Error Link Detection and Correction in Wikipedia

Chengyu Wang, Rong Zhang, Xiaofeng He*, Aoying Zhou
School of Computer Science and Software Engineering
East China Normal University, Shanghai, China
chywang2013@gmail.com, {rzhang,xfhe,ayzhou}@sei.ecnu.edu.cn

ABSTRACT

The hyperlink structure of Wikipedia forms a rich semantic network connecting entities and concepts, enabling it as a valuable source for knowledge harvesting. Wikipedia, as crowd-sourced data, faces various data quality issues which significantly impacts knowledge systems depending on it as the information source. One such issue occurs when an anchor text in a Wikipage links to a wrong Wikipage, causing the error link problem. While much of previous work has focused on leveraging Wikipedia for entity linking, little has been done to detect error links.

In this paper, we address the error link problem, and propose algorithms to detect and correct error links. We introduce an efficient method to generate candidate error links based on iterative ranking in an *Anchor Text Semantic Network*. This greatly reduces the problem space. A more accurate pairwise learning model was used to detect error links from the reduced candidate error link set, while suggesting correct links in the same time. This approach is effective when data sparsity is a challenging issue. The experiments on both English and Chinese Wikipedia illustrate the effectiveness of our approach. We also provide a preliminary analysis on possible causes of error links in English and Chinese Wikipedia.

Keywords

error link; Wikipedia; LinkRank; pairwise learning

1. INTRODUCTION

Wikipedia serves as a valuable data source for knowledge sharing to accomplish various tasks, such as knowledge base population [12, 25], taxonomy construction [13, 20], entity linking [7, 24], etc. Although the collaboratively generated data in Wikipedia contains the "wisdom of the crowds" and is updated on a daily basis, a number of data quality issues exist in Wikipedia, which negatively impact the credibility of Wikipedia, particularly when it is used as the data source to build other knowledge systems. Some of the well-known data quality issues include the lack of cross-lingual links between articles of different language versions [29], missing links between Wikipages [26], vandalism behaviors which intentionally destroy the contents of Wikipages [27], and the controversy issues where contributors have different viewpoints on a certain subject [3]. Research efforts have been focused on these fields to improve the data quality of Wikipedia.

In this paper, we pay attention to a different problem, the error link issue. Error link occurs when an anchor text in one Wikipage points to another Wikipage whose description of the entity is not what anchor text actually means. Error link phenomenon is mostly due to the multiple senses (polysemy) or ambiguity of the anchor text in which a link is created between anchor text and the entity with different meaning. Take a case from English Wikipedia as an example. Wikipage *Facebook*[1] gives a brief introduction to Facebook. Anchor text "Java" in this page links to Wikipage *Java* [2] (an island in Indonesia). But based on the context, we are pretty sure that the contributor of this Wikipage actually refers to the JAVA programming language when mentioning "Java" in sentence "The backend is written in Java ". With high confidence, we treat the link from *Facebook*[1] to *Java*[2] as an error link. This error link can be corrected easily by pointing the anchor text to *Java (programming language)*[3] .

High linking quality can be achieved by frequent checking and validation by contributors of Wikipedia, but manual checking is prohibitively costly because the number of entities and links increases rapidly as new Wikipages are continuously added. For instance, approximately 30,000 new articles are created per month in Wikipedia [3]. Furthermore, as Weaver et al. estimates, the average error rate of Wikipedia statements is 2.8% [30]. Therefore, without automatic checking, linking errors are almost inevitable.

The task of identifying error links and correcting them is important for several reasons: i) it helps to maintain high quality and credibility of Wikipedia contents; ii) it refines the semantic relations between entities in Wikipedia and potentially improves the performance of tasks such as semantic computing; and iii) applications (e.g., knowledge bases such as YAGO [25]) which take Wikipedia as input will benefit from higher quality data source.

To solve the error link problem, we need to study the semantic relations between anchor texts and entities in Wikipedia, which is similar to Entity Linking (EL) [7, 22, 24]. However, existing EL techniques are undesirable to solve the error link task due to the serious *data sparsity issue* and non-existent *ground truth assumption*. There are only a small number of error links, while we have to take the entire Wikipedia content and hyperlink structure as input when trying to identify the errors. This data sparsity issue makes it computationally expensive to directly predict whether each anchor text

*Corresponding author.

CIKM'16, October 24-28, 2016, Indianapolis, IN, USA.

DOI: http://dx.doi.org/10.1145/2983323.2983705

[1]https://en.wikipedia.org/wiki/Facebook
[2]https://en.wikipedia.org/wiki/Java
[3]https://en.wikipedia.org/wiki/Java_(programming_language)

in Wikipedia is correctly linked to the target Wikipage. Besides, Wikipedia is normally treated as the "ground truth" in EL research. For instance, the prior link probability of a text mention given an entity is computed based on the Wikipedia link structure [22, 23, 24]. However, if we wish to detect and correct error links by linking anchor texts to Wikipages directly, the accuracy may be negatively affected because we need to compute linking quality metrics based on the erroneous link structure.

In this paper, we take a two-stage approach to solve the error link problem. Because error links are mostly caused by the ambiguity of anchor texts, in the first stage, we identify ambiguous anchor texts and extract *Anchor Text Semantic Networks* (**ATSN** for short) which capture the hyperlink structure of entities related to those ambiguous anchor texts. A *LinkRank* algorithm is proposed to generate candidate error links by calculating the "goodness" of the links. In this stage, our goal is to deal with the data sparsity issue by reducing the problem space to a small set of suspicious links. In the second stage, we propose a pairwise supervised learning model on the reduced data set to single out error links with higher precision, while making link correction suggestions in the same time. In summary, we make the following contributions.

- We formalize the error link problem. Based on ATSN, we propose a *LinkRank* algorithm to generate candidate error links from entire Wikipedia link set. This reduces the problem space considerably.
- We train a pairwise supervised learning model to perform error link detection and correction with higher accuracy on the reduced data set. Graph-based features and context-based features are engineered for the model.
- Extensive experiments are conducted on both English and Chinese Wikipedia to illustrate the effectiveness of the proposed approach. We also perform a preliminary analysis on error links we identified, and present the possible causes.

The rest of this paper is organized as follows. Section 2 summarizes the related work. We define the error link problem formally in Section 3, and introduce our solution briefly. Details of our two-stage approach for addressing the error link problem are described in Section 4 and Section 5. Experimental results are presented in Section 6. We give a preliminary analysis on possible causes of error links in Section 7, and conclude our paper and discuss the future work in Section 8.

2. RELATED WORK AND DISCUSSION

The error link problem is inspired by EL and other similar tasks, which analyze semantic relations between entities and text mentions. In this section, we overview the related work, and provide a discussion on the relation between EL and error link problem.

EL focuses on linking a text mention in natural language input or semi-structured input to a named entity in a knowledge base. Given a text mention m and a collection of entities E_m, the EL system selects the entity from E_m that m most probably refers to. A recent survey on EL can be found in [22]. Various paradigms have been utilized to solve the EL problem, including machine learning-based models [1, 4, 19, 24], graph-based ranking [8, 9], probabilistic based approaches [6], etc.

Classification models solve the EL problem by predicting whether a text mention refers to a certain entity. Pilz and Paaß [19] employ the SVM classifier based on thematic features of text mention's and entity's context. Aktolga et al. [1] utilize the logistic regression model for EL to perform classification. However, the number of negative instances are far more than positive instances in the EL task. To handle this imbalance issue, many EL systems adopt the Learning to Rank framework to select the most probable entity. *LINDEN* [24] utilizes the max-margin technique to learn feature weights of a score function in order to give a rank to all the candidate entities for each text mention. It performs EL by selecting the entity with the highest rank if the score value is higher than a learned threshold. Dredze et al. [4] model the EL task as the optimization problem: given a feature function f, the correct entity y should receive a higher score $f(y)$ plus a margin than other entities.

Graph-based ranking methods can improve the effectiveness of the EL system by selecting the entity with highest ranking score. Han et al. [8] propose a referent graph to model the global topical interdependence to make different entity linking decisions in one document. Hoffart et al. [9] develop a similar model which represents the EL model as mention-entity graph. Besides, probabilistic methods are employed to perform EL as well. In [6], Han and Sun introduce an entity-mention model based on a probabilistic, generative approach. In the model, the distribution of entities in document, possible text mentions given an entity and possible context of an entity are encoded so that the model can make the linking decision based on these three evidences.

Besides adding links between Wikipages, several works focus on adding links in general documents that link to Wikipages, which is known as *Wikification*. Wikification can be treated as a generalized EL task which aims to link all the text mentions in a document to the Wikipedia knowledge base. Mihalcea and Csomai [14] develop a system called *Wikify!* to label links in the document to Wikipedia articles using keyword extraction and word sense disambiguation techniques. Milne and Witten [15] apply a supervised learning technique to link texts to Wikipedia. Granitzer et al. [5] introduce a content based strategy which aims to link Wikipages.

Due to the popularity of Wikipedia, other EL-like tasks have been addressed to study the relations between text mentions and entities in Wikipedia. For example, Wikipedia link discovery aims to add links to newly added Wikipages to existing Wikipages. Sunercan and Birturk [26] combine different approaches by considering the link information, Wikipedia category system and contextual linkness. In [16], Noraset et al. introduce a system *3W* to identify mentions in Wikipedia and then add links to their referent entities.

While previous work of missing link discovery enriches the link structure in Wikipedia, the error link problem addressed in this paper tries to correct the errors in the link structure. Wang et al. [28] discuss several data quality issues in Chinese Wikipedia, including the error link problem. Paulheim and Bizer [18] identify incorrect statements in RDF linked datasets, and evaluate the algorithm on DBPedia and NELL. *DSNotify* [21] is a system to maintain the links between dynamic linked datasets from a resource-centric perspective. Pateman and Johnson [17] propose to highlight the Wikipedia link errors and find possible alternatives by analyzing the "semantic contribution" of Wikipedia links. However, none of the prior work proposes a general framework to detect and correct error links accurately and automatically in Wikipedia. The error link problem has close relations but clear distinctions compared to existing EL (or similar) tasks, discussed as follows.

Both EL and error link problem study the relations between text mentions and entities, which can be addressed considering various signals in the contextual data, including the semantic relatedness between entities, the contextual information of text mentions and entities, external knowledge bases, etc. In this sense, the problem of error link can be regarded as a special case of entity linking. However, error link problem distinguishes itself from existing EL-related tasks in the following aspects: (i) The central task of EL is *ranking*, which assigns each candidate entity a score given a text mention. In contrast, the error link problem faces the challenge of data sparsity and thus focuses on *anomaly detection*, aiming to find a small number of error links from tens of millions of Wikipedia links. (ii) In EL research, Wikipedia is normally treated as the "ground truth" to provide link and contextual information for entities [14, 22, 23, 24], which is utilized as a knowledge base for EL. In error link problem, errors inside the Wikipedia knowledge base need to be identified. Further experimental study and analysis on this issue will be presented in Section 6.

3. ERROR LINK PROBLEM IN WIKIPEDIA

In this section, we begin to formalize the error link problem in Wikipedia. Then we give a brief introduction to the two-stage approach which solves this problem.

3.1 Problem Statement

Wikipedia consists of millions of entities and links, which can be treated as a large knowledge repository. Except irrelevant pages such as administrative pages, template pages, etc., each remaining Wikipage describes a unique entity [25]. The title of a Wikipage is regarded as the name for the entity. In many Wikipages, there are several anchor texts with links to other pages. If the anchor text m in Wikipage e_i links to Wikipage e_j, then it means there is a link $l_{i,j}$ between e_i and e_j, and m is a text mention for entity e_j. An error link occurs if an anchor text m in Wikipage e_i links to Wikipage e_j, while e_j is not the correct entity to be linked to. Because entities and Wikipages have one-to-one correspondence, for simplicity, we do not distinguish between Wikipages and entities. For example, e_i can refer to an entity or the Wikipage describing this entity.

The goal of error link detection and correction is to discover error links, and try to suggest the correct links. Given the entire Wikipedia dataset W as input, the proposed approach in this paper automatically generates triples $< l_{i,j}, l_{i,j'} >$ such that link $l_{i,j}$ is an existing error link in W and $l_{i,j'}$ is the corresponding correct link. It denotes an anchor text in Wikipage e_i links to Wikipage e_j erroneously, while it should link to Wikipage $e_{j'}$ instead.

However, we need to point out that Wikipedia is still an incomplete knowledge repository, which covers only a small portion of entities in real world. For error link correction, this work only considers the situation where the correct target entity exists in Wikipedia. Solving error link problem where correct entities are not found in Wikipedia is left to future work.

3.2 Problem Analysis and Solution Overview

The major challenge of error link detection and correction is that there are only a few error links in Wikipedia, leading to the data sparsity problem. Moreover, to the best of our knowledge, there is no prior work or benchmarks available as "ground truth". Labeling a large number of error links manually is difficult, therefore it is infeasible to apply classification methods directly on the Wikipedia link set due to the data sparsity issue. To alleviate the problem, we split the process into two sub-tasks: i) obtain a candidate error link set that has higher density of error links; and ii) perform supervised error link prediction and correction jointly on the candidate error link set. Accordingly, the solution proposed in this paper adopts a two-stage process to handle the two sub-tasks mentioned above:

- **Candidate Error Link Generation**
  Because the phenomenon of error links is rooted in the ambiguity of anchor text, we mine Wikipedia to construct a dictionary $M = \{(m, E_m)\}$ where m is an ambiguous anchor text and E_m is the set of all possible referent entities for m.

  For each $(m, E_m) \in M$, we construct an ATSN G_m to represent the hyperlink structure of entities in E_m. We propose the *LinkRank* algorithm to calculate the "goodness" of links in G_m which is used to filter the candidate error links. The candidate error link set CL is a collection of $< l_{i,j}, l_{i,j'} >$ pairs where $l_{i,j}$ is a candidate error link and $l_{i,j'}$ is the link which is most probably correct.

- **Link Classification and Correction**
  In order to solve the error link detection and correction simultaneously, we take a pairwise learning model f. Given $< l_{i,j}, l_{i,j'} >$ as input, f predicts whether or not $l_{i,j}$ is an error link and $l_{i,j'}$ is a correct link jointly. Link correction can be done according to the prediction results.

Table 1: Important notations

Notation	Description
e_i	An entity or the Wikipage describing the entity
$l_{i,j}$	A link from Wikipage e_i to e_j
$w_{i,j}$	Weight of $l_{i,j}$ calculated by *LinkRank*
M	Dictionary of ambiguous anchor texts and entities
m	An anchor text, also called mention
E_m	Collection of all possible referent entities for m
G_m	Anchor Text Semantic Network for m
CL	Candidate error link set
CL_m	Candidate error link set w.r.t. m
$SC(e_i \rightarrow e_j)$	Semantic closeness between e_i and e_j
$\vec{v}_G(l_{i,j})$	Graph-based feature vector for $l_{i,j}$
$\vec{v}_C(l_{i,j})$	Context-based feature vector for $l_{i,j}$
$\vec{v}(l_{i,j})$	Feature vector for $l_{i,j}$
$\vec{v}_{PL}(l_{i,j}, l_{i,j'})$	Feature vector for $< l_{i,j}, l_{i,j'} >$ in pairwise learning

Important notations in this paper are summarized in Table 1.

4. CANDIDATE ERROR LINK GENERATION

In this section, we describe how to generate candidate error link set from Wikipedia in detail.

4.1 Dictionary Construction

Wikipedia provides abundant features for the relations between anchor texts and entities. To construct the dictionary M consisting of ambiguous anchor texts and entities, we utilize data sources such as redirect pages, disambiguation pages and hyperlinks in Wikipages to extract all the possible referent entities E_m for anchor text m. The etailed construction method can be found in [24].

After the the dictionary M is created, it can be used to filter links with ambiguous anchor texts. This greatly reduces the search space of error links. An example of the dictionary is shown in Table 2. For example, if a link with anchor text "New York" points to *New York City*, then it is likely that this is an error link, because anchor text "New York" can refer to the magazine *New York* as well.

The dictionary has been heavily used in EL tasks [22, 23, 24]. Compared to existing approaches, the dictionary in the error link task has two significant differences: i) instead of extracting all mention-entity relations, we only focus on ambiguous anchor texts; ii) in EL, the count information for each entity is recorded as prior knowledge [24]. In these approaches, Wikipedia is treated as the "ground truth". While in this paper, we do not make this assumption, but will check whether links in Wikipedia are correct or not.

4.2 Anchor Text Semantic Network

Let $G_W = (V_W, L_W)$ denote the Wikipedia link-graph where V_W is the entity set in Wikipedia and L_W is the link set among entities. Based on the dictionary M constructed, for each anchor text m, we construct an ATSN based on G_W.

Table 2: An example of dictionary M

Anchor Text m	Possible Referent Entity Collection E_m
Java	Java Java (programming language) . . .
New York	New York City New York (magazine) New York (film) . . .

Formally, an ATSN w.r.t. anchor text m is a weighted directed graph $G_m = (V_m, L_m, W_m)$ where V_m is the node set consisting of entities, L_m is the set of links (directed edges), and W_m is the set of weights for the links in L_m.

For an entity e_i, its inlink and outlink nodes are defined as follows: $InLinkNode_i = \{e_j | l_{j,i} \in L_W\}$ and $OutLinkNode_i = \{e_j | l_{i,j} \in L_W\}$. Let $Neighbor(e_i)$ denote the union of inlink and outlink nodes of e_i (i.e., e_i's neighbors). V_m consists of two types of nodes: all entities in E_m and all neighbors of these entities, defined as: $V_m = \bigcup_{e \in E_m} Neighbor(e) \cup E_m$. A link $l_{i,j} \in L_m$ exists iff $e_i \in V_m$, $e_j \in V_m$, and $l_{i,j} \in L_W$. Therefore, there are two types of links in L_m: inlinks and outlinks of entities in E_m, and links that connect neighbors of these entities.

Fig. 1 shows part of the ATSN w.r.t. anchor text "Java". Entities of the ATSN include i) entities that can be referred as "Java" (e.g., *Java*, *Java (programming language)*, see Table 2), and ii) the neighbors of them (e.g., *Facebook*, *PHP*, etc.). It includes links among these two types of nodes, such as *Facebook*→*Java*, *Java (programming language)*→*PHP*, etc.

The ATSN represents the link structure of entities related to the anchor text, therefore the characteristics of error links. It can help us identify candidate error links. The weights W_m can be calculated by *LinkRank* algorithm, described in Section 4.3.

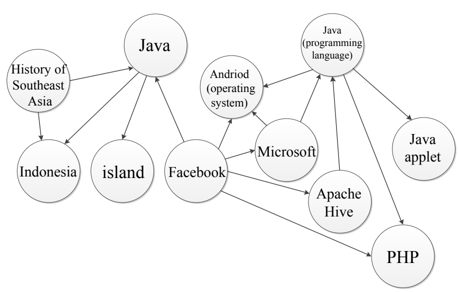


Figure 1: Part of the structure of the ATSN w.r.t. anchor text "Java".

4.3 LinkRank Algorithm

In this section, we propose the *LinkRank* algorithm that calculates the "goodness" of links in G_m. The result is used as the weights W_m for G_m.

In previous work, measurements such as Wikipedia Link-based Measure (WLM) [31] reveal the semantic closeness between two entities, but they only consider the local property of the graph (such as inlinks). However, the link structure in Wikipedia is relatively sparse. For instance, in Chinese Wikipedia, approximately 15% of the entity pairs $< e_i, e_j >$ have no common inlinks for all $l_{i,j} \in L_W$, which makes it impossible for WLM to measure the semantic closeness. The *LinkRank* we introduce ranks all links by exploiting the global structure of an ATSN. Similar to PageRank [2] and HITS [11], *LinkRank* employs an iterative ranking process. but the ranking subjects and the algorithm itself are different from PageRank and HITS. For error link problem, it is important to rank links instead of nodes (entities). The result reveals the "goodness" of a link in an ATSN, instead of a Wikipage.

4.3.1 *Iterative Ranking Process*

The detailed iterative procedure of *LinkRank* is shown below, and summarized in Alg. 1.

In the algorithm, the initial weights $\mathbf{w}^{(0)}$ of all links are assigned uniformly, i.e., $\mathbf{w}^{(0)} = \mathbf{1}$ (Line 3). To update the weights, we use a weight propagation approach according to the link structure of G_m. Let $InLink_i$ and $OutLink_i$ denote the collections of inlinks and outlinks of e_i in G_m, respectively, defined as: $InLink_i = \{l_{j,i} \in L_m | j \neq i\}$ and $OutLink_i = \{l_{i,j} \in L_m | j \neq i\}$. In each iteration n, every link $l_{i,j}$ passes its weight uniformly to its outlinks (Line 11). Thus, the transition weight of $l_{i,j}$ in n^{th} iteration is:

$$u_{i,j}^{(n)} = \frac{1}{|OutLink_j|} \cdot w_{i,j}^{(n-1)}$$

Denote the links that have zero out-degree in G_m as $\overline{L}_m$ (Line 5). These weights can not be passed to other parts of the graph. To deal with this issue, the weights of these links are distributed equally to all the links. Hence, in each iteration, every link $l_{i,j}$ receives the transition weights from $InLink_i$ and weights from $\overline{L}_m$ (Line 14). Thus, the weight update rule for $l_{i,j}$ is expressed as:

$$w_{i,j}^{(n)} = \sum_{l_{k,i} \in InLink_i} u_{k,i}^{(n)} + \frac{1}{|L_m|} \sum_{l_{p,q} \in \overline{L}_m} w_{p,q}^{(n-1)}$$

The iterative process converges if the difference of $\mathbf{w}^{(n)}$ and $\mathbf{w}^{(n-1)}$ is smaller than a small threshold ϵ (Line 16). We take the weight $w_{i,j}^{(n)}$ for $l_{i,j}$ as rank value, denoted as $w_{i,j}$ (Line 22).

Algorithm 1 LinkRank Algorithm

Input: Unweighted ATSN $G_m' = (V_m, L_m)$, threshold ϵ.
Output: ATSN $G_m = (V_m, L_m, W_m)$.
1: $\overline{L}_m = \emptyset$;
2: **for** each $l_{i,j} \in L_m$ **do**
3: $\quad w_{i,j}^{(0)} = 1$;
4: $\quad$ **if** $|Outlink_j| = 0$ **then**
5: $\quad\quad \overline{L}_m = \overline{L}_m \cup \{l_{i,j}\}$;
6: $\quad$ **end if**
7: **end for**
8: $n = 1$;
9: **while** true **do**
10: $\quad$ **for** each $l_{i,j} \in L_m \setminus \overline{L}_m$ **do**
11: $\quad\quad u_{i,j}^{(n)} = \frac{1}{|OutLink_j|} \cdot w_{i,j}^{(n-1)}$;
12: $\quad$ **end for**
13: $\quad$ **for** each $l_{i,j} \in L_m$ **do**
14: $\quad\quad w_{i,j}^{(n)} = \sum_{l_{k,i} \in InLink_i} u_{k,i}^{(n)} + \frac{1}{|L_m|} \sum_{l_{p,q} \in \overline{L}_m} w_{p,q}^{(n-1)}$;
15: $\quad$ **end for**
16: $\quad$ **if** $|\mathbf{w}^{(n)} - \mathbf{w}^{(n+1)}| < \epsilon$ **then**
17: $\quad\quad$ break;
18: $\quad$ **end if**
19: $\quad n = n + 1$;
20: **end while**
21: **for** each $l_{i,j} \in L_m$ **do**
22: $\quad w_{i,j} = w_{i,j}^{(n)}$;
23: **end for**
24: **return** $G_m = (V_m, L_m, W_m)$;

4.3.2 *Matrix Interpretation*

The iterative ranking process can be represented as matrix computation. We assign each link an integer index from 1 to $|L_m|$ and use this index to represent the link. Let $\mathbf{w}^{(i)}$ be an $|L_m| \times 1$ weight vector for all links in G_m. $\mathbf{M}$ is the $|L_m| \times |L_m|$ weight transition matrix. $M_{i,j}$ is the proportion of weight that is passed from the i^{th} link $l_{p,q}$ to the j^{th} link $l_{r,s}$. If $q = r$, $M_{i,j} = \frac{1}{|OutLink_q|}$; otherwise, $M_{i,j} = 0$. To deal with the links with zero out-degree, we add an additional term to $\mathbf{M}$ to represent the weights that are distributed among all links. So the weight transition matrix becomes:

$$\mathbf{S} = \mathbf{M} + \mathbf{a}^T(\frac{1}{|L_m|}\mathbf{1})$$

where $\mathbf{a}$ is an $|L_m| \times 1$ vector. If the i^{th} link $l_{p,q}$ in G_m has zero out-degree, then $a_i = 1$; otherwise, $a_i = 0$. The weight vector $\mathbf{w}$ can be computed recursively as: $\mathbf{w}^{(i+1)} = \mathbf{S} \cdot \mathbf{w}^{(i)}$. Similar to PageRank [2], the equation has a closed-form solution, i.e., the principal eigenvector of transition matrix $\mathbf{S}$.

From the random walk perspective, a random surfer stands at one link $l_{p,q}$ at a time. Each time the algorithm selects a link $l_{q,r}$ from $OutLink_q$ with probability $\frac{1}{|OutLink_q|}$ to make a transfer. If there are no outlinks, the algorithm randomly picks a link, each with probability $\frac{1}{|L_m|}$. Therefore, we can also view the *LinkRank* algorithm as the process of calculating the distribution of location of random surfers on the links.

4.4 Candidate Error Link Detection

In this section, we propose a method to generate candidate error link set CL based on *LinkRank*.

4.4.1 Measuring Semantic Closeness

In Wikipedia, links between Wikipages have correlation with the semantic closeness between entities. If Wikipage e_i should link to Wikipage $e_{j'}$ instead of e_j, then entity e_i should be semantically closer to entity $e_{j'}$ rather than e_j. However, the difficulty is that we can not directly use the "goodness" of link $l_{i,j}$ (such as $w_{i,j}$ calculated by *LinkRank*) to measure the semantic closeness between e_i and e_j. Assume $l_{i,j}$ is an existing error link and $l_{i,j'}$ is the correct one, but not present in Wikipedia. In this case, $w_{i,j}$ is not much meaningful, and $w_{i,j'}$ does not even exist. Other measurements have their own limitation for a sparse link-graph.

In this paper, we take an indirect approach. If entity e_i is semantically closer to entity $e_{j'}$ than e_j, then Wikipage e_i is more likely to connect to the neighbors of Wikipage $e_{j'}$. Consider the case in Fig. 1. In the graph, Wikipage *Facebook* links to Wikipage *Java* instead of *Java (programming language)*. However, Wikipage *Facebook* links to a lot of *Java (programming language)*'s neighbors (i.e., *Microsoft*, *Apache Hive* and *PHP*) while it does not link to *Java*'s neighbors. This signals that the link from Wikipage *Facebook* to Wikipage *Java* is likely to be an error link.

We define the Semantic Closeness (SC) $SC(e_i \rightarrow e_j)$ as the sum of weights of all links from e_i to e_j's neighbors, denoted as:

$$SC(e_i \rightarrow e_j) = \sum_{e_{j'} \in Neighbor(e_j) \wedge l_{i,j'} \in L_m} w_{i,j'}$$

The meaning of semantic closeness $SC(e_i \rightarrow e_j)$ implies that: i) large rank value of links indicates there are "good" links from e_i to e_j's neighbors; ii) larger number of links from e_i to e_j's neighbors is a sign of close connection between e_i and e_j.

4.4.2 Algorithm for Candidate Error Link Detection

We now introduce our algorithm in detail. The procedure is illustrated in Alg. 2.

For an ambiguous anchor text m, let CL_m denote the candidate error link set w.r.t. m. Consider a link $l_{i,j}$ in ATSN G_m. If $e_i \notin E_m$ and $e_j \in E_m$, then we calculate the semantic closeness between e_i and all the entities in E_m (Line 5). Let $e_{j'}$ denote the entity in E_m that e_i is semantically closest to (Line 7). To check whether $l_{i,j}$ is an error link, we compare the semantic closeness between $< e_i, e_j >$ and $< e_i, e_{j'} >$ pairs. Here, we employ a heuristic rule: if the following inequality holds

$$\frac{SC(e_i \rightarrow e_{j'}) - SC(e_i \rightarrow e_j)}{SC(e_i \rightarrow e_{j'})} > \tau$$

where τ is a predefined threshold ($\tau \in (0, 1)$), then $l_{i,j}$ is regarded as an candidate error link, and the most probably correct link is $l_{i,j'}$. The link pair $< l_{i,j}, l_{i,j'} >$ is added to CL_m (Line 8). For example, given the anchor text "Java", we search for all possible referent entities from the dictionary and retrieve all the links that point to an entity with the surface name "Java" from the ATSN. For each link (e.g. $Facebook \rightarrow Java$), we decide whether it is an error link and enlarge the candidate error link set.

The method avoids direct processing on the entire big link-graph of Wikipedia. We only need to process each ATSN G_m related to each ambiguous anchor text m in dictionary M. The final candidate error link set CL is the union of all CL_m.

Algorithm 2 Candidate Error Link Detection Algorithm

Input: ATSN $G_m = (V_m, L_m, W_m)$, entity set E_m, threshold τ.
Output: Candidate error link set CL_m w.r.t. m.

1: $CL_m = \emptyset$;
2: **for** each $l_{i,j} \in L_m$ **do**
3:   **if** $e_j \in E_m$ and $e_i \notin E_m$ **then**
4:     **for** each $e_k \in E_m$ **do**
5:       $SC(e_i \rightarrow e_k) = \sum_{e_{j'} \in Neighbor(e_k) \wedge l_{i,j'} \in L_m} w_{i,j'}$;
6:     **end for**
7:     $e_{j'} = \mathrm{argmax}_{e \in E_m \setminus \{e_j\}} SC(e_i \rightarrow e)$;
8:     **if** $\frac{SC(e_i \rightarrow e_{j'}) - SC(e_i \rightarrow e_j)}{SC(e_i \rightarrow e_{j'})} > \tau$ **then**
9:       $CL_m = CL_m \cup \{< l_{i,j}, l_{i,j'} >\}$;
10:     **end if**
11:   **end if**
12: **end for**
13: **return** CL_m;

5. LINK CLASSIFICATION AND CORRECTION

The candidate error link set contains higher density of error links and corresponding possibly correct links. In this section, we propose a supervised pairwise learning technique to predict error links with high precision. We also provide link correction suggestions based on prediction results in the same time.

5.1 Feature Definition

Several signals are useful for identifying error links, including graph-based and context-based features.

5.1.1 Graph-Based Features

The graph-based features can be directly derived from the hyperlink structure of Wikipedia. We do not take the semantic closeness of entities as a feature. This is because we have utilized semantic closeness to identify candidate links. If it is false positive, it will only enforce the error in the training process. Instead, we define the following features from the graph.

Inlink Similarity Feature. The inlink similarity between e_i and e_j can be measured as the Jaccard similarity between $InLinkNode_i$

and $InLinkNode_j$. In our statistical analysis, about 15% of entity pairs $< e_i, e_j >$ have no common inlinks for all links $l_{i,j}$ in Chinese Wikipedia, which results in the zero value in Jaccard similarity. However, we have noticed that the number of inlinks of these entities are different. To emphasize the difference, we propose the smoothed Jaccard similarity as the feature, defined as follows:

$$ILS(i,j) = \frac{|InLinkNode_i \cap InLinkNode_j| + 1}{|InLinkNode_i \cup InLinkNode_j| + 1}$$

Outlink Similarity Feature. Similarly, we define the outlink similarity as follows:

$$OLS(i,j) = \frac{|OutLinkNode_i \cap OutLinkNode_j| + 1}{|OutLinkNode_i \cup OutLinkNode_j| + 1}$$

We observe that, if e_i should not link to e_j, e_i's neighbors have a low probability to connect to e_j. For example, in Fig. 1, the neighbors of *Facebook* (e.g. *Microsoft*, *Apache Hive*, *PHP*) link to *Java_(programming language)*, rather than *Java*. Based on the intuition, we define inlink/outlink relatedness features. In Fig. 2, we present the inlink and outlink relatedness distributions of all Wikipedia links and error links in Chinese Wikipedia.

Inlink Relatedness Feature. Inlink relatedness is defined as the fraction of number of entities in $InLinkNode_i$ that link to e_j and the size of $InLinkNode_i$ in total, shown as follows:

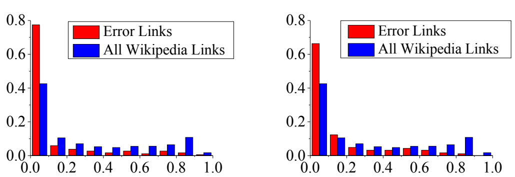
$$ILR(i,j) = \frac{|\{e_k \in InLinkNode_i | l_{k,j} \in L_m\}|}{|InLinkNode_i|}$$

Outlink Relatedness Feature. Similarly, the observation can be applied to outlinks, too. The outlink relatedness w.r.t. e_i and e_j is:

$$OLR(i,j) = \frac{|\{e_k \in OutLinkNode_i | l_{k,j} \in L_m\}|}{|OutLinkNode_i|}$$

The inlink/outlink relatedness features can reveal the characteristics of error links. Fig. 2 shows that error links tend to have lower values of inlink and outlink relatedness. Therefore, the graph-based feature vector for $l_{i,j}$ is:

$$\vec{v}_G(l_{i,j}) =< ILS(i,j), OLS(i,j), ILR(i,j), OLR(i,j) >$$



(a) Inlink relatedness (b) Outlink relatedness

Figure 2: Link relatedness distributions.

5.1.2 Context-Based Features

The contextual information is important in distinguishing error vs. correct links. In this section, we introduce context-based features to measure the semantic similarity between two entities e_i and e_j.

Contextual Similarity Feature. In Wikipedia, we can extract words in the description text of Wikipage e_i as the contextual content for the entity e_i. We remove punctuations, stop words and other meaningless symbols in the text. We denote the multiset of n-grams in Wikipage e_i as S_i, which represents the context for e_i.

Similar to [23], for a link $l_{i,j}$, the contextual similarity between e_i and e_j can be calculated as the cosine of the word vector representations of S_i and S_j:

$$CS(i,j) = \frac{S_i^T \cdot S_j}{\|S_i\|_2 \cdot \|S_j\|_2}$$

Frequent Contextual Similarity Feature. Using all the n-grams in $S_i \cup S_j$ to generate word vectors will lead to high dimensionality due to the large number of n-grams. We only take the top-k most frequent n-grams in $S_i \cup S_j$ to form word vector representation. Denote FS_i and FS_j as the multisets of frequent n-grams in Wikipage e_i and e_j, respectively. The frequent contextual similarity feature is defined as:

$$FCS(i,j) = \frac{FS_i^T \cdot FS_j}{\|FS_i\|_2 \cdot \|FS_j\|_2}$$

Thus, context-based feature vector for $l_{i,j}$ is represented as:

$$\vec{v}_C(l_{i,j}) =< CS(i,j), FCS(i,j) >$$

5.2 Pairwise Learning

One approach to detect error links is to treat the problem as binary classification on each link $l_{i,j}$. This requires an absolute measure of "goodness". In this way, we need to build a model to classify a link first, then design an algorithm to make link corrections.

In the pairwise ranking approach, the model input is a link pair $< l_{i,j}, l_{i,j'} >\in CL$. Candidate error link $l_{i,j}$ can be evaluated together with other links (i.e., a probably correct link $l_{i,j'}$) for the same anchor text, which avoids being mapped to a global scale of "goodness" [10]. A set of graph-based and context-based features w.r.t. to a link $l_{i,j}$ are engineered, and transformed as follows to fit the pairwise model. Let $\vec{v}(l_{i,j}) =< \vec{v}_G(l_{i,j}), \vec{v}_C(l_{i,j}) >$ denote the feature vector for the link $l_{i,j}$. Given a candidate error link $l_{i,j}$ and a probably correct link $l_{i,j'}$, besides $\vec{v}(l_{i,j})$ and $\vec{v}(l_{i,j'})$, we generate another feature vector based on the subtraction of the previous two feature vectors, defined as:

$$\vec{v}_S(l_{i,j}, l_{i,j'}) = \vec{v}(l_{i,j}) - \vec{v}(l_{i,j'})$$

As a result, the feature vector for a data instance $< l_{i,j}, l_{i,j'} >$ for pairwise learning can be represented as:

$$\vec{v}_{PL}(l_{i,j}, l_{i,j'}) =< \vec{v}(l_{i,j}), \vec{v}(l_{i,j'}), \vec{v}_s(l_{i,j}, l_{i,j'}) >$$

After the learning process, we can perform link correction by extracting all the positive data instances. We take all the correct links $l_{i,j'}$ as the corrections for error links $l_{i,j}$. The pairwise learning approach introduced above is a general framework in that any classification algorithm can be employed to train the model f. In this paper, we employ Support Vector Machine as the classifier due to its strong discrimination power and wide application.

6. EXPERIMENTS

In this section, we conduct comprehensive experiments on English and Chinese Wikipedia datasets to evaluate the performance of our approach. We first illustrate the effectiveness of the candidate error link generation process, then evaluate the pairwise learning model in various aspects. Comparison between our approaches with baseline approaches is also conducted.

6.1 Datasets

In the experiments, we use two datasets: English and Chinese Wikipedia dumps[4]. We prepossess the datasets by first removing all irrelevant pages such as administrative and template pages because they do not provide information about entities. For remaining Wikipages, we take the titles as names of entities and extract all the hyperlinks between these Wikipages. The detailed statistics are shown in Table 3.

[4]Download website: http://download.wikipedia.com/
English version: 20140903 Chinese version: 20140912

Table 3: Dataset statistics

Dataset	#Entities	#Links
English Wikipedia	3,555,829	91,658,488
Chinese Wikipedia	924,422	11,361,229

6.2 Candidate Error Link Generation

In this section, we present the results on the candidate error link generation process, as well as the comparison results between our method with three baselines.

6.2.1 Baselines

To the best of our knowledge, there is no prior work addressing the candidate error link generation problem. To show the effectiveness of our approach, we set up the following baselines to generate candidate error link set.

- **Simple**: It utilizes Wikipedia disambiguation information to generate candidate error links.[5]

- **AnchorText**: Error link problem is mostly due to the ambiguity of anchor texts. Wikipedia links with ambiguous anchor texts are treated as candidate error links.

- **Unweighted**: Candidate error link pair $< l_{i,j}, l_{i,j'} >$ is generated by measuring whether or not e_i is more closely to $e_{j'}$ than e_j. The major difference between this approach and ours is that we do not use the *LinkRank* weighting technique.

6.2.2 Experiments and Results

In the experiments, we use different methods to generate candidate link sets, and estimate the percentage of error links. Higher percentage means the method is effective to generate candidate error links. However, it is infeasible to obtain the "ground truth" (i.e., all the error links in Wikipedia) to calculate the percentage. For each experiment, we randomly sample 500 links from candidate error link set, and ask human annotators to check whether they are error links based on the content of Wikipedia. We perform the same experiments using baselines **Simple**, **AnchorText** and **Unweighted**, and our method (denoted as **LinkRank**) under different values of threshold τ. The results are shown in Table 4.

From the experimental results, results from **Simple** and **AnchorText** are not comparative with the result of our method. If we use the simple method or consider links with ambiguous anchor texts only, it is difficult to generate candidate error links with high density, which shows the serious data sparsity problem in our task.

Unweighted and **LinkRank** can greatly increase the density of the candidate error links by considering the link structure of an ATSN. **LinkRank** outperforms **Unweighted** in all the settings of τ for both English and Chinese datasets. It shows the effectiveness of our link weighting technique. Moreover, when τ becomes larger, the density of error links increases simultaneously, from 5.6% to 11.6% for English and from 4.0% to 8.4 % for Chinese.

Another finding is that the effectiveness of our algorithm is related to the different language versions of Wikipedia. The density of error links for English Wikipedia is higher than that for Chinese

Table 4: Density of error links in candidate error link sets generated by various methods

Method	# Error links in sample set	Density of error links
Dataset: English Wikipedia		
Simple	0	0% (approx.)
AnchorText	0	0% (approx.)
Unweighted	21	4.2%
LinkRank ($\tau = 0.2$)	28	5.6%
LinkRank ($\tau = 0.4$)	34	6.8%
LinkRank ($\tau = 0.6$)	43	8.6%
LinkRank ($\tau = 0.8$)	**58**	**11.6%**
Dataset: Chinese Wikipedia		
Simple	0	0% (approx.)
AnchorText	1	0.2%
Unweighted	17	3.4%
LinkRank ($\tau = 0.2$)	20	4.0%
LinkRank ($\tau = 0.4$)	26	5.2%
LinkRank ($\tau = 0.6$)	38	7.6%
LinkRank ($\tau = 0.8$)	**42**	**8.4%**

Wikipedia in every group of experiments using **LinkRank**. In Table 3, we can see the average link per article for English Wikipedia is 26, and 12 for Chinese, which means the hyperlink structure of English Wikipedia has higher quality. Because our method is mostly based on the analysis of the hyperlink structure from ATSN, the denser hyperlink structure makes the characteristics of error links in English Wikipedia easier to detect.

6.3 Link Classification and Correction

In this section, we evaluate the performance of the pairwise learning model we proposed in this paper, and compare it with baselines.

6.3.1 Experimental Settings

To apply the link classification and correction models on candidate error links, we sample the candidate set to generate the dataset for training and validation. To address the imbalanced classification issue, we over-sample positive instances by three times for training. The sizes of two datasets (i.e., English and Chinese Wikipedia) are four thousand and two thousand, respectively. Each instance is a tuple $< l_{i,j}, l_{i,j'} >$, denoting that $l_{i,j}$ is an candidate error link and that $l_{i,j'}$ is a probably correct link. For each instance, we ask human annotators to check the corresponding links in Wikipedia dataset and label them as positive or negative. We use the WEKA[6] toolkit for classification models. For content analysis and the extraction of context-based features, we build up dictionaries containing stop words and meaningless symbols in English and Chinese, respectively. We use the open source Ansj[7] toolkit to perform Chinese NLP analysis such as Chinese word segmentation before generating n-grams.

6.3.2 Overall Performance Evaluation

We first evaluate the methods for link classification and correction. We use 10-fold cross validation on the dataset. *Precision*, *Recall* and *F-Measure* are employed as the evaluation metrics. We introduce methods for evaluating the performance of link classification and correction as follows:

- **PL-Full**: It is the pairwise learning approach for link classification and correction using all the features (Section 5).

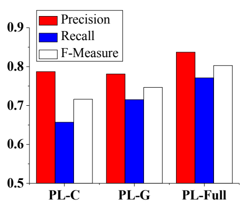
[5]Here is a simple example w.r.t. *Java*. We extract the disambiguation page (e.g. *Java_(disambiguation)*), find entities related to "Java" in that page (e.g. *Java*, *Java_(programming language)*, etc.), and treat all the links that point to these entities as candidate error links (e.g. $Facebook \rightarrow Java$). In this way, we process all the disambiguation pages from Wikipedia and then generate the candidate error link set.

[6]http://www.cs.waikato.ac.nz/ml/weka/

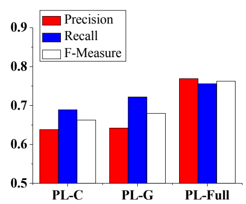
[7]http://nlpchina.github.io/ansj_seg/

- **PL-G**: It is an variant of the approach **PL-Full**. We only use graph-based features in the implementation.
- **PL-C**: It is an variant of the approach **PL-Full**. We only use context-based features in the implementation.

We set up experiments for all the methods mentioned above. The results for English and Chinese Wikipedia datasets are illustrated in Fig. 3. **PL-G** achieves higher performance than **PL-C**. It can be seen that the engineered graphed-based features have stronger discriminative power than context-based features. The possible reason is that in Wikipedia, the contents of some Wikipages are relatively incomplete. The vector space based n-gram similarity method is not sufficient for distinguishing error/correct links. Combining all the features together, in **PL-Full**, with proper parameter tuning process, the polynomial kernel SVM with degree $p = 4$ and tolerance parameter $C = 100$ achieves the highest F-measure 80.3% for English Wikipedia, and 76.2% for Chinese Wikipedia with degree $p = 3$ and tolerance parameter $C = 100$. The results show the engineered features along with the pairwise learning approach achieve higher accuracy than baselines.



(a) English Wikipedia



(b) Chinese Wikipedia

Figure 3: Results for link classification and correction.

6.3.3 *Comparison with Other Methods*

In this section, we make a comprehensive comparison between different approaches for the error link problem. We divide these methods into three categories and compare them with our pairwise learning method. The experimental results are shown in Table 5.

VSM Based Methods. VSM based methods are simple approaches which adopt the Vector Space Model to represent the contents of Wikipages. For an instance $< l_{i,j}, l_{i,j'} >$, if the content of Wikipage $e_{j'}$ is more similar to e_i than that of e_j, the link $l_{i,j}$ is regarded as an error link. We denote **VSim** and **IntroVSim** as the approaches which compare the contents in the whole article and in the introduction part (regarded as the entity summary) to correct error links. The low performance shows the simple method of context similarity comparison can not solve the error link task effectively due to the high-dimensional, sparse representation of the contents.

EL Based Methods. The error link problem can be regarded as a special case of EL, which is discussed in Section 2. We apply EL techniques to correct error links, and experimentally prove that they are not directly capable of correcting error links.

We obtain the implementation of the EL system **LINDEN** [24] from the authors[8] and re-implement the **Wikify!** [14] system to correct error links. For an error link $l_{i,j}$, it regards the content of Wikipage e_i as the context, and links the anchor text m in e_i to an entity e_{j^*} in Wikipedia. If $e_{j^*} \neq e_j$, it predicts $l_{i,j}$ as an error link. We say the EL system successfully correct an error link $l_{i,j}$ if it outputs $e_{j^*} = e_{j'}$ where $l_{i,j'}$ is the correction for the link $l_{i,j}$.

[8]Note that the YAGO-related features in **LINDEN** are not added for the Chinese Wikipedia error link set, because there is no Chinese version of YAGO or its equivalence available [28].

From the experiments, we can see that **LINDEN** has a low accuracy to correct error links in English and Chinese Wikipedia. To link text mentions to entities correctly, a lot of measurements need to be computed based on the link structure, e.g. the prior probability of an entity given a text mention, the semantic associativity between entities, etc [23, 24]. Error links affect the performance of these measurements negatively when predicting the correct link by EL. Another finding is that cases of linking errors tend to happen between "tailed" entities where few inlinks/outlinks are added in these Wikipages. This also causes the missing link problem [26], further making the semantic relatedness between entities unavoidably inaccurate. On the contrary, our link correction method is not based on the ranking of candidate entities. It directly predicts the relative "goodness" between two links, considering both link structure and content similarity. Thus it is less sensitive to the missing and error link issue.

While EL systems such as **LINDEN** employ semantic link-based features to perform EL, we also employ the **Wikify!** system as a baseline for the error link problem, of which the key technique is word sense disambiguation based on the contextual information. The results show that the performance is unsatisfying because as we have discussed before, error links occur mostly between Wikipages with incomplete information (e.g. links and contents). The contents with low quality make it difficult for **Wikify!** to perform EL directly based on context overlap. In summary, although existing EL techniques are effective to solve the general EL tasks, they are not suitable for the error link problem.

Error Link Detection Based Methods. We first re-implement Pateman and Johnson's method [17] for error link detection (denoted as **LS**), which try to make the correction based on Wikipedia link structure itself. It has higher performance than content-based methods. However, if the link structure is sparse in a subgraph where the error link is involved, this method is more likely to fail.

Next, we discuss the reason why our pairwise learning approach (i.e., **PL-Full**) combines two subtasks together: (i) error link detection and (ii) error link correction, instead of solving them separately. We first modify our algorithm for the purpose of error link detection model. We predict whether a link $l_{i,j}$ in CL is a real error link by a classifier and denote this method as **ELD**. To make it comparable with ours, we use the same feature set for each link in Section 5.1 and datasets in Section 6.3.1. We train SVM classifiers for error link detection in English and Chinese Wikipedia. Experiments show that the performance of our approach **PL-Full** is much better than the simple error link detection method (**ELD**).

The causes behind the phenomenon are discussed as follows. Links in Wikipedia have varied graph and content related characteristics. It is difficult to distinguish error/correct links only based on the characteristic of the link itself. In error link classification and correction, for a link pair $< l_{i,j}, l_{i,j'} >$, iff $l_{i,j}$ is an error link and $l_{i,j'}$ is the correct link, it is treated as a positive instance. The classification model focuses on the comparison between two links. In this way, with a (probably) correct link available, the decision is easier to make. Therefore, our method is more effective than performing error link detection and correction in separate tasks.

6.4 Discussion

The error link problem is a seemingly trivial problem due to the fact that there is abundant research on text mentions, entities and their semantic links in the task of EL, word sense disambiguation, link analysis, etc. However, in the previous experiments, we argue that it is difficult to provide a solution for detecting and correcting error links based on existing approaches. We have explored the simple method of detecting error links based on Wikipedia dis-

Table 5: Comparison between different methods

Category	Method	Precision	Recall	F-Measure
Dataset: English Wikipedia				
VSM based	**VSim**	53.2%	40.8%	46.2%
	IntroVSim	57.9%	53.2%	55.5%
EL based	**Wikify!** [14]	45.4%	48.9%	47.1%
	LINDEN [24]	46.5%	61.4%	52.9%
Error link detection based	**LS** [17]	71.4%	58.6%	64.4%
	ELD	76.9%	47.3%	58.6%
	PL-Full	**83.7%**	**77.1%**	**80.3%**
Dataset: Chinese Wikipedia				
VSM based	**VSim**	50.1%	42.1%	45.8%
	IntroVSim	56.3%	51.2%	53.6%
EL based	**Wikify!** [14]	48.2%	41.5%	44.6%
	LINDEN [24]	43.8%	38.6%	41.0%
Error link detection based	**LS** [17]	68.5%	62.3%	65.3%
	ELD	54.7%	39.7%	46.0%
	PL-Full	**76.9%**	**75.6%**	**76.2%**

ambiguation pages, and also presented the performance of state-of-the-art techniques, such as the EL method *LINDEN* [24] and the Wikipedia error link detection approach [17]. The cause behind the ineffectiveness of these methods is not the inefficacy of the algorithms but the lack of research on the characteristics of error links. Without paying special attention to the data sparsity and data quality issues, these methods are unlikely to achieve high performance although they work well in general cases. To the best of our knowledge, our method outperforms previous approaches for the Wikipedia error link problem.

Another issue that needs to be discussed is that the techniques proposed in this paper can be easily extended for error link detection for other data sources. For example, the ambiguity phenomenon is very common in networks such as knowledge graphs and heterogeneous information networks. After proper modification, our candidate error link generation method is capable of finding subgraphs containing suspicious links from any large-scale graphs. The pairwise learning technique is especially useful in the situation where direct classification of links is ineffective.

7. ANALYSIS OF ERROR LINKS

In this section, we present the preliminary analysis on error links we have discovered in English and Chinese Wikipedia.

7.1 Categorization of Error Links

Briefly, the phenomenon of error links stems from the ambiguity of anchor texts. More specifically, some Wikipages describe specific named entities, such as *Bob Gibson*, while others explain meanings of common concepts, such as *Steam*, *Lipstick*, etc. We randomly sample error links generated from the previous experiments and manually group them into categories, indicating different types of ambiguity, introduced as follows.

Multiple Senses of Named Entities (MSNE). Different named entities can share the same text mention. There is a possibility for contributors to point anchor text to the wrong entity. For example, Wikipage *Bob Gibson (musician)* introduces an American musician; however, "Bob Gibson" is also the name for a retired American baseball pitcher, described in Wikipage *Bob Gibson*. In English Wikipedia, the link from *Josh White*, an American singer who influenced lots of musicians including Bob Gibson, links to *Bob Gibson*, the baseball pitcher. It can be corrected by linking *Josh White* to *Bob Gibson (musician)*.

Multiple Senses of Concepts (MSC). A concept can have several senses. In Wikipedia, each sense of the concept usually has a unique page associated with it, instead of listing all the senses in a single page. Error links occur when the contributor links a page to the wrong sense of the concept. For example, Wikipage *Cheltenham Town F.C.* introduces an English football club in Cheltenham Town. "Administration" in this Wikipage is used as a term related to English football clubs, which refers to a situation where a football club is unable to pay off outstanding debts. But it links to the term "administration" in a general legal sense. Thus, the link should be treated as an error link.

Ambiguity Between Concepts and Named Entities (ACNE). Some word phrases can refer to common concepts or named entities according the context. Wikipage *Tactical role-playing* introduces a type of video games, which links to Wikipage *Steam* (water in the gas phase). It should link to Wikipage *Steam (software)*, a software platform.

We present some of the error links we have found in English and Chinese Wikipedia in Table 7 and Table 8. For each error link, if it has not been explained above, we give an explanation how the error is occurred and the correct link predicted by our approach.

Table 6: Percentages of different categories of error links

Dataset	Category of error links		
	MSNE	ACNE	MSC
Wikipedia Error Link Set (English)	**75.8%**	20.8%	3.4%
Wikipedia Error Link Set (Chinese)	**83.6%**	11.8%	4.6%

7.2 Distributions of Categories of Error Links

Based on the preliminary analysis, the distribution of categories of error links is shown in Table 6. For both English and Chinese Wikipedia, error links in the category *MSNE* account for the majority of all the error links, with the percentage of 75.8% and 83.6%, respectively. The rest of the error links are in the categories *ACNE* and *MSC*. The probable causes for the skewed distribution are discussed as follows: i) Wikipedia contains abundant entities but few concepts [25]. Most links tend to point to named entities rather than concepts. As a consequence, most error links are related to named entities. ii) Different senses of named entities can be very similar. In the previous example, both entities related to *Bob Gibson* are person names (musician and baseball pitcher respectively). In contrast, the senses of the latter two categories are very different, making it difficult for contributors to make the wrong decision when adding links.

8. CONCLUSION AND FUTURE WORK

In this paper, we propose to detect and correct error links in Wikipedia effectively. More specifically, the task can be divided into two steps: candidate error link generation, and error link classification and correction. We propose a *LinkRank* algorithm to detect candidate error links based on ATSN. We employ a pairwise learning technique to determine which are error links and make correction suggestions simultaneously. The experimental results on English and Chinese Wikipedia demonstrate that the proposed approach achieves accurate results. We further present a preliminary analysis based error links in English and Chinese Wikipedia.

There are two pieces of future work. Our work only focuses on error links where correct entities exist in Wikipedia. A more challenging problem would be detecting error links where there are no correct links, and finding correct entities from the Web. Besides, although our approach is mainly Wikipedia-centric, it has reasonably wide application for error link detection for Web-scale networks.

Table 7: Cases of error links in English Wikipedia

Category	Source Wikipage	Target Wikipage	Correct Wikipage
MSNE	Augustus of Prima Porta[1]	Mars	Mars (mythology)
	Josh White	Bob Gibson	Bob Gibson (musician)
MSC	Cheltenham Town F.C.	Administration (law)	Administration (British football)
ACNE	Tactical role-playing game	Steam	Steam (software)
	Ireland in the Eurovision Song Contest 2011[2]	Lipstick	Lipstick (Jedward song)

[1] Augustus of Prima Porta is a marble statue of Augustus Caesar, which has the bas-relief of the Roman god of war, Mars. The anchor text "Mars" points to the Wikipage describing the planet Mars.

[2] Wikipage *Lipstick (Jedward song)* describes a song by Irish pop duo Jedward. But Wikipage *Ireland in the Eurovision Song Contest 2011* links to Wikipage *Lipstick* (a cosmetic product) when the contributor refers to the song.

Table 8: Cases of error links in Chinese Wikipedia

Category	Source Wikipage	Target Wikipage	Correct Wikipage
MSNE	Theodore Beza[1] (泰奥多尔·贝扎)	Baden (巴登)	Baden (Switzerland) (巴登 (瑞士))
	Light Rail 705 & 706[2] (香港轻铁705、706线)	Ginza Station (银座站)	Ginza Stop (Hong Kong) (银座站 (香港))
MSC	Unit sphere[3] (单位球面)	Boundary (边界)	Boundary (topology) (边界 (拓扑学))
ACNE	Donnie Yen[4] (甄子丹)	Hero (英雄)	Hero (film) (英雄 (电影))
	Zhou Yang (actress)[5] (周扬 (演员))	Tea house (茶馆)	Tea House (TV series) (茶馆 (电视剧))

[1] Theodore Beza was a Swiss reformer and scholar, whose hometown was Baden in Switzerland. The link to Beza's hometown points to a location in Germany whose name is also Baden.

[2] Ginza Station is a subway station in Tokyo, Japan. It has the same Chinese name with a light rail stop in Hong Kong. Light Rail 705 & 706 actually goes past the stop Ginza in Hong Kong.

[3] Wikipage *Boundary* describes the dividing line or location between two areas, such as two countries. The word is also a Mathematical term in topology, introduced in Wikipage *Boundary (topology)*. In Wikipage *Unit sphere*, the contributor uses the word as an anchor text in a topological sense, but carelessly links to the wrong page.

[4] Donnie Yen is a Hong Kong actor who starred in the film Hero in 2002. Wikipage *Hero* describes the general concept.

[5] Zhou Yang is a Chinese actress who starred in the TV series Tea House. Wikipage *Tea house* describes the place where people drink tea.

We will also extend the error link detection technique to heterogeneous information networks (e.g. DBLP network), knowledge graphs (e.g. DBPedia) and other types of data in the future.

Acknowledgements. This work is supported by the National Key Research and Development Program of China under Grant No. 2016YFB1000904.

9. REFERENCES

[1] E. Aktolga, M. Cartright, and J. Allan. Cross-document cross-lingual coreference retrieval. In *CIKM*, pages 1359–1360, 2008.

[2] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30(1-7):107–117, 1998.

[3] S. Bykau, F. Korn, D. Srivastava, and Y. Velegrakis. Fine-grained controversy detection in wikipedia. In *ICDE*, pages 1573–1584, 2015.

[4] M. Dredze, P. McNamee, D. Rao, A. Gerber, and T. Finin. Entity disambiguation for knowledge base population. In *COLING*, pages 277–285, 2010.

[5] M. Granitzer, C. Seifert, and M. Zechner. Context based wikipedia linking. In *INEX*, pages 354–365, 2008.

[6] X. Han and L. Sun. A generative entity-mention model for linking entities with knowledge base. In *ACL*, pages 945–954, 2011.

[7] X. Han and L. Sun. An entity-topic model for entity linking. In *EMNLP-CoNLL*, pages 105–115, 2012.

[8] X. Han, L. Sun, and J. Zhao. Collective entity linking in web text: a graph-based method. In *SIGIR*, pages 765–774, 2011.

[9] J. Hoffart, M. A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum. Robust disambiguation of named entities in text. In *EMNLP*, pages 782–792, 2011.

[10] T. Joachims. Optimizing search engines using clickthrough data. In *KDD*, pages 133–142, 2002.

[11] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

[12] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, and C. Bizer. Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

[13] J. Li, C. Wang, X. He, R. Zhang, and M. Gao. User generated content oriented chinese taxonomy construction. In *APWeb*, pages 623–634, 2015.

[14] R. Mihalcea and A. Csomai. Wikify!: linking documents to encyclopedic knowledge. In *CIKM*, pages 233–242, 2007.

[15] D. N. Milne and I. H. Witten. Learning to link with wikipedia. In *CIKM*, pages 509–518, 2008.

[16] T. Noraset, C. Bhagavatula, and D. Downey. Adding high-precision links to wikipedia. In *EMNLP*, pages 651–656, 2014.

[17] B. M. Pateman and C. Johnson. Using the wikipedia link structure to correct the wikipedia link struture. In *Proc of 2nd Workshop on Collaboratively Constructed Semantic Resources*, page 10—18, 2010.

[18] H. Paulheim and C. Bizer. Improving the quality of linked data using statistical distributions. *IJSWIS*, 10(2):63–86, 2014.

[19] A. Pilz and G. Paaß. From names to entities using thematic context distance. In *CIKM*, pages 857–866, 2011.

[20] S. P. Ponzetto and M. Strube. Deriving a large-scale taxonomy from wikipedia. In *AAAI*, pages 1440–1445, 2007.

[21] N. Popitsch and B. Haslhofer. Dsnotify - A solution for event detection and link maintenance in dynamic datasets. *J. Web Sem.*, 9(3):266–283, 2011.

[22] W. Shen, J. Wang, and J. Han. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Trans. Knowl. Data Eng.*, 27(2):443–460, 2015.

[23] W. Shen, J. Wang, P. Luo, and M. Wang. LIEGE: : link entities in web lists with knowledge base. In *KDD*, pages 1424–1432, 2012.

[24] W. Shen, J. Wang, P. Luo, and M. Wang. LINDEN: linking named entities with knowledge base via semantic knowledge. In *WWW*, pages 449–458, 2012.

[25] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *WWW*, pages 697–706, 2007.

[26] O. Sunercan and A. Birturk. Wikipedia missing link discovery: A comparative study. In *AAAI Spring Symposium*, 2010.

[27] K. Tran and P. Christen. Cross-language learning from bots and users to detect vandalism on wikipedia. *IEEE Trans. Knowl. Data Eng.*, 27(3):673–685, 2015.

[28] C. Wang, M. Gao, X. He, and R. Zhang. Challenges in chinese knowledge graph construction. In *ICDE Workshops*, pages 59–61, 2015.

[29] Z. Wang, J. Li, Z. Wang, and J. Tang. Cross-lingual knowledge linking across wiki knowledge bases. In *WWW*, pages 459–468, 2012.

[30] G. Weaver, B. Strickland, and G. R. Crane. Quantifying the accuracy of relational statements in wikipedia: a methodology. In *JCDL*, page 358, 2006.

[31] I. Witten and D. Milne. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *Proc of AAAI WikiAI Workshop*, pages 25–30, 2008.