DC Proposal: Model for News Filtering with Named Entities

Ivo Lašek

Czech Technical University in Prague, Faculty of Information Technology,
Prague, Czech Republic
lasekivo@fit.cvut.cz

Abstract. In this paper we introduce the project of our PhD thesis. The subject is a model for news articles filtering. We propose a framework combining information about named entities extracted from news articles with article texts. Named entities are enriched with additional attributes crawled from semantic web resources. These properties are then used to enhance the filtering results. We described various ways of a user profile creation, using our model. This should enable news filtering covering any specific user needs. We report on some preliminary experiments and propose a complex experimental environment and different measures.

Keywords: Information Filtering, User Modelling, Evaluation Methods.

1 Introduction

We are flooded with information nowadays. No one is able to keep track of all news articles in the world. Tools enabling us to filter the every day information stream are required.

This paper addresses the problem of news information filtering. We propose a framework, which maintains a complex user profile in order to perform content based news filtering. The user profile is put together not only based on traditional information retrieval techniques. We count also with semantic information hidden behind named entities that can be extracted from the article text. The additional semantic information and traditional information retrieval techniques are put together to form a unified news filtering framework.

2 Main Contributions

- *More detailed articles filtering*, not only based on predefined categories. The categorization is determined based on a concrete user feedback.
- *Model combining textual (unstructured) and semantic information.* Rather than representing an article as a bag of words, we represent it as a bag of contained named entities and their properties.
- We implemented *a prototype of the framework for news filtering based on extracted named entities.*
- We propose *evaluation methods* to test our model and scenarios for user experiments.

L. Aroyo et al. (Eds.): ISWC 2011, Part II, LNCS 7032, pp. 309–316, 2011.

3 Related Work

There are two main approaches to news information filtering. To some extent domain independent collaborative filtering was used in [2] and earlier in [3]. However in the case of news filtering, the collaborative filtering has significant drawbacks as described in [1]. This approach tends to recommend generally popular topics at the expense of the more specific ones. Also, it takes some time, before the system learns the preferences on a new article.

Contrary, the other approach - content based filtering - is able to recommend a new article practically immediately. Good comparison of content based filtering approaches is given in [4]. Inspirational examples of news filtering using Bayesian classifier are described in [5,6,7]. In [6] and [7], authors distinguish long-term and short-term interest. To learn the long-term interest a Bayesian classifier is used. For the short-term, the nearest neighbour algorithm is used. The articles are transformed to tf-idf vectors and then compared based on a cosine similarity measure.

Another related problem is a recommendation of related articles mentioned in [18]. Articles are recommended not only based on their similarity, but also according to their novelty and coherence. A comparison of various information retrieval techniques for such recommendation is provided in [8]. Used metrics for modelling the relatedness of articles include apart from traditional cosine similarity of term vectors also BM25 [19] and Language Modelling [20]. Here, often one document is used as a query to find related documents. We use different approach in this sense. In our case, the user profile serves as a query to filter relevant incoming articles.

An interesting example of exploitation of semantic information in connection with unstructured text is presented by BBC [9]. In this case, semantic data obtained from DBpedia serve to interlink related articles, through identifying similar topics. The topics are determined based on extracted named entities, mapped to DBpedia ontology. We extend this idea and build a user profile, using the data obtained from semantic web resources.

4 Proposed Methodology

4.1 Articles Processing Pipeline

The articles processing pipeline is shown in Fig. 1. First, we collect news articles using RSS[1] feeds. Sometimes RSS feeds contain only a fragment of the whole article. In this case, the rest of the article is automatically downloaded from its original web page. We analyse the DOM tree of the web page and locate non-repeating blocks, containing bigger amount of text. We build on heuristics introduced in the work of our colleagues [10].

[1] Really Simple Syndication - a family of web feed formats used to publish frequently updated works.

Fig. 1. The articles processing pipeline

In downloaded articles, named entities are identified. A ready made tool is used. Currently, we delegate this task to OpenCalais[2]. Apart from named entities themselves, OpenCalais provides often their basic attributes and links to other Linked Data[3] resources containing additional information too.

Currently, we evaluate the tool made by our colleagues for annotation of named entities in news articles [22]. Thus users get the possibility to mark additional entities, the system was not able to identify. The annotation tool is currently implemented as a plug-in to a web browser.

Extracted entities and their properties (if available) are used to query additional Linked Data resources (e.g. DBpedia or Freebase). If there are some relevant resources, additional information about extracted entities is crawled.

Articles modelling and user profile creation is cowered in detail in the following Section 5.

4.2 User Feedback Collection

In order to build a user profile, we need to collect the user feedback about filtered articles. In the initial learning phase, the user may provide general information about her interests. This is done by providing RSS feeds of news portals, he usually reads. As if she was using an ordinary RSS reader.

This initial setup partially overcomes the cold start problem.

When we talk about user feedback, we mean user rating of the article at the scale from 1 to 5, assuming an explicit user feedback.

5 Modelling News Articles

To model the content of an article, we distinguish three types of features: Subject-Verb-Object triples (SVO), terms (like in the information retrieval vector model) and named entities together with their properties.

For weighting of *terms*, we use ordinary tf-idf metric [12]. The results are normalized to range from 0 to 1. In case of *SVO triples*, we use only the binary measure. Either the triple is present in an article (1) or it is not (0).

In our previous work, we used an adaptation of tf-idf for entities too. Entity identifiers were used instead of terms. The *entity frequency* (we denote it as *ef*) was then counted based on the number of occurrences of the entity in the article. However, during the course of our experiments, we observed that the idf part disqualifies some popular entities, because they are often mentioned. But the fact, that an entity is often mentioned does not mean it is less important for the user.

[2] OpenCalais. `http://www.opencalais.com/`
[3] Linked Data. `http://linkeddata.org/`

Often the opposite is true. This problem is the subject of our future experiments. Currently, we tend to omit the idf part and count only with entity frequencies as the weight. The frequency of an entity i in an article j is counted as follows:

$$ef_{i,j} = \frac{e_{i,j}}{\sum_k e_{k,j}} \tag{1}$$

In equation 1 $e_{i,j}$ is the number of occurrences of the considered entity in a particular article and the denominator is the sum of the number of occurrences of all entities identified in the document.

We use entities identified during the named entity extraction phase. Additionally, we gather their properties in the crawling phase. The properties of an entity are presented in the form of a predicate object pair. Each such a pair has its own identifier. Weights of properties correspond to frequencies of entities. The weight of property k in the context of an article j is computed as follows:

$$pf_{j,k} = \sum_{ef_{i,j} \in E_{j,k}} \alpha \times ef_{i,j} \tag{2}$$

Where $E_{j,k}$ is the set of entities contained in an article j, having property k. And α is the proportion of the importance of entity properties to the importance of the entity. In following examples, we count with $\alpha = 1$. Thus properties are equally important as entities.

Additionally, *an important feature of an article is its rating*, given by each user.

Consider for example following two sentences representing two articles:

A1: Google launches a new social site.

A2: Microsoft recommends reinstalling Windows.

A possible representation of these two articles is denoted in Table 1, 2 and 3.

Table 1. Normalized term weights in an article

Article	google	launch	new	social	site	microsoft	recommend	reinstall	windows
A1	0.8	0.8	0.8	0.8	0.8	0	0	0	0
A2	0	0	0	0	0	1	1	1	1

Table 2. Subject-Verb-Object representation of an article

Article	Google-to-launch-site	Microsoft-to-recommend-reinstalling
A1	1	0
A2	0	1

Table 3. Entities and their properties representing an article

				Google, Microsoft		Windows
Article	Google	Microsoft	Windows	type:Company	locatedIn:USA	type:Product
A1	$ef_{1,1} = 1$	$ef_{2,1} = 0$	$ef_{3,1} = 0$	$pf_{1,1} = 1$	$pf_{1,2} = 1$	$pf_{1,3} = 0$
A2	$ef_{1,2} = 0$	$ef_{2,2} = 0.5$	$ef_{3,2} = 0.5$	$pf_{2,1} = 0.5$	$pf_{2,2} = 0.5$	$pf_{2,3} = 0.5$

6 User Profile Creation

With this representation of articles, we may use various machine learning approaches to identify user needs. For some of the algorithms, the described representation using numeric feature weights is fine. Some of the algorithms (e.g. apriori) require features (or attributes) to be nominal.

To transform the model to suitable representation, we can use binary representation - simply register the presence or absence of a given feature. The other option preserving the semantic of various weights is to discretize weights, using predefined bins (e.g. low, medium, high).

Naive Approach. First approach, we were evaluating in our previous work [13], counted with only two types of user feedback (positive and negative). The user profile was composed of features extracted from articles rated by the user and is divided in two parts: P^+ (set of features extracted from positively rated articles) and P^- (set of features extracted from negatively rated articles).

So far, we counted only with entities and omitted SVO triples and terms. The rank of a new article j is then computed as follows:

$$rank_j = \sum_{entity_i \in P^+} ef_{i,j} - \sum_{entity_i \in P^-} ef_{i,j} \tag{3}$$

If the rank is higher than a certain threshold, the article is considered as interesting for the user. This approach worked good for simple profiles. But with more complex user needs, only summing the weights is not flexible.

Apriori Algorithm. Apriori algorithm [14] may be applied to articles rated by a user. Having the representation described in Section 5, we can try to find association rules having the user rating of an article on its right side. A user profile is then composed of these rules. Sample rules may look like this:

```
type:Company ^ locatedIn:USA => rating4 (confidence 0.20)
subject:Google ^ Google type:Company => rating4 (confidence 0.80)
```

The first rule gives us the information that an article containing information about entity of type `Company` with the property `locatedIn` set to `USA` would the user rate with rating 4 with the confidence of 20%. The second rule reflects SVO triples.

Clustering. Interesting results achieved by using centroid based approach to document classification [15] inspired us to consider clustering as another method of creation of a user profile. Clustering may help to identify rather abstract concepts than concrete entities, the user is interested in. Given the model introduced in Section 5, clustering of articles rated by a particular user is performed. K-Means algorithm [16] is used for clustering. The cosine function is used to measure the similarity of vectors (SVO, terms and entities vectors) representing a particular articles.

For each cluster the average rating of articles contained in this cluster is computed. Any new article gets the average rating of the cluster it belongs to. We assign a new article to appropriate cluster separately for each type of features and then combine computed ratings:

$$rating_j = \alpha * rating_j^{SVO} + \beta * rating_j^{entities} + \gamma * rating_j^{terms} \quad (4)$$

The coefficients α, β and γ sum to one.

Formal Concept Analysis. Application of formal concept analysis [17] to articles rated by a particular user to a common grade may bring interesting results. We propose to analyse properties of named entities contained in articles in order to identify common concepts. The user profile creation and evaluation composes of following steps:

- Collect articles rated by the user as interesting.
- Identify properties and entities contained in all the collected articles.
- Use the identified concepts to identify new articles, that would be rated on the same grade.

An opened question remains, if the formal concept analysis is not too restrictive in this scenario. In this context application of fuzzy concept lattices [21] may bring interesting results.

7 Test Data Collection

We intend to test the whole system in three different ways:

- *Golden standard* - We collect data that represent our golden standard using web browser plug-in to save user ratings of arbitrary articles. Precision and recall metrics as well as F-measure metric and Kendall's tau coefficient are used to evaluate results.
- *Explicit feedback collection* - While using the system and rating recommended articles, users provide an important feedback. It can be used to evaluate results of the system. Same metrics as for golden standard apply.
- *Implicit feedback collection* - Finally, the system may collect the implicit feedback too. One of possible metrics is the really opened (user have clicked on them) to total recommended articles ratio:

$$succ = \frac{\#opened\_articles}{\#recommended\_articles} \quad (5)$$

8 Research Progress

We developed and tested a simple form of the proposed model consisting of named entities [13]. The user profile was constructed using the naive approach

described in Section 6. Each entity was represented by its ef-idef (entity frequency - inverse document entity frequency) weights. We evaluated its ability to recommend one particular topic. The results were promising. However, the use case was constrained to one particular topic. We used only the extracted entities, without employing their properties. Such a system filters the news accurately, but it is too constrained. We believe, the use of entity properties may add the necessary generalization of extracted concepts. We implemented a framework to collect news articles, to identify named entities using OpenCalais and to crawl additional information about identified entities from semantic web resources. For crawling of semantic web resources, we use LDSpider [11].

9 Conclusion

In this paper, we introduced the idea of news information filtering, using not only the text of news articles, but also information hidden behind named entities. We introduced the unified model of articles for news filtering. We believe, our approach can be combined with current information retrieval methods and improve their results. In Section 6 we described our plan of future work and named various approaches to user profile creation, using the proposed model. Several evaluation methods were described.

Acknowledgements. This work has been partially supported by the grant of The Czech Science Foundation (GAČR) P202/10/0761 and by the grant of Czech Technical University in Prague registration number SGS11/085/OHK3/1T/18.

References

1. Liu, J., Dolan, P., Pedersen, E.R.: Personalized News Recommendation Based on Click Behavior. In: IUI 2010: Proceedings of the 2010 International Conference on Intelligent User Interfaces, pp. 31–40 (2010)
2. Das, A.S., Datar, M., Garg, A., Rajaram, S.: Google news personalization: scalable online collaborative filtering. In: Proceedings of the 16th International Conference on World Wide Web (WWW 2007), pp. 271–280. ACM, New York (2007)
3. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work (CSCW 1994), pp. 175–186. ACM, New York (1994)
4. Pazzani, M., Billsus, D.: Content-Based Recommendation Systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 325–341. Springer, Heidelberg (2007)
5. Carreira, R., Crato, J.M., Gonalves, D., Jorge, J.A.: Evaluating adaptive user profiles for news classification. In: Proceedings of the 9th International Conference on Intelligent User Interfaces, pp. 206–212. ACM (2004)
6. Billsus, D., Pazzani, M.: A Hybrid User Model for News Story Classification (1999)
7. Billsus, D., Pazzani, M.J.: User Modeling for Adaptive News Access. In: User Modeling and User-Adapted Interaction, vol. 10, pp. 147–180. Springer, Netherlands (2000)

8. Bogers, T., Bosch, A.: Comparing and evaluating information retrieval algorithms for news recommendation. In: Proceedings of the ACM Conference on Recommender Systems (2007), pp. 141–144 (2007)
9. Kobilarov, G., Scott, T., Raimond, Y., Oliver, S., Sizemore, C., Smethurst, M., Bizer, C., Lee, R.: Media Meets Semantic Web – How The BBC Uses DBpedia and Linked Data to Make Connections. In: Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvönen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E. (eds.) ESWC 2009. LNCS, vol. 5554, pp. 723–737. Springer, Heidelberg (2009)
10. Maruščák, D., Novotný, R., Vojtáš, P.: Unsupervised Structured Web Data and Attribute Value Extraction. In: Proceedings of 8th Annual Conference Znalosti 2009, Brno (2009)
11. Robert, I., Jurgen, U., Christian, B., Andreas, H.: LDSpider: An open-source crawling framework for the Web of Linked Data. In: Proceedings of 9th International Semantic Web Conference (ISWC 2010). Springer, Heidelberg (2010)
12. Robertson, S.E., Jones, K.S.: Relevance weighting of search terms. Journal of the American Society for Information Science, 129–146 (1976)
13. Lašek, I., Vojtáš, P.: Semantic Information Filtering - Beyond Collaborative Filtering. In: 4th International Semantic Search Workshop (2011), `http://km.aifb.kit.edu/ws/semsearch11/11.pdf` (accessed June 13, 2011)
14. Agrawal, R., Imielienski, T., Swami, A.: Mining Association Rules between Sets of Items in Large Databases. In: Proceedings of Conference on Management of Data, pp. 207–216. ACM Press, New York (1993)
15. Han, E.-H., Karypis, G.: Centroid-Based Document Classification: Analysis and Experimental Results. In: Zighed, D.A., Komorowski, J., Żytkow, J.M. (eds.) PKDD 2000. LNCS (LNAI), vol. 1910, pp. 424–431. Springer, Heidelberg (2000)
16. MacQueen, J.B.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Symposium on Math, Statistics, and Probability, pp. 281–297. University of California Press, Berkeley (1967)
17. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer, Heidelberg (1999)
18. Lv, Y., Moon, T., Kolari, P., Zheng, Z., Wang, X., Chang, Y.: Learning to model relatedness for news recommendation. In: Proceedings of the 20th International Conference on World Wide Web, WWW 2011, pp. 57–66. ACM Press (2011)
19. Robertson, S., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: Proceedings of SIGIR 1994, pp. 232–241. ACM Press, New York (1994)
20. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proceedings of SIGIR 1999, pp. 275–281. ACM Press, New York (1998)
21. Krajci, S., Krajciova, J.: Social Network and One-sided Fuzzy Concept Lattices. In: Proceedings of FUZZ-IEEE 2007, IEEE International Conference on Fuzzy Systems, pp. 1–6. Imperial College, London (2007)
22. Fišer, D.: Sémantická anotace doménově závislých dat. Katedra softwarového inženýrství, MFF UK (2011)