RetroC – A Corpus for Evaluating Temporal Classifiers

Filip Graliński[1(✉)] and Piotr Wierzchoń[2]

[1] Faculty of Mathematics and Computer Science,
Adam Mickiewicz University in Poznań, Poznań, Poland
filipg@amu.edu.pl
[2] Institute of Linguistics, Adam Mickiewicz University in Poznań,
Poznań, Poland
wierzch@amu.edu.pl
http://amu.edu.pl
http://re-research.pl/en/

Abstract. We present a corpus for training and evaluating systems for the dating of Polish texts. A number of baselines (using year references, knowledge of spelling reforms and birth years) are given for the temporal classification task. We also show that the problem can be viewed as a regression problem and a standard supervised learning tool (Vowpal Wabbit) can be applied. So far, the best result has been achieved with supervised learning with word tokens and character 5-g as features. In addition, error analysis of the results obtained with the best solution are presented in this paper.

Keywords: Temporal classification · Optical character recognition
Vowpal Wabbit · Supervised learning

1 Introduction

In recent years, more and more historical material (such as old newspapers, books no longer under copyright and archival documents) has been digitised and made available online. Unfortunately, metadata, in particular creation/publication dates, is not always present. Moreover, old textual material is often made available on the Internet in an unstructured manner and mixed with contemporary Web texts.

The task of *automatic document dating* or *temporal text classification* consists in assigning a creation or publication date to a given text relying solely on its content – that is, without the need to use explicit metadata [4]. It can be viewed as a text classification problem (which period does it come from? – with, for instance, a yearly or decadal resolution) or as a regression problem (guess the time stamp as precisely as possible treating it as a continuous value). The task can be approached using either knowledge-based methods (through knowledge of the history of the orthography of a given language or using Wikipedia or

Z. Vetulani et al. (Eds.): LTC 2015, LNAI 10930, pp. 101–111, 2018.
https://doi.org/10.1007/978-3-319-93782-3_8

other external resources) or learning-based methods (supervised learning from a corpus of time-stamped texts).

In this paper, we present (1) two releases of *RetroC* – a publicly available corpus for evaluating and training systems for the automatic dating of Polish texts and (2) some baseline results obtained using the corpus.

In Sect. 2, we discuss previous work and the state of the art as regards temporal classification. Section 3 presents the rationale behind the RetroC(2) corpus, its source materials and scope. Section 4 discusses the availability of the corpus. Basic baselines and more advanced supervised methods are outlined in Sect. 5. Finally, an error analysis is presented in Sect. 6.

2 Previous Work

Although the problem of temporal classification is of significant importance for text processing and information retrieval, as well as in terms of its numerous applications (language chronologisation, support for the digitisation of cultural heritage), relevant literature is not abundant. This is probably a result of the lack of large, freely available, resources which might be used to train and test automatic dating systems.

The research problem was first raised by de Jong et al. [9]. Those authors presented an ambitious programme of using temporal unigram language models not only for the automatic dating of historic texts, but also for linking contemporary keywords with their historic variations. Since, unfortunately, there was no extensive diachronic corpus available, de Jong et al. carried out the experiment based on a fairly large but time-limited (1999–2005) corpus of Dutch-language press materials. Kanhabua and Nørvåg [10] developed further the method of de Jong et al. by applying semantic-based pre-processing (tagging parts of speech, excerpting collocations and filtering out words) and using statistical extensions of language models (word frequency interpolation, temporal entropy, the use of Google Zeitgeist). In order to learn and test the methods, a corpus of archival websites from an approximately 8-year period was used.

Evaluation of automatic dating methods was one of the objectives of the DEFT2010 workshop. To this end, a time-extensive (1800–1944), though relatively small (about 6300 texts) corpus of French newspaper texts was used. The best system obtained an F-measure of 0.338 [1]. Use was made of information about spelling reforms, birth dates of famous people and a module which learnt to chronologise vocabulary with conditional random fields.

The evaluation task was repeated during the DEFT2011 campaign. An advanced system based on information gained from external resources (birth dates, archaisms, neologisms, dates of spelling reforms) and on classification methods making use of a training corpus (classification based on the cosine distance, with modelling using support vector machines) was then constructed by Garcia-Fernandez et al. [5].

In order to improve the automatic dating results, Chambers [2] made use of a discriminant classifier, taking into account explicit temporal references in

Table 1. Summary of work on temporal classification

Paper	Language	Time span	Corpus/Size	Methods
de Jong et al. [9]	Dutch	1999–2005	2 GB raw text (train)/500 articles (test)	Unigram language models
Kanhabua and Nørvåg [10]	English	Web pages	?	POS tagging, collocations, filtering; word frequency interpolation, temporal entropy, Google Zeitgeist
Albert et al. [1]	French	1800–1944	DEFT2010 (6300 newspaper texts)	Spelling reforms, birth dates of famous people, CRF learning
Garcia-Fernandez et al. [5]	French	1801–1944	DEFT2011 (6050 newspaper texts)	External resources (birth dates, archaisms, neologisms, spelling reforms), SVM-based classification
Chambers [2]	English	1994–2002	Gigaword Corpus (New York Times section)	Discriminant classifier on temporal references and verb tenses
Ciobanu et al. [3]	Romanian	5 centuries	?	Learning-based methods
Guo et al. [8]	English	1502–2002	Hathi Trust (250K volumes)	
this paper	Polish	1814–2013	RetroC1, 59K texts, 212M	Linear regression
this paper	Polish	1814–2013	RetroC2, 153K texts, 537M	Linear regression

the dated text and parameters such as verb tense. Kumar et al. [11] applied language models learned from Wikipedia biographies to classify stories obtained from the Gutenberg Project, and Ciobanu et al. [3] trained a classifier based on a Romanian corpus, containing data from five centuries, to date contemporary historical novels. (This is a more difficult task than dating, for instance, press articles, which usually refer to events that are not distant in time from their publication dates.)

More recently, Guo et al. [8] applied various machine learning methods (e.g. SVMs) to a large dataset extracted from the HathiTrust digital library.

A summary of previous work is given in Table 1.

3 The RetroC Corpus

RetroC is a Polish-language diachronic corpus, spanning two centuries (1814–2013) and intended for training and testing automatic dating systems. It is mostly based on publications available in Polish digital libraries [7,13], plus some old textual material from other online sources.

There have been two releases of the corpus so far: the first one (RetroC1) in 2015 and the second one (RetroC2) in 2017. RetroC2 is not only larger (being a superset of RetroC1), but also contains extra features in the training set.

The corpus was designed with the following goals in mind:

- to be a collection of Polish texts;
- to be large enough to enable the use of statistical methods;
- to be time-extensive – not just modern Web-based texts, but also old printed materials;
- to cover relatively short fragments rather than whole books, for which the dating task is much easier.

In the second release of the corpus, some new objectives were considered:

- to treat time truly as a continuous variable
- and in the same time to take into account the fact that time granularity varies for publications (yearly for books, monthly for magazines, daily for newspapers, etc.);
- to make use of the fact that publications are usually clustered into collections, sources, etc.

Consequently, whereas the training set for RetroC1 contains just texts and years (given as integers) for each item, the training data in the RetroC2 corpus is a list of quintuples:

1. the beginning of the time span given as year with a fraction (e.g. 1933.7479 for a monthly published in October 1933),
2. the end of the time span given as year with a fraction (e.g. 1933.8328 for a monthly published in October 1933),
3. title of the publication,
4. identifier of the source of the publication (usually a digital library),
5. text fragment.

(3) and (4) are given only for the training set, so this information could not be used directly as a simple feature when testing. Motivation is that it could be used to detect unreliable features (e.g. words that occur only in one magazine or one source) while training. Also, the expected value for the test set is not a time span, but a single year with a fraction — mid-point of a given time span, e.g. 1933.7903 for a monthly published in October 1933, or 1921.5 for a book for which only the publication year (1921) is known.

RetroC corpora are divided into a training set, development sets and a test set. Their sizes are given in Table 2.

Table 2. Number of text fragments for each data set

	Train	dev-0	dev-1	Test
RetroC1	40,000	9,910	N/A	10,000
RetroC2	107,471	20,000	11,563	14,220

The RetroC1 dev-0 test set was incorporated into the RetroC2 training set and the RetroC1 test set became the RetroC2 dev-1 test set to avoid overfitting when switching to RetroC2 (the RetroC2 test set was formed using a completely new collection of digital libraries).

Each set is composed of 500-word fragments taken from random publications (500-word portions were also used in the DEFT corpus [5]). For instance, the following is a dev-set item taken from an 1855 publication from the e-library of Warsaw University[1] (which is the largest source of texts for the training and development sets):

> przeprawę. Szron ten zwiększa się w skutku przymrozków i śniegu, na czem w tych dniach zupełnie nam niebrak. Zapowiedziany Toro IV i ostatni dzieła p.n. Opisanie lasów Królestwa Polskiego i Gubernji Zachodnich CE- SARSTWA Rossyjskiego, już wyszedłz droku i znajduje [omitted for brevity] ubioru damskiego zastosowane, wyszły na r. 1855 nakładem i w litografji K. Romanowicza, przy ulicy Długiej Nr 578, przechodni dom na Bielańską. Nabyć ich także możua w składzie ryciu przy ulicy Senatorskiej, wdomuW 7 Neubauera. Nakładem Xięgarni Jana Breslauera, wyszła z druku powieść historyczna: Zamek Warszawski czjfli Rodzina Konrada, w 3ch tomach, przez J. N. Cżarnomskiego. Powieść ta opisuje w sposób nader zajmujący, ostatnie chwile Xięztwa Mazowieckiego i jego wcielenie do Korony. Cena exem: rs.2 k. 70 Rzadko takiego kursu sanek jak w dniu onegdajszyro, bo też dzień>był potemu, gdyż i dość mroźny, zatem pogodny i śnieg

As can be seen, a text in the RetroC corpus is given as it was found in the text layer of a DjVu/PDF file (with possible OCR noise and errors) – only minimal post-processing was applied (joining words separated with hyphens and new lines, removing end-of-lines and other non-printing characters, UTF-8 sanitisation). In contrast to the DEFT dataset [5], dates were not removed from texts (see *1855* in the example above); this was motivated by the fact that year references are obviously a useful (though not perfect) feature for temporal classification (and we aim to use classifiers trained with RetroC to find old textual material in large Web corpora where dates are available), although it is not as important as in [8], where whole volumes, including copyright and title pages, are taken into account.

The development and test sets are balanced with respect to publication year: 50 and 100 publications per year for, respectively, RetroC1 and RetroC2. We

[1] http://ebuw.uw.edu.pl.

were not able to find very many dev-set items for some years (in particular, during the early 19th century and World War II), hence the size of some sets is smaller than 200 (years) × 50/100 (texts). The development set and the test set are also balanced (as much as possible) with respect to their sources, in order to avoid the data set being overwhelmed by one large digital library. The training set is not balanced; the distribution of publication years therein is presented in Fig. 1.

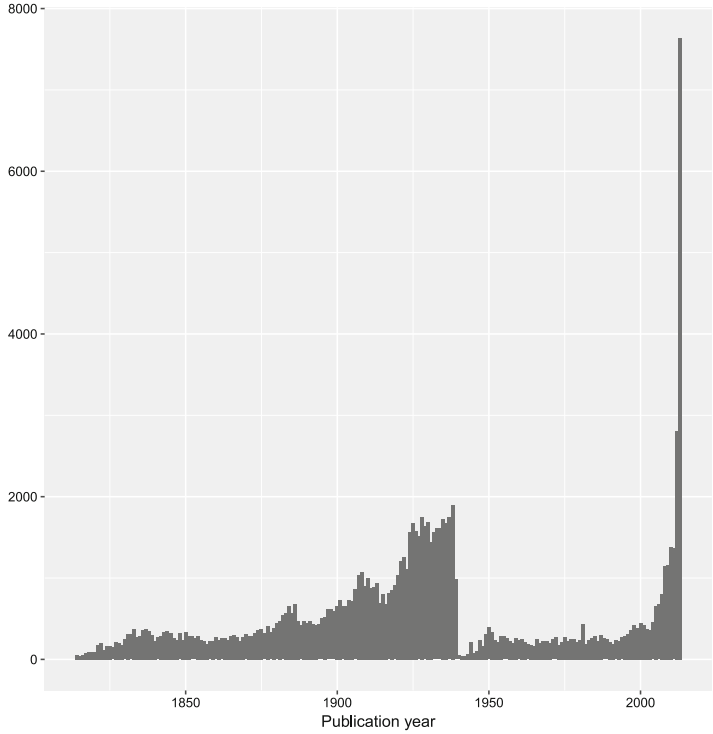

Fig. 1. Number of items in the training set

The training and dev-0 development sets are composed of texts from the same set of digital libraries. In order to make the challenge more difficult, the texts in the test set were taken from a separate set of sources (i.e. different digital libraries). This is a more realistic approach, as we would require that a temporal classifier work reliably on texts from new, unknown sources. To assess the quality of generalisation, an extra dev-1 development set taken from yet another set of sources was added in RetroC2.

The publication dates were extracted from the metadata from the digital libraries; no manual verification was performed, and there is no guarantee that all of the dates given in RetroC are correct (we also ignore whether it is in fact a publication date or creation date that is given).

4 RetroC as a Machine Learning Challenge

RetroC(2) data sets are available at Gonito.net (see http://gonito.net/challenge/retroc and http://gonito.net/challenge/retroc2). Gonito.net is an open source, web- and Git-based platform for hosting challenges for researchers in the field of machine learning (in particular: natural language processing) [6].

The key design feature of Gonito.net is using Git for managing challenges and solutions of the problems submitted by competitors. Thus, the corpus is freely available, simply from a Git repository (see repositories `git://gonito.net/retroc.git` and `git://gonito.net/retroc2.git`). In other words, there is no need to log in to Gonito.net web application to just download the data, Git command-line tool is enough.

Gonito.net web application is used to submit solutions and keep track of the effort of a given research community and progress in terms of clear evaluation metrics. Submitters are encouraged (but not forced) to upload source codes along with the test outputs as this allows for research transparency and reproducibility. For each solution described in this paper, a Gonito.net reference is given, for instance the null model that always returns 1913.5 (midpoint for the whole RetroC time span) is available at Gonito.net at {2ef3f0} (in case you are reading a physical copy of this paper, go to http://gonito.net/q and enter the reference number there). The Gonito.net reference is basically a Git commit ID, so even if the Gonito.net platform ceases to exist, the results and source codes may still be available as a regular Git repository to be cloned and inspected with standard Git tools, no external database is needed.

The Gonito.net machine learning task defined along with the RetroC(2) corpus is configured to use root-mean-square error (RMSE) as the evaluation metric, e.g. the null model yields RMSE = 52.5.

5 Baseline Solutions

5.1 Simple Baselines

A very simple baseline is to return the latest year reference found in the text (and back up to the null model if no year reference is found). This simple solution yields RMSE = 37.7 for RetroC2 (Gonito.net reference: {c9c6ce}), which is surprisingly much better than the null model.

Another simple method would be to use hand-crafted rules using the knowledge of spelling changes in Polish (*-dz*/*-c* ending for verbs, *-ya*/*-ja*/*-ia* ending for nouns, *-emi*/*-ymi* ending for adjectives, see Fig. 2). We obtained RMSE = 44.2 with this simple solution ({9f55cd}).

Both simple methods chained (first checking year references, then hand-crafted rules) yielded RMSE = 35.8 ({bd8665}), which could be treated as a simple rule-based baseline.

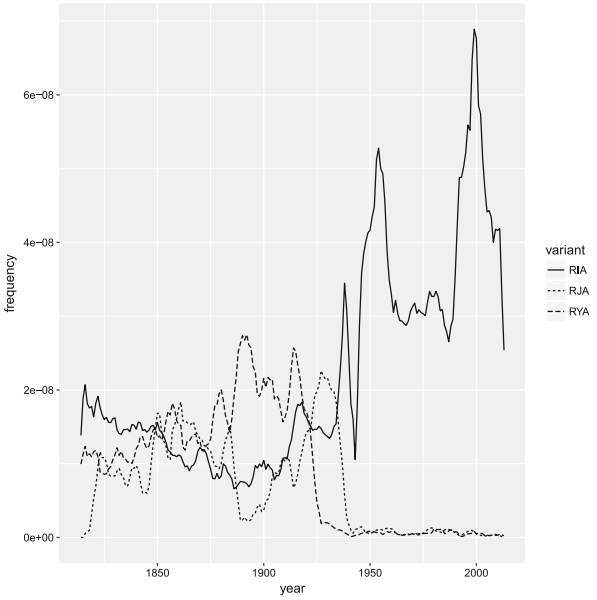

Fig. 2. Frequency of orthographic variants

5.2 Supervised Learning

As time could be treated as a continuous variable, temporal classification could be viewed as a regression problem. We trained a regressor using the Vowpal Wabbit open-source learning system [12] and RetroC training set. Both characters 5-g (as suggested in [5]) and word tokens were used as features. In addition, a small neural network (6 units) was used. This way, the best results so far were obtained: RMSE = 24.8 years for RetroC1 {9dcf6a} and RMSE = 19.5 years for RetroC2 {6ab497}. Better results for RetroC2, though not strictly comparable, might suggest that more training data might still improve the results.

The character n-grams with the highest weights (for RetroC1, with neural network switched off) are presented in Table 3. Some of the most informative features are quite obvious (e.g. year references), others less so – for instance, frequent words *ktoś* (*somebody*) and *czym* (*with what*) are informative as they were spelled as *któś* and *czem* during the 19th century, *tzw.* (*so-called*) is a relatively new abbreviation, *tal.* is a abbreviation for a monetary unit used in the 19th centry (*talar*), *aig* is a very frequent Polish word *się* mis-recognised by OCR.

In order to test the assumption that giving publication time with the highest resolution possible brings improvement when training a temporal classifier (an extra assumption introduced in RetroC2), the best solution was re-trained with publication time-stamps rounded to full years. As expected, the results were slightly, but significantly worse for the test set (RMSE = 19.7, {60e217}), which confirms the assumption.

Table 3. The features with the highest scores

	Positive	Negative
1	stori	tém
2	czym	aig
3	wtedy	»
4	*dash*	il
5	”	i5
6	tzw	)”
7	‘	téj
8	ktoś	tal
9	2009	1837
10	1985	storj

6 Error Analysis

In order to (1) learn of any defects in the corpus and (2) get insights how to improve the temporal classifier, we compared dates returned by the best solution obtained so far ({6ab497}) with the expected dates – see Fig. 3. As can be seen, whereas there is a clear tendency for the oldest texts to be misclassified as belonging to the later period, more noise can be observed for the texts from the late 20th century. In addition, there are a number of outliers. The top 100 outliers (the test cases with the highest discrepancy between publication date expected and the value returned) were inspected manually:

- 5 texts were assigned incorrect temporal metadata (e.g. 1983 instead of 1893) and the publication dates returned by the best temporal classifier were actually, more or less, correct;
- 19 text fragments were written (all or nearly all) in a language other than Polish – Russian, German, French and Latin texts were usually misclassified as earlier ones (as it was more common to found such texts among Polish publications in the 19th and early 20th centuries), whereas English texts – as later (this is partly also a metadata problem, as texts in foreign languages *are* filtered out using language metadata);
- 32 texts were misclassified due to high level of OCR noise (even though some heuristics had been used to remove such texts);
- 11 text fragments were lists of items (words, surnames, football clubs, book titles), which made them difficult to classify;
- 11 texts referred to earlier periods (e.g. excerpts from historical journals);
- for 22 texts no clear reason for the discrepancy was identified, in some cases it seemed that old texts were too “clean” (novels from the 19th century manually re-typed, no OCR noise).

The conclusion is that (1) temporal classifiers could be used to detect defects and anomalies in the temporal metadata (running the classifier on a text with

known publication year and checking manually if the discrepancy is too high) and (2) there is still some room for improvement (e.g. by better filtering texts with high level of OCR noise) for RetroC data.

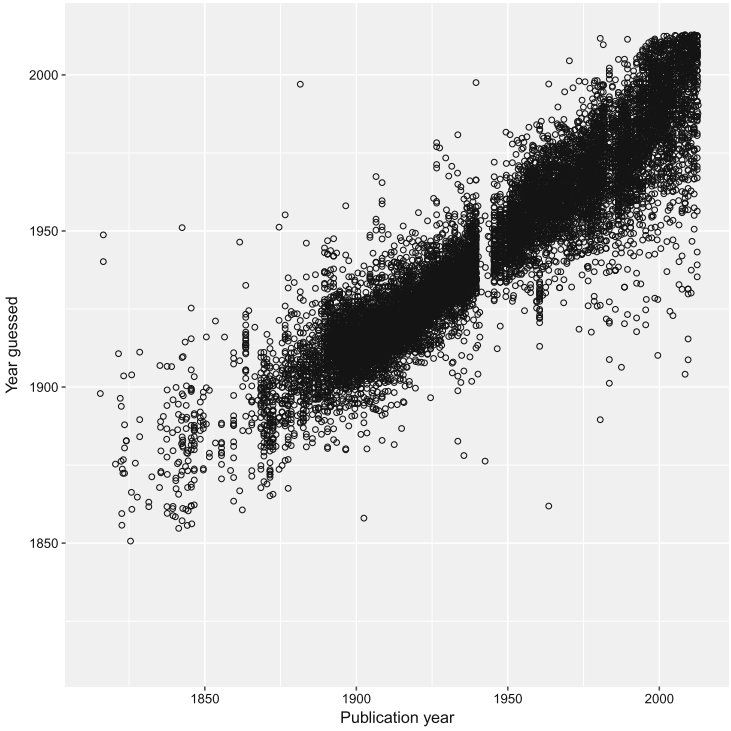

Fig. 3. Publication dates – years expected vs guessed (dev-1 set)

7 Conclusions and Further Work

We have presented two releases of RetroC, a Polish corpus for evaluating temporal classifiers, and reported initial results for certain methods. It has been shown that automatic dating can be treated as a regression problem, and that a standard machine learning tool (Vowpal Wabbit) can be used to obtain fairly good results.

For future work, we plan to implement all the advanced classification methods known in the literature for other languages, and compare and combine them with the regression methods. Also, we plan to use the temporal classifier trained on RetroC data for detecting old text fragments in large Web corpora.

References

1. Albert, P., Badin, F., Delorme, M., Devos, N., Papazoglou, S., Simard, J.: Décennie d'un article de journal par analyse statistique et lexicale. In: Proceedings of Traitement Automatique des Langues Naturelles (TALN), pp. 85–97 (2010)
2. Chambers, N.: Labeling documents with timestamps: learning from their time expressions. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers, vol. 1, pp. 98–106. Association for Computational Linguistics (2012)
3. Ciobanu, A.M., Dinu, L.P., Sulea, O.M., Dinu, A., Niculae, V.: Temporal text classification for Romanian novels set in the past. In: RANLP, pp. 136–140 (2013)
4. Dalli, A., Wilks, Y.: Automatic dating of documents and temporal text classification. In: Proceedings of the Workshop on Annotating and Reasoning about Time and Events, pp. 17–22. Association for Computational Linguistics (2006)
5. Garcia-Fernandez, A., Ligozat, A.-L., Dinarelli, M., Bernhard, D.: When was it written? Automatically determining publication dates. In: Grossi, R., Sebastiani, F., Silvestri, F. (eds.) SPIRE 2011. LNCS, vol. 7024, pp. 221–236. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-24583-1_22
6. Graliński, F., Jaworski, R., Borchmann, Ł., Wierzchoń, P.: Gonito.net - open platform for research competition, cooperation and reproducibility. In: Branco, A., Nicoletta, C., Khalid C. (eds.), Proceedings of the 4REAL Workshop: Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language, pp. 13–20 (2016)
7. Graliński, F.: Polish digital libraries as a text corpus. In: Proceedings of 6th Language and Technology Conference, Poznań, pp. 509–513 (2013)
8. Guo, S., Edelblute, T., Dai, B., Chen, M., Liu, X.: Toward enhanced metadata quality of large-scale digital libraries: estimating volume time range. In: iConference 2015 Proceedings (2015)
9. Jong, d.F., Rode, H., Hiemstra, D.: Temporal language models for the disclosure of historical text. Royal Netherlands Academy of Arts and Sciences (2005)
10. Kanhabua, N., Nørvåg, K.: Using temporal language models for document dating. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (eds.) ECML PKDD 2009. LNCS (LNAI), vol. 5782, pp. 738–741. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04174-7_53
11. Kumar, A., Baldridge, J., Lease, M., Ghosh, J.: Dating texts without explicit temporal cues, CoRR abs/1211.2290 (2012). http://arxiv.org/abs/1211.2290
12. Langford, J., Li, L., Zhang, T.: Sparse online learning via truncated gradient. In: Advances in Neural Information Processing Systems, pp. 905–912 (2009)
13. Wierzchoń, P.: Fotodokumentacja 3.0. Language, Communication. Information **4**, 63–80 (2009)