UDC 004; 004.93
DOI: 10.21122/2309-4923-2022-4-60-64

SLAVIN O. A. ^{1,2}

OPTIMIZING THE PERFORMANCE OF A SERVER-BASED CLASSIFICATION FOR A LARGE BUSINESS DOCUMENT FLOW

[1]Federal State Institution “Federal Research Center “Informatics and Management” of the Russian Academy of Sciences”
[2]LLC “Smart Engines Service”

The document categorization problem in the case of a large business document flow is considered. Textual and visual embeddings were employed for classification. Textual embeddings were extracted via OCR Tesseract. The Viola and Jones method was applied to generate visual embeddings. This paper describes the performance optimization technology for the implemented classification algorithm. Servers with Intel CPUs were used for the algorithm execution. For single-threaded implementation, high-level and low-level optimizations were performed. High-level optimization was based on the parametrization of the recognition algorithms and the employment of intermediate data. Low-level optimization was carried out via compiler tools allowing for an extended set of SIMD instructions. The implementation of parallelization with several multithreaded applications on multiple servers was also described. The proposed solution was tested using own test data sets of business documents. The proposed method can be applied in modern information systems to analyze the content of a large flow of digital document images.

Keywords: text analysis; document recognition; document classification; speedup.

Introduction

Document image recognition is a relevant problem since the number of documents printed on paper constantly grows. For example, the volume of incoming and outgoing document flows in large organizations can reach O(106) pages per day. The number of documents in the archive of a large bank may reach O(1010) pages. Despite the introduction of digital document flow, the number of paper documents and hard copies of digital documents is growing. A Russian publication reports that «according to experts and business estimates, the volume of paper document turnover is now growing in the country by 10% a year or more» [1]. However, an ever-growing paper workflow requires the storage of digitized document pages within an electronic archive. Image recognition of pages is an effective way to extract attributes.

For convenience, a document flow is segmented into individual documents (single-page or multi-page), and the latter are categorized. Works [2, 3] describe classification methods that are based on textual and visual embeddings. When categorizing heterogeneous documents, combination methods are useful, for example, using the BERT model [4]. Effective methods for textual information analysis include the application of probabilistic topic models. Such models are designed to determine the thematic structure of documents by representing each topic by a discrete distribution of word probabilities, and each document by a discrete probability distribution of topics [5]. When analyzing a document, a topic model assigns a document to a certain topic. Implicitly, it is assumed that the document contains a sufficient number of words to construct a discrete word probability distribution. The additive regularization of BigARTM topic models is described in [6]. In [6], a multi-objective optimization of the weighted criteria sum was employed. This was necessary to pre-define and ensure the stability of a topic model construction over a collection of documents. The following datasets derived from recognized images or articles can be used to train text-based methods:

– the NIST 2 database (NIST-SPDB2) which includes 5,590 binary images of 20 classes of tax forms with printed or handwritten content [7];

– the Tobacco-3482 dataset which inclues 3,482 images of 10 classes: report, memo, resume, scientific, letter, news, note, advertisement, form, and e-mail address [8].

Problem statement

We addressed the problem of business document classification for the banking industry. Business documents have several peculiarities. For example, the dictionary of acceptable static words is limited. The document template is editable for filling and printing. Both visual and textual features were used. Visual features were extracted from documents within the «Internal Russian passport» category. Textual features were extracted for the documents of the following classes: Agreement, Contract, Articles of association, User Questionnaire, Supplementary Agreement, Application, Invoice, Specimen Signature and Seal Card, and others. The images of the «Internal Russian passport» document pages were categorized using the Viola and Jones method [9]. Images of other documents were recognized using OCR Tesseract [10]. The classification of the recognized document pages was performed using the method based on special text point descriptors [11, 12]. We employed a hierarchical classifier with classes arranged into three levels, for example, «Agreement» - «Supplementary Agreement» - «Supplemental Account Bank Agreement». If the document did not match the category of «Supplemental Account Bank Agreement», then the categories «Contract» or «Additional agreement» were assigned instead. The classifier included 45 terminal classes. We should note that this number of classes significantly exceeds that of the public datasets [7, 8]. It was assumed that each class was introduced using one or more templates similar to one another. The similarity was evaluated by a set of keywords. A set of keywords could be a shingle [13] or a more general sequence of words with specified associations between word pairs. In addition to the classification accuracy set for each class, the most important requirement was speed. A multi-core computing system was required to process 300,000 pages in 8 hours (not including the time for scanning and transferring the results into the electronic archive).

Classification method

The classification method was based on determining the closeness between the array of recognized words within the digitized page and one or more document models. The models were constructed as follows. We employed special text points that corresponded to some words within the template. The special text point W, as defined in [11, 12], includes a few components of the word T(W) given within the document template, as well as B(W), a boundary consisting of the coordinates of a quadrilateral bounding the special text point image exactly or with a specified margin; the coordinates were scaled with the height and width of the normalized page. In addition, for each W we specified the parameters of the modified Levenshtein distance [11], L(W): mandatory character masks, the comparison cost of similar characters, and the threshold distance $\mathrm{d}_{\mathrm{LEV}}(\mathrm{W})$. The triplet of {T(W), B(W), L(W)} defines a special text point descriptor, and L(W) is optional. We employed the chains

$$\mathrm{C}_i = \{\mathrm{W}_1(\mathrm{C}_i), \mathrm{W}_2(\mathrm{C}_i), \dots \mathrm{W}_n(\mathrm{C}_i)\}.$$

as a set of descriptors for special text points $\mathrm{W}_1(\mathrm{C}_i)$, $\mathrm{W}_2(\mathrm{C}_i))$, … and several thresholds $d_1(\mathrm{W}_1(\mathrm{C}_i))$, $d_2(\mathrm{W}_1(\mathrm{C}_i))$ …. for classification. The thresholds allow for the comparison of relative positions of a point $\mathrm{W}_j(\mathrm{C}_i)$ and previous point $\mathrm{W}_{j\text{-}1}(\mathrm{C}_i)$, where $j>1$. The relation between terms is not necessary. Each of the specified thresholds $d_k(\mathrm{W}_j(\mathrm{C}_i))$ is a parameter in condition

$$\mathrm{r}^k_{\mathrm{T}}(\mathrm{Tm}_{i\text{-}1}(\mathrm{C}),\mathrm{Tm}_i(\mathrm{C}))<d_k(\mathrm{Tm}_i(\mathrm{C})),$$

where $\mathrm{r}^k_{\mathrm{T}}$ is some metric for evaluation of distance between two terms. Such metrics are, for example, the number of words between two terms or the distance calculated using the boundaries of the recognized words which correspond to the terms.

The detection of a special text point candidate was performed via the recognition, and a special text point $\mathrm{W}_j(\mathrm{C}_i)$ was compared to a detected candidate $\mathrm{W}^{\mathrm{REC}}$ via a modified Levenshtein distance [11]. When comparing words it is necessary to take into account the possibility of a significant number of recognition errors. The term of a special text point $\mathrm{W}_j(\mathrm{C}_i)$ is mapped to multiple candidates among words within the recognized document: the word is considered a candidate if the distance between the word and $\mathrm{W}_j(\mathrm{C}_i)$ is less than $d_{\mathrm{LEV}}(\mathrm{W}_j(\mathrm{C}_i))$. Candidates $\mathrm{W}^{\mathrm{REC}}_{\mathrm{q}}$ to be associated with $\mathrm{W}_j(\mathrm{C}_i)$ are verified in terms of consistency based on terms sequence in chain C_i. Namely, among all candidates, we selected the ones that allowed for a minimum value of

$$\mathrm{d}(\mathrm{C}_i) = \max(\mathrm{r}_{\mathrm{LEV}}(\mathrm{T}(\mathrm{W}_j(\mathrm{C}_i)), \mathrm{W}^{\mathrm{REC}}_{\mathrm{q}}))\ \text{à}\ \min.$$

When calculating the chain match score, all terms should be linked, and if the match is ideal, the penalty $\mathrm{d}(\mathrm{C}_i)$ is zero.

The page type classification was based on predefined page descriptors. For each descriptor $\mathrm{D}(\mathrm{P}_k)=\{C_{11}(\mathrm{P}_k), C_{12}(\mathrm{P}_k),\dots\}$ of class P_k, the link to the recognized page was established, and match scores for each document class were calculated. The evaluation of match to the type k was chosen as the minimum among match scores of chains $C_{k1}(\mathrm{P}_k)$, $C_{k2}(\mathrm{P}_k)$,…. After ranking the scores, we selected either the minimum which corresponded to the closest class or several lowest scores in the case of multi-class classification.

System structure

The described algorithm was implemented in the developed system, which in addition to the above-mentioned Viola and Jones algorithm included the OCR Tesseract version. The system includes the following components:

– image input;
– visual embeddings extraction;
– textual embeddings extraction;
– classification via visual embeddings;
– classification via textual embeddings;
– combining classification results;
– transfer of results to the electronic archive.

The system was implemented with Visual Studio Community 2017 and Intel C++ Compiler. The testing was performed on an Intel® Core(TM) i7-4790 CPU 3.60 GHz, 16.0 GB, Windows 7 prof 64-bit. A custom dataset was created. The test dataset consisted of 300 pages of business documents of different classes. The recognition of all 300 pages with the original parameters took t=4928,53 seconds, with the average recognition time per page t_{cp} of 16,43 seconds, the minimum recognition time per page t_{min} was 0,99 seconds, and maximum t_{max} was 143,21 seconds.

The profiling was performed using the built-in Visual Studio Performance Profiler. The majority of time was spent on OCR Tesseract on the test dataset and accounted for more than 50% of the total runtime. About 10% of the total time was spent on the bilateral filter. The Viola and Jones methods and classification accounted for less than 1% of the total time. Several types of performance optimization were applied to speed up the system: high-level optimization, low-level optimization, and parallel programming.

High-level and low-level performance optimization

With a fixed algorithm, high-level optimization is based on the following techniques:

– parameter selection;
– memoization (the use of intermediate data) [14];
– lookup tables [15];
– application of approximate calculations and other methods.

The parameter selection and data representation for the OCR Tesseract component were employed for the optimization. The limiting the recognition area in each page allowed for a significant performance improvement. To implement this restriction, in training set of 5000 pages, we selected a region which contained all key words necessary for classification of all documents. With a margin, the following recognition area limit was chosen: 70% in page height and 90% in width. This limitation allowed for an improvement of Tesseract runtime, which led to overall progress: t=2649,57 s., t_{cp}=8,83 s., t_{min}=0,83 s., t_{max}=77,64 s. . The page images binarization prior to recognition also proved to be effective. The initial goal of binarization was the recognition accuracy gain since it removes the complex background and eases morphological operations. However, we observed runtime improvement because of binarization: t=2293,20 s., t_{cp}=7,64 s., t_{min}=0,98 s., t_{max} = 59,83 s. . These values were measured in the case of recognition in limited area. The measurements account for the performance of both OCR Tesseract recognition and binarization operation.

Low-level optimization was performed by selecting an extended set of CPU instructions. The experiments were carried out with Visual Studio 13 compiler. When compiling with Intel C++ Compiler XE 15.0, we achieved a significant speedup compared to the previous version: t=2156,00 s., t_{cp}=7,19 s., t_{min}=0,76 s., t_{max}=59,66 s. . The compilation was optimized for AVX2 architecture. Intel compiler allowed to optimize OCR Tesseract performance as well as bilateral filter performance. Table 1 illustrates the described experiments. Due to the adopted optimization methods, the total time of recognition of all pages was reduced by more than half compared to the initial implementation. Hence, with the optimized approach, during 8 hours approximately 8*60*60/7,19 ≈ 4005 pages can be recognized.

Performance optimization via parallel computing

Parallelization of recognition was implemented using standalone components, hence cluster with several multi-core nodes could be employed. Numerous applications processed pages in multiple threads, and the input flow of pages was assigned to these applications using a load-balancing manager. The system was implemented via micro-tasking, i.e., parallelism was applied without using explicit control. Experiments were conducted to process 300 test pages on a single node with the characteristics described above. Table 2 illustrates the results.

Table 1. Computational tests of a single thread mode

№	Optimization approach	Total time t (s.)	Average time t_{cp} (s.)
1	Original implementation	4928,53	16,43
2	Version 2 with limited recognition area	2649,57	8,83
3	Version 3 with preliminary binarization	2293,20	7,64
4	Version 4 with optimization for AVX2	2156,00	7,19

Table 2. Computational tests of a multi-threaded mode

Number of processes	Number of threads within a process	Total time t (s.)	Average time t_{cp} (s.)
1	1	2361	7,87
1	2	1318	4,39
1	3	829	2,76
1	4	649	2,16
2	1	1298	4,33
2	2	789	2,63
2	4	485	1,62
2	8	500	1,67

The results suggest that the best average time (1.67 seconds per page) is achieved by running 2 applications, with 4 threads for page processing in each. Such configuration allows for 8*60*60/1,67 ≈ 17245 pages to be recognized in 8 hours.

In other words, we have increased the page processing speed by about 10 times. The processing of the desired 300000 pages in 8 hours would require a cluster of 4 similar nodes.

Conclusion

The considered optimization approaches for document categorization problem included:

– setting the number of servers;
– choosing classification algorithm;
– parameterization of the OCR component;
– selecting and configuring compilation tools.

Parallelization is the most effective tool for increasing processing speed. However, the other optimization approaches we considered in single-threaded mode reduced runtime by half. In other words, optimization in a single-threaded mode reduced the number of nodes almost by half.

In the case of real document flows, the number of nodes was reduced by about 40%. This reduction in the number of nodes for the system implementation simplifies the creation and maintenance of technical support as well as significantly decreases the power consumption of the entire system.

The described approaches to software performance optimization can be applied not only to business document processing systems but also to large image processing systems.

ЛИТЕРАТУРА

1. **Башкатова, А.** Цифровая экономика плодит все больше бумаг: Россияне не скоро перестанут носить в организации справки // Независимая Газета. – 2019 – 14 ноя. [Электронный ресурс] – Режим доступа: https://www.ng.ru/economics/2019-11-14/4_7727_paper.html, – Загл. с экрана – Яз. рус. Дата доступа – 08.11.2022.

2. **Liu, L., Wang, Z., Qiu, T., Chen, Q., Lu, Y., Suen, C.Y.** Document image classification: Progress over two decades, Neurocomputing 2021, 453: 223-240.

3. **Byun, Y., Lee, Y.** Form classification using DP matching. ACM Symposium on Applied Computing 2000; 1: 1–4.

4. **Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.** BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2019. [Электронный ресурс] – Режим доступа: https://arxiv.org/abs/1810.04805/, – Загл. с экрана – Яз. англ. Дата доступа – 08.11.2022.

5. **Rubin, T.N., Chambers, A., Smyth, P., Steyvers, M.** Statistical topic models for multi-label document classification. Machine Learning – 2011, Vol. 88, № 1, 157–208. https://doi.org/10.1007/s10994-011-5272-5.

6. **Vorontsov, K.V., Potapenko, A.A.** Tutorial on probabilistic topic modeling: Additive regularization for stochastic matrix factorization. Communications in Computer and Information Science – 2014, Vol. 436, pp. 29-46. https://doi.org/10.1007/978-3-319-12580-0_3.

7. NIST Special Database 2 [Электронный ресурс] – Режим доступа: https://www.nist.gov/srd/nist-special-database-2/, – Загл. с экрана – Яз. англ. Дата доступа – 08.11.2022.

8. Tobacco-3482 [Электронный ресурс] – Режим доступа: https://www.kaggle.com/patrickaudriaz/tobacco3482jpg/, – Загл. с экрана – Яз. англ. Дата доступа – 08.11.2022.

9. OCR Tesseract [Электронный ресурс] – Режим доступа: https://github.com/tesseract-ocr/tesseract/, – Загл. с экрана – Яз. англ. Дата доступа – 08.11.2022.

10. **Tereshin, A.A., Usilin, S.A., Arlazarov, V.V.** Performance Improvement of Multi-class Detection Using Greedy Algorithm for Viola-Jones Cascade Selection. Proceedings Volume 10696, Tenth International Conference on Machine Vision (ICMV 2017); 106960D (2018). https://doi.org/10.1117/12.2310101

11. **Slavin, O.A., Farsobina, V., Myshev, A.V.** Analyzing the content of business documents recognized with a large number of errors using modified Levenshtein distance. Cyber-Physical Systems: Intelligent Models and Algorithms. – 2022, Springer Nature Switzerland AG., Vol. 417, pp. 267 – 279. https://doi.org/10.1007/978-3-030-95116-0

12. **Slavin, O.A.** Using Special Text Points in the Recognition of Documents. Studies in Systems, Decision and Control. – 2020, Springer Nature Switzerland AG., Vol 259. pp. 43–53. https://doi.org/10.1007/978-3-030-32579-4_4

13. **Konaka, F., Miura, T.** Semantic similarity for sequenced shingles, – 2015 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), pp. 12-17. https://doi.org/10.1109/PACRIM.2015.7334801.

14. **Acar, U.A., Blelloch, G.E., Harper, R.** Selective memorization. ACM SIGPLAN Notices, – 2003, Vol. 38, Issue 1, pp 14–25. https://doi.org/10.1145/640128.604133

15. **Tatarowicz, A.L., Curino, C., Jones, E. P. C. and Madden, S**. Lookup Tables: Fine-Grained Partitioning for Distributed Databases. – 2012 IEEE 28th International Conference on Data Engineering, pp. 102-113. https://doi.org/10.1109/ICDE.2012.26

REFERENCES

1. **Bashkatova, A.** Cifrovaya ekonomika plodit vse bol'she bumag: Rossiyane ne skoro perestanut nosit' v organizacii spravki // Nezavisimaya Gazeta – 2019 – 14 nov. . [Электронный ресурс] – Режим доступа: https://www.ng.ru/economics/2019-11-14/4_7727_paper.html, – Загл. с экрана – Яз. рус. Дата доступа – 08.11.2022.

2. **Liu, L., Wang, Z., Qiu, T., Chen, Q., Lu, Y., Suen, C.Y.** Document image classification: Progress over two decades, Neurocomputing 2021, 453: 223-240.

3. **Byun, Y., Lee, Y.** Form classification using DP matching. ACM Symposium on Applied Computing 2000; 1: 1–4.

4. **Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.** BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2019. [Электронный ресурс] – Режим доступа: https://arxiv.org/abs/1810.04805/, – Загл. с экрана – Яз. англ. Дата доступа – 08.11.2022.

5. **Rubin, T.N., Chambers, A., Smyth, P., Steyvers, M.** Statistical topic models for multi-label document classification. Machine Learning – 2011, Vol. 88, № 1, 157–208. https://doi.org/10.1007/s10994-011-5272-5.

6. **Vorontsov, K.V., Potapenko, A.A.** Tutorial on probabilistic topic modeling: Additive regularization for stochastic matrix factorization. Communications in Computer and Information Science – 2014, Vol. 436, pp. 29-46. https://doi.org/10.1007/978-3-319-12580-0_3.

7. NIST Special Database 2 [Электронный ресурс] – Режим доступа: https://www.nist.gov/srd/nist-special-database-2/, – Загл. с экрана – Яз. англ. Дата доступа – 08.11.2022.

8. Tobacco-3482 [Электронный ресурс] – Режим доступа: https://www.kaggle.com/patrickaudriaz/tobacco3482jpg/, – Загл. с экрана – Яз. англ. Дата доступа – 08.11.2022.

9. OCR Tesseract [Электронный ресурс] – Режим доступа: https://github.com/tesseract-ocr/tesseract/, – Загл. с экрана – Яз. англ. Дата доступа – 08.11.2022.

10. **Tereshin, A.A., Usilin, S.A., Arlazarov, V.V.** Performance Improvement of Multi-class Detection Using Greedy Algorithm for Viola-Jones Cascade Selection. Proceedings Volume 10696, Tenth International Conference on Machine Vision (ICMV 2017); 106960D (2018). https://doi.org/10.1117/12.2310101

11. **Slavin, O.A., Farsobina, V., Myshev, A.V.** Analyzing the content of business documents recognized with a large number of errors using modified Levenshtein distance. Cyber-Physical Systems: Intelligent Models and Algorithms. – 2022, Springer Nature Switzerland AG., Vol. 417, pp. 267 – 279. https://doi.org/10.1007/978-3-030-95116-0

12. **Slavin, O.A.** Using Special Text Points in the Recognition of Documents. Studies in Systems, Decision and Control. – 2020, Springer Nature Switzerland AG., Vol 259. pp. 43–53. https://doi.org/10.1007/978-3-030-32579-4_4

13. **Konaka, F., Miura, T.** Semantic similarity for sequenced shingles, – 2015 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), pp. 12-17. https://doi.org/10.1109/PACRIM.2015.7334801.

14. **Acar, U.A., Blelloch, G.E., Harper, R.** Selective memorization. ACM SIGPLAN Notices, – 2003, Vol. 38, Issue 1, pp 14–25. https://doi.org/10.1145/640128.604133

15. **Tatarowicz, A.L., Curino, C., Jones, E. P. C. and Madden, S.** Lookup Tables: Fine-Grained Partitioning for Distributed Databases. – 2012 IEEE 28th International Conference on Data Engineering, pp. 102-113. https://doi.org/10.1109/ICDE.2012.26

Славин О. А., доктор технических наук, Главный научный сотрудник Федерального государственного учреждения “Федеральный исследовательский центр “Информатика и управление” Российской академии наук”, г. Москва, Россия. Количество печатных работ: 2 монографии, 137 статей, 3 учебных пособия, 64 патента. Область научных интересов: распознавание образов, информационные системы.

Slavin O. A., Doctor of Technical Sciences, Chief Researcher of the Federal State Institution “Federal Research Center “Informatics and Management” of the Russian Academy of Sciences”, Moscow, Russia. Number of publications: 2 monographs, 137 articles, 3 textbooks, 64 patents. Research interests: pattern recognition, information systems.

E-mail: oslavin@isa.ru