RedFame

Applied Economics and Finance
Vol. 8, No. 5; September 2021
ISSN 2332-7294 E-ISSN 2332-7308
Published by Redfame Publishing
URL: http://aef.redfame.com

Transformer-Based Patent Novelty Search by Training Claims to Their Own Description

Michael Freunek[1] & André Bodmer[1,2]

[1] Mathematical Institute, University of Bern, Bern, Switzerland

[2] Department of Economics, University of Bern, Bern, Switzerland

Correspondence: André Bodmer, Department of Economics, University of Bern, Bern, Switzerland.

Received: March 5, 2021 Accepted: April 14, 2021 Available online: October 13, 2021

doi:10.11114/aef.v8i5.5182 URL: https://doi.org/10.11114/aef.v8i5.5182

Abstract

In this paper we present a method to concatenate patent claims to their own description. By applying this method, bidirectional encoder representations from transformers (BERT) train suitable descriptions for claims. Such a trained BERT could be able to identify novelty relevant descriptions for patents. In addition, we introduce a new scoring scheme: relevance score or novelty score to interprete the output of BERT. We test the method on patent applications by training BERT on the first claims of patents and corresponding descriptions. The output is processed according to the relevance score and the results compared with the cited X documents in the search reports. The test shows that BERT score some of the cited X documents as highly relevant.

Keywords: patent, novelty search, natural language processing, transformer

1. Introduction

Patent searches fulfill important tasks in the patent system. Patent applications are usually checked for novelty and inventive step in order to meet the requirements of the corresponding patent law. In the event of disputes between patent owners and alleged patent infringers, patents that have already been granted are usually checked again by the alleged patent infringer for legal validity. Relevant claims are searched again as the examiner may have disregarded relevant information when the patent has been granted. Based on classic Boolean methods, patent searches tend to be more and more extensive and complex. On one hand new patent applications in complex topics are rapidly growing. On the other hand technological interdisciplinarity is increasing. In addition, some claims include a large number of features and are difficult to search in the whole corpus. With advancing development of artificial intelligence the number of projects and companies that investigate and offer search solutions is increasing. Different models and methods are used such as bag of words or words2vec combined with unsupervised and supervised methods. With the groundbreaking publication of bidirectional encoder representations from transformers (BERT) by huggingface (Note 1), this method enjoys great popularity in the field of natural language processing. BERT applications in the field of patents are already known.

This work is inspired by the publication of Risch et al. (2020). The authors provide a data set concerning patent examinations at the european patent office (EPO). Patent claims are matched against text passages which are assessed with X (document of particular relevance) or A (document defining general state of the art). Our tests on this data set did not produce good results. We suspect that the data set is not extensive enough for a given technology field and that the cited text passages are too fragmented.

For this reason we have developed a new approach and trained BERT in a new way for novelty searches. The main idea is to relate or concatenate the patent claims to their own description. Assuming that the description of the claim were not taken from its own but from another patent. However, that other patent would be very likely to destroy the novelty. By training BERT to identify a description of a claim that is destroying its novelty the description of a claim should provide the ideal training basis. Such a trained BERT should be able to identify novelty destroying descriptions for a claim in other patents or in other non-patent literature documents. In addition, there is an almost unlimited amount of data through this method, since virtually any patent can be used for training. It may be beneficial to train BERT specifically on a technology field or patent class which can be easily implemented.

In this paper we describe this method in detail by applying BERT to five patent applications and compare the result with

the cited X patents in the search reports (Note 2). The method also includes a scoring scheme called relevance score or novelty score. To our knowledge the method described in this work has not been published yet. But the presented method offers the following advantages:

- Basically, every patent (in an appropriate language) is suitable for training. Almost any amount of data is available for any technology. This means that BERT can be trained very precisely for any searched claim.
- Independence from published data sets.
- Data quality is very good and independent on subjective examiner evaluations.
- Handling of described procedure is quite simple, efficient, and straightforward.

The paper is structured as follows: first, we identify related literature in chapter 2. Then we introduce our developed method in chapter 3. In chapter 4 we apply the described method with real patent data and demonstrate the results. In chapter 5 we conclude and discuss future research.

2. Related Literature

There are several reasons for applying artificial intelligence (AI) in search procedures for patent applications. Setchi et al. (2021) developed a platform to compare state of the art AI techniques in patent searches (feature extraction, query expansion, etc.) and concluded that up to now AI is not sufficient successful to fully automate the patent application and filing process. A similar finding is reported in Demey and Golzio (2020). Further, Aristodemou and Tietze (2018) identifies four categories: (a) knowledge management, (b) technology management, (c) economic value and extraction, and (d) management of information.

Currently, BERT is one of the most popular models for tasks in natural language processing (NLP). BERT is based on the so called "*self-attention mechanism*" (Vaswani et al. (2017)). Self-attention means that different positions in a sentence or input sequence are related to each other. BERT is pre-trained on two tasks: the masked language model and next sentence prediction (Devlin et al. (2018)). Applying BERT means fine-tuning the pre-trained BERT to a task like patent classiffcation (Sun et al. (2019) and Lee & Hsiang (2020a)). Patents are classified according to standards as international patent classiffcation (IPC, Note 3) and cooperative patent classiffcation (CPC, Note 4) by patent offices according to technical features characterizing the invention.

Srebrovic and Yonamine (2020) do research with pre-training BERT on patents. An advantage of training BERT to patents from scratch compared to a standard BERT is that patent language typically differs compared to standard language. Many words, typically occurring in patents, are therefore not split during tokenization which should increase the performance on patent tasks applying NLP. This is also true for tokens specialized to patents as claim tokens whereby BERT can localize the text within a patent. BERT can be applied for patent searches as well in combination with generative pre-trained transformers (Lee & Hsiang (2020b)), or based on a data set matching claims of european patent applications to relevant or non-relevant text passages (Risch et al. (2020)). In a preliminary test Risch et al. (2020) achieved an accuracy between 0.52 to 0.54 for two labels according to A or X patents. The application of transformer models on patent landscaping has been reported by Choi et al. (2019). The authors use patent landscaping to identify related patents during research and development projects to avoid risk of patent infringement.

3. Method

The method described below is based on the following idea: Ffr almost every claim in patent data exists a novelty matching text passage. Its own description of the claim that describes the invention in the patent itself. We match (concatenate) claims and its descriptions within the patents to train BERT which learns to identify novelty relevant descriptions to claims. Such a trained BERT (claim-to-description-BERT) should have the capability to identify novelty relevant patents for a searched claim.

3.1 Data Generation

BERT has a capacity of processing input lengths of size 512 tokens. A typical patent description counts several thousand words. Therefore, it is impossible to concatenate a claim to the whole description within one input sequence. By cutting description into sections or pieces we concatenate a claim to such a description piece. Thus, which description piece should be concatenated to the claim? Here, we make use of the following observation from the patent search practice: most of the description passages fit very well to the claims within a patent. Moreover, we found that a very good approximation is related to the whole patent and its claim by concatenating all sliced description pieces to the claim. Of course this does not always apply in every single case but from a statistical point of view this approximation is very useful.

More formal: each description piece *k* of patent *i* is combined with the claim of the same patent *i* according to the first label (here label 1, see figure 1):

$$Input_{i,k} = claim(patent_i) <> SEP <> description\text{-}piece(patent_i)_k,$$

where SEP means separation token and <> is an operator that displays a new token set. Be aware to cut the description into pieces of a size that input length needs not to be truncated to fit the input length of BERT.

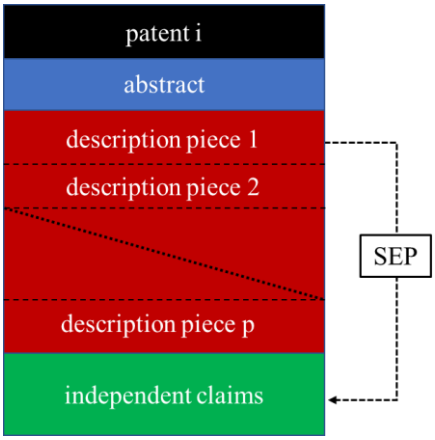

Figure 1. Generation of the first label training data for novelty search by concatenating the claims to the description of the same patent. The description of patent *i* is sliced into *p* pieces yielding *p* claim and description piece concatenations

Now, having the data for the first label we need to complete the training data set by generating data with the second label (here label 0) where no match between claim and description piece is available. Consequently, we have to concatenate claims to description pieces which are of no novelty relevance. Here, we make use of a second observation from the patent search practice: most description pieces of a patent *j* are of no novelty relevance of a searched claim of patent *i*, with $i \neq j$. Applying this observation we generate the second label according to (here label 0, see figure 2):

$$Input_{i,j,k} = claim(patent_i) <> SEP <> description\text{-}piece(patent_j)_k,$$

where $i \neq j$. Description pieces and claims are combined in a random manner subject to the limitation that claim and description piece belong to different patents.

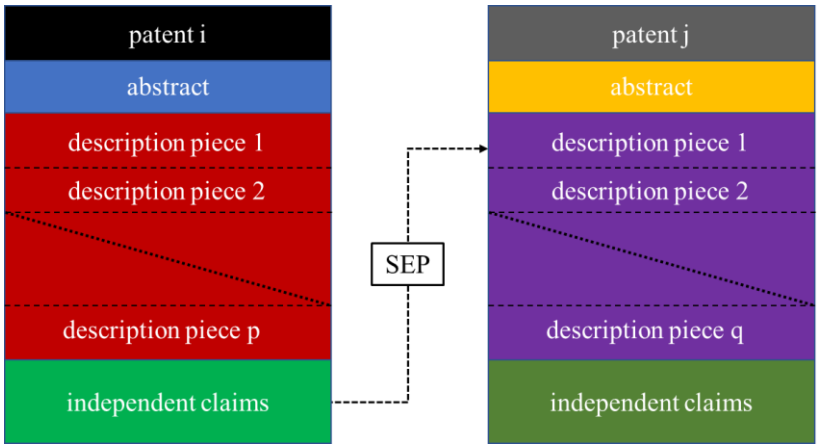

Figure 2. Generation of the second label training data for novelty search by concatenating claims to description of other patents. Description of patent *j* is sliced into *q* pieces yielding *q* claim and description piece concatenations

By taking the same claim and description pieces for the second label as for the first label we make sure that both labels have the same frequency and BERT learns to distinguish between relevant and non-relevant description pieces for a claim. Possibly, there is a probability of concatenating a claim randomly to a well fitting description piece of another patent. But the effect of confusing during training should be very low.

Eventually, each input has the structure:

$$CLS <> claim <> SEP <> text\text{-}piece <> PAD <> SEP <> <> label,$$

where CLS is the classification token and PAD is the length of the input padded during tokenization to the defined maximum input length if the length of the concatenated claim and description piece is smaller than the maximum input length. The operator <><> displays the relationship between the label and the corresponding text piece. Otherwise we get:

$$CLS <> claim <> SEP <> text\text{-}piece <> SEP <> <> label.$$

The maximum input length is not necessarily 512 token but can also be set to a lower value beforehand. It is conceivable that size of description pieces is dynamically adopted if the concatenated claim of input length of the concatenated claim and description piece fits exactly the maximum input length. Beyond that we expect the best result when description pieces are at least as long as the claim or even longer. The claims include technical features in a very compact form. A description is particularly relevant if it contains all the features of the claim., So it can be expected that size of the description piece has at least the same length as the claim.

3.2 BERT Model and Training

The following procedure is standard to train BERT. The first step is the tokenization of training inputs. We applied the tokenizer bert-base-uncased, lower-case developed by huggingface. An important parameter is maximum input sequence length. BERT allows a maximum input length of 512 tokens. It is intuitively obvious to go close to this limit with this model provided computer capacity allows it. In our experiment in chapter 4 we set the input length to 500 tokens.

The maximum input length could be a significant limitation to BERT: is the length of the searched claim in that range or even larger (which should not happen often in practice) we divide the claim into subclaims. But practice shows that usually claims are much shorter than 500 words and there are good capabilities to concatenate a claim via separation token SEP to a description piece.

Hyperparameters as learning rate, batch size, and number of epochs have to be adjusted and a BERT model has to be selected. A model of choice is Bert For Next Sentence Prediction. We train the data with BERT versions Bert For Next Sentence Prediction and Bert For Sequence Classification (both as bert-base-uncased) and astonishingly achieved better results with Bert For Sequence Classification. So in the following we will focus on that model.

Before applying a trained model to the application of interest there is a validation during training or testing after training to estimate overfitting or to estimate the performance of the model. We made the experience that the significance of the validation of concatenated claim and description pieces which are not trained explicitly but in other combinations is low. A better way for validating training is using complete different patents. Therefore, we applied this approach after training BERT in a pre-test on 100 patents (see subsection 4.2.1).

3.3 Applying Trained BERT to Novelty Search

Once BERT is trained (fine-tuned) BERT can be applied to novelty searches for claims of interest. In our studies we focused on train BERT on selected technology fields. The required computer capacity and volume of data is therefore significantly lower. In addition, it cannot be ruled out that a technology-specialized BERT is superior to a general trained BERT (trained to all or at least to several technology fields). The disadvantage, however, is that BERT must be trained specifically for a task provided that the technology has not been trained before.

To prepare the input we prepare the patents similar to data generation procedure described in section 3.1: the descriptions of the patents which we want to analyze according to novelty relevance of the claim of interest are sliced into description pieces. Again, the length of the description pieces has to fit to the length of the claim of interest and the chosen maximum input length to BERT. As before for training we expect the best results when description pieces are at least as long as the claim or even longer. Then, the claim of interest – $claim_{oi}$ – is concatenated to the description pieces and we get the following structure (by neglecting the CLS and PAD token, see figure 3):

$$\ldots$$
$$Input_m = claim_{oi} <> SEP <> description\text{-}piece(patent_j)_k$$
$$Input_{m+1} = claim_{oi} <> SEP <> description\text{-}piece(patent_j)_{k+1}$$
$$Input_{m+2} = claim_{oi} <> SEP <> description\text{-}piece(patentj)_{k+2}$$
$$\ldots$$
$$Input_n = claim_{oi} <> SEP <> description\text{-}piece(patent_{j+1})_l$$
$$Input_{n+1} = claim_{oi} <> SEP <> description\text{-}piece(patent_{j+1})_{l+1}$$
$$Input_{n+2} = claim_{oi} <> SEP <> description\text{-}piece(patent_{j+1})_{l+2}$$

…

It is important to track the patent number for each description piece to finally link the search result of BERT with the corresponding patent.

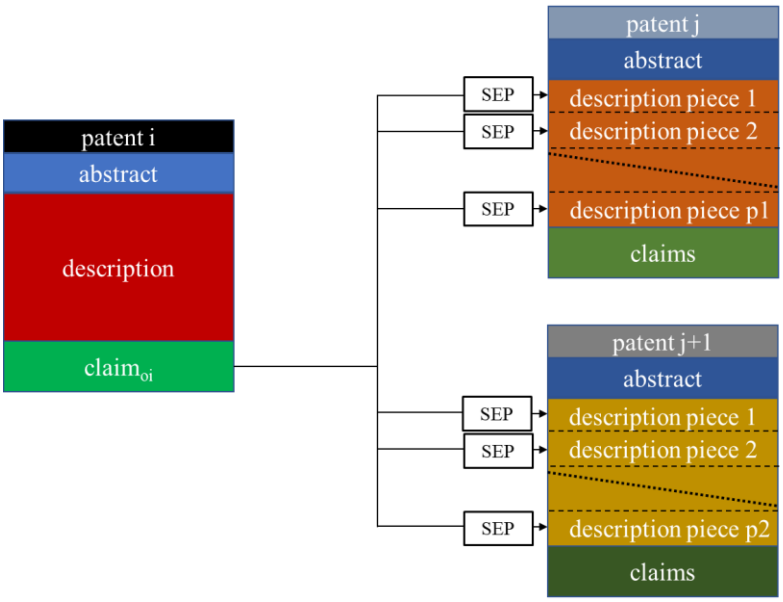

Figure 3. Novelty search of a claim of patent *i*: the searched claim is concatenated to the descriptions of patents which are analyzed according to novelty relevance

3.4 Postprocessing of BERT Results: Noise and Relevance Scoring

Since a typical description in a patent counts to several thousand words a claim would be concatenated to tens of description pieces per patent. This, of course, can be affected by the length of description pieces. There is a variation of the length of descriptions in patents dependent to the technology field or the country code of the patent. Finally, there is a considerably number of concatenations per claim. The experiments have shown that BERT often labels a single description piece as relevant. In practice, it is very inefficient to regard every patent with at least one description piece labeled as relevant. In some sense these single relevance labeling by BERT has to be understood as noise. But the great difference between novelty relevant and non-relevant patents is that the number of relevant labeled description pieces in relevant patents is much higher compared to non-relevant patents. For the description of a claim within a patent BERT identifies nearly every description piece as relevant. To get rid of this noise and to distinguish relevant from at least less relevant patents we calculated a relevance or novelty score in two ways.

3.4.1 Relevance Scoring According to the Label

The relevance or novelty score $R(claim_{oi}, patent_i)$ for a claim of interest – $claim_{oi}$ – is calculated for a patent *i* by simply sum relevant labeled claim and description piece concatenations of patent *i* divided by the total number of claim and description piece concatenations of patent *i*. The total number of claim and description piece concatenations of a patent *i* is simply the sum of its label 0, $\sum label\_0(patent_i)$, and label 1, $\sum label\_1(patent_i)$. Because of division the scoring has the property of a density. As a result of which non-relevant patents with individual description pieces classified as relevant are given a low rating and relevant patents with several or even many description pieces classified as relevant are given a high rating. We finally get:

$$R(claim_{oi}, patent_i) = \frac{\sum label\_1(patent_i)}{\sum label\_0(patent_i) + \sum label\_1(patent_i)}. \quad (1)$$

Here, we have arbitrarily assumed that label 1 classifies description pieces as relevant and label 0 as not relevant. In chapter 4 we will apply this scoring method instead of the method described in the following chapter 3.4.2.

3.4.2 Relevance Scoring According to Sigmoid Values of Label 1

An alternative way to calculate the relevance score $R(claim_{oi}, patent_i)$ for a claim of interest – $claim_{oi}$ – for a patent *i* with the same density properties according to subsection 3.4.1 is simply to sum up all sigmoid values given for label 1.

Even if BERT assigns label 0, i.e. the probability for label 1 is below 50 %, the sigmoid value is added. The sigmoid values can be calculated with the logits output by BERT yielding a relevance score for patent *i* according to:

$$R(claim_{oi}, patent_i) = \frac{\sum sigmoid_{label\_1}(patent_i)}{\sum label\_0(patent_i) + \sum label\_1(patent_i)}, \quad (2)$$

with the well known relation

$$sigmoid_{label\_0}(patent_i) + sigmoid_{label\_1}(patent_i) = 1. \quad (3)$$

It is conceivable that this method works better than the method in subsection 3.4.1 since the assignment to label 0 and label 1 is not binary but has probabilities. According the method in subsection 3.4.1 no distinction is made between whether a label 1 assignment has only been made to 51 % or 99 %. But in the numerator of equation 2 the sigmoid values are added up if the probability is below 50 % and label 0 has been formally assigned. Both methods are shown in figure 4.

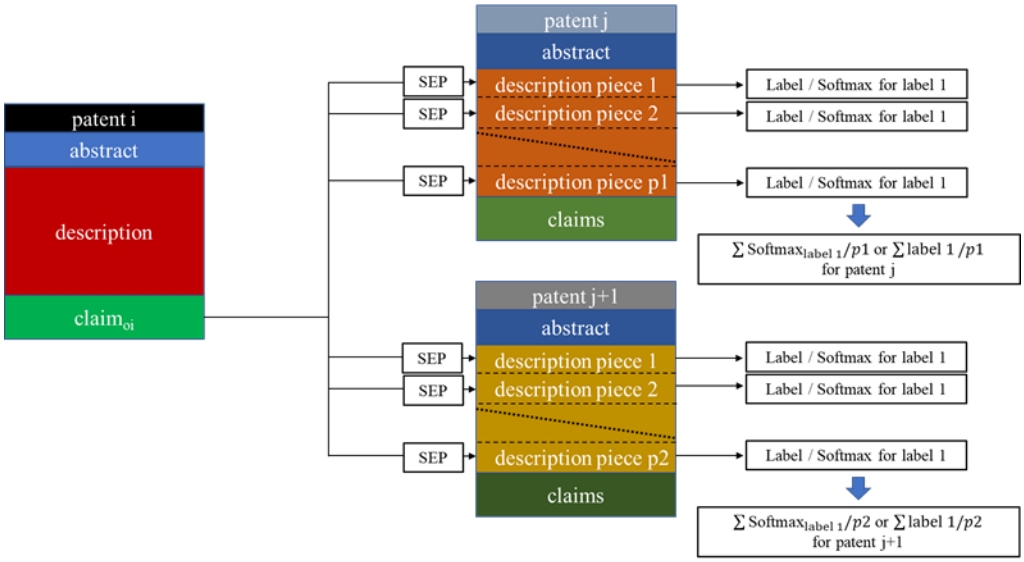

Figure 4. Novelty scoring by counting the label 1 data or by adding the sigmoid values for label 1 predictions divided by number concatenations

4. Experiments and Results

We applied the method described above to a group of patent applications (reference patents) to answer the question whether BERT is able to identify the X patents cited in the corresponding search reports. Thus, does BERT identify the X patents as novelty relevant by assigning them a high relevance score according to subsection 3.4.1 compared to non-relevant patents. The cited X patents do not necessarily have to be the best reference for a test by BERT as there could be further or even better prior art. Moreover, an examiner will select a patent according to the dependent claims which we have not taken into account. But this test is intended to give an initial indication of whether the method described above can be used as a first benchmark to return the relevant existing patents. To this end we perform the following procedure:

Step 1: Generation of 1059 training patents of label 0 and label 1 according to section 3.1 by a random selection of patents which refer to the same patent class (here on IPC main-group level). These patents form group 1.

Step 2: Training BERT on the training patents (group 1).

Step 3: Random selection of a second group (group 2) of 100 patents chosen from the same patent class as group 1. This group will be used for the pre-test.

Step 4: Random selection of a third group (group 3) of 1011 patents chosen from the same patent class as group 1. This group has to contain the cited X patents. This group is called to-be-searched group. BERT analyze this to-be-searched group and score every patent according to the relevance or novelty score according to subsection 3.4.1.

Step 5: Identifying cited X patents in relevance and novelty score hierarchy.

Remark: We do intentionally not perform any text pre-processing steps.

4.1 Data

The novelty search of the first claims is performed for the following patent applications (reference patents): WO 2019/064177 A1, EP 3 593 317 A1, EP 3 382 640 A1, EP 3 438 918 A1 and EP 3 499 493 A1. For these five patent applications exist a search report with X citations given in table 1.

Table 1. Reference and cited patents

Reference patent	Cited X patent
WO 2019/064177 A1	US 2017/109857 A1
	US 2011/194726 A1
	WO 2005/029390 A1
EP 3 593 317 A1	WO 2016/025631 A1
	US 2015/156369 A1
EP 3 382 640 A1	WO 2015/185944 A1
	US 2005/193205 A1
	US 2016/328398 A1
EP 3 438 918 A1	US 2014/029812 A1
EP 3 499 493 A1	US 2015/123887 A1

Reference patents are given in the first column and the search report cited X patents in the second column. Novelty search is performend on five reference patents.

All reference patents are assigned to the IPC patent class G06T1/00. Most of cited X patents are classified in G06T1/00 and below. For this reason the 1059 training patents (group 1), the 100 pre-test patents (group 2), and the 1011 to-be-searched patents (group 3) are as well randomly selected from G06T1/00 and below. The cited X patents were added to group 3 (if not already included randomly). All randomly chosen patents in groups 1 to 3 are US, EP, GB and AU patents in english language. It was checked that BERT was not trained on reference as well not trained on cited X patents. Although this is not a condition or limitation of the presented method. Between randomly chosen training patents and randomly chosen to-be-searched group (group 3) patents is an random overlap of 58 patents. There is no overlap to the pre-test (group 2) documents. The descriptions of the patents in groups 1 to 3 were sliced into pieces of random size in the range of 100 to 200 words. Future research could investigate a dynamic slicing to fit exactly the size of the concatenated claim and description piece to the maximum sequence length as described in section 3.1. The maximum sequence length for BERT has been chosen to 500 tokens. Slicing the descriptions of the training patents (group 1) and the generation of label 0 and label 1 claim and description pieces according to section 3.1 yield 74 498 training input sequences. We split randomly 9250 concatenated claim and description pieces for validation. As mentioned in section 3.2: although the concatenated claim description pieces are not trained the significance of the validation set is rather low since BERT is trained on the same claims and description pieces. It would have been more reasonable to generate a validation set by taking claims and descriptions not identified by BERT during the training. This is done after the training in a pre-test. Slicing the pre-test patents of group 2 and generation of the label 0 and label 1 by concatenating the claims to their description pieces (label 1) and the same claims and descriptions in a random manner to label 0 according to section 3.1. This yields 3582 input sequences of concatenated claim and description pieces. Slicing the descriptions for novelty search of to-be-searched patents of group 3 and generation the input sequences according to section 3.3 yields 40 016 concatenated claim and description pieces. We then trained BERT (Bert For Sequence Classification, bert-base-uncased) for two epochs.

4.2 Results

4.2.1 Pre-test on 100 Patents

In the pre-test we applied the trained BERT on 100 patents classified in IPC G06T1/00 (group 2). As described in section 4.1 we have 3582 input sequences of concatenated claim and description pieces. In contrast to the novelty search claims in these 100 documents are also concatenated with their own descriptions. This pre-test should therefore show whether BERT is able to identify its own descriptions for claims. The pre-test in 100 patents yields a F1 score of

0.936. BERT very well identified the correct descriptions of the claims. In this pre-test we do not apply the relevance score according to subsections 3.4.1 or 3.4.2. The given F1 score refers to all 3582 tested claim and description piece concatenations. The novelty search in the to-be-searched group (group 3) with 1011 patents of the trained BERT yields the results shown in table 2. The results are discussed in chapter 5.

Table 2. Reference and cited patents with positive relevance scores

Reference patent	Cited X patent	Positive relevance score
WO 2019/064177 A1	US 2017/109857 A1	198
	US 2011/194726 A1	42
	WO 2005/029390 A1	25
EP 3 593 317 A1	WO 2016/025631 A1	4
	US 2015/156369 A1	1
EP 3 382 640 A1	WO 2015/185944 A1	242
	US 2005/193205 A1	18
	US 2016/328398 A1	345
EP 3 438 918 A1	US 2014/029812 A1	1
EP 3 499 493 A1	US 2015/123887 A1	top score, but…

Reference patents are given in the first column, search report cited X patents in the second column, and hierarchy position given by BERT in the third column. BERT searched in 1011 patents. Position 1 in column 3 means that BERT gave the patent in column 2 the highest relevance score of all 1011 patents regarding the first claim of reference patent in column 1. Details to patent EP 3 499 493 A1 can be found in chapter 5.

5. Discussion

The pre-test yield a F1 score of 0.936. Hence, BERT very well identified the correct descriptions of claims. To emphasize once again BERT identified the individual description pieces independently of another with a very high degree of accuracy while at the same time recognizing description pieces of other patents are not assigned to the claim. This result gives reason to be confident that BERT can also be used to identify description pieces relevant to a claim in other patents or other documents of the non-patent literature and thus applied successfully to patent novelty searches. Such a novelty search was tested on five patent applications according to table 2. The relevance score shows that BERT rates some of the cited X patents as well as highly relevant. The position given in the third column in table 2 shows the position of cited X patents in the sample of the 1011 patents according to relevance score for reference patents in column 1. The X patent US 2015/156369 A1 is on position 1 for the reference patent EP 3 593 317 A1. In the sample of 1011 patents this patent was identified by BERT as the most relevant. On the other hand there are X patents with low rating as US 2016/328398 A1 on position 345 for reference patent EP 3 382 640 A1. There could be several reasons: in the test we searched only the first claim. Possibly this patent has been selected as X patent due to dependent claims that BERT did not search. The X patent US 2015/123887 A1 for reference patent EP 3 499 493 A1 was high rated by BERT as well. But some patents were highly rated as well and indistinguishable from X patent in terms of the relevance score. This can be explained by the following: a method of displaying images at a display area of a display device the method comprising. Thus, displaying visible content at a central portion of the display area and simultaneously displaying invisible content only at one or more edges of the display area. For this reason the better strategy here may be to include dependent claims for the novelty search as well. There are some differences that our search differs from a real novelty search. (a) We searched only the first claim. The dependent or further independent claims were not searched. (b) We did not take into account the priority application and publication dates. The effect could be that the relevance score of the cited X patents compared to other analyzed patents are lower since patents published after the application date could be of high relevance. (c) The sample of 1011 patents were randomly chosen in the same technology field of the reference patents (with the exception of some X patents). In real search BERT analysis could cover the entire patent class or patents filtered according to keywords. However, it is possible that a complete consideration of the entire patent class would have worsened the X documents in their position in the score. (d) As described above some of the training and to-be-searched patents have been truncated due to limits in Excel whereas the X documents were completely analyzed. It should be examined how these deviations influence the result. It must also be emphasized that the method proposed here will probably not work if the text pieces which are of novelty relevance appear only in a very isolated manner in an analyzed patent. In this case a relevance score by BERT according to the described method could be interpreted as noise.

6. Conclusion and Future Research

We presented a new method to train BERT for patent novelty search by concatenating patent claims to their own descriptions (claim-to-description-BERT). The descriptions of patents are sliced into description pieces of a certain length which should be adopted to the length of trained or searched claims. In our tests we sliced the description into pieces with a random length in the range of 100 to 200 tokens. We applied the trained BERT in a pre-test on 100 patents where BERT had to identify the descriptions to a corresponding claim of the same patent. We got an F1 score of 0.936. Eventually, we applied the trained BERT to a patent novelty search for five patent applications and compared the result to the corresponding search reports. The results showed that BERT could identify some cited X patents as highly relevant out of a group of 1011 patents in the same technology field. We have identified some possibilities on how this work can be continued:

(a) Length of the text pieces: we expect the best result when description pieces are at least as long as the claim or even longer. Finally, claims include technical features in a very compact form. A description is particularly relevant if it has all features of the claim. It is therefore to be expected that a piece of a description of relevance will have at least the same length as the corresponding claim. For training BERT dynamically adapting the length of the description to the respective claim is recommended.

(b) It would be interesting to compare the alternative relevance score according to subsection 3.4.2.

(c) We trained BERT particularly on B patents (granted patents). The independent claims of B patents tend to have more features than independent claims of the corresponding applications (A patents). The question arises whether BERT should not be better trained on A patents as the novelty search is applied to A patents. A mixture of A and B patents for training may also be the best strategy.

(d) We have taken the approach of training BERT technically very close to the features to be searched. However, it may be sufficient to train the rough technical environment which would simplify the selection of training patents.

(e) We trained and applied the method only to the first claims. An adoption to dependent claims would be very instructive.

(f) Extending the described method to search patents to attack the inventive step of a reference patent.

(g) Discussion with patent offices for practical implementation should be conducted. The effective time saving effect or increase in quality should be straightforward.

Acknowledgements

We would like to thank Jochen Spuck, (EconSight, Basel, Switzerland) and Carsten Guderian (PatentSight, Bonn, Germany) for their helpful and valuable discussions and support on patent data.

References

Aristodemou, L., & Tietze, F. (2018). The state-of-the-art on intellectual property analytics (ipa): a literature review on artificial intelligence, machine learning and deep learning methods for analysing intellectual property (ip) data. *World Patent Information*, *55*, 37-51. https://doi.org/10.1016/j.wpi.2018.07.002

Choi, S., Lee, H., Park, E. L., & Choi, S. (2019). Deep patent landscaping model using transformer and graph embedding. arXiv preprint arXiv:1903.05823.

Demey, Y. T., & Golzio, D. (2020). Search strategies at the european patent office. *World Patent Information*, *63*, 101989. https://doi.org/10.1016/j.wpi.2020.101989

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pretraining of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Lee, J. S., & Hsiang, J. (2020a). Patent classification by fine-tuning BERT language model. *World Patent Information*, *61*, 101965. https://doi.org/10.1016/j.wpi.2020.101965

Lee, J. S., & Hsiang, J. (2020b). Prior art search and reranking for generated patent text. arXiv preprint arXiv:2009.09132.

Risch, J., Alder, N., Hewel, C., & Krestel, R. (2020). PatentMatch: data for matching patent claims & prior art. arXiv preprint arXiv:2012.13919.

Setchi, R., Spasi ¢, I., Morgan, J., Harrison, C., & Corken, R. (2021). Artificial intelligence for patent prior art searching. *World Patent Information*, *64*, 102021. https://doi.org/10.1016/j.wpi.2021.102021

Srebrovic, R., & Yonamine, J. (2020). Leveraging the bert algorithm forpatents with tensorflow and bigquery. Retrieved from https://services.google.com/fh/files/blogs/bert_for_patents_white_paper.pdf

Sun, C., Qiu, X., Xu, Y., & Huang, X. (2019). How to fine-tune bert for text classification? *China National Conference on Chinese Computational Linguistics*, 194-206.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. arXiv preprint arXiv:1706.03762.

Notes

Note 1. https://huggingface.co/

Note 2. When we write about training BERT in the following, we mean fine tuning pre-trained BERT.

Note 3. https://www.wipo.int/classifications/ipc/en.

Note 4. https://www.epo.org.