3iA Côte d'Azur
Interdisciplinary Institute for Artificial Intelligence

Ph.D. research topic

- Title of the proposed topic: **Hybridation of Constraint Programming and Language Model**
- Research axis of the 3IA: 1 (Core element of AI)
- **Supervisor (name, affiliation, email): jean-charles.regin@univ-cotedazur.fr**
- Potential co-supervisor (name, affiliation):
- The laboratory and/or research group: I3S

Apply by sending an email directly to the supervisor.
The application will include:

- Letter of recommendation of the supervisor indicated above
- Curriculum vitæ.
- Motivation Letter.
- Academic transcripts of a master's degree(s) or equivalent.
- At least, one letter of recommendation.
- Internship report, if possible.

⇨ **All the requested documents must be gathered and concatenated in a single PDF file named in the following format: LAST NAME of the candidate_Last Name of the supervisor_2023.pdf**

- Description of the topic:

Text generation is one of the major successes of Artificial Intelligence. More and more text generators with impressive performances are appearing. Most of them are based on Language Model (LM) and there are a lot of transformers-based LM (i.e., LM implementing Transformers architecture based on attention block [Vasmani et al.,2017] due mainly to their effectiveness. An LM is a probabilistic model used to predict the sequence of words in a text or speech using large natural language training data (several tens of gigabytes). These texts are cut into chunks of 1024 tokens, and LMs process these sequences of tokens to learn a text distribution. Once trained, generative models, like GPT-2, can generate sequences using a part of existing sequences or a token-of-start to compute a distribution of potential successors.

GPT-2 is also suited to assess sentence quality to a certain degree by assigning a score to sequences. This score can also be seen as a marker of the meaning of a sentence. The better a sentence is scored, the more likely it is to make sense. Thanks to such a tool, one can select

among a set of generated sentences those that make sense or those that seem better than the others. This allows to solve important problems of text generation under constraints.

Currently, systems generate text and process the text sequentially to satisfy constraints which can be done by tokens (words) filtering [Roush et al 2022] or search mechanisms such as Beam Search (BS) [Liu et al., 2021; Post and Vilar, 2018; Hokamp and Liu, 2017] or more recently A* [Lu et al., 2022]. These search-based heuristics involve generating the text word-by-word while maximizing the likelihood of the sequence computed, exploring solutions space, and checking constraints. This methodology is appropriate as long as the solution space is weakly constrained, i.e. when it is easy to find a sequence satisfying the constraints. This is usually the case for lexical or semantic constraints since they are local requirements in the output and are almost formalized as output preferences.

When the solutions space is strongly constrained (i.e., when it is hard to find a solution that satisfies the constraints), that is when constraints are not local anymore (e.g., length constraint, display constraint), in other words when they are defined on the whole text, this solution is no longer effective because the sequences we are looking for are rare. In this case, a completely different methodology must be considered. The idea is to check the constraints first and then select the best solutions with an LM because there is no reason that the sequences that an LM considers likely would satisfy the constraint first. This means that some constrained text generation problems are seen as discrete combinatorial optimization problems, where variables are words, the domain of variables is the vocabulary of text, and constraints are the set of rules the text must satisfy. And we use Constraint Programming to solve them.

This approach produces sentences satisfying the constraints, but most of them have no meaning. The selection of good sentences can be achieved by using the score given by GPT-2 as we mention it. This approach has been successfully applied to generate standardized sentences in the field of vision screening. The sentences of the MNREAD medical test have to satisfy a set of display constraints that are difficult (60 characters sentence displayed without hyphenation on three lines and justified with strongly limited spacing) and there are extremely few of them in the existing literature. We were able to generate hundreds of sentences while there were only 38 which was much too few to avoid memory bias.

The goal of this phD thesis is to integrate the use of language model before the end of the text generation, otherwise during the generation. In Constraint Programming this means defining new constraints (called global constraints) to deal with this problem. This constraint will be used to reject texts whose meaning is not close enough to a given language

One can imagine a constraint based on the score computed by GPT-2 and on the maximum value it can reach before a sentence becomes meaningless or weird. The score given by GPT-2 to a sentence, i.e., a word sequence, is very close to the perplexity measure of this sequence. Perplexity is an entropy measure derived from Shannon's information theory [Brown et al., 1992]. It can be expressed as the geometric mean of the inverse conditional likelihood of the sequence [Jurafsky and Martin, 2009]. Given S_n the sequence of a succession of words of size n, so $S_n = w_1w_2...w_n$. The perplexity (PPL) of S_n is computed as follows:

$$\mathrm{PPL}(S_n) = \sqrt[n]{\frac{1}{P(w_1 w_2 w_3 \ldots w_n)}}$$

where probabilities P(…) are given by the LM.
PPL can be interpreted as the "uncertainty" of the model with respect to a sample. Usually, it is used to evaluate the LM itself by checking that good samples are recognized as such (i.e., low PPL values).

To define a constraint of perplexity PPL(X) < K, it is necessary to be able to decide that for a partial assignment of variables (e.g., the assignment of variables $x_1 \ldots x_p$ to words), one can ensure that the total perplexity (for the variables from x_1 to x_n) will necessarily exceed K. Thus, in this case, we will reject the partial assignment by not continuing it.

This is not an easy task for several reasons:

- The transformer's valuation is time-consuming due to the quadratic complexity architecture. Even though some work is currently done to build a new architecture to reduce its algorithmic complexity (e.g., forecasting time series [Zhou et al., 2021]). Evaluating millions of sentences, even on a powerful machine with GPT-2, can take several hours or days.
- The perplexity is the average mean of the inverse conditional probability of the sequences. It means that we need the entire sentence to obtain the exact valuation. This also means that if we want to score all subsequences (subsentences), it will produce a tremendous number of requests to the LM at the generation stage because millions of them are produced.
- The perplexity is not monotonic and can vary greatly and improve significantly.

References:

[Vaswani et al. 2017] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin, "Attention is All you Need", In Advances in Neural Information Processing Systems, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan and R. Garnett, Curran Associates, Inc., Vol 30, 2007.

[Roush et al., 2022] A. Roush, S. Basu, A. Moorthy and D. Dubovoy, "Most language models can be poets too: An AI writing assistant and constrained text generation studio", Proceedings of the Second Workshop on When Creative AI Meets Conversational AI, pages 9–15, Gyeongju, Republic of Korea, 2022.

[Liu et al., 2021] Y. Liu, L. Zhang, W. Han, Y. Zhang, and K. Tu, "Constrained text generation with global guidance - case study on commongen", CoRR abs/2103.07170, 2021.

[Post and Vilar, 2018] M. Post and D. Vilar, "Fast lexically constrained decoding with dynamic beam allocation for neural machine translation", Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1314–1324, New Orleans, Louisiana, USA, 2018.

[Hokamp and Liu, 2017] C. Hokamp and Q. Liu, "Lexically constrained decoding for sequence generation using grid beam search", Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1535–1546, Vancouver, Canada, 2017.

[Lu et al., 2022] X. Lu, S. Welleck, P. West, L. Jiang, J. Kasai, D. Khashabi, R. Le Bras, L. Qin, Y. Yu, R. Zellers, N. Smith, Y. Choi, "NeuroLogic A*esque Decoding: Constrained Text Generation with Lookahead Heuristics", Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp 780-799, Seattle, Washington, USA, 2022.

[Brown et al., 1992] P. Brown, S. Della Pietra, V. Della Pietra, J. Lai, and R. Mercer, "An estimate of an upper bound for the entropy of English", Computational Linguistics, 18(1):31–40, 1992.

[Jurafsky and Martin, 2009] D. Jurafsky and J.Martin, "Speech and language processing: an introduction to natural language processing", Computational Linguistics and Speech Recognition. Pearson Prentice Hall, 2009.
[Zhou et al., 2021] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting", Proceedings of the AAAI Conference on Artificial Intelligence, 35(12):11106–11115, 2021.