Classification of Wikipedia articles based on category name patterns

Aleksander Smywiński-Pohl*

Jagiellonian University
Department of Computational Linguistics
ul. Łojasiewicza 4, 30-348 Kraków, Poland
aleksander.pohl@uj.edu.pl

Abstract. In this article we show how Wikipedia category name patterns might be useful for the classification of Wikipedia articles. We provide an alternative approach to the method known in literature, which employs category name parsing. Our method is based on much simpler regular expression patterns and does not require a parser.

Keywords: Entity classification, Wikipedia, pattern discovery

1 Introduction

The classification of Wikipedia articles is a process where the articles are attributed with classes taken from a selected classification scheme. Many of the Wikipedia articles might be treated as describing particular objects or classes of objects. Thus the classification indicates the types of these objects or superclasses of those classes. E.g. if the classification scheme contains a *Person* class, the algorithm should establish that `en.wikipedia.org/wiki/Ayn_Rand` is an instance of that class, while `en.wikipedia.org/wiki/Novelist` is its subclass. The result could also only vaugly indicate that both of these Wikipedia aricles are linked by an ambiguous *is-a* relation with that class. DBpedia [1], YAGO [2] and Tipalo [3] are some of the projects aimed at that goal.

The presented approach is much similar to the way YAGO classifies the articles. Namely the algorithm employed in YAGO detects the syntactic heads of the names of the Wikipedia categories, the article belongs to (only categories with plural noun heads are treated as indicating the type). Then it disambiguates these names against WordNet synsets [4] and assigns these synsents as the types of the article. That way *Ayn Rand* is classified as $novelist_1$ and $philosopher_1$[1], since it belongs to *20th-century American novelists* and *20th-century philosophers* Wikipedia categories.

There are two primary differences between YAGO and our approach. First is the fact that we use Cyc [5] as the reference classification scheme. The second –

* This work is partially sponsored by the Faculty of Management and Social Communication of the Jagiellonian University.

[1] Synset subscripts are taken from WordNet 3.1.

more important – is the fact that we *do not use a parser* to detect the syntactic heads of the category names for establishing the relationship between them and the corresponding Cyc classes. This is motivated by three facts: some languages do not have good enough parsers developed for them to be employed in that task; parsers have problems with lengthy category names such as *UCLA Bruins men's basketball players* and there are category names that do not indicate the type of the articles, e.g. *2011 deaths* which contains people rather than events, yet might be very useful for the classification.

2 Pattern discovery

Our approach is based on the fact that there exist many Wikipedia categories that share certain name patterns. E.g. there are *People from Indiana*, *People from Arkansas* and *People from Ontario* categories in Wikipedia. That fact is to some extent exploited in YAGO – the authors manually created templates for the most popular patterns and used them to classify the entities. Although this process might be scaled up by using crowd-sourcing, the DBpedia mapping efforts[2] show that it is not as easy to engage people in such and undertaking.

To avoid the manual construction of the patterns we **search for** the names of **the Wikipedia articles** within the names of **the Wikipedia categories** and **replace the matched names** with a universal match. Current English Wikipedia edition contains more than 4 million articles and it is not much surprising that it contains articles covering *Indiana*, *Arkansas* and *Ontario* that appear in the names of the before-mentioned categories. Indeed it contains articles covering much more obscure objects such as *Vircator*, *Meryeurus* and *WWIZ*. Thus we can automatically create a *People from .** pattern from those three categories.

To find the article names in the category names we divide them by spaces, generate all continuous word sub-sequences starting with a capital letter, join them back with spaces and look-up in the list of Wikipedia articles. E.g. for *University of Montana alumni* the following sequences are generated: *University*, *University of*, *University of Montana*, *University of Montana alumni*, *Montana* and *Montana alumni*. **University**, **University of Montana** and **Montana** have their corresponding Wikipedia articles. By replacing the name of the discovered article by a universal match we receive the following patterns: `.* of Montana alumni`, `.* alumni` and `University of .* alumni`.

To overcome the problem of inflection present in inflected languages like Polish, we extend that approach by searching for the names on the list of internal Wikipedia links leading to the Wikipedia articles. That way we can create an `Absolvenci .*` (Eng. `Alumni .*`) pattern from *Absolwenci Petersburskiego Uniwersytetu Państwowego* (Eng. *Saint Petersburg State University alumni*), even though *Petersburski**ego** Uniwersytet**u** Państwow**ego*** is an inflected form of the *Petersburski Uniwersytet Państwowy* article title.

[2] `http://mappings.dbpedia.org/index.php/Main_Page`

3 Pattern mapping

When the patterns are automatically constructed, we can map them to the classification scheme of our choice. In the conducted experiments we used OpenCyc as the reference classification scheme. The mapping was done automatically using the following approach: for each pattern a list of matching Wikipedia categories was computed. That list was further converted to a list of articles that directly belong to these categories – so for each pattern a list containing indirectly referenced (unique) articles was constructed. E.g. for the *People from .** pattern, articles such as *Albert Einstein* (as he belongs to the *People from Berlin* category) and *Isaac Newton* (via *People from South Kesteven*) were collected.

We used a classification of Wikipedia articles to the OpenCyc ontology from our previous classification research [6][3] to establish a correspondence between the patterns and the Cyc classes. This was done by simply counting the number of times given Cyc class was used as a type for the articles that are covered by a given category name pattern and selecting the class with the highest count. *Albert Einstein* and *Isaac Newton* are classified as Cyc's `#$Person` and `#$Scientist` (among other types), but since the pattern matches thousands of category names, each containing hundreds of articles, the correct type (`#$Person`) is selected.

However, there are cases when there are two or more competing types with a high number of evidences. Thus assigning the type that appeared the largest number of times in all cases, would ignore the fact that some patterns are ambiguous (e.g. `A .*`). In order to filter out such patterns we have computed a histogram of the entropy of the pattern mapping and ignored all mappings where the entropy was higher than the entropy for the first minimum appearing on the histogram. That way we have kept mappings that were not completely unambiguous, but were indicating one dominating candidate mapping.

4 Results

We have compared our approach with our previous results of classifying the Wikipedia articles to the Cyc ontology, where the categories were mapped using the same method as in YAGO, namely the category name syntactic heads were mapped automatically to Cyc terms. We also compared the method with the performance of Tipalo [3], which treats the first sentences of the Wikipedia articles as their definitions and assigns a class from the Dolce Ultra Light ontology. To measure the performance we used the Tipalo validation set (we mapped the Dolce Ultra Light classes to Cyc classes manually). The results of the comparison are given in Table 1. It turns out that the method outperforms both the method based on syntactic head identification and the method based on the first sentence parsing. It should be noted, however, that it requires some initial classification to be available. As such it might be only useful as a method for improving the classification.

[3] Available at `http://klon.wzks.uj.edu.pl/wiki-types`.

Table 1. Comparison of the results for three methods of Wikipedia article classification.

Method	Precisions	Recall	F1
Tipalo	68.0	66.0	67.0
Syntactic head mapping	74.4	50.0	59.8
Pattern mapping	**94.7**	**60.0**	**73.5**

The presented method of pattern mapping was applied directly only for the English Wikipedia, but we have also conducted experiments on mapping patterns mined from one Wikipedia to its the other editions. That way the mapping established for the English Wikipedia could be transfered to the other editions, based on the fact that not only the articles, but also the Wikipedia categories have interlingual links. Although the results are promising, there is no much space in that document to discuss them in greater detail.

5 Questions

1. Do you think the pattern discovery algorithm might be useful in other tasks regarding data mining from Wikipedia?
2. Do you think OpenCyc is a good choice as a reference classification scheme?
3. Can the pattern discovery algorithm be applied to language resources other than Wikipedia?
4. Do you have any ideas how the classification method could be improved?

References

1. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: A nucleus for a web of open data. The Semantic Web (2007) 722–735
2. Suchanek, F., Kasneci, G., Weikum, G.: YAGO: a core of semantic knowledge. In: Proceedings of the 16th international conference on World Wide Web, ACM (2007) 697–706
3. Gangemi, A., Nuzzolese, A.G., Presutti, V., Draicchio, F., Musetti, A., Ciancarini, P.: Automatic typing of DBpedia entities. In: The Semantic Web–ISWC 2012. Springer (2012) 65–81
4. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press (1998)
5. Lenat, D.B.: CYC: A large-scale investment in knowledge infrastructure. Communications of the ACM **38**(11) (1995) 33–38
6. Pohl, A.: Classifying the Wikipedia Articles into the OpenCyc Taxonomy. In Rizzo, G., Mendes, P., Charton, E., Hellmann, S., Kalyanpur, A., eds.: Proceedings of the Web of Linked Entities Workshop in conjuction with the 11th International Semantic Web Conference. (2012) 5–16