Patent Classification using Extreme Multi-label Learning

A Case Study of French Patents

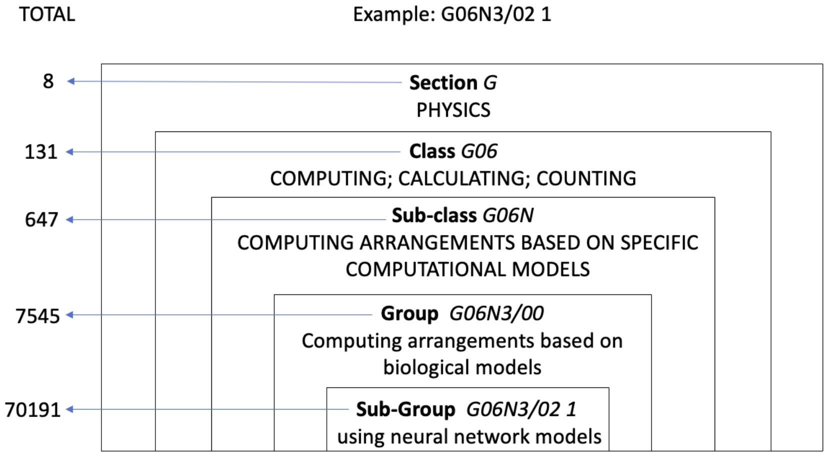
You Zuo[1], Houda Mouzoun[2], Samir Ghamri Doudane[2], Kim Gerdes[3], Benoît Sagot[1]

1. Inria Paris, 2. Institut national de la propriété industrielle, 3. University Paris Saclay

PatentSemTech 2022 - July 15, 2022

Introduction

- Task: **IPC** classification of **French Patent** documents



Introduction

- Task: **IPC** classification of **French Patent** documents
- Problems:
  1. not enough data in French (EPO, WIPO Delta dataset)
  2. limited to subclass level (with less than 700 labels)

Solutions

We present a French Patents corpus, named INPI-296k

- extracted from the INPI(French Intellectual property Office) internal database
- contains all parts of patent texts (title, abstract, claims, description) published from 2002 to 2021
- with IPC labels at all levels from sections to subgroups

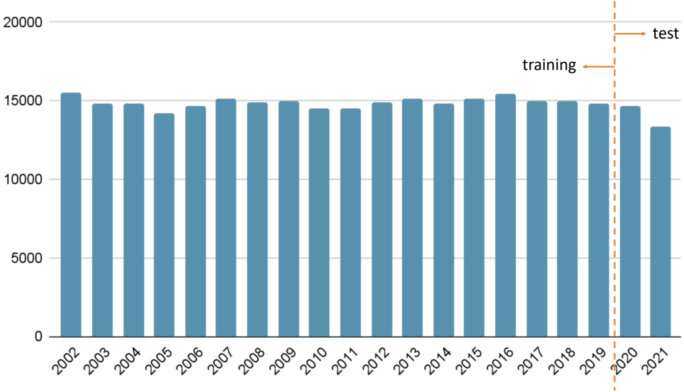
section	title	abstract	description	claims
# items	296 270	295 421	296 216	291 539
# tokens (average)	11	111	4202	725

Table: Description of our French corpus INPI-296k

Solutions

Dataset	N	L_4	$\bar{L}_4$	$\hat{L}_4$	L_6	$\bar{L}_6$	$\hat{L}_6$	L_8	$\bar{L}_8$	$\hat{L}_8$
Train	268254	638	1.73	420.46	6788	2.21	39.52	48932	2.73	5.48
Test	28017	583	1.77	48.06	4351	2.20	6.44	19593	2.64	1.43

Table: Basic Statistics of INPI-296k dataset



- N: number of patents
- L: number of labels
- $\bar{L}$: average number of IPC labels of a document
- $\hat{L}$: average number of documents per label
- The subscripts of 4,6,8 represent respectively IPC's subclass, group, and subgroup levels (4, 6, and 8 correspond to the number of characters used to encode the class)

Solutions

$\hookrightarrow$ to implement eXtreme Multi-label Learning (XML) methods to handle large numbers of classes (at the subclass and group level in our experiments)

Definition

The XML addresses the problem of learning a classifier which can automatically tag a data sample with the most relevant subset of labels from a large label set.

Examples

1) Wikipedia article page categories
2) Amazon product recommendation system

Properties

large number of labels (> 1000) + "long-tailed" label distribution

Experiments and Results

↪ Datasets

- INPI-296k
  subclass, group
- USPTO-2M [Li et al., 2018]
  subclass

Experiments and Results

↪ Datasets

- INPI-296k
- USPTO-2M [Li et al., 2018]

↪ Baselines

- Logistic Regression
- FastText [Mokolov et al., 2016]
- Bert [Devlin et al., 2018]
- Bert for Patents
  [Srebrovic and Yonamine, 2020]

Experiments and Results

↪ Datasets

- INPI-296k
- USPTO-2M [Li et al., 2018]

↪ Baselines

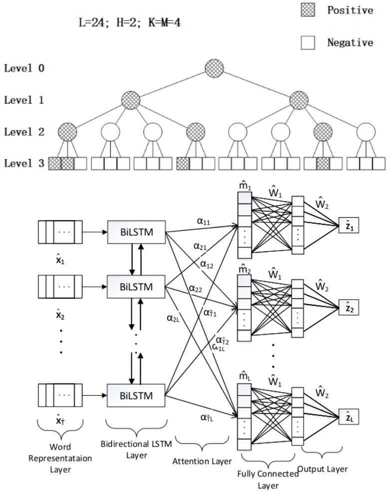
- Logistic Regression
- FastText [Mokolov et al., 2016]
- Bert [Devlin et al., 2018]
- Bert for Patents [Srebrovic and Yonamine, 2020]

↪ Selected XML models

- XML-CNN [Liu et al., 2017]
- Parabel [Prabhu et al., 2018]
- AttentionXML [You et al., 2019]
- LightXML [Jiang et al., 2021]

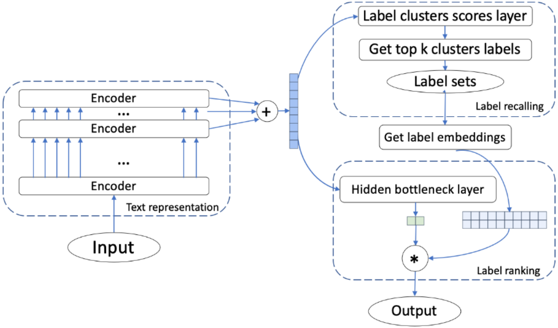
Experiments and Results

- AttentionXML [You et al., 2019]



Experiments and Results

- AttentionXML [You et al., 2019]
- LightXML [Jiang et al., 2021]



Experiments and Results

$\hookrightarrow$ Evaluation: *Precision@k* ($k = 1, 3, 5$)

For a predicted score vector $\hat{y} \in \mathbb{R}^L$ and ground truth label vector $y \in \{0,1\}^L$

$$Precision@k := \frac{1}{k} \sum_{l \in rank_k(\hat{y})} y_l$$

Experiments and Results

↪ Main results

Model	P@1	P@3	P@5
Logistic Regression	74.63	41.66	28.82
FastText	73.89	40.55	28.02
bert-large	83.77	46.27	31.37
Bert for Patents	84.31	46.73	31.73
XML-CNN	57.00	31.22	22.08
Parabel	74.43	41.49	28.50
AttentionXML	82.49	45.15	30.82
LightXML	**84.43**	**46.81**	**31.91**

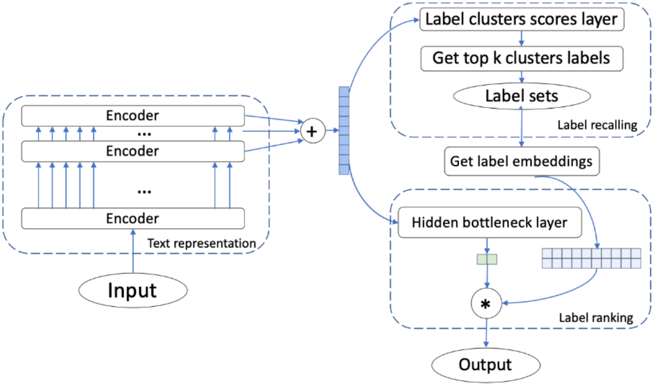
Table: Overall Performance (%) on IPC subclass on USPTO-2M (title + abstract)

Model	subclass			group		
	P@1	P@3	P@5	P@1	P@3	P@5
Logistic Regression	65.87	37.63	26.02	49.12	30.32	22.06
FastText	53.76	30.64	21.31	36.21	22.32	16.35
XML-CNN	43.43	25.50	18.23	17.74	10.20	6.96
Parabel	65.13	36.87	25.32	48.93	30.61	22.28
AttentionXML	72.54	40.68	27.63	54.83	33.78	24.49
LightXML	**76.45**	**42.82**	**29.05**	**60.60**	**36.95**	**26.65**

Table: Overall Performance (%) on IPC subclass and group on INPI-296k (title + abstract)

Experiments and Results

1. LightXML ourperforms all the other models
   - powerful text representation from transformer encoder
   - dynamic negative sampling
   - ensemble learning



Experiments and Results

1. LightXML ourperforms all the other models
2. best combinations of input = title + description

Model	IPC level	P@1	P@3	P@5
AttentionXML	subclass	76.60 (+4.06)	42.45 (+1.77)	28.71 (+1.08)
AttentionXML	group	59.27 (+4.44)	36.23 (+2.45)	26.03 (+1.54)
LightXML	subclass	76.72 (+0.27)	43.01 (+0.19)	29.17 (+0.12)
LightXML	group	62.33 (+1.73)	38.11 (+1.16)	27.48 (+0.83)

Table: Certain model performances (%) on IPC subclass and group on INPI-296k (**title + description**) (The numbers in parentheses indicate the boost relative to the model trained on (title+abstract)).

Experiments and Results

1. LightXML ourperforms all the other models
2. best combinations of input = title + description
3. error analysis
   - weaker models perform worse in learning to classify those "long-tailed" labels
   - tendency to mistake "long-tailed" labels for those more frequent labels

Conclusions

1. proposed a French Patent corpus INPI-296k
2. XML methods can be applied for IPC classification in deeper levels
3. Patents with "low resource" patent language can also achieve good results in patent classification
4. github: https://github.com/ZoeYou/Patent-Classification-2022

Onging and Future Works

1. Limitations on the length of the input text: extracting key sentences and passages as model input
2. "long-tailed" labels: using label descriptions and correlation between labels as input information
3. Evolution of the annual label distribution: adjustment of the training data order during training

Thank you for your attention!

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. (2018)
Bert: Pre-training of deep bidirectional transformers for language understanding.
CoRR, abs/1810.04805.

Ting Jiang, Deqing Wang, Leilei Sun, Huayi Yang, Zhengyang Zhao, and Fuzhen Zhuang. (2021)
Lightxml: Transformer with dynamic negative sampling for high-performance extreme multi-label text classification.
arXiv preprint arXiv:2101.03305 (2021).

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. (2016)
Bag of Tricks for Efficient Text Classification.
arXiv:1607.01759[cs.CL]

Shaobo Li, Jie Hu, Yuxin Cui, and Jianjun Hu. (2018)
DeepPatent: Patent Classification with Convolutional Neural Networks and Word Embedding.
Scientometrics 117, 2 (nov 2018), 721–744.

References

Jingzhou Liu, Wei-Cheng Chang, Yuexin Wu, and Yiming Yang. (2017)
Deep Learning for Extreme Multi-label Text Classification.
Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (2017).

Yashoteja Prabhu, Anil Kag, Shrutendra Harsola, Rahul Agrawal, and Manik Varma. (2018)
Parabel: Partitioned Label Trees for Extreme Classification with Application to Dynamic Search Advertising
WWW '18 Proceedings of the 2018 World Wide Web Conference, 993–1002.

Rob Srebrovic and Jay Yonamine. (2020)
Leveraging the BERT algorithm for Patents with TensorFlow and BigQuery.
Technical Report. Global Patents, Google
https://services.google.com/fh/files/blogs/bert_for_patents_white_paper.pdf

Ronghui You, Zihan Zhang, Ziye Wang, Suyang Dai, Hiroshi Mamitsuka, and Shanfeng Zhu. (2019)
Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification.
Advances in Neural Information Processing Systems 32 (2019).