ACTA TECHNICA JAURINENSIS
Vol. 15, No. 1, pp. 15–21, 2022
10.14513/actatechjaur.00628

Comparison of data augmentation methods for legal document classification

Gergely M. Csányi[1,2,*], **Tamás Orosz**[1,3]

[1]Montana Knowledge Management Ltd.,
Hársalja Str. 32., H-1029, Budapest, Hungary
[2]Budapest University of Technology and Economics, Department of Electric Power Engineering
J. Egry Str. 18., H-1111, Budapest, Hungary
[3]University of West Bohemia,
Univerzitni 26, 306 14 Pilsen, Czech Republic
[*]e-mail: csanyi.gergely@montana.hu

Submitted: 20/05/2021; *Accepted: 09/07/2021;* *Published online: 16/07/2021*

Abstract: Sorting out the legal documents by their subject matter is an essential and time-consuming task due to the large amount of data. Many machine learning-based text categorization methods exist, which can resolve this problem. However, these algorithms can not perform well if they do not have enough training data for every category. Text augmentation can resolve this problem. Data augmentation is a widely used technique in machine learning applications, especially in computer vision. Textual data has different characteristics than images, so different solutions must be applied when the need for data augmentation arises. However, the type and different characteristics of the textual data or the task itself may reduce the number of methods that could be applied in a certain scenario. This paper focuses on text augmentation methods that could be applied to legal documents when classifying them into specific groups of subject matters.

Keywords: *text augmentation; augmenting legal cases; legal document classification; data augmentation*

I. INTRODUCTION

The digitalization of the judicial systems needs to process, categorize and pseudonymize many sensitive legal documents before they are published online [1–3]. This paper focuses only on the automatic categorization of legal documents. Text classification or text categorization is an essential branch in Natural Language Processing (NLP) [4–6], where the role of the different machine learning-based automatic text classification procedures is to automatically assign predefined labels for different documents (**Fig. 1**). For instance, the different legal documents can be sorted into different classes by their subject matter, such as theft, embezzlement, fraud, etc. [7].

Nevertheless, the classification of legal documents belongs to the class of multi-labeled categorization, which means that a legal document can belong to more than one legal category. This is a recent and relevant topic of research [7, 8]. There are different mathematical methods proposed to handle this task. Some of them use strategies – such as label powerset or binary relevance transformations [7, 9] – to convert back this selection into a single label classification task, while others extend the numerical methodologies to handle these kinds of tasks. Some examples of machine learning techniques from this latter group are: multi-label k-nearest neighbours, multi-label Naive Bayes, or multi-label AdaBoost [10].

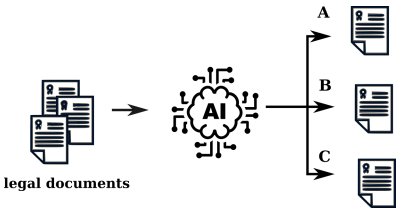

Figure 1. General process of legal document categorization

Fig. 1. shows the general process of legal document categorization, where the A, B, and C categories refer to specific subject matters, e.g., a crime of theft, the invalidity of a contract, causing a traffic accident, etc. Subject matters are generally highly imbalanced, which is illustrated on the left side of **Fig. 2**. Here, the diagram represents the number of documents (training samples) belonging to different categories. Generally, machine learning models tend to perform better when having approximately the same amount of training data for each class. However, as **Fig. 2**. shows, in practice, usually this is not the case. In this case, when the dataset is highly imbalanced, or the minority class has only few members, data augmentation techniques can help balancing the dataset.

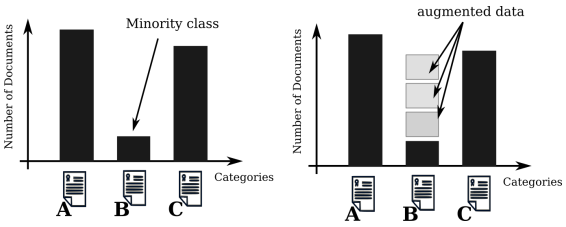

Figure 2. Augmenting imbalanced datasets

Text augmentation is a useful technique, which can automatically generate training samples. Hence, it can help to improve the performance of machine learning applications in this case. Data augmentation [11] is a well-known technique for training more robust machine learning models that have been successfully applied in the field of computer vision [12]. Generally, augmenting images is done by mirroring, rotating, cropping, scaling, flipping, etc., but these methods often cannot be applied on texts [13]. This is because the order of words is important in texts, and changing it could end up in sentences with completely different meanings. However, the need for augmenting data is not present only in the field of computer vision since imbalanced or small datasets occur in all types of machine learning applications. In case of imbalanced or small datasets, it is likely that the machine learning models overfit or do not fit well on training data. Hence augmentation can improve the robustness and performance of the models. Recently, many studies have been published to tackle the problem of data augmentation in the NLP field [14–16]. Some approaches depend more on the language or language models [14, 17], while others are (almost) independent [15, 18]. However, when applying text augmentation, one must pay attention to the characteristics of the text and the problem to be solved, since both of these may affect what type of augmentation techniques can be applied.

The main contribution of this work is providing a comprehensive survey on the possible text augmentation techniques in the case of legal document classification. Legal cases are relatively long (around 1,000-2,000 words), semi-structured texts. Therefore, certain parts of a case can be found in every document, generally well-spelled but not faultless documents. It is important to point out that matter of facts often consist of legal terms. Hence, certain words cannot be replaced or removed by any means. For example, theft, embezzlement, and fraud might be considered relatively close synonyms in general language, but they refer to three completely different types of crime in the legal context. These specialties of legal texts require special attention during the selection of the appropriate text classification methodology.

The paper is organized as follows: Section II. shows similar studies, Section III. presents a brief overview of typical text augmentation techniques and in Section IV a discussion of the useful techniques can be found.

II. SIMILAR STUDIES

Yan et al. presented a solution for augmenting legal documents [13]. Their work aimed to tackle a crime prediction problem, predicting the accusations of a case when the matter of fact is given. They applied three different techniques on sentence level:

- randomly scramble sentences in the sample,
- randomly delete sentences in the sample,
- randomly insert the sentences with the same label in other samples

The neural-based classifiers gained a lot of performance (11 %, F1-score) by the above-mentioned augmentation techniques when the training data was relatively small (10 thousand documents) but significantly

less when 10-15 times more training data was available.

Another solution for augmenting legal documents is the TauJud framework, designed for augmenting Chinese legal cases [19]. The solution performs a two-step augmentation process, namely the Universal Augmentation and Judicial Augmentation steps. The former includes stop word deletion, back translation (RTT), and clipping, while the latter includes counterfactual data augmentation [20], and synonym replacement. It is possible to set what kind of augmentation steps have to be done and to choose multiple from these steps simultaneously. However, protecting words from augmentation is missing from the framework's repertoire.

III. TYPICAL AUGMENTATION TECHNIQUES

1. Easy Data Augmentation

Easy Data Augmentation (EDA) has gained interest after defining four simple methods for augmenting textual data and showing the efficiency of this approach on five different classification tasks [15]. The methods were the following:

- Synonym Replacement (SR) – Select n pieces of words from the sentence that are not stopwords. Replace each of these words with one of its randomly chosen synonyms.
- Random Insertion (RI) – Pick a random word in a sentence. Add the randomly selected synonym of this word to a random position of the sentence.
- Random Swap (RS) – Swap position of randomly chosen word pairs n times.
- Random Deletion (RD) – Randomly delete words from the document with probability p.

The latter two methods are completely language-independent, while the first two require a language-dependent WordNet database [21]. Nevertheless, these methods do not require a pretrained language model like GPT-2 [22] or word embeddings [23–25]. The power of these techniques lies in the simplicity of the solution, while the authors reported significant gain (around 3% on average) by using these techniques on text classification tasks.

2. Round-trip translation

Round-trip Translation (RTT) is an augmentation technique that harnesses the fact that translating a text to a random language and translating back to the original one in the majority of the cases (depending on the length of the text) results in a slightly different text, yet preserving the original meaning [26–30]. The technique is also known as recursive, back-and-forth, and bi-directional translation.

3. Semantic similarity augmentation

By means of distributed word representations, semantically similar words can be identified [23]. Hence, textual data can be easily augmented by replacing a fraction of the original text with the nearest neighbours of the chosen words. This approach requires either pre-trained word embedding models for the language in question or enough data from the target application to build the embedding model [16]. Thus, this approach does not require access to a dictionary or thesaurus for a language to find synonyms [16]. This can be advantageous for languages where such resources are more difficult to obtain, but there is enough unsupervised text data to be able to build the embedding models [16]. As word embedding models e.g. Word2Vec [23], GloVe [25], FastText [24] could be used, but these models may not handle words that are homonyms (multiple-meaning words) properly. By transformer-based language models such as BERT [31], the representation for words can be obtained in a context-dependent manner, providing a better solution for homonyms but with significantly more effort.

4. Text generation

Another approach for text augmentation is to use pre-trained language models to generate random texts. While this could be made by an LSTM-based encoder-decoder network, this type of solution would require a significant amount of training data and would generate grammatically incorrect sentences [32]. Another, more sophisticated approaches would be using generative adversarial networks (GANs) [33], variational autoencoders (VAE) [34], or paraphrasing [35].

GPT [36], GPT-2 [22] models are capable of producing grammatically correct, high-quality texts even when fine-tuned on small training data [14]. Nevertheless, the lack of ability to preserve or protect certain words from the original text cannot be assured by this method either.

5. Synthetic Minority Oversampling Technique

This technique is somewhat different from the already mentioned ones. Synthetic Minority Oversampling Technique (SMOTE) cannot be applied of the original text, but on its representations [37].

The basic idea of this method is, that by assuming that in the representation space the points between samples from minority class also belong to the minority class. Hence, creating synthetic samples is done by selecting random points from the lines between original data points as **Fig. 3**. shows. These synthetic data points serve as extra data for training a classifier.

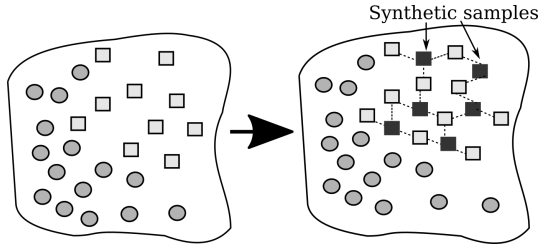

Figure 3. Applying Synthetic Minority Oversampling Technique on an imbalanced dataset to generate new training samples and create a more balanced training dataset.

IV. EVALUATION OF AUGMENTATION TECHNIQUES FOR SUBJECT MATTER CLASSIFICATION

Due to the special characteristics of legal cases, namely the length of the documents and the presence of legal terms, some methods are mentioned in Section III. cannot be applied. In the case of Easy Data Augmentation solutions **Random Deletion** could be used but with a restriction of not deleting specific words, called protected words. These are the words that refer to a certain subject matter, but in other applications, the need for having a list of protected words may also arise. **Random Swap** could be used without restrictions. However, one may have to keep in mind that in certain cases, this would not make much sense, e.g., when using word-order-independent document representation forms like tf-idf vectorization, or calculating document vectors by averaging word embeddings [38, 39]. Both **Synonym Insertion** and **Synonym Replacement** can be applied, but only with caution since adding synonyms of legal terms may end up in changing labels that are not wanted during augmenting data. One solution for this would be the same as mentioned before: by defining a list of protected words, these words cannot be used during random selection, ensuring that the most important words will not change. WordNet [21] provides different domains that can be used to restrict the scope of words for searching synonyms. For instance, synonyms for time-related data, colors, geographical locations, etc., will not change the label of the augmented legal case, yet provide another case that is somewhat different from the original, in other words augmenting the original case. It is important to point out that the quality of methods based on synonyms is highly dependent on whether the words in a text are stemmed or lemmatized or left as they are. This is true especially for highly inflected languages (e.g. German, Spanish, Russian, Hungarian), since WordNet databases contain lemmas only, so the number of words that can be selected for synonym modification (replacement or insertion), is usually significantly higher when the input text has been lemmatized beforehand.

The method of **Round-trip Translation** cannot be used as an augmentation method since there is no control over the words in the document. One cannot define protected words. As mentioned before, this would be important, since this way, after the double translation, the augmented document would not have the same label as the original one. The same problem arises with **text generation** solutions that can be useful in many applications, but there are no guarantees that the legal terms will be generated properly or will be kept intact.

Semantic similarity augmentation method is also an option that could be used during the augmentation of legal cases to classify the subject matters. The principle of using a list of protected words is also important here, since the most similar words to a given word depend on factors, like what kind of text the model was trained on etc., so keeping certain words intact otherwise would be practically impossible. These embedding models capture semantic similarities by assuming that words occurring in the same or similar contexts have similar meanings. An advantage of this method is that nearest neighbours of a certain word are not only synonyms, but words that occur in similar contexts, so a wider, more general augmentation is possible with these methods. However, it is important to emphasize that the nearest neighbours are highly dependent on the corpus on which the data was trained on and the size of the embedding vector. Another drawback of this method is the question of out-of-vocabulary (OOV) words. If the given document contains many OOV words, the quality of the augmented document may be affected. There are so-

lutions to handle this problem. One of them is using FastText embeddings that can map any word into the embedding space, even if they are OOV words, by harnessing the power of subword information. Moreover, this type of embedding proved to perform well on highly inflected languages [24]. However, using FastText embeddings can be a double-edged sword since applying it on OOV words can result in useful embeddings and completely useless ones. Hence, careful analysis has to be made before deciding for or against using FastText to deal with OOV words.

V. CONCLUSIONS

Data augmentation is a very important technique that has proven its effectiveness in a high variety of machine learning tasks. This paper provided a brief overview of textual data augmentation techniques, putting more emphasis on classification of legal cases and comparing the available solutions. Current solutions can be applied on short texts e.g. on sentence level and may be ineffective on longer texts in terms of run time. Legal texts contain a lot of legal terms that have to be handled with caution when augmenting data since synonyms of these terms may refer to a completely different legal term, adding a significant bias to the augmented data. It can be stated that none of the mentioned solutions deal with this issue, however, some of them can be extended to solve it but not all of them. Generally, the more augmentation steps are applied, the better the expected quality of the augmented dataset will be reached, but there is no golden rule, which can be applied for every single problem.

REFERENCES

[1] E. Comission, "Digitalisation of justice." "https://ec.europa.eu/info/policies/justice-and-fundamental-rights/digitalisation-justice_en", "(accessed: 01.02.2021)".

[2] E. Hyvönen, M. TAMPER, E. IKKALA, S. SARSA, A. OKSANEN, J. TUOMINEN, and A. HIETANEN, "Lawsampo: a semantic portal on a linked open data service for finnish legislation and case law," in *Proceedings of ESWC*, 2020.

[3] "Pseudonymization according to the gdpr [definitions and examples]." "https://dataprivacymanager.net/pseudonymization-according-to-the-gdpr/", "(accessed: 02.15.2021)".

[4] A. Dhar, H. Mukherjee, N. S. Dash, and K. Roy, "Text categorization: past and present," *Artificial Intelligence Review*, vol. 54, no. 4, pp. 3007–3054, 2021.

[5] P. Jackson and I. Moulinier, *Natural language processing for online applications: Text retrieval, extraction and categorization*, vol. 5. John Benjamins Publishing, 2007.

[6] P. J. Hayes, L. E. Knecht, and M. J. Cellio, "A news story categorization system," in *Second Conference on Applied Natural Language Processing*, pp. 9–17, 1988.

[7] R. B. Pereira, A. Plastino, B. Zadrozny, and L. H. Merschmann, "Categorizing feature selection

ACKNOWLEDGEMENT

This project (no. 2020-1.1.2-PIACI-KFI-2020-00049) has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the 2020-1.1.2-PIACI KFI funding scheme.

AUTHOR CONTRIBUTIONS

G. M. Csányi: Conceptualization, Software, Writing, Review, Editing.

T. Orosz: Supervision, Software, Writing, Review and Editing.

DISCLOSURE STATEMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

ORCID

G. M. Csányi http://orcid.org/0000-0001-8475-5969

T. Orosz http://orcid.org 0000-0002-8743-3989

methods for multi-label classification," *Artificial Intelligence Review*, vol. 49, no. 1, pp. 57–78, 2018.

[8] M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[9] G. E. Tsekouras, C. Anagnostopoulos, D. Gavalas, and E. Dafhi, "Classification of web documents using fuzzy logic categorical data clustering," in *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pp. 93–100, Springer, 2007.

[10] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," in *Data mining and knowledge discovery handbook*, pp. 667–685, Springer, 2009.

[11] S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: when to warp?," in *2016 international conference on digital image computing: techniques and applications (DICTA)*, pp. 1–6, IEEE, 2016.

[12] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation policies from data," *arXiv preprint arXiv:1805.09501*, 2018.

[13] G. Yan, Y. Li, S. Zhang, and Z. Chen, "Data augmentation for deep learning of judgment documents," in *International Conference on Intelligent Science and Big Data Engineering*, pp. 232–242, Springer, 2019.

[14] A. Anaby-Tavor, B. Carmeli, E. Goldbraich, A. Kantor, G. Kour, S. Shlomov, N. Tepper, and N. Zwerdling, "Do not have enough data? deep learning to the rescue!," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 7383–7390, 2020.

[15] J. Wei and K. Zou, "Eda: Easy data augmentation techniques for boosting performance on text classification tasks," *arXiv preprint arXiv:1901.11196*, 2019.

[16] V. Marivate and T. Sefara, "Improving short text classification through global augmentation methods," in *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pp. 385–399, Springer, 2020.

[17] Y. Li, T. Cohn, and T. Baldwin, "Robust training under linguistic adversity," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 21–27, 2017.

[18] S. Kobayashi, "Contextual augmentation: Data augmentation by words with paradigmatic relations," *arXiv preprint arXiv:1805.06201*, 2018.

[19] Z. Guo, J. Liu, T. He, Z. Li, and P. Zhangzhu, "Taujud: test augmentation of machine learning in judicial documents," in *Proceedings of the 29th ACM SIGSOFT International Symposium on Software Testing and Analysis*, pp. 549–552, 2020.

[20] R. Zmigrod, S. J. Mielke, H. Wallach, and R. Cotterell, "Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (Florence, Italy), pp. 1651–1661, Association for Computational Linguistics, July 2019.

[21] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[22] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[24] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *arXiv preprint arXiv:1607.04606*, 2016.

[25] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[26] A. Gerasimov, D. Nogueira, K. Semolini, S. Firoozkoohi, R. A. Rivera, T. a Patent, K. K. Zerling, L. García-Santiago, M.-D. Olvera-Lobo, M. Aiken, *et al.*, "The efficacy of round-trip translation for mt evaluation,"

[27] S. T. Aroyehun and A. Gelbukh, "Aggression detection in social media: Using deep neural networks, data augmentation, and pseudo labeling," in *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pp. 90–97, 2018.

[28] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," *arXiv preprint arXiv:1511.06709*, 2015.

[29] M. Fadaee, A. Bisazza, and C. Monz, "Data augmentation for low-resource neural machine translation," *arXiv preprint arXiv:1705.00440*, 2017.

[30] A. W. Yu, D. Dohan, M.-T. Luong, R. Zhao, K. Chen, M. Norouzi, and Q. V. Le, "Qanet: Combining local convolution with global self-attention for reading comprehension," *arXiv preprint arXiv:1804.09541*, 2018.

[31] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[32] S. Sharifirad, B. Jafarpour, and S. Matwin, "Boosting text classification performance on sexist tweets by text augmentation and text generation using a combination of knowledge graphs," in *Proceedings of the 2nd workshop on abusive language online (ALW2)*, pp. 107–114, 2018.

[33] F. H. K. d. S. Tanaka and C. Aranha, "Data augmentation using gans," *arXiv preprint arXiv:1904.09135*, 2019.

[34] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[35] A. Kumar, S. Bhattamishra, M. Bhandari, and P. Talukdar, "Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 3609–3619, 2019.

[36] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[37] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[38] W. Zhang, T. Yoshida, and X. Tang, "A comparative study of tf* idf, lsi and multi-words for text classification," *Expert Systems with Applications*, vol. 38, no. 3, pp. 2758–2765, 2011.

[39] D. Mekala, V. Gupta, B. Paranjape, and H. Karnick, "Scdv: Sparse composite document vectors using soft clustering over distributional representations," *arXiv preprint arXiv:1612.06778*, 2016.