Expert Systems With Applications 207 (2022) 117942

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

PaTRIZ: A framework for mining TRIZ contradictions in patents

Guillaume Guarino *, Ahmed Samet, Denis Cavallucci

ICUBE (UMR CNRS 7357), INSA Strasbourg, 24 Boulevard de la Victoire, 67000 Strasbourg, France

ARTICLE INFO

Keywords:
Patent
Deep learning
NLP
Contradiction
TRIZ

ABSTRACT

Patents are a significant source of information about inventions. However, understanding the content of a patent with the aim of using it for an automatic solution search is still an unsolved challenge. To achieve this purpose, a model based on the TRIZ theory (Altshuller, 1984) has been developed. This theory introduces the notion of contradiction, which is a reliable and domain-independent technique to formulate the problem solved by each patent through an opposition between parameters of a system. Each patent is considered a solution concept to a contradiction. Mining contradictions, therefore, means characterizing solution concepts.

In this paper, we propose a new approach called PaTRIZ, a complete framework for patent analysis based on a combination of sentences and word-level deep neural networks. The word-level network, called ParaBERT, comprises a novel Conditional Random Field structure, developed to integrate syntactic information. The idea is to mine the patent's motivating problem (aka *contradiction*), which is fundamental to understanding the invention and identifying for which purpose it could be used. The models are evaluated on built-in real-world datasets.

1. Introduction

Patents are a massive source of information when it comes to solving a problem. Manual expert analysis and exploitation of patents' data to answer a problem is a tedious task due to the gigantic volume of patents filed each year (1.5 million according to the World Intellectual Property Organization). This is why automatic analysis of patent content is of paramount importance for genuinely assisting R&D engineers and increasing their innovation capabilities when they solve problems.

Patents can be analyzed to establish a state of the art on a specific problem from a specific field. But, they can also be a source of inspiration by broadening the field of research to other fields. New ideas can then be discovered by studying patents from other technical fields. This is what Genrich Altshuller, a USSR engineer, highlighted after analyzing 40,000 inventive patents from all fields (Altshuller, 1984). The conclusion of Altshuller's analysis was that the problem-solution pairs were very similar no matter the technical field, and that the inventive solutions often came from the use of a scientific effect from another domain. This analysis, therefore, shows the importance of exploiting data from different fields to inventively solve a problem. Thus, finding similar problems solved in other domains would be the way to innovate. These findings laid the foundation for his theory called TRIZ (the Russian acronym for "theory of inventive problem solving").

However, the volume of data to be processed grows exponentially as one moves away from the starting problem's domain. Therefore, the purpose of this article is to introduce an approach that allows the uniform and automatic analysis of patent data from all fields.

The main challenge in comparing problems from different domains is the formulation of the problems. It is difficult, for instance, to measure the similarity between a mechanical problem and a problem from the automation field. This is why Altshuller introduced the notion of *contradiction*. All problems can theoretically be described by a contradiction. The definition of a contradiction relies on 2 an opposition between two system parameters. When the first one is improved, the second one is degraded. For instance, if one wants to *increase the volume of a car*, one will *increase the size of the chassis*, which will logically lead to an *increase in the weight of the car*, which is not desired. The two conflicting parameters, called *Evaluation Parameters* (EP) are therefore the volume and the weight. The parameters of the system that can be modified, such as the size of the chassis, in this case, are called *Action Parameters* (AP).

In this paper, we aim to mine the contradiction sentences from which we mine the contradictory EPs and APs. Thanks to the advances in Natural Language Processing (NLP) techniques and deep learning tasks, automating this mining process has never been so reachable. Unfortunately, to the best of our knowledge, the literature is limited in terms of its contribution to mining TRIZ contradictions using supervised learning and in retrieving contradictory parameters.

* Corresponding author.
E-mail addresses: guillaume.guarino@insa-strasbourg.fr (G. Guarino), ahmed.samet@insa-strasbourg.fr (A. Samet), denis.cavallucci@insa-strasbourg.fr (D. Cavallucci).

https://doi.org/10.1016/j.eswa.2022.117942
Received 24 October 2021; Received in revised form 9 June 2022; Accepted 20 June 2022
Available online 25 June 2022

To solve these challenges, we propose the PaTRIZ approach based on three distinct contributions. First, sentences that may contain evaluation parameters (contradiction sentences) are mined using an automatic summarization model. The summarization model is an improved version of the approach proposed in Guarino et al. (2020). It aims to rank the sentences by the probability of containing the improved evaluation parameter. The sentence with maximum probability constitutes the *First part of the contradiction*. The same process is used to retrieve the sentence with the degraded evaluation parameter when the parameter of the *First part of the contradiction* is improved. This second sentence is called the *Second part of the contradiction*. We aim to boost the performance of the contradiction mining approach by adding a document classifier module that filters documents containing TRIZ contradictions. Once both parts of the contradiction are extracted, the parameters are much easier to mine via a Named-Entity Recognition (NER) approach. A NER approach was chosen as the parameters can be a single word like "accuracy" but also a phrase containing several words e.g. "pressure on the fan's lower part". Therefore classifying tokens seems to be the best solution to extracting parameters of varying lengths. A combination of a Conditional Random Field (CRF (Lafferty et al., 2001)) and a deep neural network is implemented for the Named Entity Recognition task. To integrate syntactic information and help the decision process, a new constrained, Part-Of-Speech-adapted CRF has been developed. It constitutes the second stage and contribution of the paper. Finally, all the outputs of these intermediate steps are used to draw a conclusion on the quality of the extraction. These two models are integrated using an approach called PaTRIZ , which scraps all TRIZ contradictions and parameters from patent data.

Our models are trained over two real-world distinct datasets (for contradiction and for parameter mining) that have been manually labeled. Finally, a demonstrator of PaTRIZ has been developed.[1]

2. State of the art

2.1. Extractive summarization

Automatic summarization is a very common task in the NLP community. The aim is to extract salient information from a document. The two existing types of automatic summarization are extractive and abstractive summarization. Extractive summarization consists of selecting the sentences that contain the needed information. The advantage is that the information is not altered since it is returned as it is in the document. However, the summary resulting from these selected sentences will lack coherence and the links between the sentences remain to be established by the reader. The abstractive summary, on the contrary, aims to solve this problem by synthesizing new sentences from the document to be summarized. The drawback of these approaches is that several pieces of information may be distorted and this may sometimes lead to misunderstandings.

The purpose of this study is to extract of the parameters that form a contradiction. Therefore, the alteration of the information with an abstractive summary is not acceptable since the parameters of the contradiction could be modified or even removed. The extractive summary, therefore, seems more relevant since the parameters would remain unchanged. Furthermore, when parameters are not explicitly quoted but simply described, e.g. *"prevent fluid from entering..."* for the parameter *"sealing"*, the selected sentence contains the whole description.

Numerous techniques have been employed for extractive summarization, ranging from graph-based methods (Kleinberg, 1999; Litvak & Last, 2008; Mihalcea, 2004; Page et al., 1998), to Naive Bayesian approaches (Aone et al., 1998; Kupiec et al., 1995), Hidden Markov models (Conroy & O'leary, 2001) or Conditional Random Field-based models (Shen et al., 2007).

Neural networks have now taken over from other techniques as they allow better results, requiring, nevertheless, much more data. Extractive summarization usually consists of a binary classification of sentences. A representation of each sentence is computed, and then a classifier decides whether the sentence will be part of the summary or if it does not provide any relevant information. The main research focus is on the quality of these representations, which should contain as much explicit information as possible to facilitate decision-making by the classifier.

Recurrent neural networks are ubiquitous in the field of summarization and more generally in the field of NLP thanks to their ability to model temporal dependencies in sequences of words of variable length. In particular, LSTMs (Long Short Term Memory networks) have made it possible to eliminate the problem of gradient vanishing. However, the quality of captured dependencies decreases as the length of the input document increases (Nallapati et al., 2017; Zhou et al., 2018).

To overcome this problem, transformers have been developed (Vaswani et al., 2017a). They are bidirectional by nature (whereas recurrent networks are traditionally unidirectional) and they are able to build qualitative representations regardless of the length of the input document. In practice, as the number of parameters is proportional to the square of the number of tokens in the input document, a maximum length of 512 tokens has been defined. These networks are now well-established thanks to their capacity to be pre-trained (BERT Devlin et al., 2019), XLNet (Yang et al., 2019). This pre-training allows them to be applied with good performance to domains with little data. Extractive summarization models based on transformers are efficient due to high-quality sentence representations (Liu & Lapata, 2019). Recently, improvements have been made to bidirectional recurrent networks via syntactic compression (Xu & Durrett, 2019), making them as efficient as transformers.

2.2. Contradiction and parameter mining

Souili and Cavallucci (2013) and Souili et al. (2015) proposes to extract evaluation and action parameters from linguistic markers. The syntactic structures that most commonly include parameters are identified. The markers are generic (nouns, verbs, adverbs) which does not compromise their use in various domains.

In Cascini and Russo (2007), the constituent elements of a system are extracted by similarity with concepts contained in a pre-established database or thanks to keywords. The search for the contradiction and the contradictory parameters is a multi-step process that relies mainly on patterns. If one of the nouns contained in the description of the innovative element located in the claims is not an element of the system, it is considered a parameter. The closest verb gives its sense of variation. The degraded parameter is assumed to be in the state of the art and is also extracted via keywords.

Chang et al. (2017) proposes to extract the parameters and inventive principles via a similarity calculation for the inventive principles and patterns for the parameters. Thus, a word is considered a parameter as long as it is located before or after a particular expression such as *"is prevented from worsening"*.

Berduygina and Cavallucci (2020) proposes to draw inspiration from the structure of patent claims to improve the quality of the extraction by keywords and patterns.

As these approaches are very generic, they rely on very strong assumptions about the structure of patents, which are strongly dependent on the writer and domain. This leads to a great number of false positives in the extraction of parameters. More recent techniques, notably deep neural networks, seem to be a major avenue for improvement.

Chen et al. (2020) present a data mining approach based on patents. This approach is not dedicated to the extraction of parameters, but the tools used can be adapted for use in the TRIZ framework. Two neural

[1] The SummaTRIZ demonstrator which relies on our PaTRIZ engine website is available through this link: https://summatriz.inventivedesign.unistra.fr/. The login e-mail is *test@test.fr* and the password is *test*.

networks, a BILSTM and a BIGRU-HAN are used for entity identification and semantic relation extraction and the authors demonstrate the contribution of neural networks compared to methods based on patterns, for example.

We will focus on supervised training for both summarization and parameter mining as no unsupervised task could bring the information needed to compare problems from different domains. The different terminologies used in each domain make clustering methods useless. The purpose here is to target key elements (contradictory parameters from the viewpoint of TRIZ theory).

3. PaTRIZ

The two sub-modules for document, sentence, and word-level analyses that will be discussed in Sections 4 and 5 are integrated using a global framework called PaTRIZ. This framework gathers patent retrieval, sub-module interfacing, and result visualization in the demonstrator.

3.1. Patent scrapping

A few patent databases are freely available. Therefore, only US patents from the United States Patent Trademark Office (USPTO) are used. Approximately 9 million US patents are accessible. It can therefore be assumed that this pool of 9 million patents is a good estimation of the statistical characteristics of all patents filed worldwide. The large size of this database also means that it is a sufficient source of information for the problem-solving process.

Patents are downloaded automatically via the official USPTO storage platform https://bulkdata.uspto.gov/ with weekly updates. The database is indexed with Elasticsearch which allows at the same time to be able to retrieve patents in mass but also to make targeted searches like any search engine on patents sorted by domain, title, keywords, etc...

3.2. Approach overview

PaTRIZ allows the extraction of a contradiction (i.e. two parameters, one parameter per part of the contradiction) from a patent and its validate in a three-steps process. A sentence is first extracted for each of the two parts of the contradiction using the summarization model presented in Section 4. The parameters present in these two sentences are then extracted using the CRF-based model presented in Section 5. The presence of a contradiction is also evaluated with a document classifier. This allows to validate the extraction of a contradiction. The complete model is shown in Fig. 1. The output of the model is therefore the parameters contained in both parts of the contradictions. The parameter(s) contained in the first sentence is(are) supposed to be improved. The parameter(s) contained in the second sentence need(s) to be degraded with the first parameter(s) being improved. Theoretically, there should be only one parameter per sentence, but there is sometimes more than one. This does not affect the process if we extend the notion of contradiction to a set of parameters that cannot be improved at the same time.

The PaTRIZ model was developed in order to process documents with a size of less than 1600 tokens, i.e. containing less than 50 sentences. If the input document is longer than this limit, it is separated into chunks of length less than or equal to 1600 tokens. The chunks are then processed independently. When choosing the "parts of the contradictions" sentences after the sentence classifiers, all the scores of all sentences in the document are taken into account. This ensures that the contradiction search is done on the whole document.

A demonstrator has been developed to implement this complete contradiction extraction model. It is able to search the patent database indexed on one of our servers or load user's patents. A compilation

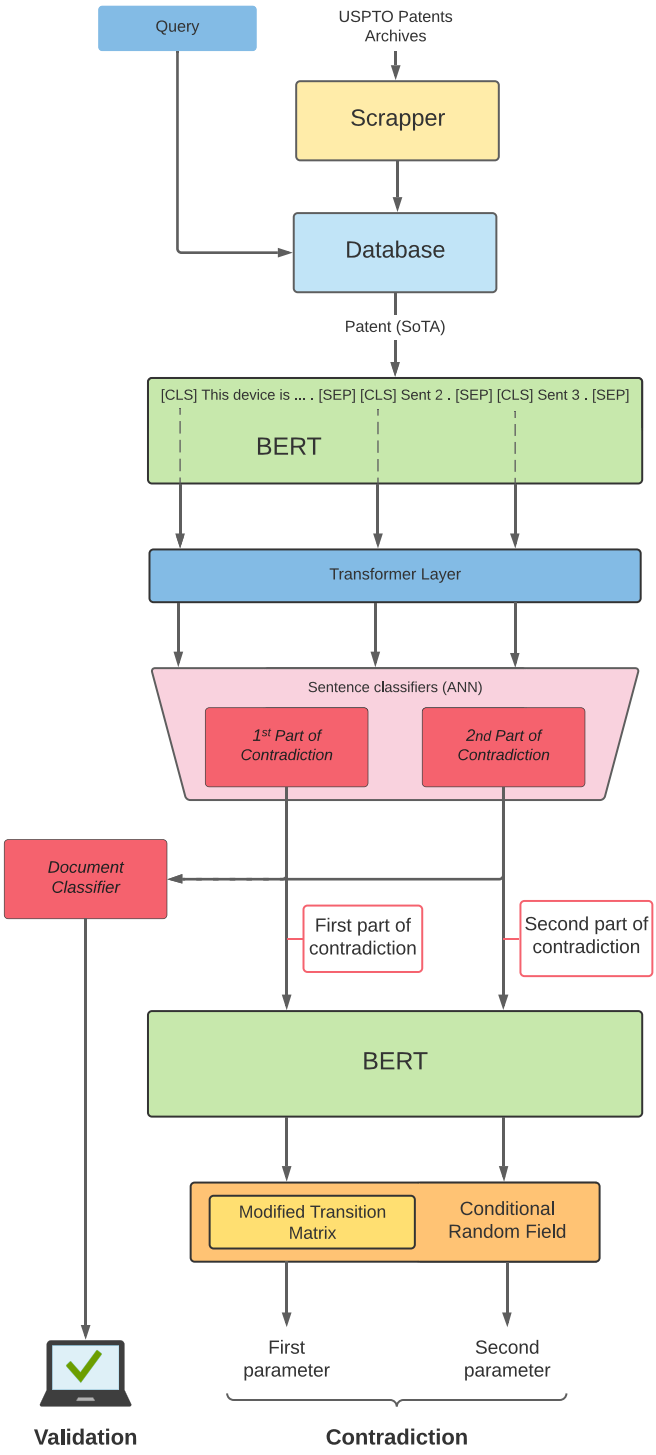

Fig. 1. PaTRIZ global framework.

of patents can be downloaded from this link and then loaded into the web interface to process each patent. The inference time is greater in the demonstrator than in real use since the parameters are extracted from all the sentences of the document to prevent the user from modifying the contradiction's sentences and not getting any results for the parameters. In optimal use, only the two sentences that contain the parameters are processed.

Table 1
Details on the summarization dataset.

Patents	*Sent.* – *Sent./doc*	1st – 1st/*doc*	2nd – 2nd/*doc*
1600	28 732–17.96	2265–1.42	3714–2.32

4. TRIZ summarization

In this section, we present the data used for the training of the "TRIZ summary" model. A dataset was created to address the lack of data in the TRIZ domain concerning contradiction extraction. The architecture and design choices of this model are detailed as well.

4.1. TRIZ's summarization dataset

The state of the art parts of 1600 patents from the United States Patent Trademark Office (USPTO) was labeled to train the sentence classification model. The labeled patents come from all domains. As the patents do not always contain a proper state of the art and do not always contain a contradiction, these 1600 patents are sampled from an initial pool of about 15 000 patents. Sentences can belong to three different classes: *First part of the contradiction*, *Second part of the contradiction* and *rejection class*. Sentences, in the first part of the contradiction, contain parameters that are improved by the application of a *solution* (often called a *partial solution* in TRIZ theory). The parameters that are degraded by the application of this partial solution are contained in sentences labeled Second part of contradiction. Finally, the sentences containing no evaluation parameters are assigned to the rejection class. The patents were labeled by a team of human experts. Details on the average number of sentences and labels in the dataset can be found in Table 1. Here is an example of an annotation with the patent US6938300B2:

This invention relates to a stroller, and more particularly to a wheel assembly for a stroller, which includes a single wheel.

...

In each of the front wheel assemblies 11, since the forward force A is located midway between two frictional forces B that are generated between the ground and the front wheels 13 and since the direction of the forward force A is parallel to those of the frictional forces B, the stroller 1 can advance along a straight path 16. When the stroller 1 moves over a lawn or uneven road surfaces, it is necessary for the stroller wheels to have a large diameter so as to ensure the comfort of the baby. ***However, if each of the front wheel assemblies 11 has two large-diameter front wheels 13, the total volume and weight of the stroller 1 will increase significantly so that it is difficult to push the stroller 1.***

In this patent one can spot a first parameter in the sentence in capital letters, comfort, which is improved by the use of large diameter wheels. Another parameter is then degraded, the ability to push the stroller, which belongs to the second part of the contradiction, in bold.

Several sentences containing the same parameters of the contradiction may be found in a patent. In such cases, all these sentences are extracted. Thus, the conflicting parameters can possibly be extracted from several different pairs of sentences. Finally, if a sentence contains both parts of the contradiction, i.e. both parameters in the contradiction, it is classified as "First part of the contradiction" but also as "Second part of the contradiction". Patents do not necessarily contain a contradiction, either because the writer does not give the necessary information or because the patent is not a solution to a contradiction. It is therefore necessary to filter the patents through the prism of the presence or absence of a contradiction. This analysis at the document level is made possible by selecting 1600 patents that do not contain a contradiction. This makes it possible to train a "filter" model on the patents. The dataset containing both contradiction and no contradiction patents can be accessed via this link.

4.2. Sentence-level analysis: SummaTRIZ model

Mining contradictions requires modeling the relationships between sentences, which has motivated the use of an extractive summarization model. Extractive summarization allows selecting all sentences that contain the document's salient information. Unlike a common summary, a TRIZ summary is different since only the sentences that contain the parameters of the contradiction are kept. The similarity between these two summarization tasks is exploited to overcome the lack of data for the extraction of contradictions. Indeed, the dataset of 1600 patents alone is not sufficient to learn a model from scratch. One can therefore take advantage of transfer learning to pre-train the model on classical summarization tasks (CNN-DailyMail dataset) before specializing it on TRIZ summarization. The architecture of the model, called SummaTRIZ, is inspired by the summarization network of Liu and Lapata (2019) which constitutes the baseline summarization model.

Patents contain a large number of references to other patents, codes and quantities to name just a few. Tokenization is therefore a very important step in the processing of this information. The Standford CoreNLP tokenizer is used for this purpose. The contents are then preprocessed to comply with the constraints on BERT's inputs. After applying BERT's Wordpiece tokenizer, the token embeddings, the segment embeddings and the position embeddings are summed. The token embeddings are simply the indices of the tokens in the Wordpiece dictionary (which contains over 30,000 words). The segment embeddings are used to identify the different sentences. Finally, the positional embeddings provide information on the relative positions of the tokens in the sentence. Segment and positional embeddings are learned by BERT and are therefore automatically generated at the input of the network. Special tokens are also added. A classification token CLS is inserted at the first position. This token is used for document classification. Finally, between each sentence, a special token called SEP is inserted. In the particular case of this summary model, a special CLS token is inserted between each sentence as well. Bert's output representations of these special tokens are the sentence representations. These representations are then used for the decision over the presence or absence of contradictory parameters in the sentences.

BERT is used to generate contextual representations of tokens and sentences in our case via CLS tokens. The advantage of encoders based on the attention mechanism is the ability to model dependencies between tokens of different sentences with a limited influence of the input document's size on the quality of the dependencies. In the case of LSTMs, for example, the length of the input will have a negative effect on the quality of the dependency modeling. The attention mechanism introduced in Vaswani et al. (2017b) allows for a better understanding of the internal structure of content from projections between searched information and currently available information or within current information (Self-attention). The attention mechanism is defined as follows: with Queries Q (what information the layer needs), Keys K and Values V (what information it actually has):

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d}})V \quad (1)$$

with d a rescale factor linked to the dimension of the input. In the case of self-attention, Queries are identical to Keys and Values. These modeled inter-tokens links allow the integration of contextual information in the vector output from BERT. In the case of the TRIZ summary, the aim is to integrate information on certain sentence structures such as "Nevertheless this leads to the reduction of" which will contain the parameters of the contradiction.

A global attention layer on top of BERT helps in the search for contradiction since links between sentences can be established. This also allows to free oneself from the constraint of BERT input's maximum length (512 tokens) since this attention allows to apply BERT to several input sequences and then process them all together in this layer. We

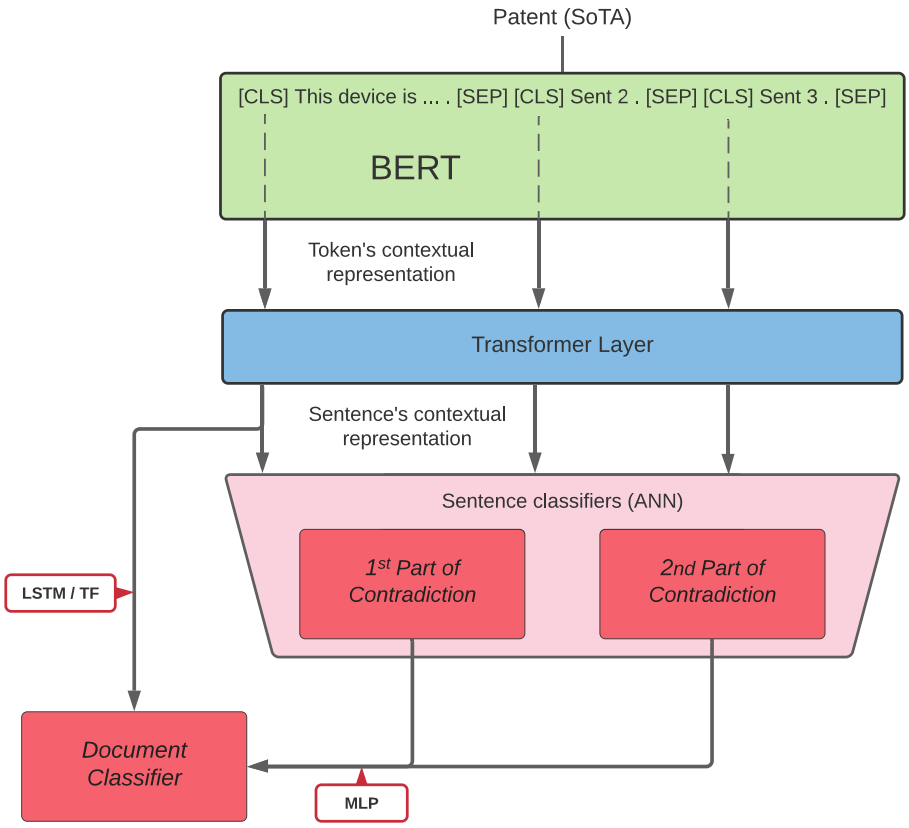

Fig. 2. Improved SummaTRIZ (Guarino et al., 2020) with a document classifier.

then set the input size limit at 1500 since 75% of the patents are shorter than 1500 tokens.

Two MLP (Multi-Layer Perceptron) classifiers are used to decide whether the sentences belong to the contradiction (first part or second part). The choice of a classifier per part of a contradiction is based on the possibility that a sentence can be both the first and second part of a contradiction. This situation is difficult to manage with a single classifier since, as this situation is rare in the dataset, the classes would be very unbalanced.

The addition of CLS tokens for each sentence and the addition of the global attention layer require significant training. This is why the model is pre-trained to summarize news articles with the CNN/DailyMail dataset. The model learns to generate and combine information at the word and sentence level. It is then fine-tuned (Guarino et al., 2021) on extracting contradictions with the 1600 labeled patents.

4.3. Improved SummaTRIZ for patent document analysis

The baseline model provides sentence-level analysis via the classification of the First part of contradiction and the Second part of contradiction. However, this information remains difficult to interpret when it comes to determining whether or not there is a contradiction in the document. Indeed, it ranks sentences for both parts of the contradictions but does not clearly indicate when both parts constitute a plausible contradiction. It, therefore, appears necessary to add to this sentence analysis a more global analysis of the document in order to validate the extraction of sentences (validate when there is a contradiction to mine and reject when there is not).

Several different approaches have been explored to validate the extraction. The approaches include: a probabilistic model, a transformer, a recurrent network, and a Multi-Layer Perceptron (see Fig. 2).

The probabilistic model is based solely on the probabilities resulting from the classification of the n sentences contained in the document. In other words, we consider the sentence with the maximum probability $max_{1\leq i\leq n} P_{c1}(S_i)$ to belong to the First part of the contradiction and the sentence having the maximum probability $\max_{1\leq i\leq n} P_{c2}(S_i)$ of belonging to the Second part of the contradiction are sufficient for the decision. Thus, the probability that the document contains a contradiction is evaluated as follows:

$$P_c(S_1 ... S_n) = \max_{1\leq i\leq n}(P_{c1}(S_i)) * \max_{1\leq i\leq n}(P_{c2}(S_i)) \quad (2)$$

This approach is naïve because it does not allow to link sentences in order to establish the presence of contradiction. This is why feature-based approaches seem more relevant.

Models based on a recurrent network and a transformer use the whole document to conclude whether or not contradictions are present. However, the recurrent network shows limitations when the length of the document increases, since the *state vector* is not sufficient to retain all the features characteristic of a contradiction. The transformer does not have this limitation, but the dilution of key information in the whole text still makes the decision difficult.

This is why taking advantage of the sentence classification already established by the sentence classifiers makes sense in order to select only the decisive clues on the presence of a contradiction. In this case, the representations of the two sentences showing the maximum probabilities of belonging to the "First part of contradiction" and the "Second part of contradiction" are fed into a Multi-Layer Perceptron which predicts whether the document contains a contradiction (see Eq. (3)). The input to this module is therefore dependent on the sentence classifiers, and the selected sentences are therefore potentially incorrect. However, this forces the encoder to incorporate information about the contradiction into each of the sentence representations.

For a fair comparison between variants and the baseline model, we add a document classifier (MLP) on top of the summarization baseline. This model will be called $SummaTRIZ_B$.

$$P_c(S_1 ... S_n) = MLP(\arg\max_{1\leq i\leq n}(P_{c1}(S_i)), \arg\max_{1\leq i\leq n}(P_{c2}(S_i))) \quad (3)$$

Table 2
Details on the parameters dataset.

Patents	$PE-PE/doc$	$PA-PA/doc$	$Avg.\ words_{PE}$	$Avg.\ words_{PA}$
1093	8719–7.98	1651–1.51	3.79	3.29

To limit the impact of the lack of labeled data, semi-supervised learning with a Generative Adversarial Network (GAN, Goodfellow et al. (2014)) is implemented (Guarino et al., 2021). This semi-supervised learning consists of generating vector representations of sentences from noise using a recurrent network (LSTM). The sentence classifiers are trained to distinguish the representations generated by the recurrent network from those of real unlabeled patents. As the prediction error is back-propagated through SummaTRIZ's encoder, the encoder will be pushed to modify the representations it generates for them to be as different as possible from those generated by the recurrent network. The quality of the encoder's representations will, thus, increase. This will make the decision on the presence of contradictions much easier.

5. TRIZ's parameter mining

This section details the data and models used for extracting TRIZ parameters from patent sentences. A dataset is created for this purpose. The developed models are also being described. In particular, Conditional Random Fields (CRFs), coupled with deep neural networks, are modified to integrate syntactic information.

5.1. Dataset

A contradiction is defined by two evaluation parameters but also by an action parameter. The aim of the dataset is, therefore, to build a classification model to mine both of these entities. The classification of the evolution of the parameters (positive or negative) is not useful since this model is destined to be associated with the summarization model of contradiction sentences. The summarization model already predicts the evolution of the parameters contained in the sentences (improved if the sentence is the "First part of contradiction" and degraded if the sentence is in the "Second part of contradiction"). Therefore, a model to extract all parameters, whatever their evolution, is sufficient to retrieve both contradictory parameters and their evolution. Almost 9000 parameters are labeled from 1100 USPTO patents of all technical domains. The dataset is available here. The labeling was performed by a human expert. Details on the dataset can be found in Table 2. An example of annotation with patent US7010885B2 is shown below (action parameters are in high-case letters, evaluation parameters in bold):

Known barriers, however, are unsatisfactory for a variety of reasons. Sealing plugs, which were a step forward over other barriers, utilize snap-fit clips to ***hold the plug in place****, i.e., in an orifice of a panel member. However, snap-fit clips on a sealing plug, without more, are insufficient because the clips cannot* ***produce a contaminant-tight seal*** *between the plug and the panel member. To overcome this, a sealer material, such as compressible rubber, adhesive, caulk or mastic, has been used in combination with a carrier to form the sealing plug. The sealer material may create a* ***contaminant-tight seal*** *between the carrier and the panel member.*

...

Thus, the SIZE OF THE BARRIER *must be closely matched to the size of the orifice to ensure that there are no* ***gaps between the carrier and the panel member****. Therefore,* ***expensive precision manufacturing techniques*** *are required in the formation of the orifice and the carrier to ensure that the barrier* ***cannot be installed incorrectly****, i.e., off-center. Consequently, the inventor hereof has recognized a need for a physical barrier that overcomes one or more of these problems.*

5.2. Conditional random field

A CRF (Lafferty et al., 2001) is a statistical model to model the relationships between neighboring variables. In the case of a classification task, the idea is to model the conditional probabilities $P(Y_k|X)$ with Y_k the labels and X the observations which will be, in this case, a set of features extracted from a textual content. A linear chain CRF is used in this study whose graph representation is shown in Fig. 3. Each label and observation are nodes on the graph. The edges show the dependencies between variables. Each label will thus depend on the current observation as well as on the preceding and following labels. This is the Markov property: $P(Y_k|X, Y_v, k \sim v) = P(Y_k|X, Y_v, k \sim v)$ with $k \sim v$ meaning that k and v are neighbors in the graph.

Let Y be a sequence of l labels. Let X be the sequence of l corresponding observations. The computation of $P(Y|X)$ is deduced from each label and observation of the sequence (considering that the labels are independent of their neighbors at first):

$$\begin{aligned} P(Y|X) &= \prod_{k=0}^{l-1} P(Y_k|X_k) \\ &= \prod_{k=0}^{l-1} \frac{\exp(U(X_k, Y_k))}{Z(X_k)} \\ &= \frac{\exp(\sum_{k=0}^{l-1} U(X_k, Y_k))}{Z(X)} \end{aligned} \tag{4}$$

with $Z(X)$, the partition function, i.e. the normalization factor computed from the sum of all possible numerators (for each possible label sequence). $P(Y_k|X_k)$ is therefore modeled with a normalized exponential as in a classical softmax output of a neural network.

If one introduces a dependency between labels, i.e. the $k+1$th label depends on the kth label, one can rewrite $P(Y|X)$ by adding a term linking the successive labels:

$$\begin{aligned} P(Y|X) &= \prod_{k=0}^{l-1} \frac{\exp(U(X_k, Y_k))\exp(T(Y_{k+1}, Y_k))}{Z(X_k)} \\ &= \frac{\exp(\sum_{k=0}^{l-1} U(X_k, Y_k) + \sum_{k=0}^{l-2} T(Y_{k-1}, Y_k))}{Z(X)} \end{aligned} \tag{5}$$

with T a matrix giving the transition potentials (called pairwise potentials) between the labels. This matrix T is called the transition matrix. The pairwise potentials $T(Y_{k-1}, Y_k)$ refer to the likelihood that label Y_k is followed by label Y_{k+1}. When we associate a neural network with a CRF, the unary potentials $U(X_k, Y_k)$ are given by the last layer of the neural network. These potentials refer to the likelihood that the label would be Y_k given X_k. The goal is then to maximize $P(Y|X)$ with respect to the parameters of the neural network and the parameters contained in the transition matrix which are learnable as well.

5.3. ParaBERT model

5.3.1. Baseline

The parameters consist of one or more words. The extraction of parameters is therefore similar in form to a Named-Entity Recognition task. The labels are therefore adapted in accordance with the BIO policy with B (Begin) for the parameter start token, I (Interior) for parameters consisting of a single token or for tokens belonging to parameters and located after the start token, and O (Out) for tokens not belonging to a parameter. BERT provides contextual representations of each token. A softmax classifier with five output neurons (B and I for the action parameters and for the evaluation parameters, and O for the other tokens) predicts the label of each token. As the parameters are, a priori, not case-dependent, the "no-cased" BERT-Large model is used.

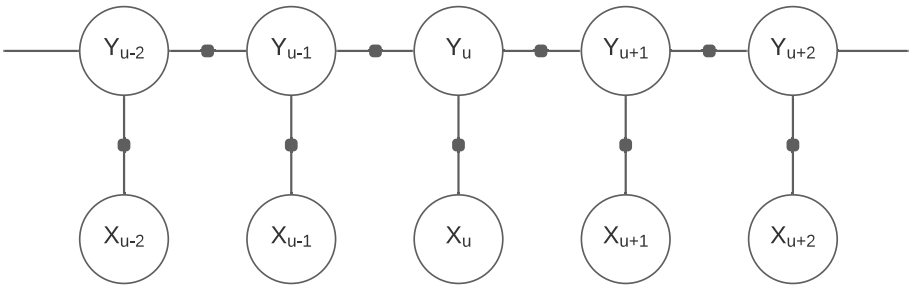

Fig. 3. Linear chain Conditional Random Field (CRF).

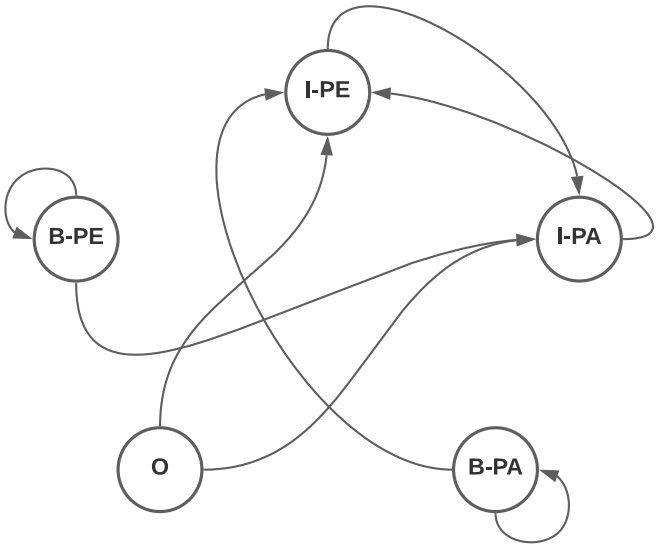

Fig. 4. Impossible label transitions.

5.3.2. Constrained CRF

Several transitions between labels are impossible, such as EP-B/EP-B or EP-B/AP-B. Indeed, since the action and evaluation parameters are nominal groups, they cannot be placed consecutively; they have to contain one verb in between. These dependencies are not easily modeled with a linear classifier. This is why a *Conditional-Random-Field* (CRF) is used. A CRF has a transition matrix that stores the information learned from these dependencies (cf. Part 5.2). As the impossible transitions are known, this transition matrix is initialized in a manner to respect the diagram of impossible transitions shown in Fig. 4. The pairwise potentials corresponding to impossible transitions are initialized with a high negative value in log space which corresponds to a probability of transition close to 0.

5.3.3. Syntax-based CRF

The evaluation and action parameters follow syntactic patterns which are dependant on the patents' domain and writer. This explains the failure of pattern-based methods for their extraction. However, a few sentence constructions occur regularly, such as "The modification of this + AP + allows to improve + EP". Taking into account the syntactic structures can therefore bring additional information on the possible labels. Indeed, in the previous example, even without having read the EP, we can easily make the hypothesis that there will be an EP after the infinitive verb "to improve".

Part-Of-Speech (POS) tagging is a very common task in NLP which aims to associate each word with its grammatical class. POS tagging being a popular task, very efficient models already exist : Brown et al. (2020), Irie et al. (2019), Melis et al. (2020) and Radford et al. (2019). Therefore, no models are re-trained for this part. In order not to increase the inference time, a relatively fast tagger is chosen from the spacy library. Indeed, a very high tagging quality does not appear necessary to recognize common syntactic structures.

Two different approaches are introduced to take into account the POS tagging information in the classification of tokens. The transition matrix gives the transition probabilities from the L_t label assigned to token t to L_{t+1}, label of token $t+1$. In a classical CRF, this matrix is unique. A new matrix structure is presented in this section to model label dependencies while taking into account syntactic information. Two configurations are presented. The first configuration takes into account the label of the considered token as well as the tokens which directly precede and follow the considered token. Three POS tags are therefore used. The second configuration takes into account the label of the considered token as well as the two preceding and following labels. In this case, five POS tags are exploited.

A first category of models (Fig. 5), assumes that the transition matrix should be dependent on the syntactic structure. Indeed, if a token is found in a common grammatical structure for an evaluation parameter, for instance, the probability of transition to the EP label should be increased. Conversely, if the structure has little chance of being associated with an evaluation parameter this transition probability should decrease. A transition matrix is then initialized for each possible configuration of part-of-speech tags. For each label prediction, only the matrix corresponding to the series of three or five tags (depending on the configuration) is used. A tensor of dimension (N_P, N_P, N_P, N_C, N_C) or (N_P, N_P, N_P, N_P, N_P, N_C, N_C) with N_P the number of Part of Speech classes and N_C the number of labels for parameters mining task is used to index the transition matrices. In the example presented in Fig. 5 we assume that a series of three POS labels C_u, C_v, C_w carry the syntactic information. This series of tags will directly correspond to a transition matrix at position (u,v,w) in the tensor presented above.

A second category of models (Fig. 6) aims at reducing the number of transition matrices. The principle is to initialize a constant (and small) number N of transition matrices T and to create a pointing mechanism towards the most adapted transition matrix from the series of parts of speech returned by the tagger. Thus, a few parameters are added and the transition matrices model the most emblematic cases only for the transition. The first step is the encoding of the combination of parts of speech tags. An encoding matrix E is therefore introduced. Hadamard products between the tags' one hot matrices (one hot vector for the POS tag with an additional dimension related to the position in the tag sequence (0,1,2) or (0,1,2,3,4)) and the encoding matrix allows the creation of an embedding V_{emb} containing the information on the tags and their position:

$$V_{emb} = \sum_j \sum_i E \odot \delta_i \delta^T_{j=tag_i} \tag{6}$$

with i the position in the tag sequence (from 0 to 2 if three tags are used for instance), j the POS class and tag_i the POS class of ith tag.

V_{emb} then passes through a fully-connected neural network (FC):

$$V'_{emb} = FC(V_{emb}) \tag{7}$$

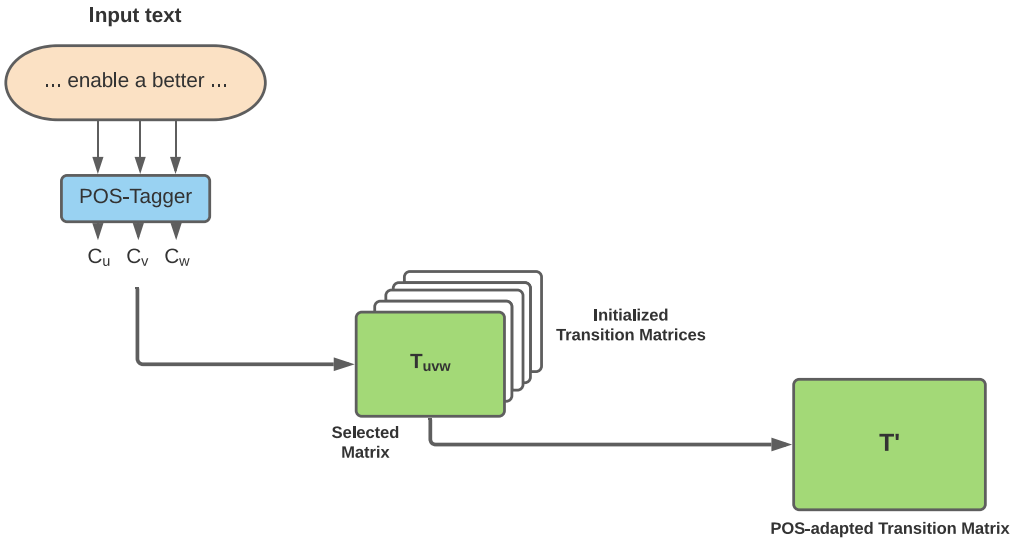

Fig. 5. Multiple indexed transition matrices for POS information integration.

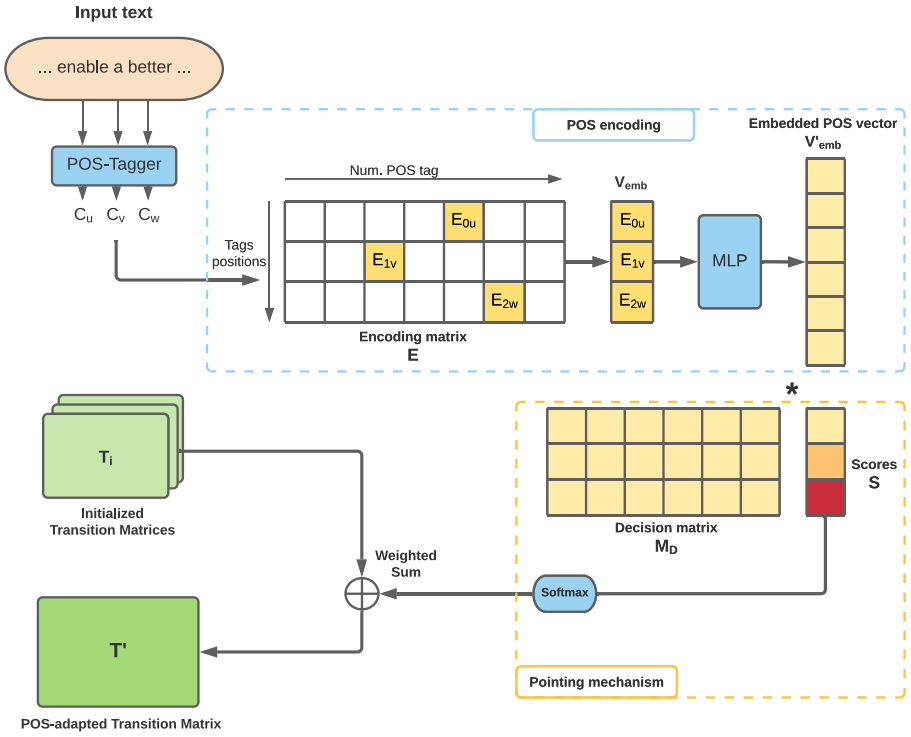

Fig. 6. Pointing structure to integrate POS information in label.

The product between a decision matrix M_D and the embedding V'_{emb} is then performed (Eq. (8)). Line i of M_D represents the "kind" of embedding that should be processed with transition matrix T_i. The product of M_D and V'_{emb} is therefore a scalar product between the targeted embeddings (meaning the embeddings meant to be processed by one of the initialized transition matrices) and the actual embedding. The product results in a vector S containing the scores associated to each transition matrix. A maximum score at position i means that the ith targeted embedding is the closest to the actual embedding V'_{emb}. It implies that V'_{emb} should be processed with the ith transition matrix.

$$S = M_D V'_{emb} \tag{8}$$

In order for the gradient to be back-propagated through all the transition matrices' parameters, M_D and E, the choice of the new transition matrix T' is modeled by a weighted sum of the matrices by the score vector (after application of a softmax function):

$$S' = Softmax(S) \tag{9}$$

$$T' = \sum_{i=0}^{N} S'_i T_i \tag{10}$$

with T_i the ith initialized transition matrix, S'_i the score associated to matrix T_i and N the number of transition matrices chosen by the user.

The transition matrix T' will thus be unique for each sequence of tags but it will be very close to one of the existing matrices thanks to the application of the softmax function on the scores which puts the emphasis on the chosen matrix.

6. Experiments

In this section, the results of the selection of sentences containing the contradictory parameters through the TRIZ summarization model are presented. The results of the parameter extraction module are discussed as well. 4-fold cross-validation was performed for both models.

6.1. TRIZ summarization

6.1.1. Metrics

The usual classification metrics (Accuracy, Precision, Recall) are used to evaluate the performance of the model at the sentence and document level. However, new metrics, more adapted to the extraction of contradictions, need to be defined. For the sentence-level analysis, the S metric allows evaluating the ability of the model to correctly rank sentences for the first and second parts of the contradiction. A labeled document includes n_1 sentences for the first part of the contradiction and n_2 sentences for the second part of the contradiction. The metric S for a labeled document is defined as the number of correct sentences for the first/second part of the contradiction within the n_1/n_2 sentences with maximum probabilities for the *first/second part of contradiction*.

We call $E_1 = \{S_{10}...S_{1n_1}\}$ the set of the n_1 sentences labeled as the *first part of the contradiction* and $E_2 = \{S_{20}...S_{2n_1}\}$ the set of the n_2 sentences labeled as the *second part of the contradiction*. Each pair $S_{1i}S_{2j}$ forms a contradiction. We consider that a contradiction is extracted if:

$$\arg\max_{1\le i\le n}(P_{c1}(S_i)) \in E_1 \tag{11}$$

$$\arg\max_{1\le i\le n}(P_{c2}(S_i)) \in E_2 \tag{12}$$

$$P_c(S_1...S_n) > P_{threshold} \tag{13}$$

with $P_{c1}(S_i)$ the probability that sentence i is the first part of the contradiction, $P_{c2}(S_i)$ the probability that sentence i is the second part of the contradiction and $P_c(S_1...S_n)$ the probability that there is a contradiction to mine.

A first metric CO_F evaluates the ability to extract a correct sentence at the same time for both parts of the contradiction (i.e. the first two conditions). A second metric CO_V evaluates if a contradiction is fully extracted, meaning it is also validated by the document classifier (i.e. all the above conditions are respected).

6.1.2. Results

Results are shown in Tables 4–6. $SummaTRIZ_B$ is the baseline model, without adversarial training, and a Multi-Layer Perceptron document classifier. The other four models are variations of SummaTRIZ with different document classifiers (probabilistic, Multi-Layer Perceptron, Long Short Term Memory, and Transformer), with adversarial training. Details on the different models are shown in Table 3 (TL refers to Transfer Learning using CNN/Daily Mail dataset as explained in Section 4.2, Adv. Train. refers to Adversarial Training as mentioned in Section 4.3). The adversarial training significantly improves the results with an increase in the S metric of 8% and 6% respectively for the first and second part of the contradiction. The number of contradictions found (CO_F) increases by 14.5%. The number of validated contradictions also strongly increases (+23.3%).

The performance differences between the different document classifiers are small except for the probabilistic approach, which performs even worse than the baseline $SummaTRIZ_B$. This is due to the very strong starting hypothesis. The only metric that really separates the other document classifier architectures is the validation of contradictions (CO_V). The Multi-Layer Perceptron allows validating the most extractions, with almost 2% more than the LSTM and 4% more than the Transformer. It is therefore the $SummaTRIZ_{MLP}$ model that is selected for the extraction of contradictory sentences.

Table 3
Summary of tested models.

Model	Doc. classifier	Doc. classifier input	TL	Adv. Train.
B (Liu & Lapata)	MLP	Contradiction sent.	Yes	No
$PROB$	None	Contradiction sent.	Yes	Yes
MLP	MLP	Contradiction sent.	Yes	Yes
$LSTM$	LSTM	Full document	Yes	Yes
TF	Transf.	Full document	Yes	Yes

Table 4
Sentence classification: *First Part of Contradiction*.

Model	Loss	Acc.	Prec.	Recall	F1 score	Support
SummaTRIZ $_B$ Liu & Lapata (2019)	0.115	0.97	0.54	0.25	0.35	1098
SummaTRIZ $_{PROB}$	**0.112**	0.97	0.56	0.25	0.35	1168
SummaTRIZ $_{MLP}$	**0.112**	0.97	0.56	0.23	0.33	**1187**
SummaTRIZ $_{LSTM}$	**0.112**	0.97	**0.58**	0.22	0.32	1186
SummaTRIZ $_{TF}$	0.113	0.97	0.52	**0.29**	**0.37**	1143

Table 5
Sentence classification: *Second Part of contradiction*.

Model	Loss	Acc.	Prec.	Recall	F1 score	Support
SummaTRIZ $_B$ Liu & Lapata (2019)	0.129	0.96	0.68	0.50	0.57	2500
SummaTRIZ $_{PROB}$	0.120	**0.97**	**0.69**	0.56	0.62	2619
SummaTRIZ $_{MLP}$	0.120	**0.97**	0.67	0.62	0.64	2626
SummaTRIZ $_{LSTM}$	**0.118**	**0.97**	0.67	**0.63**	**0.65**	**2645**
SummaTRIZ $_{TF}$	0.121	0.96	0.65	**0.63**	0.64	2631

Table 6
Document classification.

Model	Loss	Acc.	Pre.	Recall	F1 score	CO_F	CO_V
SummaTRIZ $_B$ Liu & Lapata (2019)	0.502	0.76	0.74	0.80	0.77	580	467
SummaTRIZ $_{PROB}$	–	0.57	0.73	0.21	0.33	**668**	192
SummaTRIZ $_{MLP}$	**0.466**	0.78	**0.76**	0.83	0.79	666	**576**
SummaTRIZ $_{LSTM}$	0.481	0.77	0.73	**0.86**	0.79	654	567
SummaTRIZ $_{TF}$	0.467	**0.79**	**0.76**	0.84	**0.80**	648	552

6.1.3. Conclusion on TRIZ summarization

An adversarial training of the model allows to build richer representations, which will integrate better quality information and be more easily interpretable by sentence and document classifiers. A global attention layer is added to refine the representation of the generated sentences and, at the same time, to analyze the content of greater length than the traditional BERT limit of 512 tokens. The validation of the extraction is then ensured by the document classifier that predicts the presence or absence of a contradiction in the document.

6.2. Parameters mining

Once the sentence analysis is performed, the contradictory parameters must be extracted from the selected sentences in order to obtain the final formulation of the contradiction with an improved and degraded parameter. The results of the parameter extraction will be presented in this section.

6.2.1. Results

Results are shown in Tables 8 and 9. The constrained-CRF models, evaluated in Tables 8 and 9 contain a cs suffix. The configurations which take into account three POS labels sequences are marked with indice 1. The configurations working with five POS labels sequences are marked with an indice 2. The models marked with * refer to the model presented in Fig. 5. The models marked with * refer to the pointing mechanism (Fig. 6). All this information are gathered in Table 7. The baseline model is unsurprisingly the least efficient since it takes less

Table 7
Tested models' characteristics for parameters extraction.

Model	Classifier	POS size	Pointer	Constrained
Baseline (Devlin et al., 2019)	MLP	None	No	No
$Baseline_{CRF}$ (Lafferty et al., 2001)	CRF	None	No	No
$ParaBERT_{CRF-cs}$	CRF	None	No	Yes
$ParaBERT^{*}_{CRF1}$	CRF	1	No	No
$ParaBERT^{*}_{CRF1-cs}$	CRF	1	No	Yes
$ParaBERT^{*}_{CRF2}$	CRF	2	No	No
$ParaBERT^{*}_{CRF2-cs}$	CRF	2	No	Yes
$ParaBERT^{**}_{CRF1}$	CRF	1	Yes	No
$ParaBERT^{**}_{CRF1-cs}$	CRF	1	Yes	Yes
$ParaBERT^{**}_{CRF2}$	CRF	2	Yes	No
$ParaBERT^{**}_{CRF2-cs}$	CRF	2	Yes	Yes

Table 8
Results for evaluation parameters (EP) mining.

Model	Loss	TP	Prec.	Recall	F1	Support
Baseline (Devlin et al., 2019)	0.43	3680	31.7	42.4	36.2	8694
$Baseline_{CRF}$ (Lafferty et al., 2001)	0.393	3870	36.9	44.5	40.3	8694
$ParaBERT_{CRF-cs}$	0.137	3893	47.5	44.8	46.1	8694
$ParaBERT^{*}_{CRF1}$	0.299	3643	32.4	41.9	36.5	8694
$ParaBERT^{*}_{CRF1-cs}$	0.141	3867	47.3	44.5	45.9	8694
$ParaBERT^{*}_{CRF2}$	0.293	3568	27.7	41.0	33.1	8694
$ParaBERT^{*}_{CRF2-cs}$	0.134	3753	46.6	43.2	44.8	8694
$ParaBERT^{**}_{CRF1}$	0.390	**4079**	46.3	**46.9**	46.6	8694
$ParaBERT^{**}_{CRF1-cs}$	0.149	3890	47.4	44.7	46.0	8694
$ParaBERT^{**}_{CRF2}$	0.420	4034	46.0	46.4	46.2	8694
$ParaBERT^{**}_{CRF2-cs}$	0.149	4010	**48.6**	46.1	**47.3**	8694

into account the sequential nature of the labels than with the addition of a Conditional Random Field at the output. As the loss computation is different between the baseline model and the CRF-based models, the comparison can only be made on the metrics. The difference between the baseline CRF model and the baseline is very clear both for the Evaluation Parameters (EP) and the Action Parameters (AP). For the F1-score metric, the difference between the baseline and the CRF model is about 10% for EPs and no less than 26% for APs. Thus, the use of a CRF has a positive influence on the extraction of TRIZ parameters.

The addition of constraints on the transitions in the CRF (Part 5.3.2) has a high influence on the loss since it is divided by 3 between the CRF model and the same model with constraints. The influence of the constraints is also clear on the metrics, with a significant increase in precision (30% for the PE and 85% for the PA). The recall is also improved. These differences are explained by the still significant number of impossible transitions predicted by the CRF model. With all these errors removed, the precision logically increases significantly.

Taking into account syntactic information has a variable benefit depending on the chosen configuration. When a transition matrix is declared for each possible sequence of Part of Speech tags (models with *), whether for sequences of 3 (models with a 1) or 5 tags (models with a 2) and with or without constraints (cs suffix) the models perform worse than with a single transition matrix. A global decrease in the metrics is observed (−9% on the F1-score for the EP between $ParaBERT^{*}_{CRF1}$ and $Baseline_{CRF}$ or −0.4% on the F1-score between $ParaBERT^{*}_{CRF1-cs}$ and $ParaBERT_{CRF-cs}$). Some losses or metrics, notably for PAs, show some improvement but this remains marginal compared to the significant decreases in the metrics related to EPs. These more than nuanced results can be explained by the extremely large number of initialized transition matrices of the order of 5000 for configurations 1 and more than a thousand for configurations 2. These matrices are therefore used very rarely, some probably never, and therefore cannot be trained correctly. It, therefore, seems justified to introduce a mechanism of pointing to transition matrices allowing to initialize much fewer transition matrices (8 for these experiments) but always functional whatever the sequence of tags. These models (**) show, indeed, clear improvements in the losses (−25% for the models without constraints, -3% for the models with constraints) and metrics associated with EP (+15% for the models without constraints, up to +3% for the models with constraints) and AP with (+44% for the models without constraints, up to +6% for the models with constraints). Some metrics show slight decreases such as −0.2% between the F1 score of the constrained CRF model and that of the $^{**}_{CRF1-cs}$ model but this remains anecdotal. This pointer thus allows a real improvement of the results while limiting the number of parameters.

Table 9
Results for action parameters (AP) mining.

Model	TP_{AP}	$Prec_{AP}$	$Recall_{AP}$	$F1_{AP}$	$Support_{AP}$
Baseline (Devlin et al., 2019)	183	19.3	11.1	13.7	1651
$Baseline_{CRF}$ (Lafferty et al., 2001)	265	23.5	16.0	18.7	1651
$ParaBERT_{CRF-cs}$	308	43.4	18.7	26.0	1651
$ParaBERT^{*}_{CRF1}$	186	19.1	11.3	14.1	1651
$ParaBERT^{*}_{CRF1-cs}$	327	40.1	19.9	26.5	1651
$ParaBERT^{*}_{CRF2}$	193	13.1	11.6	12.1	1651
$ParaBERT^{*}_{CRF2-cs}$	266	35.6	16.2	22.2	1651
$ParaBERT^{**}_{CRF1}$	**393**	37.4	**23.8**	**29.0**	1651
$ParaBERT^{**}_{CRF1-cs}$	308	**46.5**	18.7	26.6	1651
$ParaBERT^{**}_{CRF2}$	381	37.8	23.1	28.7	1651
$ParaBERT^{**}_{CRF2-cs}$	330	43.9	20.0	27.5	1651

6.2.2. Conclusion on parameter extraction

The best performing model for parameter extraction is thus based on BERT and a CRF. To take advantage of the Part of Speech sequences and thus of the relationship between a grammatical structure and the presence of EP/AP, a new CRF model is introduced. This model includes a varying transition matrix determined thanks to the sequence of part-of-speech tags in the neighborhood of the token under study. A pointing mechanism is adopted to guarantee an adaptation of the transition matrix without adding too many parameters to the model. This mechanism also guarantees better training of the transition matrices since the gradient is back-propagated in the whole set of matrices for each inference. It is worth noting that the metrics for parameter extraction artificially decrease because of the length of the description of the parameters. Indeed, a couple of words can be added or removed from the description of the parameters without affecting the quality of the extracted information. This makes the metrics decrease while the parameters are still correctly mined.

6.3. Visualization of results

Visualization in the form of links of contradictions between sentences is proposed in the demonstrator. This principle is based on the fact that even if the document classifier validates that the document contains a contradiction, this contradiction may not be extracted. Indeed, the two selected sentences may not contain the parameters of the contradiction. Thus, a particularization of the results for each pair of sentences seems to be a good solution. The idea is to predict, from all the available results, the probability that each pair of sentences forms a contradiction. The inputs of this last visualization module are five probabilities: P_{s_1}, P_{s_2} the probabilities of belonging to the TRIZ summary for the two selected sentences, P_{doc} probability that the document contains a contradiction, and P_{p_1}, P_{p_2} probabilities that the two selected sentences contain a parameter. A sigmoid classifier then

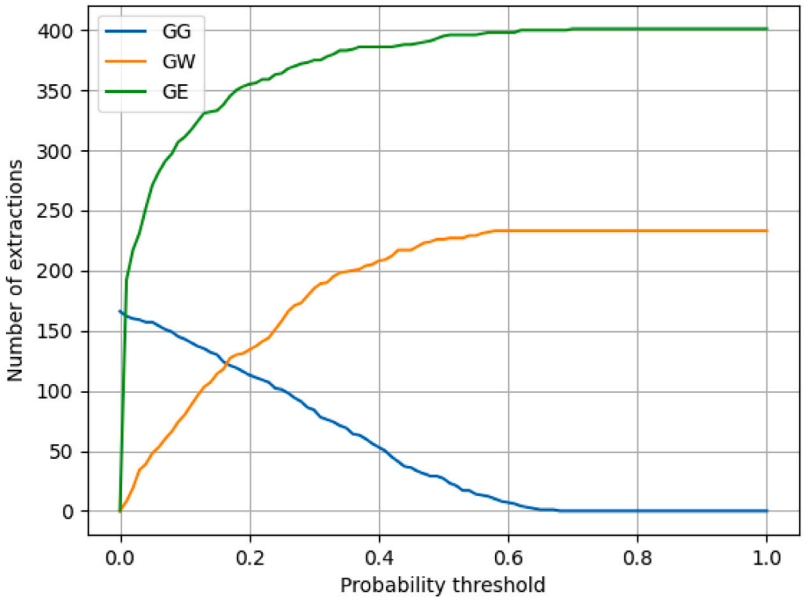

Fig. 7. Influence of the output threshold on contradiction metrics.

gives the probability that the pair of sentences and parameters gives a contradiction. This classifier is trained on the most credible (in terms of contradiction) pair of sentences for each document. Indeed, if one wants to learn the model on all possible pairs of sentences, the data is extremely unbalanced with a ratio of about $1/30$ between positive (contradiction) and negative (no contradiction) examples. The results are shown in Fig. 1. Three metrics are introduced: GG, GW, GE. GW and GG refer to the 400 documents of the test set containing a contradiction. GW is the number of documents where the sigmoid classifier predicts an absence of contradiction when the sentence pair is indeed not a contradiction i.e.:

$$P_c(S^{\bar{c}}_{1,C}, S^{\bar{c}}_{2,C}) < thresh \tag{14}$$

with $(S^{\bar{c}}_{1,C}$ and $S^{\bar{c}}_{2,C})$ two sentences from a document containing a contradiction but which do not form a contradiction and thresh the chosen probability threshold. GG is the number of documents where the classifier predicts a contradiction when the two extracted sentences are indeed a contradiction i.e.

$$P_c(S^{c}_{1,C}, S^{c}_{2,C}) > thresh \tag{15}$$

with $(S^{c}_{1,C}$ and $S^{c}_{2,C})$ two sentences from a document containing a contradiction which do form a contradiction. GE refers to the 400 documents of the test set not containing a contradiction. GE is the number of documents where the classifier predicts an absence of contradiction for a couple of sentences from a document without contradiction i.e.:

$$P_c(S^{\bar{c}}_{1,\bar{C}}, S^{\bar{c}}_{2,\bar{C}}) < thresh \tag{16}$$

with $S^{\bar{c}}_{1,\bar{C}}$ and $S^{\bar{c}}_{2,\bar{C}}$ two sentences from a document which does not contain a contradiction and, therefore, does not form a contradiction. The goal is, therefore, to have all three metrics at their maximum. Fig. 7 presents the variations of the three metrics as a function of the threshold used in the output of the classifier to determine if the pair of sentences describes a contradiction. Logically, if the threshold is very high, the precision is maximal, whereas if the threshold is close to 0, the recall is maximal. A compromise must therefore be found between precision and recall during visualization. The threshold is thus set at 0.25 in the demonstrator.

6.4. Qualitative results

In this section, results from our demonstrator are shown. The test patents are available here.

Positive example 1: Patent EP0489335

Such materials are advantageous in that they have high thermal conductivity and thus allow the melt of thermoplastic resin to cool rapidly and shorten the molding cycle time. (EP: thermal conductivity, cool rapidly, shorten the molding cycle time, AP: /)

The quick solidification of the melt combined with limited flowability of the materials makes it difficult to achieve melt flow over a large area. (EP: quick solidification of the melt, flowability, melt flow over a large area, AP: /)

Positive example 2: Patent US5316377

For limited use or lightweight applications, such as with barbecue carts, lawn mowers, trash containers and many other devices, plastic wheels can serve the same purpose, but at relatively lower costs. (EP: limited use, lightweight applications, costs, AP: /)

Inherent negatives of such wheels, however, are that the core of the wheels are hollow and thus the wheels tend to be "noisy". (EP: core of the wheels are hollow, noisy, AP: /)

Negative example: Patent US5833916

The softness and thinness of the film, however, provides the molded product with insufficient surface properties such as wear resistance, scratch resistance, surface heat resistance, weatherability, light resistance, and chipping resistance. (EP: surface properties, wear resistance, scratch resistance, surface heat resistance, weatherability, light resistance, chipping resistance, AP: /)

The post-mold painting, however, is disadvantageous insofar as it requires additional time and labor. (EP: requires additional time and labor, AP: /)

In the positive example (Patent US5833916), the contradiction is well mined. Having a quick solidification of the melt will reduce the molding cycle time. But, if the material solidifies quickly, it will become difficult to mold large parts. The sentences are the right ones. The main parameters are also mined, but few are those that could be removed like "cool rapidly" which is a synonym for "shorten the molding cycle time" in terms of evaluation parameters. We could also mention that quick solidification is present in both sentences, which makes it hard to understand where the contradiction is. The

same limitation appears in patent US5316377 where the sentences are correct. There is a contradiction, in this case, between having cheap and light wheels and being silent. The parameters are also extracted but some of them are irrelevant as “limited use” or “core of the wheels are hollow” which is the cause of the noise parameter. In the negative example (Patent US5833916), two major limitations are visible. First, too many parameters are extracted, and this will make the system less relevant when comparing contradictions. Moreover, in the second part of the contradiction, the parameters are not specific enough to really understand what the contradiction is. In this case, this “high level” potential contradiction is difficult to interpret and it is actually also difficult to verify that it is a contradiction.

The summary model performs best when the sentences to be extracted are close. However, the addition of a paragraph at the beginning of a state of the art can disturb the model in the search for contradiction. A work on the explainability of this model could allow for a better understanding of the choice of sentences.

PaTRIZ is, therefore, able to mine meaningful information in terms of contradictions and parameters from patents of all technical domains.

7. Conclusion

A two-step process is proposed in this paper to extract contradictions from patents. First, the two sentences containing the improved and degraded parameters associated with the contradiction are selected via a summary model. Then, these parameters are extracted from the two pre-selected sentences. A dataset was created for each task. The summary model includes sentence classifiers but also a document classifier, allowing to filter the patents containing a contradiction from those without. The parameter extraction model, ParaBERT, is based on the combination of BERT (Devlin et al., 2019) and a new CRF structure built to include information about the syntactic structure of sentences.

PaTRIZ opens the door for an automated deep understanding of patents’ contents. This will first allow targeted searches for patents that would contain solutions to an initial contradiction. It would also allow automatic state of the art in a very short time. The summarization model is relatively efficient since it allows to extract one contradiction out of two. Nevertheless, the consistency of the output can still be improved. The pairs of sentences that are supposed to form contradictions are sometimes unrelated. A more advanced modeling of the contradiction links still needs to be studied. An improvement could also be the addition of constraints on the relative position of the sentences forming the contradiction since in practice they are often close (with 5–7 sentences maximum).

Future work includes developing a multi-task model including summarization and parameter sub-modules, using PaTRIZ to target patents and facilitate problem-solving, and adding explainability to the summarization model. Massive contradiction extraction, coupled with the claims, containing the solutions, could also help better understand the path leading to an invention.

CRediT authorship contribution statement

Guillaume Guarino: Methodology, Software, Investigation, Data curation, Resources, Writing – original draft. **Ahmed Samet:** Conceptualization, Methodology, Writing – review & editing, Supervision. **Denis Cavallucci:** Conceptualization, Data curation, Methodology, Validation, Writing – review & editing, Supervision, Project administration.

Declaration of competing interest

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.

Data availability

Data will be made available on request.

References

Altshuller, G. (1984). *Creativity as an exact science*. CRC Press, URL: https://books.google.fr/books?id=bUFZDwAAQBAJ.

Aone, C., Okurowski, M. E., & Gorlinsky, J. (1998). Trainable, scalable summarization using robust NLP and machine learning. In *ACL ’98/COLING ’98, Proceedings of the 36th annual meeting of the association for computational linguistics and 17th international conference on computational linguistics - Volume 1* (pp. 62–66). USA: Association for Computational Linguistics, http://dx.doi.org/10.3115/980845.980856.

Berduygina, D., & Cavallucci, D. (2020). Improvement of automatic extraction of inventive information with patent claims structure recognition. In *Science and information conference* (pp. 625–637). Springer.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems, Vol. 33* (pp. 1877–1901). Curran Associates, Inc., URL: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Cascini, G., & Russo, D. (2007). Computer-aided analysis of patents and search for TRIZ contradictions. *International Journal of Product Development*, *4*(1/2), 52–67, URL: https://ideas.repec.org/a/ids/ijpdev/v4y2007i1-2p52-67.html.

Chang, H.-T., Chang, C.-Y., & Wu, W.-K. (2017). Computerized innovation inspired by existing patents. In *2017 international conference on applied system innovation (ICASI)* (pp. 1134–1137). http://dx.doi.org/10.1109/ICASI.2017.7988268.

Chen, L., Xu, S., Zhu, L., Zhang, J., Lei, X., & Guancan, Y. (2020). A deep learning based method for extracting semantic information from patent documents. *Scientometrics*, *125*, 289–312. http://dx.doi.org/10.1007/s11192-020-03634-y.

Conroy, J., & O’leary, D. (2001). Text summarization via hidden Markov models. In *Proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 406–407).

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)* (pp. 4171–4186). Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/n19-1423.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In *NIPS’14, Proceedings of the 27th international conference on neural information processing systems - Volume 2* (pp. 2672–2680). Cambridge, MA, USA: MIT Press.

Guarino, G., Samet, A., Nafi, A., & Cavallucci, D. (2020). SummaTRIZ: Summarization networks for mining patent contradiction. In *2020 19th IEEE international conference on machine learning and applications (ICMLA)* (pp. 979–986). IEEE.

Guarino, G., Samet, A., Nafi, A., & Cavallucci, D. (2021). PaGAN: Generative adversarial network for patent understanding. In *2021 international conference on data mining*.

Irie, K., Zeyer, A., Schlüter, R., & Ney, H. (2019). Language modeling with deep transformers. In *Proc. interspeech 2019* (pp. 3905–3909). http://dx.doi.org/10.21437/Interspeech.2019-2225.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, *46*(5), 604–632. http://dx.doi.org/10.1145/324133.324140.

Kupiec, J., Pedersen, J., & Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 68–73). New York, NY, USA: ACM Press.

Lafferty, J. D., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML ’01, Proceedings of the eighteenth international conference on machine learning* (pp. 282–289). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., URL: http://dl.acm.org/citation.cfm?id=645530.655813.

Litvak, M., & Last, M. (2008). Graph-based keyword extraction for single-document summarization. In *MMIES ’08, Proceedings of the workshop on multi-source multilingual information extraction and summarization* (pp. 17–24). USA: Association for Computational Linguistics.

Liu, Y., & Lapata, M. (2019). Text summarization with pretrained encoders. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 3730–3740). Hong Kong, China: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/D19-1387.

Melis, G., Kočiský, T., & Blunsom, P. (2020). Mogrifier LSTM. In *International conference on learning representations*. URL: https://openreview.net/forum?id=SJe5P6EYvS.

Mihalcea, R. (2004). Graph-based ranking algorithms for sentence extraction applied to text summarization. In *Proceedings of the 42nd annual meeting of the association for computational linguistics, companion volume (ACL 2004)*. URL: http://www.cs.unt.edu/~rada/papers.html.

Nallapati, R., Zhai, F., & Zhou, B. (2017). SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents. In *AAAI'17, Proceedings of the thirty-first AAAI conference on artificial intelligence* (pp. 3075–3081). AAAI Press.

Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). The PageRank citation ranking: Bringing order to the Web. In *Proceedings of the 7th international world wide web conference*. (pp. 161–172). Brisbane, Australia. URL: citeseer.nj.nec.com/page98pagerank.html.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners.

Shen, D., Sun, J.-T., Li, H., Yang, Q., & Chen, Z. (2007). Document summarization using conditional random fields. In *IJCAI international joint conference on artificial intelligence* (pp. 2862–2867).

Souili, A., & Cavallucci, D. (2013). Toward an automatic extraction of IDM concepts from patents. In A. Chakrabarti (Ed.), *CIRP design 2012* (pp. 115–124). London: Springer London.

Souili, A., Cavallucci, D., & cois Rousselot, F. (2015). A lexico-syntactic pattern matching method to extract idm- triz knowledge from on-line patent databases. *Procedia Engineering*, *131*, 418–425. http://dx.doi.org/10.1016/j.proeng.2015.12.437, URL: https://www.sciencedirect.com/science/article/pii/S1877705815043295. TRIZ and Knowledge-Based Innovation in Science and Industry.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017a). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems 30* (pp. 5998–6008). Curran Associates, Inc., URL: http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017b). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems, Vol. 30*. Curran Associates, Inc., URL: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Xu, J., & Durrett, G. (2019). Neural extractive text summarization with syntactic compression. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 3292–3303). Hong Kong, China: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/D19-1324, URL: https://aclanthology.org/D19-1324.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). XLNet: Generalized autoregressive pretraining for language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems 32* (pp. 5754–5764). Curran Associates, Inc..

Zhou, Q., Yang, N., Wei, F., Huang, S., Zhou, M., & Zhao, T. (2018). Neural document summarization by jointly learning to score and select sentences. In *Proceedings of the 56th annual meeting of the association for computational linguistics (Volume 1: Long papers)* (pp. 654–663). Melbourne, Australia: Association for Computational Linguistics, URL: https://www.aclweb.org/anthology/P18-1061.