Integrating Internet Directories by Estimating Category Correspondences

Yoshimi Suzuki and Fumiyo Fukumoto
Graduate Faculty of Interdisciplinary Research, University of Yamanashi, Japan

Keywords: Internet Directories, Text Categorization.

Abstract: This paper focuses on two existing category hierarchies and proposes a method for integrating these hierarchies into one. Integration of hierarchies is proceeded based on semantically related categories which are extracted by using text categorization. We extract semantically related category pairs by estimating category correspondences. Some categories within hierarchies are merged based on the extracted category pairs. We assign the remaining categories to a newly constructed hierarchy. To evaluate the method, we applied the results of new hierarchy to text categorization task. The results showed that the method was effective for categorization.

1 INTRODUCTION

With the exponential growth of information on the Internet, finding and organizing relevant materials is becoming increasingly difficult. Internet directories with classifying Web pages into predefined hierarchical categories are one of the solutions to organize a large volume of documents (Dumais and Chen, 2000; Xue et al., 2008). It improves the efficiency and accuracy of Information Retrieval on the Web (Hearst and Karadi, 1997). Because pages/documents on the Web are organized into hierarchical categories so that similar pages are grouped together.

Categories in the hierarchical structures are carefully defined by human experts and documents are well-organized. However, there are at least two issues in terms of maintenance. One is that a single hierarchy tends to have some bias in both defining hierarchical structure and classifying documents, some hierarchies are coarse-grained, while others are fine-grained. Therefore, it might be difficult for users to find relevant materials by using only one such hierarchy. The second issue is that it is often insufficient to find relevant documents as category hierarchies often evolve at a much slower pace than the documents reside in. Maintenance of a hierarchy is indispensable to keep category consistency, while it requires remarkable human efforts and is time-consuming. The above issues indicate that the methodology for the automatic construction of a hierarchy by making the maximum use of existing hierarchies is needed.

In this paper, we present a method for integrating two existing category hierarchies. We first extract semantically related category pairs by estimating category correspondences. According to the extracted category pair, we merge categories from one hierarchy and those from another hierarchy. Finally, we assign the remaining categories to a newly constructed hierarchy while keeping consistency.

2 RELATED WORK

Ontology merging on the Internet is crucial to provide intelligent Web services (Choi et al., 1999; Lee et al., 2005; Lehmberg and Hassanzadeh, 2018) and several tools such as SMART and PROMPT for mapping on ontology merging and alignment are developed (Noy and Musen, 1999). The earliest known approach is Noy et al. method which combines two ontologies represented in a hierarchical categorization (Noy and Musen, 2000). Their method is based on the similarity between words with dictionaries. McGuinness et al developed a system called CHIMAERA which is an interactive merging tool based on Ontolingual ontology editor (McGuinness et al., 2000). Stumme presented a method that uses the attributes of concepts to merge different ontologies (Stumme and Madche, 2001). It creates a new concept without regarding the original concepts in both ontologies. However, most of these approaches require human interaction for the merging process. Ichise *et al.* addressed the problem and presented a method for integrating multiple

Suzuki, Y. and Fukumoto, F.
Integrating Internet Directories by Estimating Category Correspondences.
DOI: 10.5220/0008354104270434
In *Proceedings of the 11th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2019)*, pages 427-434
ISBN: 978-989-758-382-7

Internet directories (Ichise et al., 2003). They used Enhanced Naive Bayes(E-NB) (Agrawal and Srikant, 2001) to integrate directories. They applied the κ-statistics to find similar category pairs and transferred the document categorization from a category in the source Internet directory to a similar category in the target Internet directory. They did not rely on words or word similarity in a document but instead relied on the category structure which makes the computational cost more efficient. However, their method is based on the existence of a large number of shared links. Therefore, as they reported, the performance was not better if there were fewer shared links. More recently, ontology matching becomes a key interoperability enabler for the Semantic Web and Workshop is organized to emphasis on theoretical and practical aspects of ontology matching[1].

Hierarchy construction and modification are also related to merging category hierarchies, while very few works have addressed the hierarchical modification problem (Kim et al., 2002; Tang et al., 2006; Marszalek and Schmid, 2008; Zhang et al., 2012; He and Sun, 2013; Naik and Rangwala, 2017). Tang *et al.* attempted to modify the relations between categories aiming to improve the classification accuracy (Tang et al., 2006). However, they did not change the leaf categories of the given hierarchy which leave the topically incohesive leaf categories untouched. Yuan *et al.* proposed a method for modifying a given category hierarchy by redistributing its documents into more topically cohesive categories (Yuan et al., 2012). The modification is achieved with three operations, i.e., sprout, merge, and assign. They evaluated the method by conducting text classification using real data from Yahoo! Answers and AnswerBag hierarchies, and showed that the method had a better performance of classification. However, the maintenance is conducted by using the auxiliary hierarchy, which is not always complimentary with the original one. As a result, the accuracy depends on the auxiliary hierarchy itself. Zhuge et al proposed a method of automatic maintenance of category hierarchy by incorporating the global-phase adjustments and the local-phase adjustments to improve the topical cohesion and classification accuracy of categories (Zhuge and He, 2017). Global modification is to break up inappropriate parent-child relations to make the original hierarchy satisfy with the pattern consistence of the clustering results. In contrast, local adjustment is applied to some specific nodes. It consists of three operations, i.e., pull-up, merge, and split. The experimental results by using three datasets, Reuters-25, 20Newsgroups and DMOZ shows that these two adjustments improve the topical cohesion and classification accuracy of categories, while they focused on just one category hierarchy.

In the context of text classification, many authors have attempted to apply deep learning techniques including CNN (Wang et al., 2015; Zhang et al., 2015; Zhang et al., 2017; Wang et al., 2017), the attention based CNN (Yang et al., 2016a), bag-of-words based CNN (Johnson and Zhang, 2015), and the combination of CNN, recurrent neural network (Lee and Dernoncourt, 2016; Zhang et al., 2016) and Hierarchical attention network (Yang et al., 2016b) to text classification. Most of them demonstrated that neural network models are powerful for learning features from texts, while there a few works which apply a model to the hierarchical structure of categories (Shimura et al., 2018).

In contrast with the aforementioned works, here we propose a method for integrating category hierarchies by estimating category correspondences. Our method does not need any auxiliary hierarchy, relying instead on classification technique to estimate category correspondences.

3 SYSTEM DESCRIPTION

The method consists of three steps: (i) estimating category correspondences, (ii) merging categories, and (iii) assignment of the remaining categories.

3.1 Estimating Category Correspondences

The first step for integrating two category hierarchies is to retrieve pairs of semantically relevant categories assigned to each hierarchy. Let two category hierarchies be H_a and H_b. The assumption is that if the category C_a assigned to H_a is semantically related with the category C_b from H_b, the documents assigned to C_a are semantically close to the documents assigned to C_b. We thus applied the text classification technique to identify semantically relevant categories. We used the CNN model based on (Kim, 2014a) for classifying documents with a category assigned to a hierarchy.

Let $\mathbf{x}_i \in \mathbb{R}^k$ be the k-dimensional word vector with the i-th word in a sentence obtained by applying skip-gram model provided in fastText[2]. A sentence with length n is represented as $\mathbf{x}_{1:n} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \mathbb{R}^{nk}$. A convolution filter $\mathbf{w} \in \mathbb{R}^{hk}$ is applied to a window size of h words to produce a new feature, c_i =

[1] om2018.ontologymatching.org

[2] https://github.com/facebookresearch/fastText

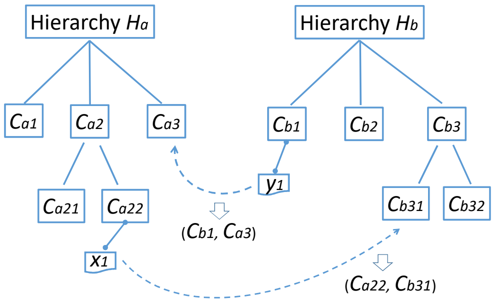

Figure 1: Classification between hierarchies.

$f(\mathbf{w} \cdot \mathbf{x}_{i:i+h-1} + b)$ where $b \in \mathbb{R}$ indicates a bias term and f refers to a non-linear activation function. We applied this convolution filter to each possible window size in the document and obtained a feature map, $m \in \mathbb{R}^{n-h+1}$. Next, we apply a max pooling operation over the feature map and obtain the maximum value $\hat{m}$ as a feature of this filter. We obtained multiple filters by varying window sizes and multiple features. These features form a pooling layer and are passed to a fully connected layer. In the fully connected layer, we applied dropout (Hinton et al., 2012). The dropout randomly sets values in the layer to 0. Finally, we obtained the probability distribution over categories. The network is trained with the objective that minimizes the binary cross-entropy (BCE) of the predicted distributions and the actual distributions by performing stochastic gradient descent.

We classified documents with a category assigned to H_a into categories with H_b by using the CNN model. Similarly, each document assigned to a category belonging to H_b was classified into categories assigned to H_a. We used hierarchical classification technique (Dumais and Chen, 2000), i.e., models were learned to distinguish each category from only those categories within the same level. We compute a Boolean scoring function (BSF) that a category can only be selected if its ancestor categories are selected. The function sets a threshold value and categories whose scores exceed the threshold value are considered for selection.

Figure 1 illustrates an example of classification between H_a and H_b hierarchies. A document x_1 assigned to category "C_{a22}" in Figure 1 is classified into category "C_{b31}" in the hierarchy H_b, and a document y_1 belonging to the category "C_{b1}" is classified into category "C_{a3} in the hierarchy H_a. As the result, we obtained two category pairs, (C_{a22},C_{b31}) and (C_{b1},C_{a3}). We estimated category correspondences based on these category pairs.

Category correspondences assume that semantically similar categories have similar statistical properties than dissimilar categories. We applied simple statistics, i.e., χ^2 statistics to the results of category

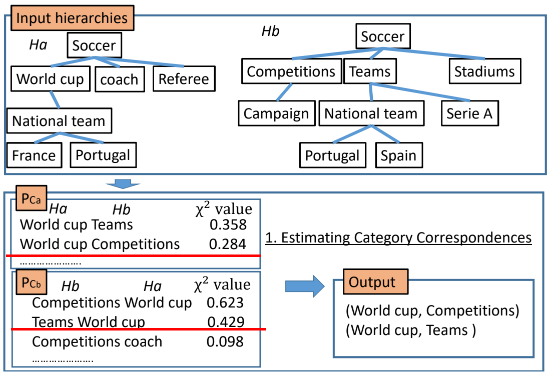

Figure 2: Estimating category correspondences.

pairs which is shown in Eq. (1).

$$\chi^2(C_a,C_b) = \frac{f(C_a,C_b) - S(C_a,C_b)}{S(C_a,C_b)},$$
$$\text{where} \quad S(C_a,C_b) = T_{C_a} \times \frac{T_{C_b}}{T_{H_a}},$$
$$T_{C_a} = \sum_{k \in H_b} f(C_a,k), T_{H_a} = \sum_{c_a \in H_a} T_{C_a}. \tag{1}$$

$f(C_a,C_b)$ in Eq. (1) refers to the co-occurrence frequency of the category C_a and C_b, and it is equal to the number of category C_b documents assigned to C_a. Similar to the H_a hierarchy, we can estimate category correspondences from the H_b hierarchy, and extract category pairs according to the χ^2 value. We note that the similarity obtained by each hierarchy does not have a fixed range because the number of categories and documents assigned to the H_a are different from those of H_b. We apply a simple normalization technique shown in Eq. (2) to the results obtained by each hierarchy to bring the similarity value into the range [0,1].

$$\chi^2_{new}(c_a,c_b) \quad = \quad \frac{\chi^2_{old}(c_a,c_b) - \chi^2_{min}(c_a,c_b)}{\chi^2_{max}(c_a,c_b) - \chi^2_{min}(c_a,c_b)} \tag{2}$$

Let P_{C_a} and P_{C_b} be a set of pairs obtained by H_a and H_b hierarchy, respectively. We created the set of category pairs, $P_{(C_a,C_b)} = \{(C_a,C_b) \mid (C_a,C_b) \in P_{C_a} \cap P_{C_b}, \chi^2(C_a,C_b) \geq L_{\chi^2}\}$, where each pair is sorted in descending order of χ^2_{new} value. L_{χ^2} refers to a lower bound. We regarded each pair of $P_{(C_a,C_b)}$ as semantically similar. Figure 2 illustrate the procedure, estimating category correspondences. In this example, the lower bound L_{χ^2} is set to 0.200, and two category pairs, ("World cup", "Competitions") and ("World cup", "Teams") are regarded as a semantically related category pair.

Input {
Two input hierarchies, and a set of pair
$P_{(C_a,C_b)} = \{(C_a,C_b) \mid (C_a,C_b) \in P_{C_a} \cap P_{C_b}, \chi^2(C_a,C_b) \geq L_{\chi^2}\}$
}
Output {
Category hierarchy
}
Merge procedure {
For $i = 1, 2, \cdots N$ {
// N is the number of pairs in the set $P_{(C_a,C_b)}$
(1) If each category of the pair is identical:
  two categories of the pair is merged.
(2) else if each category of the pair is not assigned to any other pairs in the set P:
  two categories of the pair is merged
(3) else
  apply tree edit distance and merge each category of a pair whose value of the edit distance is minimum.
}
}

Figure 3: Flow of merge procedure.

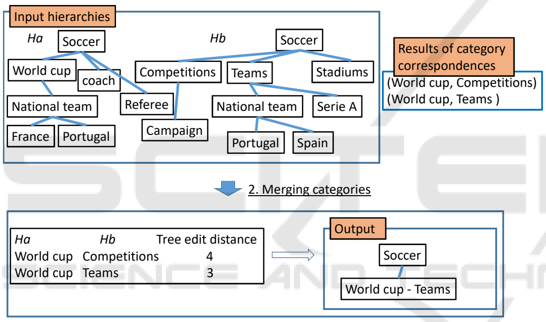

Figure 4: Merging categories.

3.2 Merging Categories

The second step for integrating category hierarchies is to merge categories. The procedure is conducted by using the resulting category pairs obtained by the first step, (i) estimating category correspondences. The flow of the merge procedure is shown in Figure 3.

(1) in Figure 3 shows that two categories within the pair is the same. (2) shows that each category is included within the pair itself. Tree edit distance in (3) of Figure 3 is a metric for estimating the similarity between two tree structures (Zhang and Shasha, 1989; Pawlik and Augsten, 2012). It is to find operations of the minimum cost that convert one tree into another. One edit-distance operation corresponds to one insert, deletion or relabels of a node. It has been widely used to many applications, *e.g.*, image analysis, pattern recognition, and NLP. We used it to merge categories.

Figure 4 illustrates tree edit distance procedure.

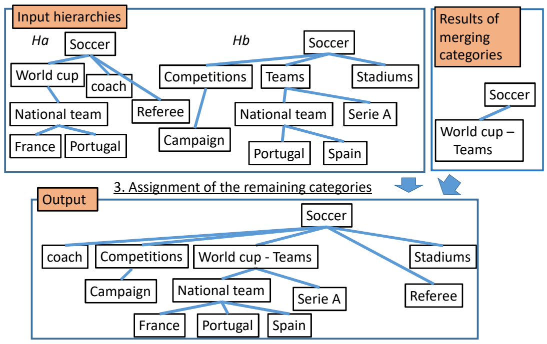

Figure 5: Assignment of the remaining categories.

We note that there are two ways to merge trees in H_a and H_b. One is "World cup" root subtree and "Competitions" root subtree. Another is "World cup" and "Teams" root subtrees. We can convert "World cup" subtrees into "Competitions" by four operations. Namely, three deletions "National team", "France", and "Portugal", and one insert operation, "Campaign". The tree edit distance is thus four. Similarly, "World cup" subtree is converted into "Teams" subtree by three operations. As a result, "World cup" and "Teams" subtrees are merged into one.

3.3 Assignment of Remaining Categories

The final step of integrating hierarchies is to assign remaining categories to the newly merged hierarchy. Parent-children relation of the original hierarchies is inherited to a new hierarchy. Figure 5 illustrates assignment of categories. "Coach" and "Stadiums" are children of "Soccer" in H_a and H_b, respectively. These categories are assigned to children of "Soccer" in the new hierarchy. "National team" is a child of both "World cup" in H_a and "Teams" in H_b. It is assigned as a child category of "World cup-Teams" in the new hierarchy.

Table 1: Categories used in the experiments.

DMOZ		Yahoo	
Cat	N	Cat	N
Athletics	6	Athletic stadium	5
Soccer	42	Soccer	42
Baseball	35	Baseball	38
Golf	11	Golf	14
Volleyball	6	Volleyball	7
Water Sports	6	Surfing	7
Comparative sports	13	Martial arts	31
Boxing	20	Boxing	7
Racing	20	Auto racing	26
Total	159	Total	177

Table 2: CNN model settings.

Description	Values	Description	Values
Input word vectors	fastText	Filter region size	(2,3,4)
Stride size	1	Feature maps (m)	128
Filters	128×3	Activation function	ReLu
Pooling	1-max pooling	Dropout	Randomly selected
Dropout rate1	0.25	Dropout rate2	0.5
Hidden layers	1,024	Batch sizes	100
Gradient descent	Adam	Epoch	30 with early stopping
Loss function	BCE loss over sigmoid activation	Threshold value for BSF	0.5

Table 3: Results of Integration.

DMOZ(159) / Yahoo(177)	Int	Not int	$F1$
Athletics(6) / Athletic stadium(5)	4	3	.724
Soccer(42) / Soccer(42)	38	8	.830
Baseball(35) / Baseball(38)	31	11	.702
Golf(11) / Golf(14)	11	3	.732
Volleyball(6) / Volleyball(7)	5	3	.881
Water Sports(6) / Surfing(7)	5	3	.748
Comparative sports(13) / Martial arts(31)	10	24	.740
Hockey(20) / Hockey(7)	3	21	.667
Racing(20) / Auto racing(26)	12	22	.823
Average			**.761**

4 EXPERIMENTS

We had an experiment category integration. For evaluating the effectiveness of category assignments, we conducted an extrinsic evaluation, i.e., text categorization by using the results of hierarchy integration.

4.1 Data

We selected DMOZ[3] and Yahoo! hierarchies[4] as an evaluation data. From these hierarchies, we chose subtrees whose root node is "Sports" and used them in the experiments. The hierarchies consist of four levels. For the second level of a hierarchy, we sorted categories in descending order according to the number of documents assigned to those categories and chose the top nine categories to evaluate our method. For the third and fourth levels, we chose categories whose number of documents is more than 20,000. Table 1 shows the number of categories and documents in each DMOZ and Yahoo! sub-hierarchies. "Cat" shows category name assigned to the second level. "N" refers to the total number of categories in the third and fourth level of a hierarchy.

For each category in each level of a hierarchy, we randomly selected 13,000 documents. Of these, we used 10,000 documents for training and the remains for test data. We divided the training data into three folds; we used 5% to tuning the parameters, and the remains to train the models. All the documents are tagged by using Tree Tagger (Schmid, 1995). We used nouns, verbs, and adjectives.

We divided the test documents into three folds: the first was used to estimate lower bound L_{χ^2}, the second was used as classification of the test data in the category integration, and the third was used as an extrinsic evaluation, i.e., the test data of the classification task. We manually evaluated category correspondences obtained by using the first fold, and set the lower bound to 0.235.

[3]http://www.dmoz-odp.org

[4]http://www.yahoo.com

4.2 CNN Model Setting and Evaluation Metrics

Our model setting for CNN is shown in Table 2. Dropout rate1 in Table 2 shows dropout immediately after embedding layer, and Dropout rate2 refers to Dropout in a fully connected layer.

We used $F1$ score as an evaluation measure. The $F1$ measure which combines recall(r) and precision(p) with an equal weight is $F1(r,p) = \frac{2rp}{r+p}$. The evaluation is made by two humans. The classification for category integration is determined to be correct if two human judges agree. We also used $F1$ as an evaluation metric for an extrinsic evaluation, i.e., text classification.

4.3 Category Integration

The results of the integration are shown in Table 3. "Int(Not int)" shows the number of categories which are merged(not merged) into one. We can see from Table 3 that the average $F1$ score was 0.761. Table 4 illustrates some examples of integrated categories. "L" refers to a category level. "X" of "X/Y" in Correct(Incorrect) shows an integrated DMOZ category, and "Y" indicates Yahoo category. "(Z)" in Incorrect refers to a Yahoo category which should be integrated. "X" and "Y" of "X/Y" in Level indicates category

Table 4: Examples of integrated categories.

DMOZ/Yahoo	Correct	Incorrect	L
Athletics/Athletic stadium	Marathon/Marathon	Race walking/Olympic(Marathon)	3/3
Soccer/Soccer	Coaching/Supervision	Freestyle/Freestyle football(Freestyle)	3/3
Baseball/Baseball	High School/High School Baseball	Tournaments/Drafts(Tournaments)	4/3
Golf/Golf	Tool/Instruction	Courses/Training field(Golf courses)	4/4
Volleyball/Volleyball	Beach/Beach volley	Tournaments/Rule(Tournament)	4/3
Water Sports/Surfing	Windsurfing/Wind-surfer	Underwater Hockey/Club(Hockey)	4/3
Martial arts/Comparative sports	Aikido/Aikido	Kick-boxing/Pro wrestling(Kick-boxing)	4/4
Hockey/Hockey	Universal hockey/Floor-ball	Players/Clubs(Players)	4/4
Racing/Auto racing	Road racing/Road racing	Events/Accident(Events)	4/4

Table 5: DMOZ and Yahoo! Categorization.

Level	Dm $\longrightarrow$ Ya			Ya $\longrightarrow$ Dm		
	Prec	Rec	$F1$	Prec	Rec	$F1$
Top	.733	.691	**.730**	.780	.749	**.764**
2nd	.681	.611	.644	.692	.652	.671
3rd	.548	.491	.518	.445	.531	.484
4thh	.243	.203	.221	.302	.339	.319

level of DMOZ and Yahoo which is integrated incorrectly. Table 4 shows that most of the error occurs to the integration of a lower level of the hierarchies. One reason is text classification performance that we used to estimate category correspondences. Table 5 shows the results of text classification. "Dm $\longrightarrow$ Ya" shows that we classified DMOZ documents into Yahoo! categories. As we can see from Table 5, the $F1$ score drops when the level of hierarchy becomes lower as the $F1$ score in the top-level attained at 0.730 (Dm $\longrightarrow$ Ya) and 0.764 (Ya $\longrightarrow$ Dm) but the fourth level were only 0.221 and 0.319. Another reason is that surface information of the categories with a lower level of hierarchies are different from each other, and thus more difficult to integrate than a higher level of a hierarchy. Accuracy can be improved if we extend our method to incorporate other category similarity measures, e.g., Word Mover Distance based on Word2Vec to calculate among categories (Kim, 2014b). This is a rich space for further exploration.

4.4 Text Classification

For evaluating the effectiveness of category assignments, we compared the results of text classification with and without hierarchy integration. Similar to integration experiments, we used $F1$ measure as an evaluation measure. The results are shown in Table 6. Each value in Table 6 indicates Micro-$F1$. "Without Int" and "With Int" shows the results without and with integrating DMOZ and Yahoo hierarchies, respectively. "Dm $\longrightarrow$ Ya" refers to the results obtained by classifying DMOZ documents into Yahoo hierarchy, i.e., the training documents to learn SVM models are Yahoo documents. "Dm $\longrightarrow$ Int" indicates the results obtained by classifying DMOZ documents into an integrated hierarchy. Each value of "2nd", "3rd", and "4th" indicates the categories from the top to second, the top to third, and whole categories, respectively.

We can see from Table 6 that the overall performance of integrating category hierarchies was better to those not integrating hierarchies in both of the DMOZ and Yahoo cases as the Macro average $F1$ in "with Integration" of whole categories was 0.611, while "without Integration" was 0.544, and the performance was improved (6.7%). As shown in Table 6, there is a drop in performance in going from the top to lower level of categories in all of the methods. However, the performance obtained by integrating hierarchies is still better than the method of not integrating hierarchies. This shows that a newly created hierarchy contributes to text classification.

5 CONCLUSION

We proposed a method for integrating two category hierarchies based on text classification. We first extract semantically similar category pairs by estimating category correspondences. According to the extracted category pair, categories are merged from one hierarchy and those from another hierarchy. Finally, we assign the remaining categories to newly constructed hierarchy. The extrinsic evaluation showed that integrating hierarchies is effective for the classification task as the improvement of classification attained at 6.7%. Future work will include (i) evaluating the classification method by using other representation learning such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2018), (ii) extending the method to make use of category similarity based on Word Mover Distance (Kim, 2014b) for further improvement, (iii) relabeling for every modified category (Yuan et al., 2012), and (iv) evaluating the method by using root categories other than "sports" for quantitative evaluation.

Table 6: Categorization results.

Without Integration						With Integration					
Method	Hierarchical level					Method	Hierarchical level				
	Top	2nd	3rd	4th	Average		Top	2nd	3rd	4th	Average
Dm $\longrightarrow$ Ya	.730	.644	.518	.221	.528	Dm $\longrightarrow$ Int	.754	.732	.623	.429	.634
Ya $\longrightarrow$ Dm	.764	.671	.484	.319	.559	Ya $\longrightarrow$ Int	.793	.701	.510	.344	.587
Average	.747	.657	.501	.270	**.544**	Average	.774	.716	.567	.386	**.611**

REFERENCES

Agrawal, R. and Srikant, R. (2001). On Integrating Catalogs. In *Proc. of the 10th International World Wide Web Conference(WWW-10))*, pages 603–612.

Choi, N., Song, I., and Han, H. (1999). A Survey on Ontology Mapping. *Newsletter ACM SIGMOD Record*, 3(35):34–41.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In *arXiv:1810.04805*.

Dumais, S. and Chen, H. (2000). Hierarchical Classification of Web Content. In *Proc. of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 256–263.

He, L. and Sun, X. (2013). Automatic Maintenance of the Category Hierarchy. In *Proc. of the 9th International Conference on Semantics, Knowledge and Grids*, pages 218–221.

Hearst, M. A. and Karadi, C. (1997). Cat-a-Cone: An Interactive Interface for Specifying Searches and Viewing Retrieval Results using a Large Category Hierarchy. In *Proc. of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 246–255.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, H., and Salakhutdinov, R. R. (2012). Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors. In *arXiv preprint arXiv:1207.0580*.

Ichise, R., Tanaka, H., and Honiden, S. (2003). Integrating Multiple Internet Directorie by instance-based learning. In *Proc. of the 18th International Joint Conference on Artificial Intelligence)*, pages 22–30.

Johnson, R. and Zhang, T. (2015). Effective Use of Word Order for Text Categorization with Convolutional Neural Networks. In *Proc of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 103–112.

Kim, D., Kim, J., and g. Lee, S. (2002). Catalog Integration for Electronic Commerce through Category-hierarchy Merging Technique. In *Proc. of the 12th International Workshop on Research Issues in Data Engineering: Engineering E-Commerce*, pages 28–33.

Kim, Y. (2014a). Convolutional Neural Networks for Sentence Classification. In *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

Kim, Y. (2014b). Convolutional Neural Networks for Sentence Classification. In *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

Lee, H., Shim, J., and Kim, D. (2005). Ontological Modeling of E-catalogs using EER and Description Logics. In *Proc. of the International Workshop on Data Engineering Issues in E-Commerce*, pages 125–131.

Lee, J. Y. and Dernoncourt, F. (2016). Sequential Short-Text classification with Recurrent and Convolutional Neural Networks. In *Proc. of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies*, pages 515–520.

Lehmberg, O. and Hassanzadeh, O. (2018). Ontology Augmentation through Matching with Web Tables. In *Proc. of the 13th International Workshop on Ontology Matching*, pages 37–48.

Marszalek, M. and Schmid, C. (2008). Constructing Category Hierarchies for Visual Recognition. In *Proc. of the 10th European Conference on Computer Vision*, pages 479–491.

McGuinness, D. L., Fikes, R., Rice, J., and Wilder, S. (2000). The Chimacra Ontology Environment. In *Proc. of the AAAI 2000*, pages 1123–1124.

Naik, A. and Rangwala, H. (2017). Integrated Framework for Improving Large-Scale Hierarchical Classification. In *Proc. of the 16th IEEE International Conference on Machine Learning and Applications*, pages 281–228.

Noy, N. F. and Musen, M. A. (1999). Automated Support for Ontology Merging and Alignment. In *Proc. of the 12th Workshop on Knowledge Acquition, Modelling, and Management)*, pages 1–24.

Noy, N. F. and Musen, M. A. (2000). Algorithm and Tool for Automated Ontology Merging and Alignment. In *Proc. of the 17th National Conference on Artificial Intelligence(AAAI'00)*, pages 450–455.

Pawlik, M. and Augsten, N. (2012). RTED: A Robust Algorithm for the Tree Edit Distance. In *Proc. of the 38th International Conference on Very Large Data Bases)*, pages 334–345.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep Contextualized Word Representations. In *Proc. of the 16th Anual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2227–2237.

Schmid, H. (1995). Improvements in Part-of-Speech Tagging with an Application to German. In *Proc. of the EACL SIGDAT Workshop*, pages 47–50.

Shimura, K., Li, J., and Fukumoto, F. (2018). HFT-CNN: Learning Hierarchical Category Structure for Multi-label Short Text Categorization. In *Proc. of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 811–816.

Stumme, G. and Madche, A. (2001). Ontology Merging for Federated Ontologies on the Semantic Web. In *Proc. of the International Workshop on Foundation of Models for Information Integration(FMII'01)*, pages 413–419.

Tang, L., Zhang, J., and Liu, H. (2006). Acclimatizing Taxonomic Semantics for Hierarchical Content Classification from Semantics to Data-Driven Taxonomy. In *Proc. of the 16th ACM SIGKDD Conference on Knowledge discovery and Data Mining*, pages 384–393.

Wang, J., Wang, Z., Zhang, D., and Yan, J. (2017). Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification. In *Proc. of the 26th International Joint Conference on Artificial Intelligence*, pages 2915–2921.

Wang, P., Xu, J., Xu, B., Liu, C.-L., Zhang, H., Wang, F., and Hao, H. (2015). Semantic Clustering and Convolutional Neural Network for Short Text Categorization. In *Proc. of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 352–357.

Xue, G.-R., Xing, D., Yang, Q., and Yu, Y. (2008). Deep Classification in Large-scale Text Hierarchies. In *Proc. of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 619–626.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. (2016a). Hierarchical Attention Networks for Document Classification. In *Proc. of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies*, pages 1480–1489.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. (2016b). Hierarchical Attention Networks for Document Classification. In *Proc. of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics in Human Language Technologies*, pages 1480–1589.

Yuan, Q., Cong, G., adn C-Y. Lin, A. S., and Mangnenat-Thalmann, N. (2012). Category Hierarchy Maintenance: A Data-Driven Approach. In *Proc. of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 791–800.

Zhang, J., Zhang, J., and Chen, S. (2012). Constructing Dynamic Category Hierarchies for Novel Visual Category Discovery. In *Proc. of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 479–491.

Zhang, K. and Shasha, D. (1989). Simple Fast Algorithm for the Editing Distance between Trees and Related Problems. *SIAM Journal on Computing*, 18(6):1245–1262.

Zhang, R., Lee, H., and Radev, D. (2016). Dependency Sensitive Convolutional Neural Networks for Modeling Sentences and Documents. In *Proc. of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics Human Language Technologies*, pages 1512–1521.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-Level Convolutional Networks for Text Classification. In *Advances in Neural Information Processing systems*, pages 649–657.

Zhang, Y., Lease, M., and Wallace, B. C. (2017). Exploiting Domain Knowledge via Grouped Weight Sharing with Application to Text Categorization. In *Proc. of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 155–160.

Zhuge, H. and He, L. (2017). Automatic Maintenance of Category Hierarchy. *ELSEVIER: Future Generation Computer Systems*, pages 1–12.