ECAI 2020
G.D. Giacomo et al. (Eds.)

doi:10.3233/FAIA200432

Toward a Generic AutoML-Based Assistant for Contracts Negotiation

Youssef Bendraou[1] and **Nathan Ramoly**[1] and **Baba Seydou Bamba**[1]

Abstract. Contracts are the cornerstone of legal agreements in the industry. As essential legal documents, contracts are subject to negotiations, demanding extensive analysis and evaluation efforts. Although the emergence of machine learning has enabled assistance tools for text analysis tasks, the specificity and constraints of each business context remain obstacles for the automation of contracts evaluation. In this paper, we propose an AutoML based approach for automated negotiation assistance that uses expert annotated contracts and the business-specific knowledge of acceptance policies. Driven by policies rules, our approach generates a classification process composed of hierarchies of complementary classifiers, each being automatically prepared according, but not limited to, feature extraction, learning model and data granularity. Experiments conducted on real-world service contracts have yielded promising results.

1 INTRODUCTION

As the legal translation of any agreement, contracts have been critical documents in the industry. In a business to business (B2B) context, contracts are prone to intense negotiations: they are analyzed, criticized and modified, as each party applies its own acceptance policies. Such processes are demanding, time consuming and error-prone for negotiators. However, the progress in Artificial Intelligence (AI) and Machine Learning (ML) offers new tools toward assistance in document analysis. In fact, recent works [1, 2, 4] have explored contract analysis and have proven the relevance of ML and Deep Learning (DL). Indeed, in order to clarify clauses to the user, approaches such as Claudette [4] or Polisis [1] have successfully applied hierarchies and/or combinations of classifiers in, respectively, *terms of services* and *privacy policies* domains.

However, these approaches are not suitable for the context of B2B contracts negotiation, that is subject to multiple challenging constraints. In fact, each editor uses its own terms and style, leading to an heterogeneity of vocabulary and structure among B2B contracts, for yet similar meanings. Furthermore, acceptance policies are company-specific, and are consequently highly heterogeneous. Naturally, domains and types of contracts are numerous and depends on the core activities of organisations. An efficient negotiation assistant shall adapt to any business, yet, heterogeneity remains a major obstacle for current approaches. Another challenging constraint is the limited amount of up-to-date example contracts, that is directly impacted by businesses size, policies evolution and negotiation records. Furthermore, as negotiations focus on particular clauses, sample data are likely to be imbalanced.

In this work, we aim to explore solutions that overcome the aforementioned challenges and to enable automated negotiation assistance. We propose a flexible ML based solution that automatically evaluates the acceptance of each clause of a contract, given a set of annotated data and policies. This can be seen as an anomaly detection problem, as we aim to detect clauses that are anomalous toward given policies. Our solution relies on two main features: Firstly, it uses a semi-automatic system for clauses annotation and generation [6] that enables an increase and a balancing of the training set, based on a limited and imbalanced original contract set. Secondly, our main contribution resides in the exploration of an Automated Machine Learning (AutoML) method that generates hierarchies and combinations of classifiers fitting the context. Our AutoML approach considers various ML models, feature extraction, but also several data granularities. Furthermore, it takes into account policies as rules in order to direct the generation process. This approach allows to overcome the heterogeneity issues of B2B contracts. As a case study, we experimented our proposition with actual service contracts.

2 AUTOMATED CONTRACT EVALUATION

Contracts are divided into independent sections, called *clauses*, whose size may vary from a single sentence to several paragraphs. During a negotiation, clauses are modified, added or removed according the *policies* of each party. *Policies* can be seen as business constraints that should be respected in collaborations. Hence, the role of the negotiator is to identify invalid clauses and to enforce these policies as much as possible. Our goal is to provide an assistance tool by automating this evaluation process. Formally, let C be the set of clauses of a contract, T the set of clauses types in the current domain, and P the set of policies. We aim to define the evaluation process $f : P \times C \to \mathbb{B}$ that provides a Boolean that asserts the validity of a clause $c \in C$ against a policy $p \in P$. In order to implement this process, we proposed a three steps sequence:

1. Classification of clauses per type. We define $\alpha : C \to T$
2. Classification of clause validity against any policies, according to its type. We define, $\forall t \in T, \beta_t : C \to \mathbb{B}$
3. Classification of the anomaly toward a violated policy. We define,
$$\forall t \in T, \gamma_t : C \to \begin{cases} P, \text{ if } \beta_{\alpha(c)} = 1 \text{ with } c \in C \\ \emptyset \text{ , if } \beta_{\alpha(c)} = 0 \text{ with } c \in C \end{cases}$$

Hence, with $c \in C$ and $p \in P$, $f(c,p) = 1 \iff \gamma_{\alpha(c)} = p$, meaning the clause c is invalid against p. This decomposition enables specific, and consequently accurate, ML classifiers for each step. According to the literature [3, 4, 7], using *ensemble classifiers*, that rely on multiple complementary sub-classifiers, leads to a more accurate classification. Consequently, as depicted in Figure 1, α, β_t and γ_t

[1] Ilyeum Insights, Ilyeum, Paris, France, emails: ybendraou@ilyeum.com, nramoly@ilyeum.com, bbamba@ilyeum.com

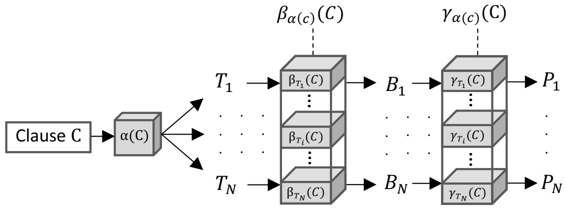

Figure 1. Evaluation process, including α, β and γ functions, where an *ensemble classifier* is represented as a grey cuboid and $N = |T|$.

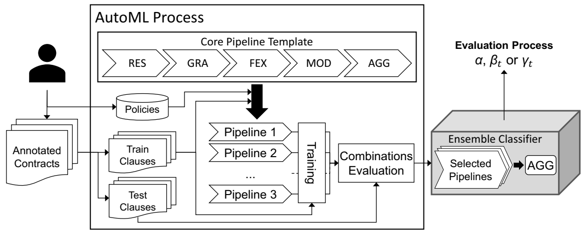

Figure 2. AutoML training process that generates an *ensemble classifier* by instantiating classification pipelines and selecting relevant combinations.

were implemented as *ensemble classifiers* relying on sub-classifiers with various characteristics including learning model, feature extraction and granularity. The outcome of α determines the instances of β and γ to apply on the clause. As setting up such hierarchies of classifiers is challenging, we adopted a dedicated AutoML process.

The AutoML training process relies on two types of inputs provided by a domain expert: policies and example contracts, both accepted and rejected. We assume example contracts to be annotated according to violated policies and multiple granularities, clause-wise and sentence-wise typically. If needed, the example contracts can be reinforced through generation of new samples [6]. Policies are formatted as rules, in our work, we used a SWRL formalism.

We proposed an AutoML approach that aims to determine close-to-optimal *ensemble classifiers* that fit the needs of the evaluation process f and the specific constraints of the B2B context. Accordingly and unlike classical AutoML solutions, our approach not only determines the suited feature extraction tool, model and hyperparameters, but also extracts the best combination of sub-classifiers, as well as the most relevant granularities of data. Our AutoML approach is displayed in Figure 2. It generates multiple instances of a template pipeline that includes: sample restriction (RES), granularity selection (GRA), features extraction (FEX), learning model and hyperparameters (MOD), and aggregation technique (AGG). The instantiating of a pipeline is restricted or directed according to the policies rules, hence reducing the research space and optimizing the pipeline creation. The generated pipelines are then combined and aggregated, thanks to a vote procedure [4], into an *ensemble classifier*. The most relevant *ensemble classifier*, determined by its F_1 score on a test set, is then selected and injected in the evaluation process. Consequently, the whole hierarchy of the evaluation process can be created and trained automatically while being adapted to any B2B context.

3 PRELIMINARY EXPERIMENTS

We implemented a negotiation assistant prototype using *Python*, *scikit-learn* and *TPOT* [5]. Multiple experiments were conducted. In this paper, we focus on the results achieved by the evaluations of the complete process on a set of service contracts issued by Ilyeum commercial department. The dataset[2] consists of 503 clauses, or 1925 sentences, from 23 contracts, with 28% of invalid clauses, including non-recurrent policy violations. 20 policies were also provided and formatted as SWRL rules by experts. Multiple bag-of-words based feature extraction approaches (*count, tf-idf, hashing*) were considered in this prototype. We measured the precision (P), recall (R) and F_1 score for both invalidity detection (β process) and classification (γ process). Measures achieved by the AutoML-based prototype were compared to SVM-based classifiers, that have proven their worth for text classification [4]. Measurements, are displayed in Table 1.

[2] The prototype source code and a sample of the dataset is available at: https://github.com/IlyeumInsights/ana

Table 1. Weighted-averaged precision (P), recall (R), and F_1 factor of our assistant prototype against single model baselines

Approach	**P**	**R**	**F_1**
Assistant - β	0.87	0.87	0.86
SVM - binary	0.77	0.78	0.78
Assistant - γ	0.74	0.76	0.73
SVM - multi.	0.71	0.74	0.72

Our prototype proves to be effective for invalid clauses detection, achieving a F_1 score of 0.86; 0.08 higher than the baseline. Regarding the identification of violated policies, a gain of 0.01 is achieved on the F_1. This last result reveals a need for improvements. Moreover, the approach should be confronted with various heterogeneous domains. Nevertheless, the achieved performance demonstrates the relevance of our evaluation process and AutoML approach.

4 CONCLUSION

Our study considers the use of rule-driven AutoML for generic and automated contract evaluation in B2B contexts. We identified key challenges, including the heterogeneity of domains, and proposed an AutoML-based approach that generates a hierarchy of classifiers, enabling flexible and accurate identification of invalid clauses and violated policies. Although further experiments are needed and will be performed on state-of-the-art datasets [1, 4], preliminary results showed our approach to be promising and applicable for contracts negotiation assistance. Yet, in order to enable a fully automated contract negotiators, various perspectives are to be explored, including the use of ontological knowledge, reasoning and NLU techniques.

REFERENCES

[1] H. Harkous, K. Fawaz, R. Lebret, F Schaub, K.G. Shin, and K. Aberer, ‘Polisis: Automated analysis and presentation of privacy policies using deep learning’, *27th USENIX Security Symposium, USENIX Security 2018, Baltimore, MD, USA.*, 531–548, (August 2018).

[2] Y. He, J. Li, Y. Song, M. He, and H. Peng, ‘Time-evolving text classification with deep neural networks’, *in the* 27^{th} *International Joint Conference on Artificial Intelligence, IJCAI-18*, 2241–2247, (July 2018).

[3] D. Isa, L. Lee, V. Kallimani, and R. RajKumar, ‘Text document preprocessing with the bayes formula for classification using the support vector machine’, *IEEE Trans. Knowl. Data Eng.*, **20**(9), 1264–1272, (2008).

[4] M. Lippi, P. Palka, G Contissa, F. Lagioia, H.W. Micklitz, G. Sartor, and P. Torroni, ‘Claudette: an automated detector of potentially unfair clauses in online terms of service’, *Artificial Intelligence and Law*, **27**(2), (2019).

[5] R. S. Olson and J. H. Moore, ‘Tpot: A tree-based pipeline optimization tool for automating machine learning’, *Automated Machine Learning*, 151–160, (2019).

[6] Z. Shi, X. Chen, X. Qiu, and X. Huang, ‘Towards diverse text generation with inverse reinforcement learning’, *in the* 27^{th} *International Joint Conference on Artificial Intelligence, IJCAI-18*, 4361–4367, (July 2018).

[7] C. H. Wan, L. H. Lee, R. Rajkumar, and D. Isa, ‘A hybrid text classification approach with low dependency on parameter by integrating k-nearest neighbor and support vector machine’, *Expert Systems with Applications*, **39**(15), 11880 – 11888, (2012).