Using Genres to Improve Search Engines

Vedrana Vidulin
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana
Slovenia
vedrana.vidulin@ijs.si

Mitja Luštrek
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana
Slovenia
mitja.lustrek@ijs.si

Matjaž Gams
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana
Slovenia
matjaz.gams@ijs.si

Abstract

Modern search engines are typically queried with keywords, which foremostly convey the topic of the sought web page. Consequently the resulting top hits are often topically relevant, but nonetheless not what the user wants. The premise of this paper is that the relevance of the hits can be improved when also searching by genre, classification criterion orthogonal to topic. To this end a genre classifier was built using machine learning methods. It was used in web page retrieval to filter out the hits not belonging to the desired genre. This approach considerably improved the relevance of the top ten hits, which indicates that genre classifier can be a useful addition to search engines.

Keywords

genre classifier, search engine, multilabeled classification, web page retrieval

1. INTRODUCTION

Modern search engines rely on queries composed of keywords and ranking algorithms to retrieve web pages and to rank them by relevance [21]. The problem with keyword search is that it often cannot precisely capture the user's intent. For example, searching for the keyword "elephant" will result in a list of web pages describing the life of elephants in various levels of detail, safari picture galleries, newspaper articles about saving the elephants in Africa etc. If the user is interested only in scientific papers on the life of elephants, specifying the genre as "scientific" would give more precise results.

A web page is used to send a message to the user. The message has a topic, for example the life of elephants, but it also tries to communicate that topic in a specific way. To a zoologist it will give a high number of objective facts about elephants. When wishing to entertain, it will present pictures and video material. In the light of the previous explanation, genre can be described as intentional styling of a web page with the objective to communicate the topic in a specific manner. For the purpose of automatic genre classification, a genre can simply be defined as the style of a web page [9].

Recent experiments with genre classification on a corpus obtained by web crawling [18] show that the problem of incorporating genres into search engines is far from resolved. Our experience with including a genre classifier into Alvis semantic search engine [1] confirmed the difficulty of the task. This paper presents the work carried out for Alvis with several modifications.

In Section 2, we describe present related work. The corpus we experimented with is presented in Section 3. Section 4 lists the features used to describe web pages. In Section 5 the experiments with several machine learning (ML) algorithms are described. Section 6 presents the web page retrieval experiment. Section 7 concludes the paper.

2. RELATED WORK

A lot of work has been done on leaning a genre classifier. Most researchers used single-labeled approach, meaning that each document belongs to only one genre. This approach is suitable in cases where the genres are clearly delineated; however [18] used multi-labeled approach, advocating that each web page can belong to zero, one or multiple genres.

Two criteria for choosing genre categories were explored. The first criterion is to choose only those genres that can be directly instantiated by formulating text in the proposed genre. For example, 7-Web-Genre Collection described in [18] uses this criterion, including genres like *Personal home page* or *FAQ*. The second criterion is to choose genres that could cover all the web pages on the Internet. Such broad categories that subsume several genres are presented in [15]. For example genre *Journalistic materials* includes press reportage, editorial and review, while genre *Informative materials* includes recipes, lecture notes and encyclopedic information. The advantage of the second approach is that it can cover more easily the diversity of the Internet. However, the disadvantage lies in the difficulty to represent common characteristics of web pages that compose such broad categories. Hence, genre classifiers learned on corpuses with broader categories showed somewhat lower performance.

Table 1. Overview of features and classification algorithms used in genre classification

Features \ ML Algorithms		Decision trees	Naïve Bayes	SVM	Discriminant analysis	Regression	Neural networks	Nearest neighbour
Surface	**Function words**	[3] [6] [10]			[11] [22]	[12] [22]		
	Genre-specific words, punctuation, classes of words	[4] [5] [6] [10]	[5]	[2] [5]	[11] [22]	[12] [22]	[12]	[15]
	Word, sentence and document length	[4] [5] [6] [10]	[5]	[5]	[11] [22]	[12] [22]	[12]	[15]
Structural	**POS**	[3] [5] [6]	[5] [19]	[5]	[11] [22]			[15]
	Sentence type							[15]
Presentation	**Token type**	[4] [5] [6] [10]	[5]	[5]	[22]	[12] [22]	[12]	[15]
	Graphical elements and other HTML tags	[4]						[15]
	Links	[4]						[15]
Other	**URL**							[15]

Various genre describing features were explored, most commonly grouped into surface, structural, presentation and other features.

Surface features pertain to the text. They are easily extractable and do not require sophisticated parsing. This group includes function words [13], genre-specific words, punctuation marks, classes of words (such as dates, times, postal addresses and telephone numbers) and word, sentence and document length.

Structural features require some form of natural language processing. They include features like parts of speech (POS) [19], phrases (e.g. noun phrase or verb phrase) and sentence types (the frequencies of declarative, imperative and question sentences).

Presentation features mainly describe the appearance of a document, although token type also pertains to the content. As such, most of them cannot be extracted from plain text document. They are most commonly used with web pages [15]. This group includes token type (like the percentage of a document taken by numbers or whitespaces), graphical elements and other HTML tags (e.g. the frequencies of images or tables) and links.

Other features can be the characteristics of URL in a web page corpus [15] etc.

The ML algorithms most commonly used for genre classification are: various types of decision trees, Naïve Bayes, SVM, discriminant analysis, regression, neural networks and nearest-neighbor methods [5].

Overview of features and classification algorithms is presented in Table 1.

Web page retrieval experiment that examines the impact of genres on the relevance of retrieved web pages is presented in [26]. Only two genres (instead of genres they use types to avoid instantiation problem) are considered, *Course Page* and *Instruction Document*.

3. CORPUS

The 20-Genre Collection corpus [23] consists of 1,539 web pages classified into 20 genres. Considering that the corpus was gathered from the Internet, where genres are far from clearly delineated, we decided for the multilabeled approach, i.e., each web page can belong to multiple genres.

Genre categories were chosen with the intention to cover whole Internet. Accordingly, we defined category

like error message (useful to filter out such web pages), although it is unlikely that a user would search explicitly for such web pages.

The composition of the corpus is presented in Table 2.

Table 2. Composition of corpus

GENRE	WEB PAGES	GENRE	WEB PAGES
Blog	77	Informative	225
Children's	105	Journalistic	186
Commercial/Promotional	121	Official	55
Community	82	Personal	113
Content Delivery	138	Poetry	72
Entertainment	76	Pornographic	68
Error Message	79	Prose Fiction	67
FAQ	70	Scientific	76
Gateway	77	Shopping	66
Index	227	User Input	84

The genre of a web page can be described through the communicational intention that shapes the page. *Blog* presents updates on what is going on with an entity. *Children's* presents content in a simple and colorful way specifically suited for children. *Commercial/promotional* web pages are intended to invoke the visitor's interest in goods or services, typically for commercial gain. *Community* type web page involves the visitor in the creation of the page and enables interaction with other visitors. *Content delivery* delivers content that is not a part of the page. *Entertainment* web pages entertain the visitor. *Error message* tells the visitor to go away. *FAQ* are intended to help a user to solve common problems by answering frequently asked questions. *Gateway* transfers the visitor to another page. *Index* transfers the visitor to a selection of multiple other pages. *Informative* conveys objective information of permanent interest suitable for general population. *Journalistic* conveys mostly objective information on current events. *Official* conveys information with legal or otherwise official consequences. *Personal* conveys subjective, personal information in an informal way. *Poetry* presents poems and lyrics with intention to evoke emotions. *Pornographic* web pages have intention to sexually arouse the visitor. *Prose fiction* presents story about real or fictional event in artistic form with intention to evoke imagination and emotions. *Scientific* conveys objective information suitable for experts. *Shopping* web pages sell goods or services online. *User input* solicits the visitor's input.

The web pages were collected from the Internet using three methods. Firstly, we used highly-ranked Google hits for popular keywords like "Britney Spears". The keywords were chosen according to the Google Zeitgeist statistics [27]. Our purpose was to build a classifier that will not have a problem with recognizing the most popular web pages. Secondly, we gathered random web pages. Finally, we specifically searched for web pages belonging to the genres underrepresented to that point to obtain a balanced corpus that represents all genres equally well (imbalance usually cause the difficulties in learning inadequately represented genres). Only web pages in English were collected. The corpus was manually labeled with genres by two independent annotators. Their labels disagreed on about a third of the web pages in the corpus, so those were reassessed by a third and sometimes even a fourth annotator.

4. FEATURES

Table 3. The feature set

	FEATURES
SURFACE	**Function words**: number of occurrences of 50 most common function words in the corpus / total number of function words. **Genre-specific words, phrases and punctuation marks**: number of occurrences of 321 selected content words / total number of content words, number of occurrences of 34 selected punctuation symbols / total number of punctuation symbols. **Classes of words or phrases**: number of named entities of the classes date, location and person / total number of words. **Word length, sentence length, document length**: average number of characters per word, average number of words per sentence, number of characters in hyperlink text / total number of characters.
STRUCTURAL	**POS tags**: number of occurrences of 36 available POS tags / total number of words, number of occurrences of 1,869 selected POS trigrams / total number of POS trigrams. **Sentence types**: number of declarative sentences, interrogative sentences, exclamatory sentences and other sentences (in most cases list items) / total number of sentences.
PRESENTATION	**Token type**: number of alphabetical tokens (sequences of letters), numerical tokens (sequence of digits), separating tokens (sequences of separator characters, such as spaces and returns) and symbolic tokens (sequences of characters excluding alphanumeric and separator characters) / total number of tokens. **Graphical elements and other html tags**: number of tags belonging to classes text formatting, document structure, inclusion of external objects, interaction and navigation / total number of tags; number of single tags / total number of tags. **Links**: number of hyperlinks to the same domain, to a different domain and containing "mailto" / total number of hyperlinks.
OTHER	**URL features**: depth of URL, document type (static HTML, script, other), top-level domain (com, org, edu, national…), presence of "www", tilde (/~), year, query (?foo) and fragment (#foo), presence of 54 most common words in URL.

Features were selected by analyzing 20-Genre Collection corpus and consulting the genre literature. In total, 2,492

Table 4. The comparisons of ML algorithms

RANK BY / ML ALG.	ACCURACY	PRECISION	RECALL	F-MEASURE	TOTAL WINS	TOTAL LOSSES	TOTAL SCORE
SMO	8 (20, 51)	7 (22, 38)	1 (49, 6)	3 (43, 8)	134	103	31
SMO – FS	3 (47, 8)	3 (46, 13)	8 (13, 47)	7 (21, 31)	127	99	28
NAÏVE BAYES	9 (0, 154)	9 (3, 120)	5 (43, 28)	8 (8, 87)	54	389	-335
NAÏVE BAYES – FS	7 (17, 22)	8 (15, 34)	7 (17, 20)	6 (21, 24)	70	100	-30
J48	6 (25, 28)	6 (22, 27)	3 (31, 4)	4 (36, 2)	114	61	53
J48 – FS	4 (32, 3)	4 (28, 4)	4 (29, 6)	2 (37, 1)	126	14	112
ADABOOST	1 (61, 0)	1 (51, 1)	6 (25, 13)	5 (37, 4)	174	18	156
ADABOOST – FS	2 (59, 4)	2 (44, 2)	2 (41, 5)	1 (54, 1)	198	12	**186**
RANDOM FOREST	5 (34, 25)	5 (23, 15)	9 (0, 119)	9 (0, 99)	57	258	-201

features were chosen belonging to all four groups shown in Table 1. The features are listed in Table 3.

Content words were selected by combining the list of most frequent content words from the corpus and manually selected words that describe the genres of web pages. They were stemmed by the Porter stemming algorithm [16].

POS tags were extracted with TreeTagger [20]. POS trigrams were selected in two steps. In the first step only trigrams that are present more than three times in a web page were extracted. In the second step, 25% of the most frequent and 25% of the least frequent trigrams in the corpus were discarded.

5. ML EXPERIMENTS

In order to fairly compare several ML algorithms, we decided to work with Weka [24], a ML suite containing a large number of such algorithms. Since Weka does not support multilabeled classification and neither do some of the algorithms under consideration, the ML problem was divided into 20 binary sub-problems, one for each genre. The task was thus to train 20 sub-classifiers, each to decide whether an input web page belongs to one of the 20 genres.

The data set was split into 67% for training the sub-classifiers and the rest for testing. Because some genres have a relatively low number of examples, it was important to preserve the ratio of the examples of each genre in both the training and the test set. However, stratified splitting of the data set was problematic due to the examples having multiple genre labels. Therefore, we assigned each example a single label for splitting in a manner that prioritized the less represented genres, since the quality of the classifier and the reliability of its testing would suffer more if those were split improperly. Weka filter Stratified Remove Folds was finally used on the single-labeled data set.

Five of the algorithms available in Weka were selected, three recommended in genre literature and two of our own choice. The recommended algorithms are support vector machines (SVM), Naïve Bayes and decision trees [5]. In Weka, SVM is implemented as sequential minimal optimization (SMO) and decision trees as J48. Boosting and Random Forest are the algorithms of our choice. In Weka, boosting is implemented as AdaBoostM1; we boosted decision trees.

Each algorithm except Random Forest was trained both on all features and on selected features only. Feature selection (FS) was carried out on all features using the training set. The space of feature subsets was searched by best-first search [17] and each subset was evaluated with CfsSubsetEval evaluation function [24]. This function favors the feature subsets composed of features highly correlated with the class and poorly intercorrelated.

FS was conducted separately for each genre and the number of selected features varied from 15 to 69.

Each algorithm was run 10 times and for each experiment the data set was split into training and test set separately. Normally we would have used ten-fold cross-validation, but a sizable test set was needed for web page retrieval experiments described in the next section, so we used 67:33 split in all the experiments. Given 20 genres, we conducted 200 experiments in each of them comparing all 9 ML algorithms. The performance was measured in terms of accuracy, precision, recall and F-Measure. The algorithms were compared to each other using the corrected resampled paired t-test [24] with significance level of 5%. The results are presented in Table 4. The first number in the brackets represents the number of wins of a given algorithm, i.e., the number of experiments where it significantly outperformed the rest. The second number in the brackets represents the number of losses, i.e., the number of experiments where the algorithm performed significantly worse than the rest. The number outside the brackets is the algorithm's rank according to the total score calculated by subtracting the number of losses from the number of wins.

AdaBoost clearly dominates the ranks. Only in the case of recall, SMO was ranked first, but even there AdaBoost with FS was ranked second. Our final choice was AdaBoost with FS, although the decision regarding FS is debatable. The choice was based on two criteria. The first criterion was the total score calculated by subtracting total losses (the sum of losses over all four measures) from total wins (the sum of wins over all four measures) for each

algorithm. The second criterion was that smaller feature sets are less prone to overfitting [25].

The final genre classifier, which was also used in the web page retrieval experiment, was trained on a single training set consisting of 67 % of the data set. It was tested on the remainder of the data set, which was also used in the web page retrieval experiment. Table 5 presents AdaBoost's accuracy, precision, recall and F-Measure for each genre. The last row gives the measures averaged over the sub-classifiers for all the 20 genres.

Table 5. Performance of genre classifiers in percent

	ACC.	PREC.	REC.	F-ME.
BLOG	96	71	56	63
CHILDRENS'	94	50	36	42
COMMERCIAL/	91	33	13	19
COMMUNITY	98	90	68	77
CONTENT	91	47	17	25
ENTERTAINMENT	94	33	27	30
ERROR MESSAGE	97	79	73	76
FAQ	99	94	77	85
GATEWAY	95	41	22	29
INDEX	85	53	42	46
INFORMATIVE	83	33	16	21
JOURNALISTIC	92	80	47	59
OFFICIAL	97	63	28	39
PERSONAL	94	64	36	46
POETRY	97	82	56	67
PORNOGRAPHIC	97	75	52	62
PROSE FICTION	96	57	32	41
SCIENTIFIC	96	89	32	47
SHOPPING	97	89	38	53
USER INPUT	97	83	61	70
AVERAGE	94	65	41	50

The average accuracy in Table 5 is 94%, which is impressive, but unfortunately it is not a good indicator of the performance in this case. By splitting the multi-labeled ML problem into 20 binary sub-problems, we got 20 unbalanced data sets with high numbers of negative and low numbers of positive examples. Sub-classifiers that would recognize only negative examples would still be highly accurate. Precision indicates the number of web pages truly belonging to a given genre out of all the web pages the classifier recognized as such. The average precision was 65%. Recall indicates the number of web pages recognized by the classifier to belong to a given genre out of all the belonging web pages. The average recall is only 41%, but for a search engine, this is often not problematic, because most queries return a much greater number of hits than a user needs. In general, recall and precision are inversely related: as you attempt to increase one, the other tends to decline [14]. Recall is also inadequate as a single indicator of performance because recall of 100% can be achieved simply by retrieving all examples [8]. To obtain a single information retrieval indicator that will consider both precision and recall, we used F-Measure. F-Measure is calculated as presented in Eq. 1.

$$\frac{2 \times recall \times precision}{recall + precision} \quad (1)$$

The average F-measure is only 50% due to the poor recall, but as mentioned before, in our particular case precision is more important, as can be seen in the next section.

6. WEB PAGE RETRIEVAL EXPERIMENTS

The web page retrieval experiments were conduced using Google Desktop [7]. Google is one of the most popular search engines, but since we experiment on corpus, we resorted to its desktop version. Our test set of web pages from the previous section was labeled with genres and indexed by Google Desktop. 18 keywords (or sets of keywords) were selected in such a way that they gave at least a hint of what genre the user was looking for. By this approach we wanted to examine if it is possible to obtain a web page of desired topic and genre through keywords alone. In the second phase of the experiment the accompanying genres were used to limit the hits to the desired genre.

The queries were entered into Google Desktop: once with and once without specifying the genre. The quality of the search hits was measured by precision at 10 (P@10), which is the fraction of relevant hits among the top 10 (or fewer, when fewer than 10 hits were returned). The results of the first experiment are presented in Table 6 and the

Table 6. Results of querying using keywords

	NUM. OF RELEVANT/ NUM. OF RETRIEVED	P@10
madonna lyrics	1/3	34
erotic story	3/4	75
hurricane	6/10	60
orlando bloom news	2/8	25
joke	4/10	40
tech blog	2/6	34
sex shop	2/8	25
horoscope	2/8	25
fanfiction	2/3	67
game faq	2/8	25
kids entertainment	2/10	20
poem	4/10	40
terms and conditions of use	4/10	40
file not found	4/10	40
online forum	3/10	30
health research	4/10	40
software download	4/10	40
porn	4/10	40
AVERAGE		39

results of the second one in Table 7. The keywords (and genre) are in the first column. In the second column, the first number represents the number of relevant hits among the top 10 and the second number the total number of hits (capped at 10).

On average, P@10 is 39% when using only keywords. The combination of keywords and genres resulted in the average P@10 of 81%, more than twice as much as without genres. It must be noted that the total number of hits when genres were specified was much lower, which can be explained by the low recall of the genre classifier. However, as mentioned before, this often does not matter to search engine users; when a user has a very specific query and the total number of relevant hits is consequently low, a search engine could automatically ignore the genre.

Table 7. Results of querying using keywords and genres

	NUM. OF RELEVANT / NUM. OF RETRIEVED	P@10
madonna lyrics + poetry	1/1	100
erotic story + prose fiction	1/1	100
hurricane + informative	2/2	100
orlando bloom news + journalistic	2/2	100
joke + entertainment	2/4	50
tech blog + blog	1/2	50
sex shop + shopping	1/1	100
horoscope + entertainment	1/3	34
fanfiction + prose fiction	2/2	100
game faq + faq	1/2	50
kids entertainment + childrens'	2/2	100
poem + poetry	6/7	86
terms and conditions of use + official	3/3	100
file not found + error message	1/2	50
online forum + community	3/3	100
health research + scientific	3/4	75
software download + content delivery	4/4	100
porn + adult	7/10	70
AVERAGE		**81**

7. CONCLUSION

Classification of web pages by genre is a feature yet to be exploited by search engines. The main reason is probably that current genre classifiers are not entirely up to the task. However, experiments presented in this paper show that it is possible to more than double the precision of a search engine by specifying genres besides keywords, despite the not entirely satisfactory performance of the genre classifier itself. This leads us to believe that genre classification could already be beneficial to search engine users. To verify this claim, we plan to experiment on the open Internet to determine if genres improve the precision of web page retrieval there as well. In addition, we intend to examine the computational expenses incurred by various steps of feature extraction, since this is a potential bottleneck in large-scale applications.

8. ACKNOWLEDGMENTS

The work presented in this paper was part of the 6FP European project ALVIS (002068), conducting research in the design, use and interoperability of topic-specific search engines with the goal of developing an open source prototype of a distributed, semantic-based search engine.

9. REFERENCES

[1] Alvis, 2007, http://www.alvis.info/alvis/ [2007-05-30].

[2] S. Argamon, and J. Dodick, "Conjunction and Modal Assessment in Genre Classification: A Corpus-Based Study of Historical and Experimental Science Writing", *AAAI Spring Symposium on Attitude and Affect in Text*, 2004.

[3] S. Argamon, M. Koppel, and G. Avneri, "Routing Documents According to Style", *First International Workshop on Innovative Information Systems*, 1998.

[4] I. Bretan, J. Dewe, A. Hallberg, N. Wolkert, and J. Karlgren, "Web-Specific Genre Visualization", *WebNet*, Orlando, USA, 1998.

[5] N. Dewdney, C. VanEss-Dykema, and R. MacMillan, "The Form is the Substance: Classification of Genres in Text", *Workshop on Human Language Technology and Knowledge Management*, ACL, 2001.

[6] A. Finn, *Machine Learning for Genre Classification*, MSc thesis, University College Dublin, 2002.

[7] Google Desktop, 2007, http://desktop.google.com/ [2007-05-10].

[8] Information retrieval – Wikipedia, 2007, http://en.wikipedia.org/wiki/Information_retrieval [2007-01-28].

[9] J. Karlgren, *Stylistic Experiments for Information Retrieval*, PhD thesis, 2000.

[10] J. Karlgren, "Stylistic Experiments in Information Retrieval", *Natural Language Information Retrieval*, 1999, pp. 147-166.

[11] J. Karlgren, and D. Cutting, "Recognizing Text Genres with Simple Metrics Using Discriminant Analysis", *Proceedings of 15th International Conference on Computational Linguistics*, Kyoto, Japan, 1994, pp. 1071-1075.

[12] B. Kessler, G. Nunberg, and H. Schuetze, "Automatic Detection of Text Genre", *Proceedings of ACL/EACL*, Madrid, Spain, 1997, pp. 32-38.

[13] M. Koppel, N. Akiva, and I. Dagan, "A Corpus-Independent Feature Set for Style-Based Text Categorization", *Proceedings of Workshop on Computational Approaches to Style Analysis and Synthesis*, IJCAI, Acapulco, Mexico, 2003.

[14] A. Large, L.A. Tedd, and R.J. Hartley, *Information seeking in the online age: Principles and practice*, Bowker, London, 1999.

[15] C.S. Lim, K.J. Lee, and G.C. Kim, "Multiple Sets of Features for Automatic Genre Classification of Web Documents", *Information Processing and Management*, 2005, pp. 1263-1276.

[16] M. Porter, “The Porter Stemming Algorithm”, 2007, http://www.tartarus.org/~martin/PorterStemmer/ [2007-01-10].
[17] S.J. Russell, and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd edition, Prentice Hall, 2002.
[18] M. Santini, *Automatic Identification of Genre in Web Pages*, PhD Thesis, University of Brighton, UK, 2007.
[19] M. Santini, “A Shallow Approach To Syntactic Feature Extraction For Genre Classification”, *Proceedings of 7th Annual Colloquium for the UK Special Interest Group for Computational Linguistics*, Birmingham, UK, 2004.
[20] H. Schmid, “Probabilistic part-of-speech tagging using decision trees”, *Proceedings of International Conference on New Methods in Language Processing*, 1994.
[21] Search engine – Wikipedia, 2007, http://en.wikipedia.org/wiki/Search_engine [2007-05-29].
[22] E. Stamatatos, N. Fakotakis, and G. Kokkinakis, “Automatic Text Categorization in Terms of Genre and Author”, *Computational Linguistics*, 2000, pp. 471-495.
[23] Web Genre Dataset, 2007, http://dis.ijs.si/mitjal/genre/ [2007-04-30].
[24] I.H. Witten and E. Frank, *Data Mining – Practical Machine Learning Tools and Techniques*, Elsevier Inc., 2005.
[25] L. Wolf, and I. Martin, “Robust Boosting for Learning from Few Examples”, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2005.
[26] J. Xu, Y. Cao, H. Li, N. Craswell, and Y. Huang, “Searching Documents Based on Relevance and Type”, *Advances in Information Retrieval, Proceeding of 29th European Conference on IT Research (ECIR 2007)*, Rome, Italy, 2007, pp. 629-636.
[27] Zeitgeist: Search patterns, trends, and surprises, 2005, http://www.google.com/press/zeitgeist.html, [2005-06-25].