# ORGANISING INCOMING KNOWLEDGE BY TEXT CLASSIFICATION

**Izidor Matušov**
Bachelor Degree Programme (3), FIT BUT
E-mail: xmatus19@stud.fit.vutbr.cz

Supervised by: Aleš Smrčka
E-mail: smrcka@fit.vutbr.cz

**Abstract**: Nowadays, the count of new articles waiting to be read is getting bigger. There is a need for an efficient way how to organise them. The approach proposed in the paper is to classify an article by its content instead of by source it comes from. This paper describes an application of a text classification and its enhancement—an usage of word sense disambiguation for the features extraction. Another feature for better user experience is determining how difficult is to read the article.

**Keywords**: RSS, text classification, readability tests

## 1 INTRODUCTION

Nowadays mankind suffers from information overload. The amount of available information increases rapidly. The proliferation is caused by availability of the Internet and by lowering barriers of publishing content. Formats for *syndicating contents* like RSS and Atoms easily deliver users new articles, blog posts, videos, and podcasts right to the aggregator. Just couple information sources can generate tens or hundreds pieces of a new content.

However, the user is not usually interested in every piece of a new information. Typically, the political articles in online version of newspapers go with sport articles which are not exciting for every user. Another situation is a big amount of similar articles reacting to the same community, national, or worldwide event. The articles came from different sources, and thus has a different point of view on the event. The reasons described above imply a need for an efficient organising tool of a syndicated content.

We introduce the reader to the topic in Section 2. In Section 3, techniques used for classification are described. In Section 4, we propose several techniques which extend the workflow described in Section 3. We conclude the paper in Section 5.

## 2 STATE OF THE ART

There are many sophisticated applications which support subscribing and working with the syndicated content. Most of them is capable of organizing content.

The basic approach is to put a source of content into a *category*. New articles are then placed into a category regardless of the actual content. A few application goes further. Users set the source of an information to multiple categories sometimes called *tags*. It is convenient especially for mixed sources. For example, a content of a newspaper could be tagged *sport* and *government*. *Google Reader*, a web-based application, has implemented such a way of organising content.

Advanced applications organise content based on a set of rules, e.g. *put in a special category called "Sport" the content from cnn.com which contains the word "goal"*. Achievable effects are various and strongly depend on possibilities of the application. To mention just few of them: title, author,

length, or presence of a word in the article are used. The feature is present in application Liferea where it is called as *Search Folder*. Although it allows organising articles based on their content, a user intervention is needed to create a set of rules. Such a definition is time consuming and not so intuitive for novice users.

## 3 A CLASSIFICATION METHOD

Organising by a content of an article uses techniques of machine learning. The existing set of articles with already assigned keywords is needed. Each article has its own *features*—frequency and presence of words, length, author, and date of publishing. Features and keywords of articles are called *training data* which is taken as the input of machine learning algorithm. The goal of algorithm is to create a set of classification rules which is performed by finding dependences between some features and keywords. Afterwards, the rules are applied on other articles (incoming knowledge) to assign keywords. The Figure 1 shows the workflow of the process.

Extracting features of an article is a domain of natural language processing (NLP). At first, the content of an article is split into words using *tokenization* which can be done just by splitting on white spaces characters or delimiters. Variants of the same words, e.g. *working*, *worked*, *work*, are represented as the same word using *stemming algorithm*. Some words with no particular sense called *stop words*, e.g. *and*, *the*, or *of*, are omitted. To simplify computation of the classification, only frequent words are taken as the features of the article. The algorithms are described in detail in [1].

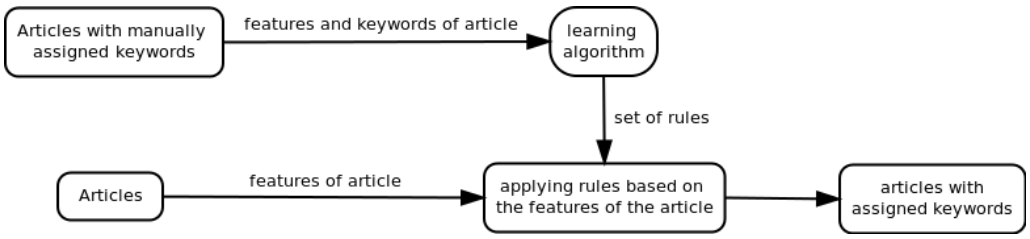


**Figure 1:** The workflow of an automatic assigning keywords.

The proposed approach is not rigorous because it usually extrapolates the small set of training data to bigger set of articles with unknown keywords. The problem can be cleared with user cooperation. When user finds an incorrectly assigned keyword or a missing one, the article and its keywords are added to the set of training data. The learning part runs again to create a new set of rules which are applied on new articles.

The algorithm works even with missing dependences between words, e.g. dependencies between objects the words represent in the reality, which are not present in the text. The algorithm just guesses connections between features and the keywords. If the text contains word *Facebook*, the article has a big chance to be about Facebook, the social network. But common words like *new* have many senses and often needs other words to determine their sense, e.g. with *York* there is a big chance the article is something about in New York.

## 4 DESIGN OF THE CLASSIFICATION TOOL

The big disadvantage of machine learning algorithm is the initial learning stage. The user must assign keywords to several articles to create the training data. To decrease the influence of this disadvantage, we use existing database of keywords assigned to articles. There is a web service which allows people to manage their links to the articles with setting tags—keywords. The service is called Delicious.com. It is mainly used by English speaking Internet users. Delicious.com offers the API suitable to base

on the training data which can greatly eliminate the learning stage. Moreover, some newspaper publishing services, like The New York Times on their website nytimes.com, also publish human edited metadata including keywords.

We also propose an usage of a technique how to improve the classification tool, in particular, word sense disambiguation. A text of an article consists of words. However, words do not exactly correspond with their meanings. There are two cases: (1) the same word represent two different meanings (*church* as "*a building*" or as "*a religious service*") and (2) the different words represent the same meaning ("*buck*" and "*dollar*" mean *money*). Word sense disambiguation algorithm [2] compare senses of words which are in relationship, and choose the most appropriate meaning for each word. The freely available lexical database WordNet provides the list of meanings for each word and data needed for the semantic similarity between two meanings.

Another interesting feature to consider are so-called readability tests [3] which are widely used in the United States. The algorithm of the readability test takes the text as an input and results with a number representing how much difficult is to read the text. Easily said, the algorithm compares how long sentences and how complicated words are, while the definition of a complicated word is based on a consequence of the Zipf's law such that the longer words are used less and not so easily understood as the shorter words. In the tool proposed here, readability tests inform the user how difficult the text is by assigning a special set of keywords. When more articles with similar content are available, the user can choose which article read first with regard to how the text is written.

## 5 CONCLUSION

This paper described an application of machine learning and NLP algorithms in the practical application for organising syndicated content. Organization of incoming knowledge is an interesting problem and several papers have been published. Although our proposed solution is a quite common with (notably) recent [4], there are major differences. We suggested a classification into multiple categories instead of a single category. If we assume the manual classification is correct, the both solutions have a various accuracy between 70 and 95% depending on a certain classification method and extracting features. We also propose the automation of the initial learning stage, an enhancement of the features extraction, and a categorization by the readability.

## REFERENCES


[1] C. D. Manning and H. Schütze. Foundations of Statistical Natural Language Processing. Cambridge : MIT Press, 1999. 680 p. ISBN 02-621-3360-1.

[2] R. Mihalcea, Unsupervised Large-Vocabulary Word Sense Disambiguation with Graph-based Algorithms for Sequence Data Labeling. In *Proceedings of the Joint Conference on Human Language Technology / Empirical Methods in Natural Language Processing (HLT/EMNLP)*, Vancouver, October, 2005

[3] W. H. DuBay. The Principles of Readability. Costa Mesa, CA: Impact Information, 2004, *online*, `www.impact-information.com/impactinfo/readability02.pdf`, [Cited: 2011-03-01]

[4] S. Saha, A. Sajjanhar, S. Gao, R. Dew, Y. Zhao. Delivering Categorized News Items Using RSS Feeds and Web Services. In *the* $10^{th}$ *IEEE International Conference on Computer and Information Technology*, pp. 698-702, 2010.