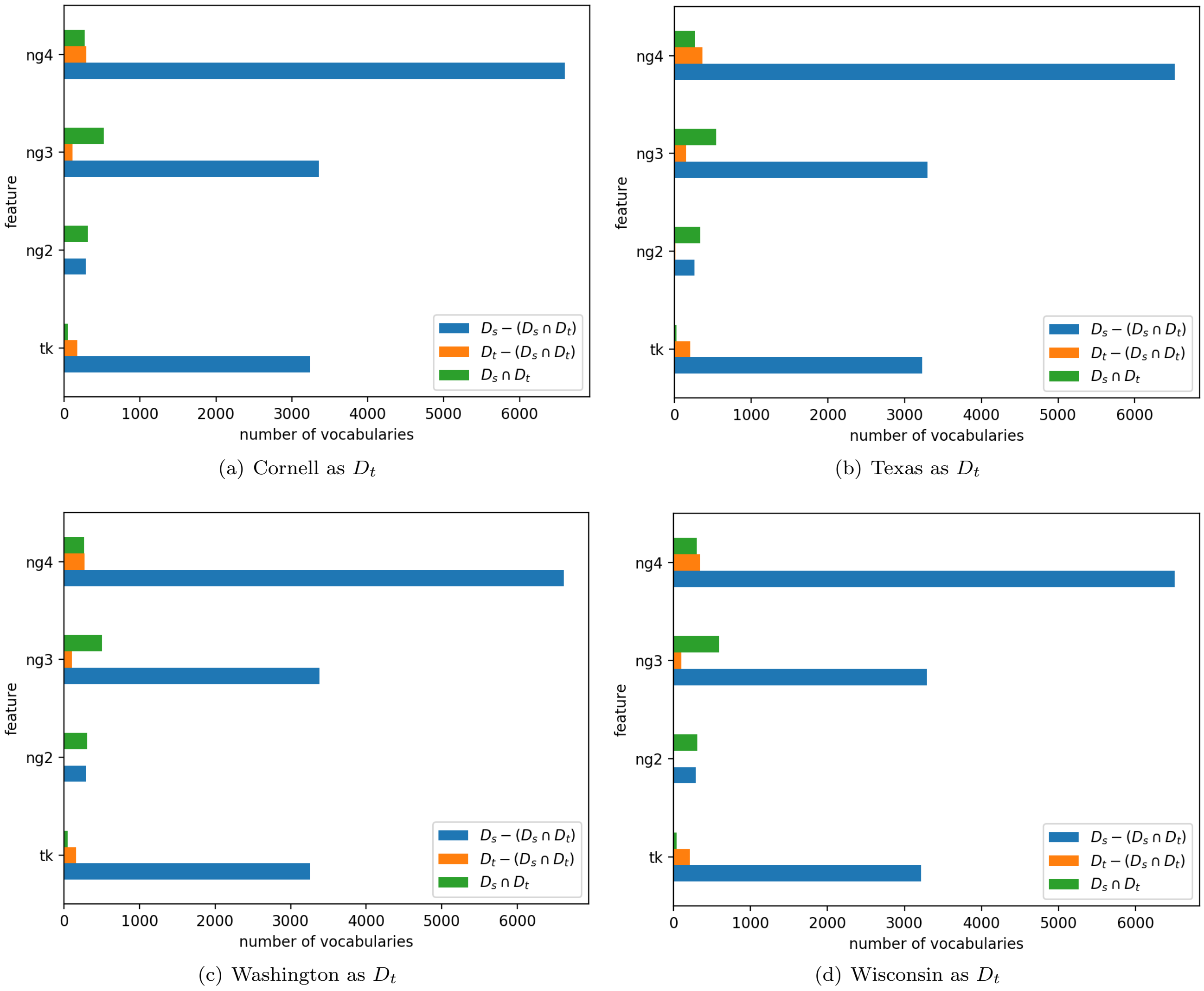
Table D.1: **Top Ten Keywords in Web-KB datasets:** The table shows top ten *tokens* and *n-grams* along with their frequencies.

D_t	Class	Token D_t	Token D_s
Cornell	Student	**people:126**, **info:126**, html:126, home:26, index:12, welcome:3, kuen:2, jiawang:2, aswin:2, ychung:2	html:441, **users:235**, homes:127, home:115, students:96, grads:79, phd:65, index:63, **people:41**, www:28
Cornell	Faculty	**info:34**, html:31, **people:23**, faculty:11, department:9, annual:9, dean:2, sam:2, lnt:2, cardie:2	html:496, Faculty:284, cs:98, **users:87**, **people:86**, **info:82**, index:68, home:65, fac:51, dept:43
Texas	Student	**users:148**, madhukar:1, cdj:1, correl:1, chuang:1, chaput:1, ckwong:1, rou:1, bayardo:1, markng:1	html:567, **people:167**, home:141, info:140, homes:127, students:96, **users:87**, grads:79, index:75, pnhd:65
Texas	faculty	**users:46**, html:15, report:14, profiles:14, miranker:1, vin:1, lavender:1, novak:1, dijkstra:1	html:512, Faculty:295, info:116, **people:109**, cs:98, index:68, home:66, fac:51, dept:43, **users:41**
Washington	Student	homes:126, dbj:1, dougz:1, dbc:1, zamir:1, sungeun:1, paul:1, speed:1, segal:1, fix:1	html:567, **users:235**, **people:167**, home:141, info:140, students:96, grads:79, index:75, phd:65, www:28
Washington	Faculty	homes:18, html:14, **people:13**, **faculty:13**, beame:2, weld:2, shapiro:1, karp:1, chambers:1, eggers:1	html:513, **faculty:282**, info:116, cs:98, **people:96**, **users:87**, index:68, home:66, fac:51, dept:43
Wisconsin	Student	html:156, shubu:2, raji:2, parker:2, moshovos:2, dzimm:2, samit:2, lloyd:2, milo:2, zeiden:2	html:441, **users:235**, **people:167**, home:141, info:140, homes:127, students:96, grads:79, index:75, phd:65
Wisconsin	Faculty	html:42, **faculty:10**, info:7, pubs:7, lumelsky:4, rrm:2, bart:2, strik:2, bach:2, olvi:2	html:485, **faculty:285**, **people:109**, info:109, cs:98, **users:87**, index:68, home:66, fac:51, dept:43

Table D.2: **Top Ten Keywords in class *student* and *faculty* in Web-KB datasets:** The table shows top ten *token* features and shared tokens (bold text) in both class *student* and *faculty* along with their frequencies for each D_t and D_s setup.

Appendix E

Token and Vocabulary Similarities between D_s and D_t



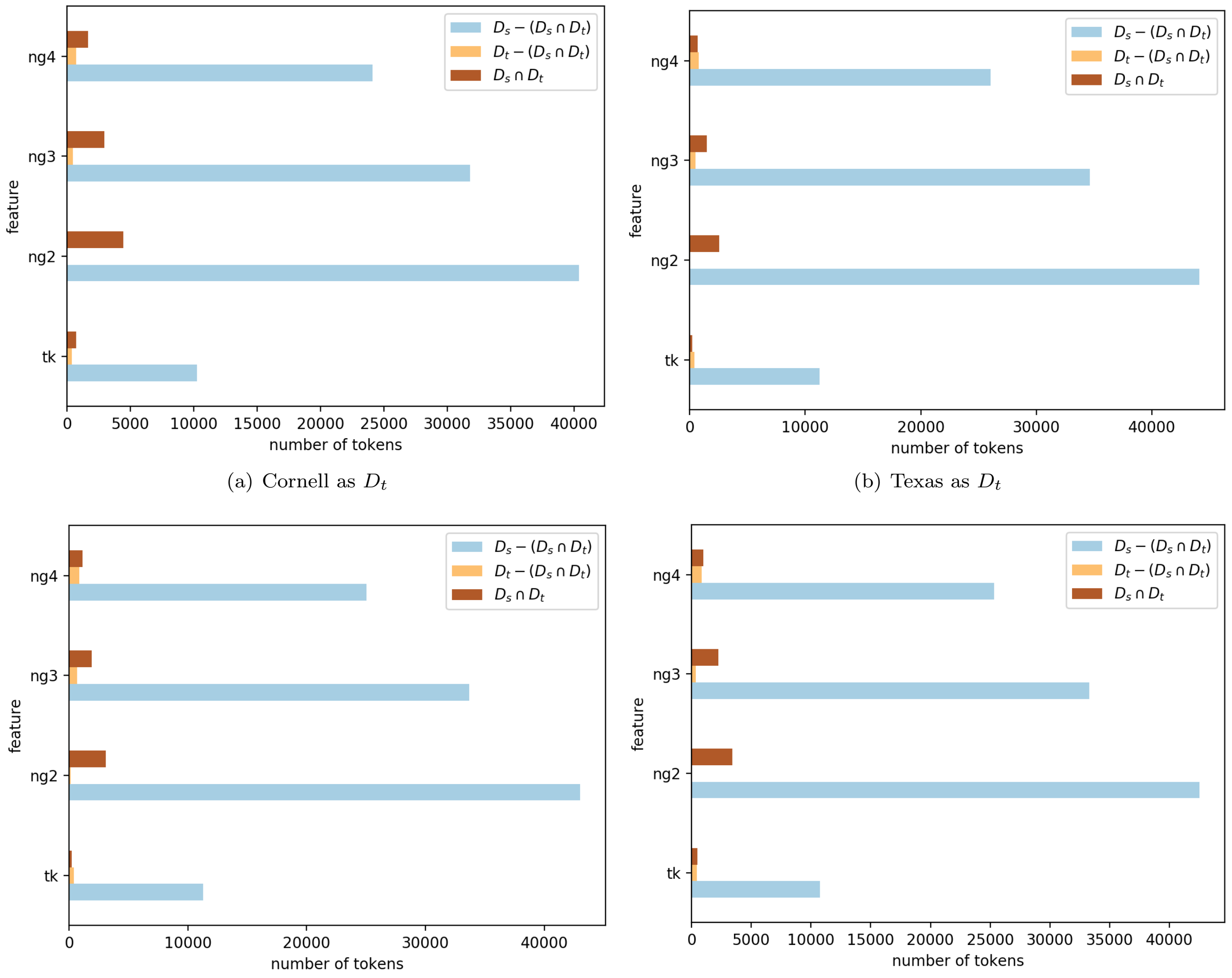
(a) Cornell as D_t

(b) Texas as D_t

(c) Washington as D_t

(d) Wisconsin as D_t

Figure E.1: **The number of vocabularies in Web-KB dataset setup :** The graph plots the number of intersection and unique vocabularies between D_s and D_t per each feature in each Web-KB dataset setup.



(a) Cornell as D_t

(b) Texas as D_t

(c) Washington as D_t

(d) Wisconsin as D_t

Figure E.2: **The number of tokens in Web-KB dataset setup :** The graph shows the number of intersection and unique tokens between D_s and D_t per each feature in each Web-KB dataset setup.

Appendix F

Token and Vocabulary Similarities over the Size of D_s

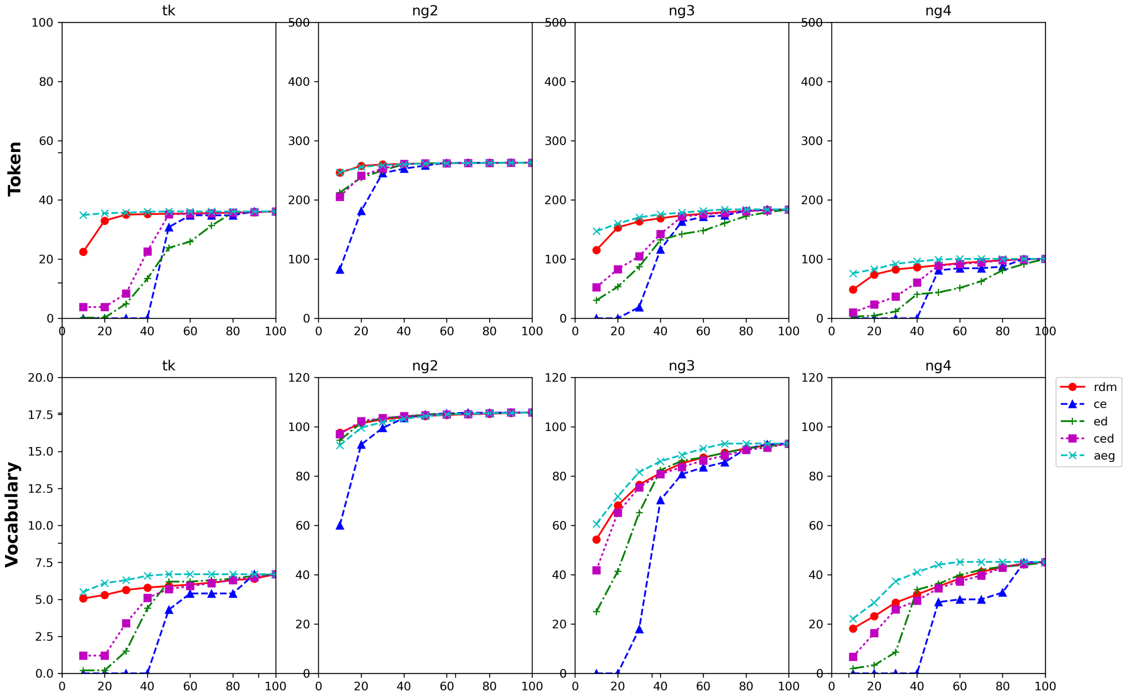


Figure F.1: **The average number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Texas as the D_t and the other four group universities as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

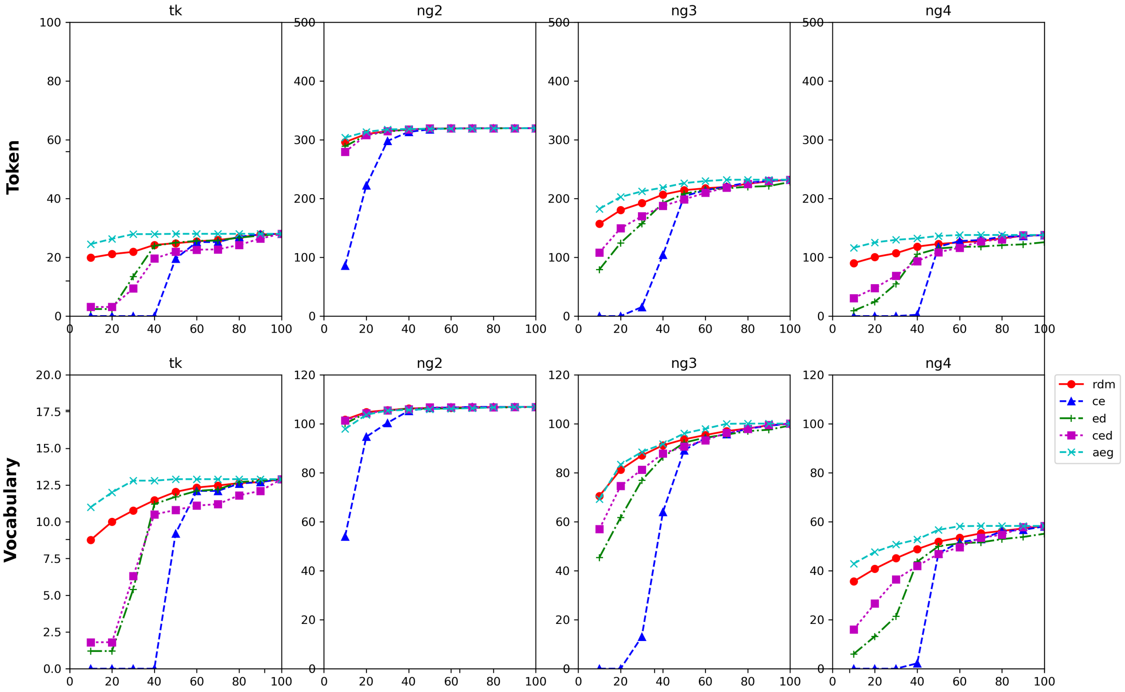


Figure F.2: **The average number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Washington as the D_t and the other four group universities as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

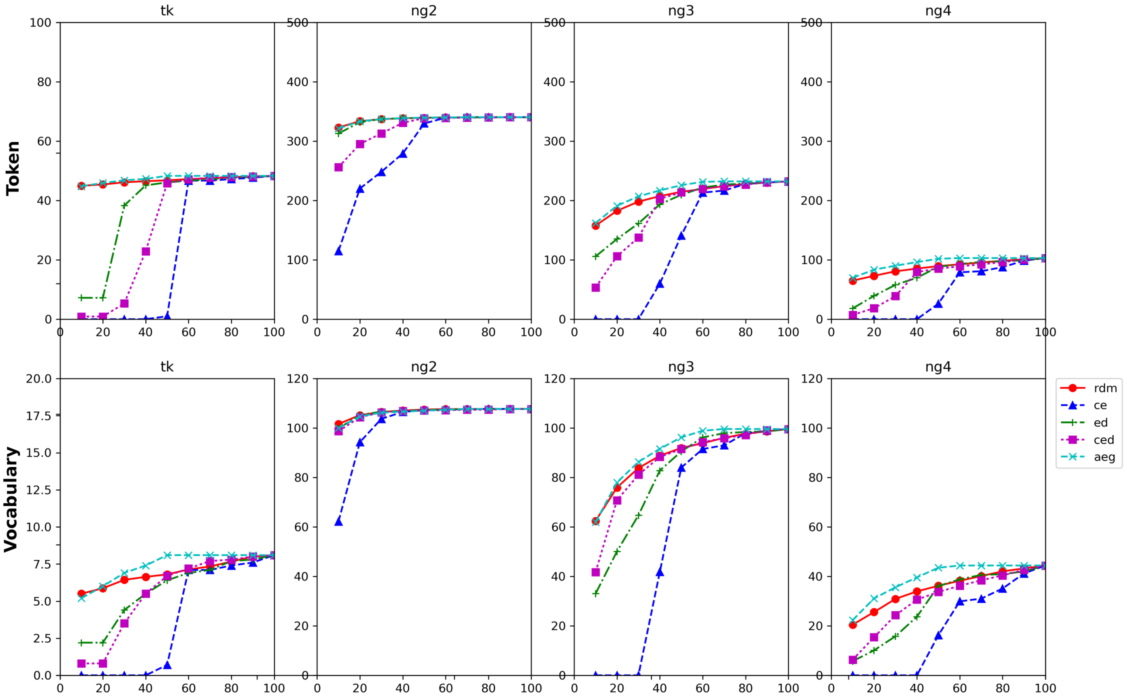


Figure F.3: **The average number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Wisconsin as the D_t and the other four group universities as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

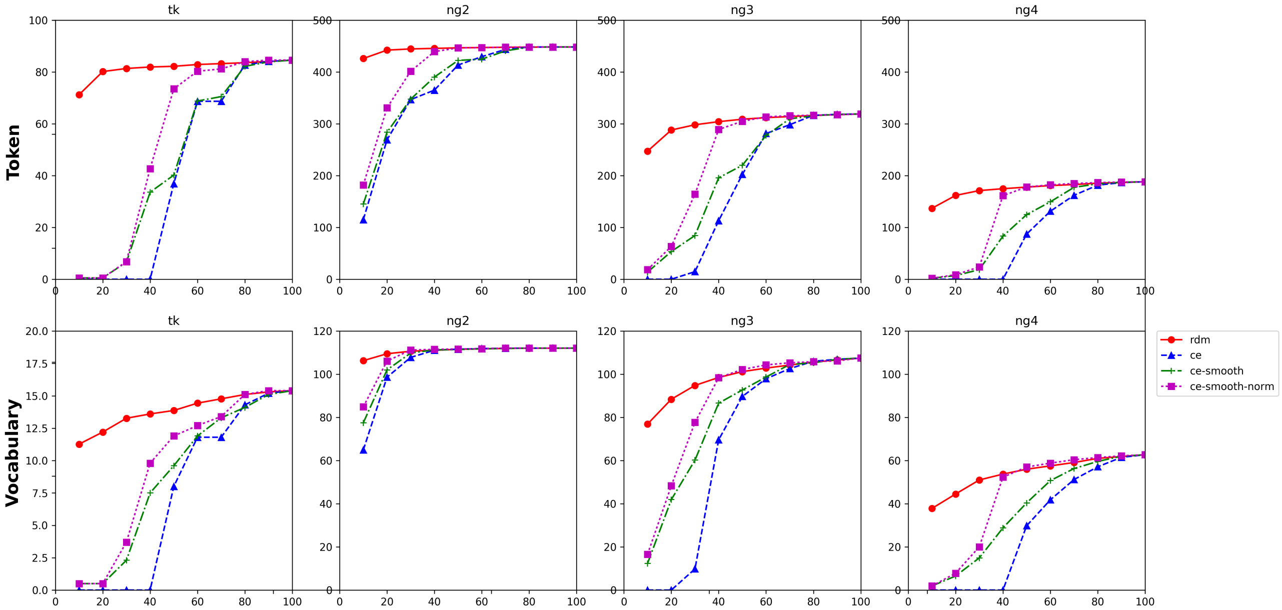


Figure F.4: **The average number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Cornell as the D_t and the other four group universities as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

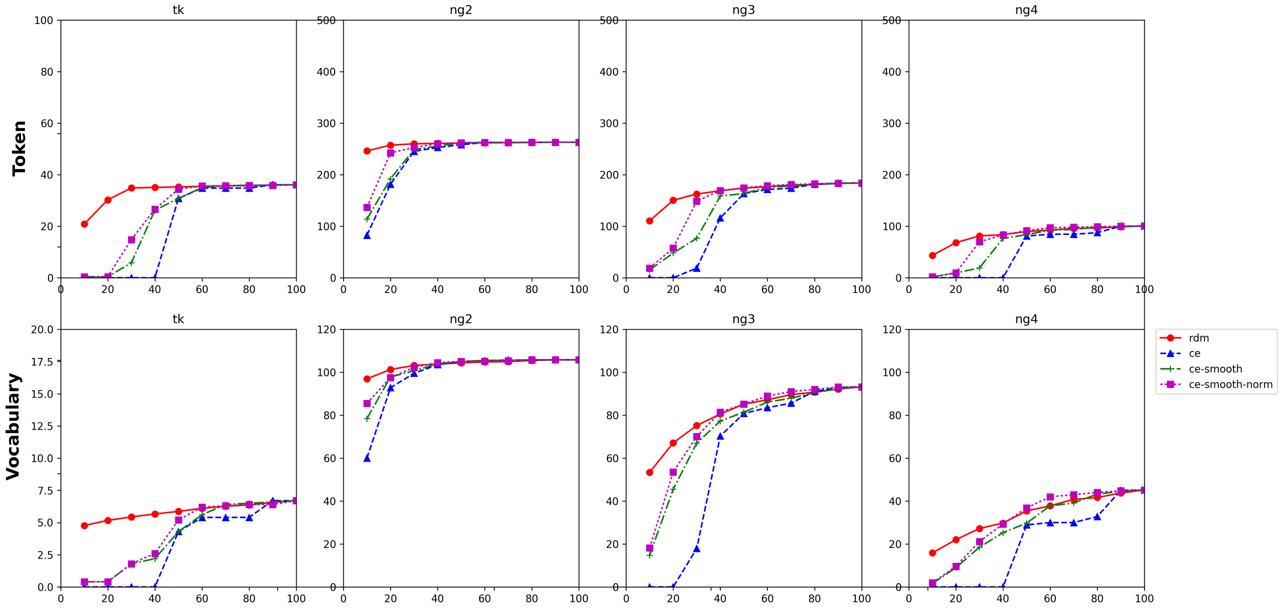


Figure F.5: **The average number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Texas as the D_t and the other four group universities as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

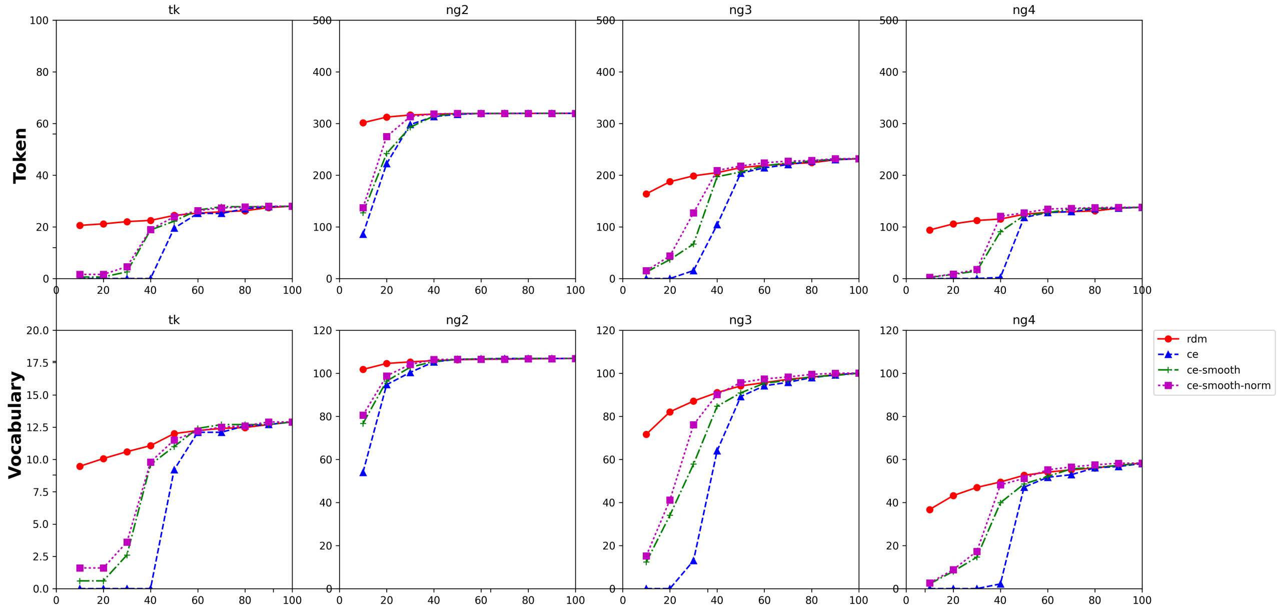


Figure F.6: **The average number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Washington as the D_t and the other four group universities as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

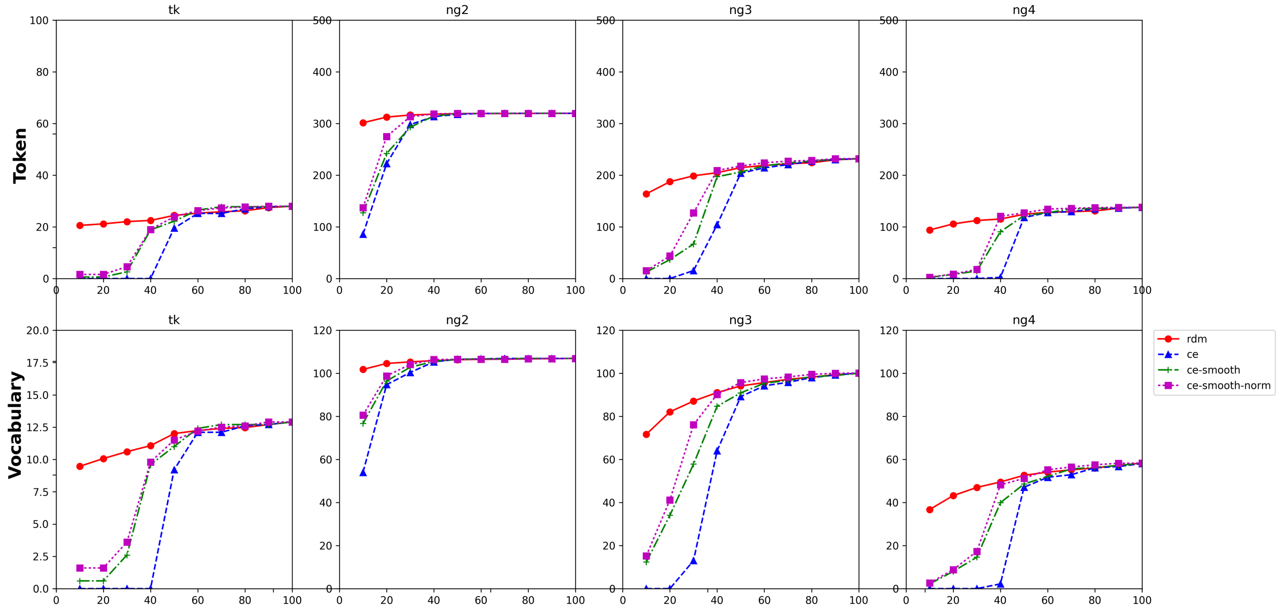


Figure F.7: **The average number of token and vocabulary similarity between D_s and D_t over the size of D_s:** Counted on Wisconsin as the D_t and the other four group universities as the D_s. The graph plots the token and vocabulary intersections between D_s and D_t over D_s selection percentage. X-axis and Y-axis represents D_s percentage and the number of token or vocabulary intersections respectively.

Appendix G

Radboud University Nijmegen URLs

Judge 1	Judge 2	URL
course	course	http://www.ru.nl/english/education/masters/medical-neuroscience/programme-outline/
course	student	http://www.ru.nl/english/education/study-radboud/housing/housing-bachelor/
course	course	http://www.ru.nl/english/education/master's-programmes/programme/science/mathematics/specialisations/foundations/
other	project	http://www.ru.nl/english/@928631/grant-worth-22-9/
course	course	http://www.ru.nl/english/education/masters/computing-security
course	course	http://www.ru.nl/english/education/masters/religiewetenschappen/pre-master/
other	student	http://www.ru.nl/english/education/masters/anthropology-0/scholarships-loans/scholarships/ots-korea/
course	course	http://www.ru.nl/english/education/master's-programmes/english-taught/computer-sciences/computing-science
other	student	http://www.ru.nl/english/education/masters/science-physical-0/scholarships-and/scholarships/ots-korea/
other	faculty	http://www.ru.nl/english/about-us/our-university/history/prime-minsters/
course	course	http://www.ru.nl/english/education/masters/molecular-life-2/
course	course	http://www.ru.nl/english/education/masters/computing-foundation
project	course	http://www.ru.nl/english/education/masters/pathobiology/our-approach-to-this/
project	course	http://www.ru.nl/english/education/masters/historical-literary/
other	other	http://www.ru.nl/english/general/90-years-radboud-0/radboud-ceremony/
course	course	http://www.ru.nl/english/education/masters/linguistics-language/deadlines
course	course	http://www.ru.nl/english/education/masters/theology-general/programme-outline/@960852/disclaimer-en/
course	course	http://www.ru.nl/english/education/masters/computing-data/deadlines/
project	project	http://www.ru.nl/english/research/radboud/themes/brain-cognition/vm/news-brain-cognition/?
course	faculty	http://www.ru.nl/english/education/masters/philosophy-social/contact/
other	student	http://www.ru.nl/english/education/master%27s-programmes/service-package/visa-and-residence/residence_permit/
faculty	faculty	http://www.ru.nl/english/education/bachelor/filosofie/anderen-filosofie/maxim-asseldonk/
course	course	http://www.ru.nl/english/education/master's-programmes/phiosophy
student	student	http://www.ru.nl/english/education/masters_student/financial_matters/student_budget_and/
faculty	faculty	https://www.ru.nl/english/about-us/facilities/confidential-advisor/
project	project	http://www.ru.nl/english/news-agenda/vm/brain-cognition/2014/decode-grant/
faculty	faculty	http://www.ru.nl/english/contact/addresses_e-mail
course	course	http://www.ru.nl/english/education/masters/physics-astrophysics/programme-outline/
course	course	http://www.ru.nl/english/education/masters/artificial/programme-outline/
other	student	http://www.ru.nl/english/education/studying_in_nijmegen/
course	project	http://www.ru.nl/english/research/radboud/themes/astronomy/vm/alma-world-largest/
course	course	http://www.ru.nl/english/education/masters/political-science/pre-masters-dutch/
course	course	http://www.ru.nl/english/education/master%27s-programmes/medical-services/
other	student	http://www.ru.nl/english/about-us/working-radboud/integrity-conduct/confidential/vm/academic_integrity/

student	student	http://www.ru.nl/english/education/exchange-phd-other/exchange-students/admission/online_application
project	project	http://www.ru.nl/english/news-agenda/vm/brain-cognition/2014/social-dominance/@960852/disclaimer-en/
other	student	http://www.ru.nl/english/education/master's-programmes/service-package/visa-and-residence/residence_permit/
course	course	http://www.ru.nl/english/education/masters/philosophy-research/
course	course	http://www.ru.nl/english/education/masters/biomedical-sciences/our-approach/
course	student	http://www.ru.nl/english/education/bachelor'-programmes/ financial-matters/working-as-student/@960852/disclaimer-en/
student	student	http://www.ru.nl/english/education/exchange_student/
project	project	http://www.ru.nl/english/education/masters/pathobiology/our-research-this/
other	student	http://www.ru.nl/english/education/study-radboud/city-nijmegen/expat_desk/
other	student	http://www.ru.nl/english/education/master's-programmes/admission-enrolment/language/
other	other	http://www.ru.nl/english/education/master's-programmes/nfp
course	course	http://www.ru.nl/english/education/masters/chemistry/
course	student	http://www.ru.nl/english/education/masters/business-analysis/tuition-and-handling/
faculty	faculty	http://www.ru.nl/english/education/masters/science-genomics/contact/
course	student	http://www.ru.nl/english/education/masters/medical-biology/admission/
course	course	http://www.ru.nl/english/education/masters/planologie/
student	student	http://www.ru.nl/english/education/bachelor'-programmes/ financial-matters/student-budget-and/@960844/information-about/
project	project	http://www.ru.nl/english/news-agenda/vm/humanities/2014/anchoring-innovation/
faculty	faculty	http://www.ru.nl/english/about-us/our-university/change-perspective/vm/anne-willemsen/
student	student	http://www.ru.nl/english/education/exchange_student/programmes/certificate/cps/nijmegen-school/
course	course	http://www.ru.nl/english/education/masters/filosofie-analytisch/voorlichting/masterdag/register-law/
course	course	http://www.ru.nl/english/education/master
faculty	faculty	http://www.ru.nl/english/research/radboud/themes/health/vm/news-health/@958727/dr-philip-poortmans/
other	course	http://www.ru.nl/english/@895556/learning-agreement/
course	course	http://www.ru.nl/english/education/masters/microbiology/specific-requirement/
course	student	http://www.ru.nl/english/education/masters/biomedical-sciences/scholarships-loans/
course	course	http://www.ru.nl/english/education/masters/pedagogische/toelating/
course	student	http://www.ru.nl/english/education/masters/mathematics/tuition-handling-fee-0/vm/tuition-fees-wizard/
course	course	http://www.ru.nl/english/education/masters/fiscaal-recht/voorlichting/masterdag/register-law/
course	student	http://www.ru.nl/english/education/masters/water-environment/tuition-and-handling/
course	project	http://www.ru.nl/english/research/radboud/themes/children-parenting/vm/academic-centre/
other	project	http://www.ru.nl/english/@678307/social_activities/
other	faculty	http://www.ru.nl/english/education/master's-programmes/contact/newsitems/nobelprize_geim/
course	student	http://www.ru.nl/english/education/masters/science-particle/scholarships-and/@957136/ots-russia/
course	course	http://www.ru.nl/english/education/master's-programmes/programme/ planning-human/human-geography/specialisations/europe-borders/
student	student	http://www.ru.nl/english/education/master's-programmes/information-your-own/greek-students/
student	student	http://www.ru.nl/english/education/masters/information-sciences/what-others-say/testimonials/deri-taufan/
course	other	http://www.ru.nl/english/education/bachelor/informatica/daarom-radboud/
course	course	http://www.ru.nl/english/education/masters/algebra-topology/
project	course	http://www.ru.nl/english/research/radboud/themes/genetics-cellular/vm/epigenetica/
project	project	http://www.ru.nl/english/news-agenda/vm/language/2014/signlanguage_0114/
course	faculty	http://www.ru.nl/english/education/masters/mls-neuroscience/contact/
course	course	http://www.ru.nl/english/education/masters/artificial/deadlines/
other	faculty	http://www.ru.nl/english/news-agenda/vm/informatics-digital/2013/radboud-university/
faculty	faculty	http://www.ru.nl/english/information/staff/?
project	project	http://www.ru.nl/english/research/radboud/themes/language/vm/news-language/@924197/'huh-'-universals/
course	course	http://www.ru.nl/english/education/masters/philosophy-research/meet-radboud/
other	project	http://www.ru.nl/english/news-agenda/agenda/all-events/@978040/famelab-science-180/

other	student	http://www.ru.nl/english/education/study-radboud/
course	course	http://www.ru.nl/english/education/masters/european-law-human/
course	course	http://www.ru.nl/english/education/masters/math-foundations/meet-radboud/
course	faculty	http://www.ru.nl/english/education/master’s-programmes/contact/newsitems/masterkeuzegids2014
student	student	http://www.ru.nl/english/education/exchange_student/admis/
course	course	http://www.ru.nl/english/education/masters/linguistics-english
course	course	http://www.ru.nl/english/education/masters/clinical-biology/meet-radboud/
faculty	faculty	http://www.ru.nl/english/vm/search/@919387/heer-prof-schulte/
course	faculty	http://www.ru.nl/english/education/masters/political-science-0/contact/
course	faculty	http://www.ru.nl/english/education/masters/linguistics-german/contact/
course	course	http://www.ru.nl/english/education/masters/chemistry-life/our-approach/
course	student	http://www.ru.nl/english/education/master’s-programmes/information-your-own/brazilian-students/@960852/disclaimer-en/
faculty	faculty	http://www.ru.nl/english/@936463/prof-nico/?
course	course	http://www.ru.nl/english/education/masters/biology/our-approach/
student	student	http://www.ru.nl/english/education/exchange_student/programmes/ects_guide_0/ects_guide/ects_guide
course	course	http://www.ru.nl/english/education/master’s-programmes/overview
course	course	http://www.ru.nl/english/education/programmes/@674089/religious_studies/
faculty	faculty	http://www.ru.nl/english/research/radboud/themes/health/vm/professor-jan/