Web Search Result Clustering with BabelNet

Marek Kozlowski
OPI-PIB
mkozlowski@opi.org.pl

Maciej Kazula
OPI-PIB
mkazula@opi.org.pl

Abstract

In this paper we present well-known search result clustering method enriched with BabelNet information. The goal is to verify how Babelnet/Babelfy can improve the clustering quality in the web search results domain. During the evaluation we tested three algorithms (Bisecting K-Means, STC, Lingo). At the first stage, we performed experiments only with textual features coming from snippets. Next, we introduced new semantic features from BabelNet (as disambiguated synsets, categories and glosses describing synsets, or semantic edges) in order to verify how they influence on the clustering quality of the search result clustering. The algorithms were evaluated on AMBIENT dataset in terms of the clustering quality.

1 Introduction

In the previous years, Web clustering engines (Carpineto, 2009) have been proposed as a solution to the issue of lexical ambiguity in Information Retrieval. These systems group search results, by providing a cluster for each specific topic of the input query. Users navigate through the clusters in order to retrieve the pertinent results. Most of clustering engines group search results on the basis of their lexical similarity, and therefore suffer from semantic lackness e.g. polysemy (different user needs expressed with the same words).

In this paper we present well-known search result clustering method enriched with BabelNet information. The goal is to verify how Babelnet/Babelfy can improve the clustering quality in the web search results domain. Our approach is evaluated on the dataset AMBIENT using four distinct measures, namely: Rand Index (RI), Adjusted Rand Index (ARI), Jaccard Index (JI) and F1 measure.

2 Related Work

2.1 Search Result Clustering

The goal of text clustering in information retrieval is to discover groups of semantically related documents. Contextual descriptions (snippets) of documents returned by a search engine are short, often incomplete, and highly biased toward the query, so establishing a notion of proximity between documents is a challenging task that is called Search Result Clustering (SRC). Search Results Clustering (SRC) is a specific area of documents clustering.

Approaches to search result clustering can be classified as data-centric or description-centric (Carpineto, 2009).

The data-centric approach (as Bisecting K-Means) focuses more on the problem of data clustering, rather than presenting the results to the user. Other data-centric methods use hierarchical agglomerative clustering (Maarek, 2000) that replaces single terms with lexical affinities (2-grams of words) as features, or exploit link information (Zhang, 2008).

Description-centric approaches (as Lingo, STC) are more focused on the description that is produced for each cluster of search results. Accurate and concise cluster descriptions (labels) let the user search through the collection's content faster and are essential for various browsing interfaces. The task of creating descriptive, sensible cluster labels is difficult - typical text clustering algorithms rely on samples of keywords for describing discovered clusters. Among the most popular and successful approaches are phrase-based, which form clusters based on recurring phrases instead of numerical frequencies of isolated terms. STC algorithm employs frequently recurring phrases as both document similarity feature and final cluster description (Zamir, 1998). Clustering in STC is treated as find-

ing groups of documents sharing a high ratio of frequent phrases. The Lingo algorithm combines common phrase discovery and latent semantic indexing techniques to separate search results into meaningful groups. Lingo uses singular value decomposition of the term-document matrix to select good cluster labels among candidates extracted from the text (frequent phrases). The algorithm was designed to cluster results from Web search engines (short snippets and fragmented descriptions of original documents) and proved to provide diverse meaningful cluster labels (Osinski, 2004).

2.2 Babel eco-system

The creation of very large knowledge bases has been made possible by the availability of collaboratively-edited online resources such as Wikipedia and Wiktionary. Although these resources are only partially structured, they provide a great deal of valuable knowledge which can be harvested and transformed into structured form.

BabelNet[1](Navigli, 2014; Flati, 2014) is a multilingual encyclopedic dictionary and semantic network, which currently covers more than 270 languages and provides both lexicographic and encyclopedic knowledge thanks to the seamless integration of WordNet, Wikipedia, Wiktionary, OmegaWiki, Wikidata and the Open Multilingual WordNet. BabelNet encodes knowledge in a labeled directed graph G=(V,E), where V is the set of nodes (concepts) and E is the set of edges connecting pairs of concepts. Each edge is labeled with a semantic relation. Each node contains set of lexicalizations of the concepts for different languages. The multilingually lexicalized concepts are Babel synsets. At its core, concepts and relations in BabelNet are harvested from the largest available semantic lexicon of English, WordNet, and the biggest open encyclopedia Wikipedia. BabelNet preserves the organizational structure of WordNet, i.e., it encodes concepts and named entities as sets of synonyms (synsets), but also the information typical for WordNet is complemented with wide Wikipedia encyclopedic coverage, resulting in an intertwined network of concepts and named entities.

BabelNet is available online as (1) a public web user interface, (2) a public SPARQL endpoint or (3) HTTP Rest API.

[1]http://babelnet.org

Babelfy[2](Navigli, 2014; Flati, 2014) is a unified graph-based approach that leverages BabelNet to jointly perform word sense disambiguation and entity linking in arbitrary languages. Babelfy is based on a loose identification of candidate meanings coupled with a densest subgraph heuristic, which selects high-coherence semantic interpretations. Babelfy WSD performance evaluations outperform the state-of-the-art supervised systems.

3 Approach

The goal is to verify how Babelnet/Babelfy can improve the clustering quality in the web search results domain. The evaluation was performed in three steps. First, we tested three algorithms (Bisecting K-Means, STC, Lingo). At this stage, we performed experiments only with textual features coming from snippets. Those methods do not exploit any external corpora or knowledge resource in order to overcome lack of data. This stage is fulfilled in order to choose the representative algorithm for the next phases. Next, we introduced new semantic features from BabelNet/Babelfy (as disambiguated synsets, categories/glosses describing synsets, or semantic edges) in order to verify how they influence on the clustering quality of the search result clustering algorithm. Finally, we verified the idea of clustering snippets without the specialized algorithm, namely only with the use of BabelNet/Babelfy systems.

4 Experiments

4.1 Experimental Setup

Test sets. We conducted our experiments on the AMBIENT data set. AMBIENT (AMBIguous ENTries[3]) consists of 44 topics, each with a set of subtopics and a list of 100 ranked documents (Carpineto, 2008).

Reference algorithms. We compared such search result clustering methods: (1) Lingo (Osinski, 2004), (2) Suffix Tree Clustering (Zamir, 1998) and (3) Bisecting K-means (Steinbach, 2000).

BabelNet modules. We used the HTTP API provided by BabelNet and Babelfy. Babelfy was used in order to disambiguate the given text [4]. Such extracted synsets were processed by BabelNet API

[2]http://babelfy.org

[3]http://credo.fub.it/ambient/

[4]https://babelfy.io/v1/disambiguate

in order to get more information about them [5] (as categories, glosses), or to get some graph relations as hypernyms[6].

4.2 Scoring

Following (Di Marco, 2009), the methods were evaluated in terms of the clustering quality. Clustering evaluation is a difficult issue. Many evaluation measures have been proposed in the literature so, in order to get exhaustive results we calculated four distinct measures, namely: Rand Index (RI), Adjusted Rand Index (ARI), Jaccard Index (JI) and F1 measure. The above mentioned measures are described in detail in (Di Marco, 2009).

4.3 Results

We conducted three level experiments on the AMBIENT data set. First, we compared three search result clustering algorithms (i.e. Lingo, STC, and Bisecting K-means [7]). Our goal was to estimate theirs quality parameters (the details in the Table 1). Lingo and STC outperform significantly the Bisecting K-means in the measures as Adjusted Rand Index and Jaccard Index. We decided further to investigate only one of those algorithms, namely Lingo. Our goal is to verify how we can improve Lingo with the information from BabelNet/Babelfy.

We performed a second experiment aimed at quantifying the impact of BabelNet/Babelfy on search result clustering algorithm.

Algorithm	**RI**	**ARI**	**JI**	**F1**
Lingo	62.52	18.09	30.76	49.01
STC	66.95	23.05	28.10	53.08
K-means	62.79	7.69	12.83	49.79

Table 1: A comparison between different search result clustering approaches (percentages) on AMBIENT data set.

Table 2 presents the potential improvements with theirs quality influences. Lingo record corresponds to the Lingo raw quality results (ours baseline). Next records define different types of data extensions, mainly adding new features to the previously existing snippet's text (title and summary). Improvements can be defined as follows:

- synsets+ - the snippet's text is disambiguated using Babelfy, and the retrieved synset ids are added as additional tokens to the snippet textual data
- categories+ - the snippet's text is disambiguated using Babelfy, and the retrieved synset ids are processed with BabelNet in order to get theirs categories, such retrieved categories are added as additional tokens to the snippet textual data
- categories+1 - the snippet's text is disambiguated using Babelfy, and the retrieved synset ids are processed with BabelNet in order to get theirs categories, the categories occurring more than once are added as additional tokens to the snippet textual data
- categories+2 - the snippet's text is disambiguated using Babelfy, and the retrieved synset ids are processed with BabelNet in order to get theirs categories, the categories occurring more than twice are added as additional tokens to the snippet textual data
- glosses+ - the snippet's text is disambiguated using Babelfy, and the retrieved synset ids are processed with BabelNet in order to get theirs glosses, the glosses are added as additional phrases to the snippet textual data
- hypernyms+ - the snippet's text is disambiguated using Babelfy, and the retrieved synset ids are processed with BabelNet in order to get theirs hypernyms, the hypernyms are added as additional tokens to the snippet textual data

The cells in bold in the table 2 show the real improvements, that beats the baseline measures of the original algorithm Lingo. There is easy to notice that synsets+ and categories+ report the best measures in the context of all extensions. The glosses+ and hypernyms+ do not provide any exclusive information for the clustering purpose.

In the last table (no. 3) we verified the idea of clustering snippets without the specialized algorithm, namely only with the use of BabelNet/Babelfy systems. The record babelCl1 represents the approach based on adding to the snippet the topic item (query), next disambiguation process is performed, the assigned synset to the topic

[5] https://babelnet.io/v3/getSynset

[6] https://babelnet.io/v3/getEdges

[7] We used the implementation of those algorithms from carrot2 project http://project.carrot2.org/download.html

Improvement	RI	ARI	JI	F1
Lingo	62.52	18.09	30.76	49.01
synsets+	**63.52**	**18.61**	29.21	**49.76**
categories+	**63.04**	17.01	27.46	**49.36**
categories+1	61.73	16.48	29.55	48.65
categories+2	62.17	17.44	30.30	48.80
glosses+	**62.69**	12.27	21.30	47.24
hypernyms+	61.52	16.35	29.44	48.32

Table 2: A comparison between different improvements (percentages) applied toward the Lingo algorithm and tested on AMBIENT data set.

is treated as the cluster signature. Next record (babelCl2) is the variation of the previous approach, the snippet's signature is not the topic's synset label, but the set of its hypernyms. The snippets are aggregated into one cluster if only there is any intersection between such hypernym's collections representing each snippet.

However, pure babelnet clustering approaches attain very low ARI measure, which disqualified such methods. Other measures are also below the baseline.

Approach	RI	ARI	JI	F1
Lingo	62.52	18.09	30.76	49.01
babelCl1	50.60	1.67	26.87	41.53
babelCl2	50.44	1.56	27.06	40.41

Table 3: The scores (percentages) reported by the clustering based on BabelNet disambiguated snippet's topics.

5 Conclusion

In this paper we presented search result clustering enriched with BabelNet/Babelfy information. During the evaluation we tested three search result clustering algorithms (Bisecting K-Means, STC, Lingo). At the first stage, we performed experiments only with textual features coming from snippets. Next, we introduced new semantic features from BabelNet/Babelfy (as disambiguated synsets, categories/glosses describing synsets, or semantic edges) in order to verify how they influence on the clustering quality of the search result clustering. The quality improvements of above semantic extensions are very poor. The best improvements concerning synsets expansions, do not overcome 1%. In the third attempt we tried to perform snippets clustering without the specialized algorithm, namely only with the use of Babelfy/Babelnet interfaces. The reported results are still below the Lingo measures.

There is also the problem connected with the time performance. Querying BabelNet/Babelfy HTTP API is a time consuming process especially when you need to acquire fifty thousand synset's informations. This limitation influences on the experiment's time spans. Therefore during experiments we decided to switch from MORESQUE data set to AMBIENT, because it is almost three times smaller.

In the future we plan to introduce more sophisticated improvements based on graph theories, because there must be a way to drastically improve the quality measures consuming such well-defined and organized semantic network as BabelNet.

Acknowledgments

I would like to express my gratitude to the team responsible for creating and maintaining the BabelNet eco-system.

References

C. Carpineto, S. Osinski, G. Romano and D. Weiss. 2009. A survey of web clustering engines. ACM Computing Surveys 41(3), pp. 1–38.

I. Maarek, R. Fagin and D. Pelleg. 2000. Ephemeral document clustering for web applications. IBM Research Report RJ 10186

X. Zhang, X. Hu and X. Zhou. 2008. A comparative evaluation of different link types on enhancing document clustering. In: Proceedings of SIGIR, pp. 555–562.

S. Osinski, J. Stefanowski and D. Weiss. 2004. Lingo: Search Results Clustering Algorithm Based on Singular Value Decomposition. In: Proceedings of the International IIS: IIPWM04 Conference held in Zakopane, pp. 359–368.

O. Zamir and O. Etzioni. 1998. Web document clustering: A feasibility demonstration. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 46–54.

R. Navigli. 2014. (Digital) Goodies from the ERC Wishing Well: BabelNet, Babelfy, Video Games with a Purpose and the Wikipedia Bitaxonomy. In: Proc. of the 2nd International Workshop on NLP and DBpedia 2014.

T. Flati and R. Navigli. 2014. Three birds (in the LLOD cloud) with one stone: BabelNet, Babelfy and the Wikipedia Bitaxonomy. In: Proc. of SEMANTiCS 2014.

C. Carpineto and G. Romano. 2008. AMBIENT dataset, http://credo.fub.it/ambient.

A. Di Marco and R. Navigli. 2013. Clustering and diversifying web search results with graph-based word sense induction. *Computational Linguistics*, pp. 709–754.

M. Steinbach, G. Karypis and V. Kumar. 2000. A comparison of Document Clustering Techniques. In: Proceedings of World Text Mining Conference, KDD2000.