SynCRF: Syntax-Based Conditional Random Field for TRIZ Parameter Minings

Guillaume Guarino[1][a], Ahmed Samet[2][b] and Denis Cavallucci[1][c]
[1]ICube-CSIP team, INSA Strasbourg, University of Strasbourg, 24 Bd de la Victoire, Strasbourg, 67000, France
[2]ICube-SDC team, INSA Strasbourg, University of Strasbourg, 24 Bd de la Victoire, Strasbourg, 67000, France

Keywords: Conditional Random Field, TRIZ, Name Entity Recognition, Text Mining.

Abstract: Conditional random fields (CRF) are widely used for sequence labeling such as Named Entity Recognition (NER) problems. Most CRFs, in Natural Language Processing (NLP) tasks, model the dependencies between predicted labels without any consideration for the syntactic specificity of the document. Unfortunately, these approaches are not flexible enough to consider grammatically rich documents like patents. Additionally, the position and the grammatical class of the words may influence the text's understanding. Therefore, in this paper, we introduce SynCRF which considers grammatical information to compute pairwise potentials. SynCRF is applied to TRIZ (Theory of Inventive Problem Solving), which offers a comprehensive set of tools to analyze and solve problems. TRIZ aims to provide users with inventive solutions given technical contradiction parameters. SynCRF is applied to mine these parameters from patent documents. Experiments on a labeled real-world dataset of patents show that SynCRF outperforms state-of-the-art and baseline approaches.

1 INTRODUCTION

Sequence tagging encompasses a large variety of tasks, e.g., Named Entity Recognition (NER) and Part-Of-Speech (POS) tagging, to cite a few. Sequence tagging is often used in Natural Language Processing (NLP) and information retrieval.

Named Entity Recognition processes have much to gain from modeling the relations between predictions. Traditionally, an encoder is used to build a contextual representation of the tokens in the input document (Saha et al., 2018). A classification of the tokens is then performed. Unfortunately, even if encoders can capture contextual information of a token, they fail to encapsulate formal constraints on the predicted sequence of labels. Conditional Random Fields (CRF (Lafferty et al., 2001)) are widely used to model the relations between the predictions, via pairwise potentials, and thus improve the consistency of the predicted tag sequence.

In this paper, we investigate the potential of an architecture combining an encoder and a CRF (i.e. a Neural Random Field (Peng et al., 2009)) for Named Entity Recognition task. Unfortunately, CRFs do not take into account the grammatical structure of sentences to increase the relevance of the predicted tags sequence. We propose a new CRF architecture, called *SynCRF*, which aims at integrating syntactic information in the prediction mechanism. The pairwise potentials are, thus, predicted from the structure of each sentence.

SynCRF is applied in a TRIZ theory-based problem (Altshuller, 1984). TRIZ offers a package of practical techniques, which helps to analyze existing products and situations, extract root problems, reveal potential opportunities for evolution, and generate new solution concepts in a systematic way. TRIZ differs from other innovation theories by considering each problem as a contradiction between two parameters. For instance, in the aircraft industry, increasing the volume of the fuselage negatively impacts the total weight which hampers the lift-off ability. Such formulation is a typical TRIZ *contradiction* between the *volume* parameter and the *weight* parameter. The purpose of this theory of innovation is to build analogies between different domains via contradictions and inventive principles (Altshuller, 1984) that are general formulations of solutions (segmentation, prior action, ...). The contradictions between parameters are formulations of problems that are independent of the do-

[a] https://orcid.org/0000-0003-3032-9125
[b] https://orcid.org/0000-0002-1612-3465
[c] https://orcid.org/0000-0003-1815-5601

Guarino, G., Samet, A. and Cavallucci, D.
SynCRF: Syntax-Based Conditional Random Field for TRIZ Parameter Minings.
DOI: 10.5220/0012411300003636

In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 3*, pages 890-897
ISBN: 978-989-758-680-4; ISSN: 2184-433X

main and the inventive principles are formulations of solutions that also are independent of the domain. In the case of the volume/weight contradiction of the aircraft fuselage one can exploit TRIZ inventive principle 40 (Composite materials), for instance, and propose to change from an aluminum to a composite-type fuselage to lighten the structure.

We aim at applying SynCRF to extract these parameters from patents. Patents are a wealth of information about inventions but still require experts to understand the described solutions. To allow the automatic processing of problems within the TRIZ framework, a system must be able to understand the content of scientific or technical documents. Understanding a patent in the sense of TRIZ means mining the parameters of the contradiction(s) that these patents are solving. The Encoder-LSTM-CRFs are a well-known and commonly used architecture (Chiu and Nichols, 2016). This architecture aims to add sequentiality to the encoder representations. However, the purpose of this paper is different. It aims to model contextual dependencies between the labels by generating pairwise potentials from syntactic and semantic information.
The contributions of this paper are: *(i)* a new CRF structure, that encapsulates two variants *SynCRF-pos* and *SynCRF-context* and takes into account the syntactic information to compute pairwise potentials between labels; *(ii)* a TRIZ-based application to better understand patents' contents with TRIZ parameter mining; *(iii)* exhaustive experiments on TRIZ parameter mining with a manually built real-world dataset.

2 RELATED WORKS

In this section, we review approaches that were proposed to mine information from patents (TRIZ and not TRIZ-based approaches). We also focus on Named Entity Recognition applications solved with the use of both deep learning and Conditional Random Fields approaches.

Patents are structured documents with more or less constant sections such as abstract, description, claims. Unfortunately, patent wording of sentences differs from classical documents such as articles due to the legal nature of patents. Prior art search is a recurrent task in the field as it it necessary to verify that a patent is describing an actual invention (Cetintas and Si, 2012). However, prior art search as implemented in these approaches do not provide information for understanding the purpose of the invention as they are based on terms frequency in the documents.

CRFs are often used in sequence labeling tasks like Named Entity Recognition (NER) (Lample et al., 2016). CRFs are also used in slot filling tasks (Saha et al., 2018) to build structured knowledge bases usable for semantic-based information retrieval. They are exploited in vision applications as well, for instance, for semantic segmentation (Zheng et al., 2015).

CRFs model the dependencies between labels and between input data and labels. Nevertheless, the ability of deep neural networks to encode information is higher. Therefore, Neural Random Fields were introduced. A CRF is placed on top of a deep neural network to take advantage of the high-quality extracted features (Peng et al., 2009). For text mining, CRF are usually used with recurrent networks: Long Short Term Memory (LSTM) networks or Gated Recurrent Unit (GRU) networks (Cho et al.,). Recurrent networks (Hochreiter and Schmidhuber, 1997) are known to be efficient for language processing as they allow information to be transmitted throughout the encoding of a sequence via a memory vector.

With the arrival of pre-trained encoders, which perform better than recurrent neural networks in NLP tasks, the trend (Li et al., 2020) is to associate a pre-trained encoder (BERT (Devlin et al., 2018), XLNet (Yang et al., 2019), etc.) with a CRF. An architecture with a pre-trained encoder and a CRF is chosen in this paper. Pre-trained encoders perform better in downstream tasks with little labeled data as is the case for the TRIZ used case detailed in Section 5.

A limitation of the classical CRF is the lack of flexibility on the pairwise potentials. The transition matrix is unique regardless of the grammatical structure of the sequence under study. Approaches were developed in vision applications to generate pairwise potentials from Convolutional Neural Networks (Vemulapalli et al., 2016) but no approaches tackled the integration of syntactic information in pairwise potentials for text mining. Nevertheless, for a NER task, the position and the grammatical class of the words have an influence on the labels.

3 CONDITIONAL RANDOM FIELD

A Conditional Random Field (CRF) (Lafferty et al., 2001) is a statistical model dedicated to the modeling of dependencies between neighboring variables (Chu et al., 2016). In classification tasks, the CRF model computes the conditional probabilities $P(Y_k|X)$ with Y_k the labels and X the observations. A linear chain CRF is used in this study. Each label depends on the current observation as well as on the preceding and the following labels (Markov property).

Assuming Y and X corresponding respectively to a sequence of l labels and their corresponding sequence of l observations. The computation of $P(Y|X)$ is computed from each label and observation of the sequence (considering that the labels are predicted independently of one another at first) with the following formula:

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

with $Z(X)$, the partition function, i.e. the normalization factor computed from the sum of all possible numerators (for each possible labels sequence) and $U(X_k,Y_k)$ the *unary potential* referring to the likelihood that label Y_k is assigned given an observation X_k. $P(Y_k|X_k)$ is modeled with a normalized exponential as in a classical softmax output of a neural network.

If the dependency between two successive labels k^{th} and $k+1^{th}$ is established, then a linking term could be added to $P(Y|X)$ and therefore could be written as follows:

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

with $T(Y_{k-1},Y_k)$ the transition potential between label Y_{k-1} and label Y_k which is called the *pairwise potential*. The pairwise potential $T(Y_{k-1},Y_k)$ refers to the likelihood of Y_k label being followed by Y_{k+1}. Pairwise potentials are usually stored in a matrix called *transition matrix*. When the CRF is associated to a neural encoder (Saha et al., 2018), the unary potentials $U(X_k,Y_k)$ are given by the last layer of the neural encoder. The purpose is then find a label sequence Y which maximizes $P(Y|X)$ with respect to the parameters of the neural network and to the pairwise potentials which are learnt as well.

4 SynCRF: SYNTACTIC CONDITIONAL RANDOM FIELD

We tackle the problem of the independence of the pairwise potentials from the grammatical structure. Our approach, SynCRF, is proposed in several mechanisms allowing us to adapt the transition matrix to the syntactic structure of the studied sentences. We introduce two different architectures. The first one, called *SynCRF-pos*, is based on the parts of speech and the other one, *SynCRF-context*, takes into account all the information extracted by the encoder to compute pairwise potentials.

4.1 SynCRF-pos: Part of Speech-Based Syntactic CRF

SynCRF-pos, shown in Fig.1, consists of two main parts: the encoding of parts of speech and the generation of pairwise potentials contained in the CRF's transition matrix. An encoding matrix E is introduced to make the transition between parts of speech and a numerical vector containing the information on the syntactic structure of the sentence. Sequences of five parts of speech are encoded (to simplify Fig.1, only three tags are considered). We, therefore, make the assumption that the label of a token is only influenced by the two preceding and following tokens. The one-hot-vectors, associated with the part of speech tags, allow selecting in E the parameters contained in the encoded vector V_{emb}. A Hadamard product is performed between the tags' one-hot matrix (one-hot vector for each of the POS tags concatenated relatively to their position in the tag sequence (0,1,2,3,4) and the encoding matrix E):

$$V_{emb} = \sum_j \sum_i E \odot \delta_i \delta_{j=tag_i}^T \tag{3}$$

with i the position in the tag sequence (from 0 to 2 if three tags are used for instance), j the index of the POS class (u, v, w in Fig. 1) and tag_i the POS class of i^{th} tag. V_{emb} is then upsampled via a fully-connected layer of neurons to give V'_{emb}:

$$V'_{emb} = FC(V_{emb}). \tag{4}$$

V'_{emb} is then used as an input for a neural network allowing the generation of these pairwise potentials. Several types of neural networks are implemented and compared in this approach: a fully-connected 2-layer network and two recurrent GRU-type networks. The fully-connected network directly integrates the syntactic information contained in the encoded vector into a new transition matrix. On the other hand, the goal of the recurrent networks is to integrate a longer-term memory of the CRF and to emulate potentials that are not only dependent on the previous label but also on the preceding ones. Two configurations of recurrent networks are implemented. The first one aims at giving more weight to the last label than to the previous ones. V'_{emb} is thus aggregated to the memory

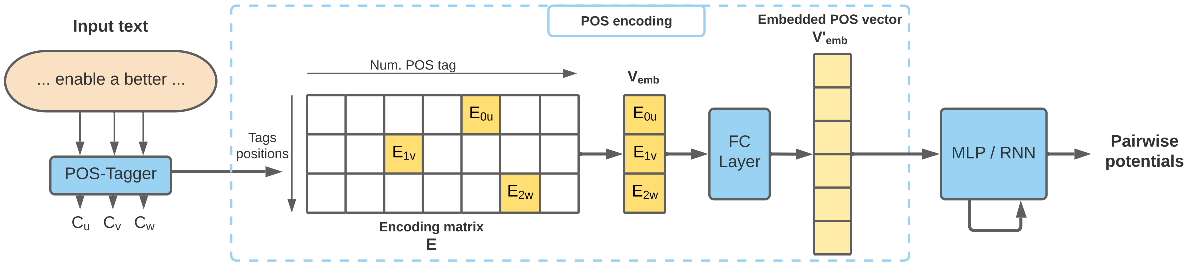

Figure 1: SynCRF-pos architecture for POS-adapted pairwise potentials generation.

vector (i.e. the hidden units, V_{hidden}) before generating the transition potentials using a fully-connected layer. The memory vector is then updated using V'_{emb}:

$$P_{i,j} = FC(V_{hidden}, V'_{emb}) \tag{5}$$

$$V_{hidden} = GRU_{update}(V'_{emb}) \tag{6}$$

with $P_{i,j}$ the pairwise potentials, FC a fully-connected neuron layer, V_{hidden} the GRU's hidden units and GRU_{update} the hidden units' update function.

In the second configuration, the memory vector is first updated with V'_{emb} and then the pairwise potentials are computed from the new memory vector as follows:

$$V_{hidden} = GRU_{update}(V'_{emb}) \tag{7}$$

$$P_{i,j} = FC(V_{hidden}). \tag{8}$$

The part-of-speech tags are generated using the python library spacy. Having an extreme quality on the part of speech does not seem to be a determining factor in the functioning of the method. The emphasis is therefore placed on the speed of tagging.

4.2 SynCRF-context: Context-Based Syntactic CRF

The use of a CRF on top of an encoder enables taking advantage of the contextual representations of tokens at the output of the encoder (Fig.2). Masked language models, due to their training, integrate rich syntactic information. It is, therefore, worth investigating generating the pairwise potentials of the Conditional Random Field from these contextual representations instead of using a part of speech tagging process. Additionally, parts of speech tagging process adds computational complexity. A neural network computes the potentials given the representations. Three different configurations are implemented for this neural network. A 1-layer and 2-layers fully connected neural networks are tested along with a recurrent neural network. A 1-cell GRU network is used. The purpose of this last configuration is building a direct link between the generated pairwise potentials to improve

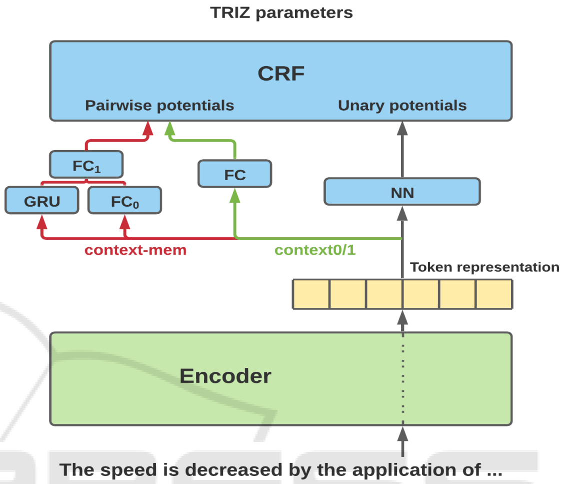

Figure 2: SynCRF-context architecture.

consistency in label sequences. The token representation V_{rep} is fed into fully connected layer FC_0 to compute V'_{rep} (Eq.9). V'_{rep} along with the recurrent network hidden units V_{hidden} are then fed into a fully connected layer FC_1 to give the output pairwise potentials (Eq.10). The hidden units are finally updated using the input representation V_{rep} (Eq.11). The memory cell is therefore used to keep track of the input representations sequence while the feed-forward networks FC_0 and FC_1 are extracting the relevant features to predict the pairwise potentials as follows:

$$V'_{rep} = FC_0(V_{rep}) \tag{9}$$

$$P_{i,j} = FC_1(V'_{rep}, V_{hidden}) \tag{10}$$

$$V_{hidden} = GRU_{update}(V_{rep}). \tag{11}$$

The generation of ”contextual” potentials is thus made possible by adding a minimum of parameters while remaining end-to-end trainable.

5 TRIZ PARAMETER MINING

5.1 TRIZ Theory: Basics

In TRIZ theory, problems are formulated as a contradiction between two parameters to ease their resolution and enhance the chances of finding an innovative solution. These two parameters are called *evaluation parameters*. A contradiction in the sense of TRIZ means that when one of the evaluation parameters is improved through an action on another parameter of the system (action parameter), the other evaluation parameter is degraded. For example, in patent US6938300B2: *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.* By increasing the diameter of the wheels the comfort is improved but the ability to push the stroller is degraded and vice-versa. *Comfort* and *ability to push* are Evaluation Parameters (EP). The wheels diameter is an Action Parameter (AP).

In TRIZ theory, the resolution of problems based on contradictions is achieved through the "TRIZ matrix". This matrix is designed to link the contradictions and the solutions. The Trizian solutions are the 40 inventive principles defined by Altshuller (Altshuller, 1984) (Segmentation, Periodic Action, Intermediary, etc...). This matrix has as many boxes as there are possible contradictions between the TRIZ parameters (39 parameters, so 39*39 boxes). These 39 parameters are, in theory, able to describe any problem from any domain. This matrix, therefore, applies to all known technical domains. In each box are indicated the inventive principles to be used to solve this type of contradiction. For example, for a contradiction between the parameters "Volume of a moving object" and "Weight of a moving object", the inventive principles proposed by this matrix are ("Taking out", "Copying", "Pneumatics and hydraulics" and "Composites"). In the example of the aircraft, provided in the introduction, the "Composites" principle could indeed be applied to solve the contradiction between the weight and the volume.

Despite the inherent variations in sentence wording due to the variety of patent drafters, these parameters (EP or AP) are, nevertheless, regularly located in sentences with similar syntactic structures. For example: "The use of tools or machines to install these barriers increases the complexity and cost of the installation process beyond that": nominal group (AP) + verb + nominal group (EP). It is, therefore, interesting to study the contribution of syntactic information in the TRIZ parameter mining process. At the same time, the parameters are regularly formed by several words (such as "cost of the installation process"). It is important to create a dependency between the predicted labels. These assumption incites to integrate syntactic information into a CRF to better model the dependencies between labels (pairwise potentials) through our SynCRF approach.

5.2 Dataset and Training

Pre-trained encoders are designed to work well in domains suffering from data deficiency. TRIZ domain and patent analysis are especially concerned by the lack of labeled data as the labeling process is tedious and can only be performed by experts. A dataset of 1100 labeled patents was created and made available[1]. It contains about 9000 labeled TRIZ parameters from abstracts, state-of-the-art, and claims parts of patents. Patents come from the United States Patent Trademark Office (USPTO). They were selected to cover all known technical domains (using CPC-IPC classification). An example of a labeled sentence is given below:

"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."

The *size of the barrier* is labeled as an action parameter (AP) while *no gaps between the carrier and the panel member* is labeled as an evaluation parameter (EP). The dataset was annotated by four engineers from industry field. In the annotation instructions, the parameters were defined as follows: an evaluation parameter is a parameter that measures the performance of a system, an action parameter is a parameter that can be modified and that influences one or more evaluation parameters. Verbs referring to changes in parameters (increase, decrease, etc.) are not included in the annotations. Two types of EP, EP+ and EP-, are defined to reflect either the positive or negative evolution of a parameter, or its positive or negative aspect (for example, a cost will fundamentally be a negative parameter). However, in this work, we do not consider the evolution of evaluation parameters and EP+ and EP- are aggregated in a single class EP. EPs are most often nominal groups (volume, power output, etc.) but verbal expressions can be annotated if no noun or nominal group allows to correctly describe

[1] The dataset can be downloaded **here.**

the parameter. For example, ”prevent fluid from entering the engine” will be annotated as it refers to sealing without the possibility of annotating a nominal group referring more directly to ”sealing”. SynCRF is trained using gradient back-propagation. The additional fully connected layers on top of the encoder and the CRF are fully trained on the patent dataset while the pre-trained encoder is fine-tuned with a decreasing learning rate to avoid overfitting. The base learning rate is set to 3e-5 for the encoder and 1e-3 for the decoding part (Conditional Random Field or Fully connected layer for the *Baseline* model. The decoder has a higher learning rate as it has to be learned from scratch. A step learning rate decay is implemented. After the first epoch, the encoder learning rate is decreased to 6e-6 and then 3e-6 after the second epoch. Adam optimizer is used with a batch size of 16. The training is performed on an RTX2080Ti[2].

6 EXPERIMENTS AND RESULTS

Classification metrics are used to evaluate the models (Precision, Recall, F1-score). The accuracy is considered as not relevant to compare the models for this task. 4-fold cross-validation is performed.

Berdyugina et al. (Berdyugina and Cavallucci, 2020) is the only state-of-the-art approach to tackle parameter mining. This approach is based on a cause-effect framework. As the Action Parameters can influence the Evaluation Parameters, they are seen as causes of a change in an EP. The EPs are, therefore, seen as effects. It was trained on a cause-effect dataset. To be able to compare with models using our data and measure the impact of our new syntactic CRF we, therefore, introduce XLNet. XLNet (Yang et al., 2019) pre-trained encoders is used in SynCRF. We add a simple classification layer with a fully-connected layer on top of the encoders to mine parameters.

SynCRF is a neural random field (neural encoder with CRF). Thus we also consider neural random fields to have a fairer comparison with SynCRF. A CRF (Lafferty et al., 2001) is placed on top of both of these neural encoders to build XLNet-CRF (Chai et al., 2022).

As the extraction of TRIZ parameters is seen as a Named Entity Recognition task with a BIO (Beginning, Inside, Outside) (Ramshaw and Marcus, 1999) label policy, several transitions are forbidden. In the case of EP and AP for TRIZ, it is, for instance, impossible to go from an evaluation parameter EP-I (Interior of EP) to an action parameter AP-I (Interior) since the action parameter should start with a label B (Begin). Constraints can be manually applied to forbid these transitions. The potentials related to the forbidden transitions can manually be set to values lower than 0 in the log space which correspond to zero transition probabilities. These transitions will, thus, never appear in the predicted label sequences. To highlight the impact of the transition constraints, we introduce a baseline approach which is basically XLNet-CRF with the constraints called XLNet-CRF-cs.

Table 1 contains the results associated with SynCRF based on XLNet encoding. The *SynCRF* prefix indicates the newly developed CRF architecture. SynCRF-pos relates to the models using parts of speech (shown on Figure 1). *mem* and *mem-o* refer to the variation on the recurrent models described in 4.1. *mem* is the model described with Eq. 5 and 6 while *mem-o* refers to Eq. 7 and 8. SynCRF-context relates to the models using token contextual representations to generate pairwise potentials (see Figure 2). The number behind *context* indicates which configuration described in 4.2 is used. SynCRF-context-mem relates to the SynCRF-context variant with the memory cell. The *cs* suffix indicates that probabilities of forbidden transitions are manually set to 0.

Table 2 compares the best SynCRF configuration versus the state of the art and baselines approaches.

6.1 SynCRF-pos Results

E and *A* suffixes in the metrics in Table 1 refer to Evaluation Parameters (EP) and to Action Parameters (AP). We can see that adding constraints on the transitions allows to slightly decrease the loss (from 1% to 2% for SynCRF-pos-mem). It also improves precision and recall by about 1% for EPs and 3% for APs. The addition of constraints to SynCRF thus allows constant but relatively limited improvements in the results.

Concerning the architecture, we highlight the relevance of adding temporal information on the previous pairwise potentials with a recurrent network. Indeed, we observe a decrease of about 20% in the loss between the non-recurrent SynCRF-pos models and the recurrent SynCRF-pos-mem models. On the metrics, we observe an increase in precision but a decrease in the recall, which keeps the F1 score at the same level. As precision is the most important metric in our case to avoid undermining bad contradictions the best SynCRF-pos model seems to be SynCRF-pos-mem-cs.

[2]The code to reproduce the results can be downloaded **here.**

Table 1: SynCRF results with XLNet encoding.

Model	Loss	TP_E	$Prec_E$	$Rec._E$	$F1_E$	$Supp._E$	TP_A	$Prec_A$	$Rec._A$	$F1_A$	$Supp._A$
SynCRF-pos	0.159	4182	50.6	47.6	49.0	8789	424	37.9	**25.0**	29.9	1692
SynCRF-pos-cs	0.157	4049	50.9	46.1	48.4	8789	410	42.1	24.3	30.4	1692
SynCRF-pos-mem	0.139	4024	51.5	45.9	48.4	8789	319	37.3	18.9	24.9	1692
SynCRF-pos-mem-cs	0.134	4071	51.3	46.3	48.7	8789	369	38.9	21.8	27.8	1692
SynCRF-pos-mem-o	0.291	1045	13.1	11.5	12.2	8789	85	9.2	5.0	6.5	1692
SynCRF-pos-mem-o-cs	0.134	4099	52.8	46.6	49.5	8789	364	39.9	21.5	27.8	1692
SynCRF-context0	0.128	4170	53.2	47.4	50.2	8789	383	**49.1**	22.6	30.8	1692
SynCRF-context1	0.122	4180	**53.4**	47.6	**50.3**	8789	378	43.8	22.4	29.5	1692
SynCRF-context-mem	**0.111**	4188	52.6	**47.7**	50.0	8789	407	43.7	24.1	**31.0**	1692

Table 2: Comparison of SynCRF with the state of the art.

Model	Loss	TP_E	$Prec_E$	$Rec._E$	$F1_E$	$Supp._E$	TP_A	$Prec_A$	$Rec._A$	$F1_A$	$Supp._A$
BERT(Devlin et al., 2018)	0.423	3769	31.6	43.3	36.5	8717	210	18.5	12.7	14.8	1651
BERT-CRF(Sun et al., 2022)	0.393	3876	37.8	44.5	40.9	8717	284	26.7	17.2	20.6	1651
BERT-CRF-cs	0.137	3939	48.5	45.2	46.8	8717	286	**45.1**	17.3	24.7	1651
XLNet(Yang et al., 2019)	0.399	4148	38.0	47.2	42.1	8789	318	26.1	18.8	21.7	1692
XLNet-CRF(Chai et al., 2022)	0.348	4222	43.7	**48.1**	45.8	8789	315	31.2	18.6	23.2	1692
XLNet-CRF-cs	0.140	3819	48.7	43.6	45.9	8789	264	42.3	15.6	21.6	1692
(Berdyugina and Cavallucci, 2020)	-	1887	11.0	21.5	14.6	8770	479	2.5	**28.9**	4.5	1656
XLNet-SynCRF	**0.111**	4188	**52.6**	47.7	**50.0**	8789	407	43.7	24.1	**31.0**	1692

6.2 SynCRF-context Results

Using the richer tokens' representations of the encoder as a source for the syntactic information shows, compared to the explicit syntactic information-based models (SynCRF-pos), a significant improvement in the results (Table 1). The loss decreases by about 10% between the best SynCRF-pos model and the best SynCRF-context model. The metrics are also positively impacted. The accuracy increases by 1% with XLNet for the EPs and by about 14% for the APs. The recall is relatively constant so it leads to an improvement in the F1 score.

The variant with the memory cell appears to be the best model in terms of loss and AP metrics while its performance on EP is as consistent as SynCRF-context0 and SynCRF-context1. SynCRF-context approaches also show globally better results than SynCRF-pos in terms of loss and metrics. This syntactic information also minimizes the impact of arbitrary constraints on certain transitions as these are learned by the network that generates the pairwise potentials. They outperform all constrained models without any external action on the pairwise potentials.

6.3 Comparison with the State of the Art

Table 2 compares SynCRF-context-mem, which is the best configuration of SynCRF, with the state-of-the-art approaches and baselines introduced. The contribution of a traditional CRF (XLNet-CRF) in the extraction of TRIZ parameters is visible in the results with a decrease of about 10% of the loss and of 4-5% of the F1-score for EP and AP compared to the encoders alone.

The addition of constraints on forbidden transitions (XLNet-CRF-cs) has a strong positive impact on the loss value compared to XLNet-CRF models (-60%) but the impact on the metrics is not constant depending on the encoder and the parameters' type. The precision is the only metric that is always improved by 5 to 10% with the additional constraints on the CRF. We, therefore, highlight that the interest in a traditional CRF is felt above all when one is aware of certain forbidden transitions which can be managed by imposing the values of the associated pairwise potentials. This impact is also much higher on a classical CRF than on our SynCRF. Berdyugina et al. (Berdyugina and Cavallucci, 2020) shows relatively weak performance compared to other models. The cause-effects framework does seem to fit well the parameters because the recall is relatively high. It shows, for instance, the best recall for APs but the precision is extremely low so it is clear that there are a lot of false positives with this methodology and we cannot rely on it to extract contradiction parameters.

SynCRF largely outperforms all these approaches. Indeed, it shows consistent performance with both encoders. The loss is three times slower than encoders only and encoder+CRF architectures. The improvement on the metrics is massive especially for APs with a 25% improvement on the F1 score compared to the best baseline but also for APs with a 7% improvement on the F1 score. The precision is the most improved metric for EPs which is exactly what we are looking for. Thus, we demonstrate that adding syntactic information to generate pairwise potentials in a Conditional Random Field is very valuable, especially in tasks where labels are strongly linked to syntax like in TRIZ contradiction modeling.

7 CONCLUSION

In this paper, we present an approach called SynCRF that allows to mine TRIZ parameters from patents. This approach is part of a solved contradiction mining process whose purpose is a fine-grained understanding of the inventions described in patents. SynCRF is built with a deep neural encoder and a Conditional Random Field. It relies on the syntactic structure of sentences to estimate pairwise potentials and improve consistency in the predicted label sequences. SynCRF shows solid improvements over the state of the art with absolute improvements of 3 to 5% for all metrics over the best baseline (XLNet-CRF-cs). It is also highlighted that SynCRF learns more easily the forbidden transitions and allows for example to improve the precision by more than 20% compared to the best baseline without constraints on the transitions (XLNet-CRF).

ACKNOWLEDGEMENTS

This research was funded in part by the French National Research Agency (ANR) under the project "ANR-22-CE92- 0007-02"

REFERENCES

Altshuller, G. (1984). *Creativity As an Exact Science*. CRC Press.

Berdyugina, D. and Cavallucci, D. (2020). Setting up context-sensitive real-time contradiction matrix of a given field using unstructured texts of patent contents and natural language processing. In *Triz Future 2020*.

Cetintas, S. and Si, L. (2012). Effective query generation and postprocessing strategies for prior art patent search. *J. Assoc. Inf. Sci. Technol.*, 63:512–527.

Chai, Z., Jin, H., Shi, S., Zhan, S., Zhuo, L., and Yang, Y. (2022). Hierarchical shared transfer learning for biomedical named entity recognition. *BMC Bioinformatics*, 23.

Chiu, J. P. and Nichols, E. (2016). Named Entity Recognition with Bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Cho, K., van Merrienboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Moschitti, A., Pang, B., and Daelemans, W., editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014*, pages 1724–1734.

Chu, X., Ouyang, W., Li, h., and Wang, X. (2016). Crf-cnn: Modeling structured information in human pose estimation. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. cite arxiv:1810.04805Comment: 13 pages.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Li, X., Zhang, H., and Zhou, X.-H. (2020). Chinese clinical named entity recognition with variant neural structures based on bert methods. *Journal of Biomedical Informatics*, 107:103422.

Peng, J., Bo, L., and Xu, J. (2009). Conditional neural fields. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C., and Culotta, A., editors, *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc.

Ramshaw, L. and Marcus, M. (1999). *Text Chunking Using Transformation-Based Learning*, pages 157–176. Springer Netherlands, Dordrecht.

Saha, T., Saha, S., and Bhattacharyya, P. (2018). Exploring deep learning architectures coupled with crf based prediction for slot-filling. In Cheng, L., Leung, A. C. S., and Ozawa, S., editors, *Neural Information Processing*, pages 214–225, Cham. Springer International Publishing.

Sun, J., Liu, Y., Cui, J., and He, H. (2022). Deep learning-based methods for natural hazard named entity recognition. *Scientific Reports*, 12:4598.

Vemulapalli, R., Tuzel, O., Liu, M.-Y., and Chellappa, R. (2016). Gaussian conditional random field network for semantic segmentation. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3224–3233.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., and Le, Q. V. (2019). *XLNet: Generalized Autoregressive Pretraining for Language Understanding*. Curran Associates Inc., Red Hook, NY, USA.

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., and Torr, P. H. S. (2015). Conditional random fields as recurrent neural networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1529–1537.