Transfer Learning Based Cross-lingual Knowledge Extraction for Wikipedia

Zhigang Wang[†], **Zhixing Li**[†], **Juanzi Li**[†], **Jie Tang**[†], **and Jeff Z. Pan**[‡]
[†] Tsinghua National Laboratory for Information Science and Technology
DCST, Tsinghua University, Beijing, China
{wzhigang,zhxli,ljz,tangjie}@keg.cs.tsinghua.edu.cn
[‡] Department of Computing Science, University of Aberdeen, Aberdeen, UK
jeff.z.pan@abdn.ac.uk

Abstract

Wikipedia infoboxes are a valuable source of structured knowledge for global knowledge sharing. However, infobox information is very incomplete and imbalanced among the Wikipedias in different languages. It is a promising but challenging problem to utilize the rich structured knowledge from a source language Wikipedia to help complete the missing infoboxes for a target language.

In this paper, we formulate the problem of cross-lingual knowledge extraction from multilingual Wikipedia sources, and present a novel framework, called WikiCiKE, to solve this problem. An instance-based transfer learning method is utilized to overcome the problems of topic drift and translation errors. Our experimental results demonstrate that WikiCiKE outperforms the monolingual knowledge extraction method and the translation-based method.

1 Introduction

In recent years, the automatic knowledge extraction using Wikipedia has attracted significant research interest in research fields, such as the semantic web. As a valuable source of structured knowledge, Wikipedia infoboxes have been utilized to build linked open data (Suchanek et al., 2007; Bollacker et al., 2008; Bizer et al., 2008; Bizer et al., 2009), support next-generation information retrieval (Hotho et al., 2006), improve question answering (Bouma et al., 2008; Ferrández et al., 2009), and other aspects of data exploitation (McIlraith et al., 2001; Volkel et al., 2006; Hogan et al., 2011) using semantic web standards, such as RDF (Pan and Horrocks, 2007; Heino and Pan, 2012) and OWL (Pan and Horrocks, 2006; Pan and Thomas, 2007; Fokoue et al., 2012), and their reasoning services.

However, most infoboxes in different Wikipedia language versions are missing. Figure 1 shows the statistics of article numbers and infobox information for six major Wikipedias. Only 32.82% of the articles have infoboxes on average, and the numbers of infoboxes for these Wikipedias vary significantly. For instance, the English Wikipedia has 13 times more infoboxes than the Chinese Wikipedia and 3.5 times more infoboxes than the second largest Wikipedia of German language.

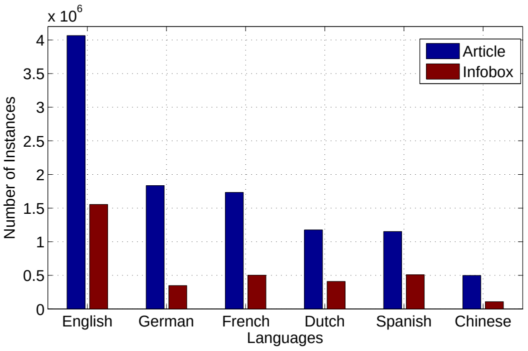


Figure 1: Statistics for Six Major Wikipedias.

To solve this problem, KYLIN has been proposed to extract the missing infoboxes from unstructured article texts for the English Wikipedia (Wu and Weld, 2007). KYLIN performs well when sufficient training data are available, and such techniques as shrinkage and retraining have been used to increase recall from English Wikipedia's long tail of sparse infobox classes (Weld et al., 2008; Wu et al., 2008). The extraction performance of KYLIN is limited by the number of available training samples.

Due to the great imbalance between different Wikipedia language versions, it is difficult to gather sufficient training data from a single Wikipedia. Some translation-based cross-lingual knowledge

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 641–650,
Sofia, Bulgaria, August 4-9 2013. ©2013 Association for Computational Linguistics

extraction methods have been proposed (Adar et al., 2009; Bouma et al., 2009; Adafre and de Rijke, 2006). These methods concentrate on translating existing infoboxes from a richer source language version of Wikipedia into the target language. The recall of new target infoboxes is highly limited by the number of equivalent cross-lingual articles and the number of existing source infoboxes. Take Chinese-English[1] Wikipedias as an example: current translation-based methods only work for 87,603 Chinese Wikipedia articles, 20.43% of the total 428,777 articles. Hence, the challenge remains: how could we supplement the missing infoboxes for the rest 79.57% articles?

On the other hand, the numbers of existing infobox attributes in different languages are highly imbalanced. Table 1 shows the comparison of the numbers of the articles for the attributes in template PERSON between English and Chinese Wikipedia. Extracting the missing value for these attributes, such as *awards*, *weight*, *influences* and *style*, inside the single Chinese Wikipedia is intractable due to the rarity of existing Chinese attribute-value pairs.

Attribute	***en***	***zh***	**Attribute**	***en***	***zh***
name	82,099	1,486	*awards*	2,310	38
birth date	77,850	1,481	*weight*	480	12
occupation	66,768	1,279	*influences*	450	6
nationality	20,048	730	*style*	127	1

Table 1: The Numbers of Articles in TEMPLATE PERSON between English(en) and Chinese(zh).

In this paper, we have the following hypothesis: *one can use the rich English (auxiliary) information to assist the Chinese (target) infobox extraction.* In general, we address the problem of cross-lingual knowledge extraction by using the imbalance between Wikipedias of different languages. For each attribute, we aim to learn an extractor to find the missing value from the unstructured article texts in the target Wikipedia by using the rich information in the source language. Specifically, we treat this cross-lingual information extraction task as a transfer learning-based binary classification problem.

The contributions of this paper are as follows:

1. We propose a transfer learning-based cross-lingual knowledge extraction framework called **WikiCiKE**. The extraction performance for the target Wikipedia is improved by using rich infoboxes and textual information in the source language.

2. We propose the TrAdaBoost-based extractor training method to avoid the problems of topic drift and translation errors of the source Wikipedia. Meanwhile, some language-independent features are introduced to make WikiCiKE as general as possible.

3. Chinese-English experiments for four typical attributes demonstrate that WikiCiKE outperforms both the monolingual extraction method and current translation-based method. The increases of 12.65% for precision and 12.47% for recall in the template named person are achieved when only 30 target training articles are available.

The rest of this paper is organized as follows. Section 2 presents some basic concepts, the problem formalization and the overview of WikiCiKE. In Section 3, we propose our detailed approaches. We present our experiments in Section 4. Some related work is described in Section 5. We conclude our work and the future work in Section 6.

2 Preliminaries

In this section, we introduce some basic concepts regarding Wikipedia, formally defining the key problem of cross-lingual knowledge extraction and providing an overview of the WikiCiKE framework.

2.1 Wiki Knowledge Base and Wiki Article

We consider each language version of Wikipedia as a ***wiki knowledge base***, which can be represented as $K = \{a_i\}_{i=1}^{p}$, where a_i is a disambiguated article in K and p is the size of K.

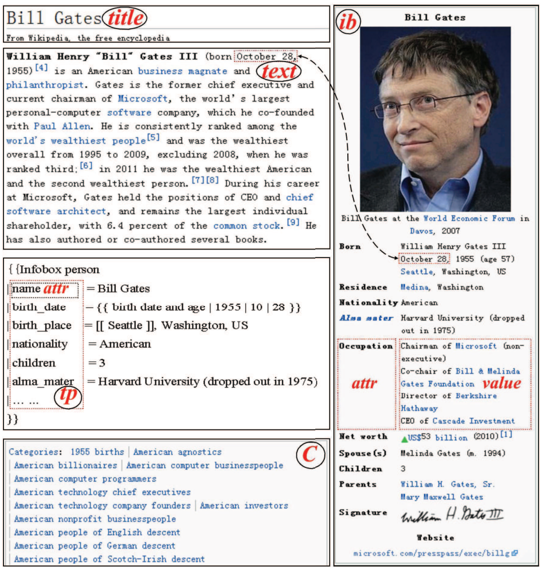
Formally we define a ***wiki article*** $a \in K$ as a 5-tuple $a = (title, text, ib, tp, C)$, where

- $title$ denotes the title of the article a,

- $text$ denotes the unstructured text description of the article a,

- ib is the infobox associated with a; specifically, $ib = \{(attr_i, value_i)\}_{i=1}^{q}$ represents the list of attribute-value pairs for the article a,

[1] Chinese-English denotes the task of Chinese Wikipedia infobox completion using English Wikipedia



Bill Gates

From Wikipedia, the free encyclopedia

William Henry "Bill" Gates III (born October 28, 1955)[4] is an American business magnate and philanthropist. Gates is the former chief executive and current chairman of Microsoft, the world' s largest personal-computer software company, which he co-founded with Paul Allen. He is consistently ranked among the world' s wealthiest people[5] and was the wealthiest overall from 1995 to 2009, excluding 2008, when he was ranked third;[6] in 2011 he was the wealthiest American and the second wealthiest person.[7][8] During his career at Microsoft, Gates held the positions of CEO and chief software architect, and remains the largest individual shareholder, with 6.4 percent of the common stock.[9] He has also authored or co-authored several books.

```
{{Infobox person
|name          = Bill Gates
|birth_date    = {{ birth date and age | 1955 | 10 | 28 }}
|birth_place   = [[ Seattle ]], Washington, US
|nationality   = American
|children      = 3
|alma_mater    = Harvard University (dropped out in 1975)
|... ...
}}
```

Categories: 1955 births | American agnostics | American billionaires | American computer businesspeople | American computer programmers | American technology chief executives | American technology company founders | American investors | American nonprofit businesspeople | American people of English descent | American people of German descent | American people of Scotch-Irish descent

Bill Gates

Bill Gates at the World Economic Forum in Davos, 2007

Born	William Henry Gates III October 28, 1955 (age 57) Seattle, Washington, US
Residence	Medina, Washington
Nationality	American
Alma mater	Harvard University (dropped out in 1975)
Occupation	Chairman of Microsoft (non-executive) Co-chair of Bill & Melinda Gates Foundation Director of Berkshire Hathaway CEO of Cascade Investment
Net worth	▲US$53 billion (2010)[1]
Spouse(s)	Melinda Gates (m. 1994)
Children	3
Parents	William H. Gates, Sr. Mary Maxwell Gates
Signature	William H. Gates III

Website

microsoft.com/presspass/exec/billg

Figure 2: Simplified Article of "Bill Gates".

- $tp = \{attr_i\}_{i=1}^{r}$ is the infobox template associated with ib, where r is the number of attributes for one specific template, and
- C denotes the set of categories to which the article a belongs.

Figure 2 gives an example of these five important elements concerning the article named "Bill Gates".

In what follows, we will use named subscripts, such as $a_{Bill\ Gates}$, or index subscripts, such as a_i, to refer to one particular instance interchangeably. We will use "*name* in TEMPLATE PERSON" to refer to the attribute $attr_{name}$ in the template tp_{PERSON}. In this cross-lingual task, we use the source (***S***) and target (***T***) languages to denote the languages of auxiliary and target Wikipedias, respectively. For example, K_S indicates the source wiki knowledge base, and K_T denotes the target wiki knowledge base.

2.2 Problem Formulation

Mining new infobox information from unstructured article texts is actually a multi-template, multi-slot information extraction problem. In our task, each template represents an infobox template and each slot denotes an attribute. In the WikiCiKE framework, for each attribute $attr_T$ in an infobox template tp_T, we treat the task of missing $value$ extraction as a binary classification problem. It predicts whether a particular word (token) from the article $text$ is the extraction target (Finn and Kushmerick, 2004; Lafferty et al., 2001).

Given an attribute $attr_T$ and an instance (word/token) x_i, $X_S = \{x_i\}_{i=1}^{n}$ and $X_T = \{x_i\}_{i=n+1}^{n+m}$ are the sets of instances (words/tokens) in the source and the target language respectively. x_i can be represented as a feature vector according to its context. Usually, we have $n \gg m$ in our setting, with much more attributes in the source that those in the target. The function $g : X \mapsto Y$ maps the instance from $X = X_S \cup X_T$ to the true label of $Y = \{0, 1\}$, where 1 represents the extraction target (positive) and 0 denotes the background information (negative). Because the number of target instances m is inadequate to train a good classifier, we combine the source and target instances to construct the training data set as $TD = TD_S \cup TD_T$, where $TD_S = \{x_i, g(x_i)\}_{i=1}^{n}$ and $TD_T = \{x_i, g(x_i)\}_{i=n+1}^{n+m}$ represent the source and target training data, respectively.

Given the combined training data set TD, our objective is to estimate a hypothesis $f : X \mapsto Y$ that minimizes the prediction error on testing data in the target language. Our idea is to determine the useful part of TD_S to improve the classification performance in TD_T. We view this as a transfer learning problem.

2.3 WikiCiKE Framework

WikiCiKE learns an extractor for a given attribute $attr_T$ in the target Wikipedia. As shown in Figure 3, WikiCiKE contains four key components: (1) **Automatic Training Data Generation**: given the target attribute $attr_T$ and two wiki knowledge bases K_S and K_T, WikiCiKE first generates the training data set $TD = TD_S \cup TD_T$ automatically. (2) **WikiCiKE Training**: WikiCiKE uses a transfer learning-based classification method to train the classifier (extractor) $f : X \mapsto Y$ by using $TD_S \cup TD_T$. (3) **Template Classification**: WikiCiKE then determines proper candidate articles which are suitable to generate the missing value of $attr_T$. (4) **WikiCiKE Extraction**: given a candidate article a, WikiCiKE uses the learned extractor f to label each word in the $text$ of a, and generate the extraction result in the end.

3 Our Approach

In this section, we will present the detailed approaches used in WikiCiKE.

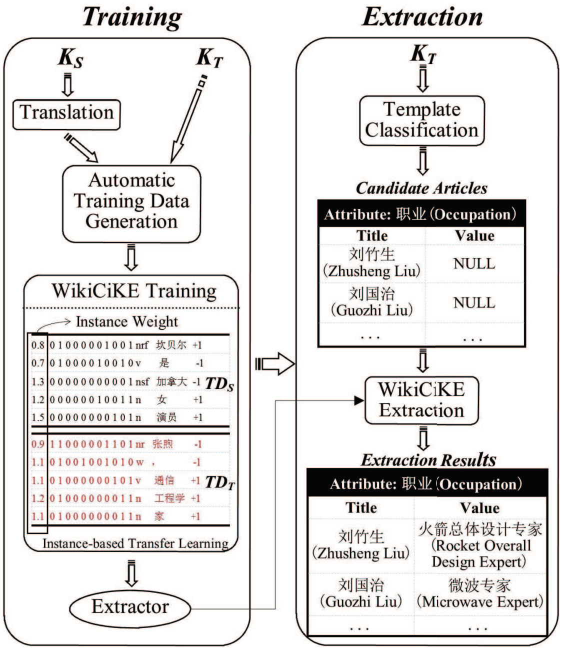

Training
Extraction
K_S
K_T
Translation
Automatic Training Data Generation
WikiCiKE Training
Instance Weight
0.8 0 1 0 0 0 0 0 1 0 0 1 nrf 坎贝尔 +1
0.7 0 1 0 0 0 0 0 1 0 0 1 0 v 是 -1
1.3 0 0 0 0 0 0 0 0 0 0 0 1 nsf 加拿大 -1
1.2 0 0 0 0 0 0 1 0 0 1 1 n 女 +1
1.5 0 0 0 0 0 0 0 0 1 0 1 n 演员 +1
TD_S
0.9 1 1 0 0 0 0 0 1 1 0 1 nr 张煦 -1
1.1 0 1 0 0 1 0 0 1 0 1 0 w ， -1
1.1 0 1 0 0 0 0 0 0 1 0 1 v 通信 +1
1.2 0 1 0 0 0 0 0 0 0 1 1 n 工程学 +1
1.1 0 1 0 0 0 0 0 0 0 1 1 n 家 +1
TD_T
Instance-based Transfer Learning
Extractor
K_T
Template Classification
Candidate Articles
Attribute: 职业 (Occupation)
Title
Value
刘竹生 (Zhusheng Liu)
NULL
刘国治 (Guozhi Liu)
NULL
. . .
. . .
WikiCiKE Extraction
Extraction Results
Attribute: 职业 (Occupation)
Title
Value
刘竹生 (Zhusheng Liu)
火箭总体设计专家 (Rocket Overall Design Expert)
刘国治 (Guozhi Liu)
微波专家 (Microwave Expert)
. . .
. . .

Figure 3: WikiCiKE Framework.

3.1 Automatic Training Data Generation

To generate the training data for the target attribute $attr_T$, we first determine the equivalent cross-lingual attribute $attr_S$. Fortunately, some templates in non-English Wikipedia (e.g. Chinese Wikipedia) explicitly match their attributes with their counterparts in English Wikipedia. Therefore, it is convenient to align the cross-lingual attributes using English Wikipedia as bridge. For attributes that can not be aligned in this way, currently we manually align them. The manual alignment is worthwhile because thousands of articles belong to the same template may benefit from it and at the same time it is not very costly. In Chinese Wikipedia, the top 100 templates have covered nearly 80% of the articles which have been assigned a template.

Once the aligned attribute mapping $attr_T \leftrightarrow attr_S$ is obtained, we collect the articles from both K_S and K_T containing the corresponding $attr$. The collected articles from K_S are translated into the target language. Then, we use a uniform automatic method, which primarily consists of word labeling and feature vector generation, to generate the training data set $TD = \{(x, g(x))\}$ from these collected articles.

For each collected article $a = \{title, text, ib, tp, C\}$ and its $value$ of $attr$, we can automatically label each word x in $text$ according to whether x and its neighbors are contained by the $value$. The $text$ and $value$ are processed as bags of words $\{x\}_{text}$ and $\{x\}_{value}$. Then for each $x_i \in \{x\}_{text}$ we have:

$$g(x_i) = \begin{cases} 1 & x_i \in \{x\}_{value}, \ |\{x\}_{value}| = 1 \\ 1 & x_{i-1}, x_i \in \{x\}_{value} \ or \ x_i, x_{i+1} \in \{x\}_{value}, \\ & |\{x\}_{value}| > 1 \\ 0 & otherwise \end{cases} \tag{1}$$

After the word labeling, each instance (word/token) is represented as a feature vector. In this paper, we propose a general feature space that is suitable for most target languages. As shown in Table 2, we classify the features used in WikiCiKE into three categories: format features, POS tag features and token features.

Category	Feature	Example
Format feature	First token of sentence	你好，世界！ *Hello* World!
	In first half of sentence	你好，世界！ *Hello* World!
	Starts with two digits	12月31日 31^{th} Dec.
	Starts with four digits	*1999*年 夏天 *1999's* summer
	Contains a cash sign	10￥or 10$
	Contains a percentage symbol	10%
	Stop words	的, 地, 这··· of, the, a, an
	Pure number	365
	Part of an anchor text	电影*导演* Movie *Director*
	Begin of an anchor text	*游戏* 设计师 *Game* Designer
POS tag features	POS tag of current token	
	POS tags of previous 5 tokens	
	POS tags of next 5 tokens	
Token features	Current token	
	Previous 5 tokens	
	Next 5 tokens	
	Is current token contained by title	
	Is one of previous 5 tokens contained by title	

Table 2: Feature Definition.

The target training data TD_T is directly generated from articles in the target language Wikipedia. Articles from the source language Wikipedia are translated into the target language in advance and then transformed into training data TD_S. In next section, we will discuss how to train an extractor from $TD = TD_S \cup TD_T$.

3.2 WikiCiKE Training

Given the attribute $attr_T$, we want to train a classifier $f : X \mapsto Y$ that can minimize the prediction error for the testing data in the target language. Traditional machine learning approaches attempt to determine f by minimizing some loss function L on the prediction $f(x)$ for the training instance x and its real label $g(x)$, which is

$$\hat{f} = \underset{f \in \Theta}{\operatorname{argmin}} \sum L(f(x), g(x)) \;\; where \; (x, g(x)) \in TD_T \tag{2}$$

In this paper, we use TrAdaBoost (Dai et al., 2007), which is an instance-based transfer learning algorithm that was first proposed by Dai to find $\hat{f}$. TrAdaBoost requires that the source training instances X_S and target training instances X_T be drawn from the same feature space. In WikiCiKE, the source articles are translated into the target language in advance to satisfy this requirement. Due to the topic drift problem and translation errors, the joint probability distribution $P_S(x, g(x))$ is not identical to $P_T(x, g(x))$. We must adjust the source training data TD_S so that they fit the distribution on TD_T. TrAdaBoost iteratively updates the weights of all training instances to optimize the prediction error. Specifically, the weight-updating strategy for the source instances is decided by the loss on the target instances.

For each $t = 1 \sim T$ iteration, given a weight vector $\boldsymbol{p_t}$ normalized from $\boldsymbol{w_t}$($\boldsymbol{w_t}$ is the weight vector before normalization), we call a basic classifier F that can address weighted instances and then find a hypothesis f that satisfies

$$\begin{gathered} \hat{f}_t = \underset{f \in \Theta_F}{\operatorname{argmin}} \sum L(\boldsymbol{p_t}, f(x), g(x)) \\ (x, g(x)) \in TD_S \cup TD_T \end{gathered} \tag{3}$$

Let ϵ_t be the prediction error of $\hat{f}_t$ at the t^{th} iteration on the target training instances TD_T, which is

$$\epsilon_t = \frac{1}{\sum_{k=n+1}^{n+m} w_k^t} \times \sum_{k=n+1}^{n+m} (w_k^t \times |\hat{f}_t(x_k) - y_k|) \tag{4}$$

With ϵ_t, the weight vector $\boldsymbol{w_t}$ is updated by the function:

$$w_{t+1} = h(\boldsymbol{w_t}, \epsilon_t) \tag{5}$$

The weight-updating strategy h is illustrated in Table 3.

Finally, a final classifier $\hat{f}$ can be obtained by combining $\hat{f}_{T/2} \sim \hat{f}_T$.

TrAdaBoost has a convergence rate of $O(\sqrt{\ln(n/N)})$, where n and N are the number of source samples and number of maximum iterations respectively.

		TrAdaBoost	**AdaBoost**
Target samples	$+$	w_t	w_t
	$-$	$w_t \times \beta_t^{-1}$	$w_t \times \beta_t^{-1}$
Source samples	$+$	$w_t \times \beta^{-1}$	No source training sample available
	$-$	$w_t \times \beta$	

$+$: correctly labelled $-$: miss-labelled
w_t: weight of an instance at the t^{th} iteration
$\beta_t = \epsilon_t \times (1 - \epsilon_t)$
$\beta = 1/(1 + \sqrt{2 \ln nT})$

Table 3: Weight-updating Strategy of TrAdaBoost.

3.3 Template Classification

Before using the learned classifier f to extract missing infobox value for the target attribute $attr_T$, we must select the correct articles to be processed. For example, the article $a_{New\ York}$ is not a proper article for extracting the missing value of the attribute $attr_{birth\_day}$.

If a already has an incomplete infobox, it is clear that the correct tp is the template of its own infobox ib. For those articles that have no infoboxes, we use the classical 5-nearest neighbor algorithm to determine their templates (Roussopoulos et al., 1995) using their category labels, outlinks, inlinks as features (Wang et al., 2012). Our classifier achieves an average precision of 76.96% with an average recall of 63.29%, and can be improved further. In this paper, we concentrate on the WikiCiKE training and extraction components.

3.4 WikiCiKE Extraction

Given an article a determined by template classification, we generate the missing $value$ of $attr$ from the corresponding $text$. First, we turn the $text$ into a word sequence and compute the feature vector for each word based on the feature definition in Section 3.1. Next we use f to label each word, and we get a labeled sequence $text^l$ as $text^l = \{x_1^{f(x_1)} ... x_{i-1}^{f(x_{i-1})} x_i^{f(x_i)} x_{i+1}^{f(x_{i+1})} ... x_n^{f(x_n)}\}$ where the superscript $f(x_i) \in \{0, 1\}$ represents the positive or negative label by f. After that, we extract the adjacent positive tokens in $text$ as the predict value. In particular, the longest positive token sequence and the one that contains other positive token sequences are preferred in extraction. E.g., a positive sequence “comedy movie director” is preferred to a shorter sequence “movie director”.

4 Experiments

In this section, we present our experiments to evaluate the effectiveness of WikiCiKE, where we focus on the Chinese-English case; in other words, the target language is Chinese and the source language is English. It is part of our future work to try other language pairs which two Wikipedias of these languages are imbalanced in infobox information such as English-Dutch.

4.1 Experimental Setup

4.1.1 Data Sets

Our data sets are from Wikipedia dumps[2] generated on April 3, 2012. For each attribute, we collect both labeled articles (articles that contain the corresponding attribute $attr$) and unlabeled articles in Chinese. We split the labeled articles into two subsets $A_T\ and\ A_{test}(A_T \cap A_{test} = \emptyset)$, in which A_T is used as target training articles and A_{test} is used as the first testing set. For the unlabeled articles, represented as A'_{test}, we manually label their infoboxes with their texts and use them as the second testing set. For each attribute, we also collect a set of labeled articles A_S in English as the source training data. Our experiments are performed on four attributes, which are *occupation*, *nationality*, *alma mater* in TEMPLATE PERSON, and *country* in TEMPLATE FILM. In particular, we extract values from the first two paragraphs of the texts because they usually contain most of the valuable information. The details of data sets on these attributes are given in Table 4.

Attribute	$\mathbf{\vert A_S \vert}$	$\mathbf{\vert A_T \vert}$	$\mathbf{\vert A_{test} \vert}$	$\mathbf{\vert A'_{test} \vert}$
occupation	1,000	500	779	208
alma mater	1,000	200	215	208
nationality	1,000	300	430	208
country	1,000	500	1,000	–

$|A|$: the number of articles in A

Table 4: Data Sets.

4.1.2 Comparison Methods

We compare our WikiCiKE method with two different kinds of methods, the monolingual knowledge extraction method and the translation-based method. They are implemented as follows:

1. **KE-Mon** is the monolingual knowledge extractor. The difference between WikiCiKE and KE-Mon is that KE-Mon only uses the Chinese training data.
2. **KE-Tr** is the translation-based extractor. It obtains the *values* by two steps: finding their counterparts (if available) in English using Wikipedia cross-lingual links and attribute alignments, and translating them into Chinese.

We conduct two series of evaluation to compare WikiCiKE with KE-Mon and KE-Tr, respectively.

1. We compare WikiCiKE with KE-Mon on the first testing data set A_{test}, where most values can be found in the articles' texts in those labeled articles, in order to demonstrate the performance improvement by using cross-lingual knowledge transfer.
2. We compare WikiCiKE with KE-Tr on the second testing data set $A^{'}_{test}$, where the existences of values are not guaranteed in those randomly selected articles, in order to demonstrate the better recall of WikiCiKE.

For implementation details, the *weighted-SVM* is used as the basic learner f both in WikiCiKE and KE-Mon (Zhang et al., 2009), and Baidu Translation API[3] is used as the translator both in WikiCiKE and KE-Tr. The Chinese texts are preprocessed using ICTCLAS[4] for word segmentation.

4.1.3 Evaluation Metrics

Following Lavelli's research on evaluation of information extraction (Lavelli et al., 2008), we perform evaluation as follows.

1. We evaluate each $attr$ separately.
2. For each $attr$, there is exactly one $value$ extracted.
3. No alternative occurrence of real $value$ is available.
4. The overlap ratio is used in this paper rather than "exactly matching" and "containing".

Given an extracted $value\ v' = \{w'\}$ and its corresponding real $value\ v = \{w\}$, two measurements for evaluating the overlap ratio are defined:

recall: the rate of matched tokens w.r.t. the real $value$. It can be calculated using

$$R(v', v) = \frac{|v \cap v'|}{|v|}$$

[2] `http://dumps.wikimedia.org/`

[3] `http://openapi.baidu.com/service`

[4] `http://www.ictclas.org/`

precision: the rate of matched tokens w.r.t. the extracted $value$. It can be calculated using

$$P(v', v) = \frac{|v \cap v'|}{|v'|}$$

We use the average of these two measures to evaluate the performance of our extractor as follows:

$$R = avg(R_i(v', v)) \;\; a_i \in A_{test}$$

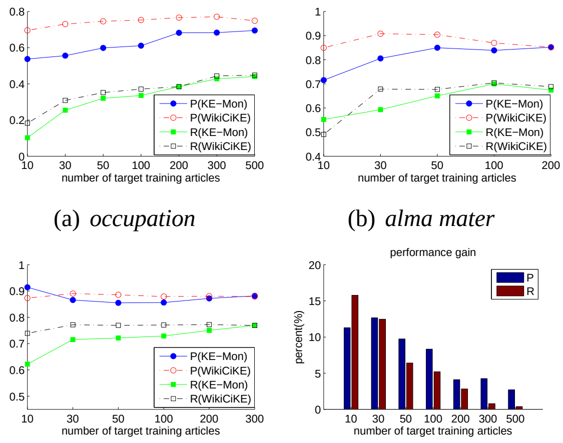
$$P = avg(P_i(v', v)) \;\; a_i \in A_{test} \; and \; v_i{}' \neq \emptyset$$

The *recall* and *precision* range from 0 to 1 and are first calculated on a single instance and then averaged over the testing instances.

4.2 Comparison with KE-Mon

In these experiments, WikiCiKE trains extractors on $A_S \cup A_T$, and KE-Mon trains extractors just on A_T. We incrementally increase the number of target training articles from 10 to 500 (if available) to compare WikiCiKE with KE-Mon in different situations. We use the first testing data set A_{test} to evaluate the results.

Figure 4 and Table 5 show the experimental results on TEMPLATE PERSON and FILM. We can see that WikiCiKE outperforms KE-Mon on all three attributions especially when the number of target training samples is small. Although the *recall* for *alma mater* and the *precision* for *nationality* of WikiCiKE are lower than KE-Mon when only 10 target training articles are available, WikiCiKE performs better than KE-Mon if we take into consideration both *precision* and *recall*.



(a) *occupation*

(b) *alma mater*

(c) *nationality*

(d) average improvements

Figure 4: Results for TEMPLATE PERSON.

Figure 4(d) shows the average improvements yielded by WikiCiKE w.r.t KE-Mon on TEMPLATE PERSON. We can see that WikiCiKE yields significant improvements when only a few articles are available in target language and the improvements tend to decrease as the number of target articles is increased. In this case, the articles in the target language are sufficient to train the extractors alone.

#	KE-Mon		WikiCiKE	
	P	***R***	***P***	***R***
10	81.1%	63.8%	**90.7%**	66.3%
30	78.8%	64.5%	87.5%	69.4%
50	80.7%	66.6%	87.7%	**72.3%**
100	82.8%	68.2%	87.8%	72.1%
200	83.6%	70.5%	87.1%	73.2%
300	85.2%	72.0%	89.1%	76.2%
500	86.2%	73.4%	88.7%	75.6%

# Number of the target training articles.

Table 5: Results for *country* in TEMPLATE FILM.

4.3 Comparison with KE-Tr

We compare WikiCiKE with KE-Tr on the second testing data set A'_{test}.

From Table 6 it can be clearly observed that WikiCiKE significantly outperforms KE-Tr both in *precision* and *recall*. The reasons why the recall of KE-Tr is extremely low are two-fold. First, because of the limit of cross-lingual links and infoboxes in English Wikipedia, only a very small set of values is found by KE-Tr. Furthermore, many values obtained using the translator are incorrect because of translation errors. WikiCiKE uses translators too, but it has better tolerance to translation errors because the extracted value is from the target article texts instead of the output of translators.

Attribute	KE-Tr		WikiCiKE	
	P	***R***	***P***	***R***
occupation	27.4%	3.40%	64.8%	26.4%
nationality	66.3%	4.60%	70.0%	**55.0%**
alma mater	66.7%	0.70%	**76.3%**	8.20%

Table 6: Results of WikiCiKE vs. KE-Tr.

4.4 Significance Test

We conducted a significance test to demonstrate that the difference between WikiCiKE and KE-Mon is significant rather than caused by statistical errors. As for the comparison between WikiCiKE and KE-Tr, significant improvements brought by

WikiCiKE can be clearly observed from Table 6 so there is no need for further significance test. In this paper, we use McNemar's significance test (Dietterich and Thomas, 1998).

Table 7 shows the results of significance test calculated for the average on all tested attributes. When the number of target training articles is less than 100, the χ is much less than 10.83 that corresponds to a significance level 0.001. It suggests that the chance that WikiCiKE is not better than KE-Mon is less than 0.001.

#	10	30	50	100	200	300	500
χ	179.5	107.3	51.8	32.8	4.1	4.3	0.3

# Number of the target training articles.

Table 7: Results of Significance Test.

4.5 Overall Analysis

As shown in above experiments, we can see that WikiCiKE outperforms both KE-Mon and KE-Tr. When only 30 target training samples are available, WikiCiKE reaches comparable performance of KE-Mon using 300-500 target training samples. Among all of the 72 attributes in TEMPLATE PERSON of Chinese Wikipedia, 39 (54.17%) and 55 (76.39%) attributes have less than 30 and 200 labeled articles respectively. We can see that WikiCiKE can save considerable human labor when no sufficient target training samples are available.

We also examined the errors by WikiCiKE and they can be categorized into three classes. For attribute $occupation$ when 30 target training samples are used, there are 71 errors. The first category is caused by incorrect word segmentation (40.85%). In Chinese, there is no space between words so we need to segment them before extraction. The result of word segmentation directly decide the performance of extraction so it causes most of the errors. The second category is because of the incomplete infoboxes (36.62%). In evaluation of KE-Mon, we directly use the values in infoboxex as golden values, some of them are incomplete so the correct predicted values will be automatically judged as the incorrect in these cases. The last category is mismatched words (22.54%). The predicted value does not match the golden value or a part of it. In the future, we can improve the performance of WikiCiKE by polishing the word segmentation result.

5 Related Work

Some approaches of knowledge extraction from the open Web have been proposed (Wu et al., 2012; Yates et al., 2007). Here we focus on the extraction inside Wikipedia.

5.1 Monolingual Infobox Extraction

KYLIN is the first system to autonomously extract the missing infoboxes from the corresponding article texts by using a self-supervised learning method (Wu and Weld, 2007). KYLIN performs well when enough training data are available. Such techniques as shrinkage and retraining are proposed to increase the recall from English Wikipedia's long tail of sparse classes (Wu et al., 2008; Wu and Weld, 2010). Different from Wu's research, WikiCiKE is a cross-lingual knowledge extraction framework, which leverags rich knowledge in the other language to improve extraction performance in the target Wikipedia.

5.2 Cross-lingual Infobox Completion

Current translation based methods usually contain two steps: cross-lingual attribute alignment and value translation. The attribute alignment strategies can be grouped into two categories: cross-lingual link based methods (Bouma et al., 2009) and classification based methods (Adar et al., 2009; Nguyen et al., 2011; Aumueller et al., 2005; Adafre and de Rijke, 2006; Li et al., 2009). After the first step, the value in the source language is translated into the target language. E. Adar's approach gives the overall precision of 54% and recall of 40% (Adar et al., 2009). However, recall of these methods is limited by the number of equivalent cross-lingual articles and the number of infoboxes in the source language. It is also limited by the quality of the translators. WikiCiKE attempts to mine the missing infoboxes directly from the article texts and thus achieves a higher recall compared with these methods as shown in Section 4.3.

5.3 Transfer Learning

Transfer learning can be grouped into four categories: instance-transfer, feature-representation-transfer, parameter-transfer and relational-knowledge-transfer (Pan and Yang, 2010). TrAdaBoost, the instance-transfer approach, is an extension of the AdaBoost algorithm, and demonstrates better transfer ability than traditional learning techniques (Dai et al., 2007). Transfer learning have been widely studied for classification, regression, and cluster problems. However, few efforts have been spent in the information extraction tasks with knowledge transfer.

6 Conclusion and Future Work

In this paper we proposed a general cross-lingual knowledge extraction framework called WikiCiKE, in which extraction performance in the target Wikipedia is improved by using rich infoboxes in the source language. The problems of topic drift and translation error were handled by using the TrAdaBoost model. Chinese-English experimental results on four typical attributes showed that WikiCiKE significantly outperforms both the current translation based methods and the monolingual extraction methods. In theory, WikiCiKE can be applied to any two wiki knowledge based of different languages.

We have been considering some future work. Firstly, more attributes in more infobox templates should be explored to make our results much stronger. Secondly, knowledge in a minor language may also help improve extraction performance for a major language due to the cultural and religion differences. A bidirectional cross-lingual extraction approach will also be studied. Last but not least, we will try to extract multiple *attr-value* pairs at the same time for each article.

Furthermore, our work is part of a more ambitious agenda on exploitation of linked data. On the one hand, being able to extract data and knowledge from multilingual sources such as Wikipedia could help improve the coverage of linked data for applications. On the other hand, we are also investigating how to possibly integrate information, including subjective information (Sensoy et al., 2013), from multiple sources, so as to better support data exploitation in context dependent applications.

Acknowledgement

The work is supported by NSFC (No. 61035004), NSFC-ANR (No. 61261130588), 863 High Technology Program (2011AA01A207), FP7-288342, FP7 K-Drive project (286348), the EPSRC WhatIf project (EP/J014354/1) and THU-NUS NExT Co-Lab. Besides, we gratefully acknowledge the assistance of Haixun Wang (MSRA) for improving the paper work.

References

S. Fissaha Adafre and M. de Rijke. 2006. Finding Similar Sentences across Multiple Languages in Wikipedia. *EACL 2006 Workshop on New Text: Wikis and Blogs and Other Dynamic Text Sources*.

Sisay Fissaha Adafre and Maarten de Rijke. 2005. Discovering Missing Links in Wikipedia. *Proceedings of the 3rd International Workshop on Link Discovery*.

Eytan Adar, Michael Skinner and Daniel S. Weld. 2009. Information Arbitrage across Multi-lingual Wikipedia. *WSDM'09*.

David Aumueller, Hong Hai Do, Sabine Massmann and Erhard Rahm". 2005. Schema and ontology matching with COMA++. *SIGMOD Conference'05*.

Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak and Sebastian Hellmann. 2009. DBpedia - A crystallization Point for the Web of Data. *J. Web Sem.*.

Christian Bizer, Tom Heath, Kingsley Idehen and Tim Berners-Lee. 2008. Linked data on the web (L-DOW2008). *WWW'08*.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim S-turge and Jamie Taylor. 2008. Freebase: a Collaboratively Created Graph Database for Structuring Human Knowledge. *SIGMOD'08*.

Gosse Bouma, Geert Kloosterman, Jori Mur, Gertjan Van Noord, Lonneke Van Der Plas and Jorg Tiedemann. 2008. Question Answering with Joost at CLEF 2007. *Working Notes for the CLEF 2008 Workshop*.

Gosse Bouma, Sergio Duarte and Zahurul Islam. 2009. Cross-lingual Alignment and Completion of Wikipedia Templates. *CLIAWS3 '09*.

Wenyuan Dai, Qiang Yang, Gui-Rong Xue and Yong Yu. 2007. Boosting for Transfer Learning. *ICM-L'07*.

Dietterich and Thomas G. 1998. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Comput.*.

Sergio Ferrández, Antonio Toral, íscar Ferrández, Antonio Ferrández and Rafael Muñoz. 2009. Exploiting Wikipedia and EuroWordNet to Solve Cross-Lingual Question Answering. *Inf. Sci.*.

Aidan Finn and Nicholas Kushmerick. 2004. Multilevel Boundary Classification for Information Extraction. *ECML*.

Achille Fokoue, Felipe Meneguzzi, Murat Sensoy and Jeff Z. Pan. 2012. Querying Linked Ontological Data through Distributed Summarization. *Proc. of the 26th AAAI Conference on Artificial Intelligence (AAAI2012)*.

Yoav Freund and Robert E. Schapire. 1997. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.*.

Norman Heino and Jeff Z. Pan. 2012. RDFS Reasoning on Massively Parallel Hardware. *Proc. of the 11th International Semantic Web Conference (ISWC2012)*.

Aidan Hogan, Jeff Z. Pan, Axel Polleres and Yuan Ren. 2011. Scalable OWL 2 Reasoning for Linked Data. *Reasoning Web. Semantic Technologies for the Web of Data*.

Andreas Hotho, Robert Jäschke, Christoph Schmitz and Gerd Stumme. 2006. Information Retrieval in Folksonomies: Search and Ranking. *ESWC'06*.

John D. Lafferty, Andrew McCallum and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *ICML'01*.

Alberto Lavelli, MaryElaine Califf, Fabio Ciravegna, Dayne Freitag, Claudio Giuliano, Nicholas Kushmerick, Lorenza Romano and Neil Ireson. 2008. Evaluation of Machine Learning-based Information Extraction Algorithms: Criticisms and Recommendations. *Language Resources and Evaluation*.

Juanzi Li, Jie Tang, Yi Li and Qiong Luo. 2009. RiMOM: A Dynamic Multistrategy Ontology Alignment Framework. *IEEE Trans. Knowl. Data Eng.*.

Xiao Ling, Gui-Rong Xue, Wenyuan Dai, Yun Jiang, Qiang Yang and Yong Yu. 2008. Can Chinese Web Pages be Classified with English Data Source?. *WWW'08*.

Sheila A. McIlraith, Tran Cao Son and Honglei Zeng. 2001. Semantic Web Services. *IEEE Intelligent Systems*.

Thanh Hoang Nguyen, Viviane Moreira, Huong Nguyen, Hoa Nguyen and Juliana Freire. 2011. Multilingual Schema Matching for Wikipedia Infoboxes. *CoRR*.

Jeff Z. Pan and Edward Thomas. 2007. Approximating OWL-DL Ontologies. *22nd AAAI Conference on Artificial Intelligence (AAAI-07)*.

Jeff Z. Pan and Ian Horrocks. 2007. RDFS(FA): Connecting RDF(S) and OWL DL. *IEEE Transaction on Knowledge and Data Engineering. 19(2): 192 - 206*.

Jeff Z. Pan and Ian Horrocks. 2006. OWL-Eu: Adding Customised Datatypes into OWL. *Journal of Web Semantics*.

Sinno Jialin Pan and Qiang Yang. 2010. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.*.

Nick Roussopoulos, Stephen Kelley and Frédéric Vincent. 1995. Nearest Neighbor Queries. *SIGMOD Conference'95*.

Murat Sensoy, Achille Fokoue, Jeff Z. Pan, Timothy Norman, Yuqing Tang, Nir Oren and Katia Sycara. 2013. Reasoning about Uncertain Information and Conflict Resolution through Trust Revision. *Proc. of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS2013)*.

Fabian M. Suchanek, Gjergji Kasneci and Gerhard Weikum. 2007. Yago: a Core of Semantic Knowledge. *WWW'07*.

Max Volkel, Markus Krotzsch, Denny Vrandecic, Heiko Haller and Rudi Studer. 2006. Semantic Wikipedia. *WWW'06*.

Zhichun Wang, Juanzi Li, Zhigang Wang and Jie Tang. 2012. Cross-lingual Knowledge Linking across Wiki Knowledge Bases. *21st International World Wide Web Conference*.

Daniel S. Weld, Fei Wu, Eytan Adar, Saleema Amershi, James Fogarty, Raphael Hoffmann, Kayur Patel and Michael Skinner. 2008. Intelligence in Wikipedia. *AAAI'08*.

Fei Wu and Daniel S. Weld. 2007. Autonomously Semantifying Wikipedia. *CIKM'07*.

Fei Wu and Daniel S. Weld. 2010. Open Information Extraction Using Wikipedia. *ACL'10*.

Fei Wu, Raphael Hoffmann and Daniel S. Weld. 2008. Information Extraction from Wikipedia: Moving down the Long Tail. *KDD'08*.

Wentao Wu, Hongsong Li, Haixun Wang and Kenny Qili Zhu. 2012. Probase: a Probabilistic Taxonomy for Text Understanding. *SIGMOD Conference'12*.

Alexander Yates, Michael Cafarella, Michele Banko, Oren Etzioni, Matthew Broadhead and Stephen Soderland. 2007. TextRunner: Open Information Extraction on the Web. *NAACL-Demonstrations'07*.

Xinfeng Zhang, Xiaozhao Xu, Yiheng Cai and Yaowei Liu. 2009. A Weighted Hyper-Sphere SVM. *ICNC(3)'09*.