Construction de modèles de TAL interprétables à l'aide de prototypes

« Explanations for NLP models from case-based reasoning »

Séminaire TALia du 6/5/2022

Référence 1

arXiv > cs > arXiv:1907.09728

Computer Science > Machine Learning

[Submitted on 23 Jul 2019]

Interpretable and Steerable Sequence Learning via Prototypes

Yao Ming, Panpan Xu, Huamin Qu, Liu Ren

ProSeNet

Référence 2

arXiv > cs > arXiv:2007.01777

Computer Science > Machine Learning

[Submitted on 3 Jul 2020 (v1), last revised 15 Aug 2021 (this version, v2)]

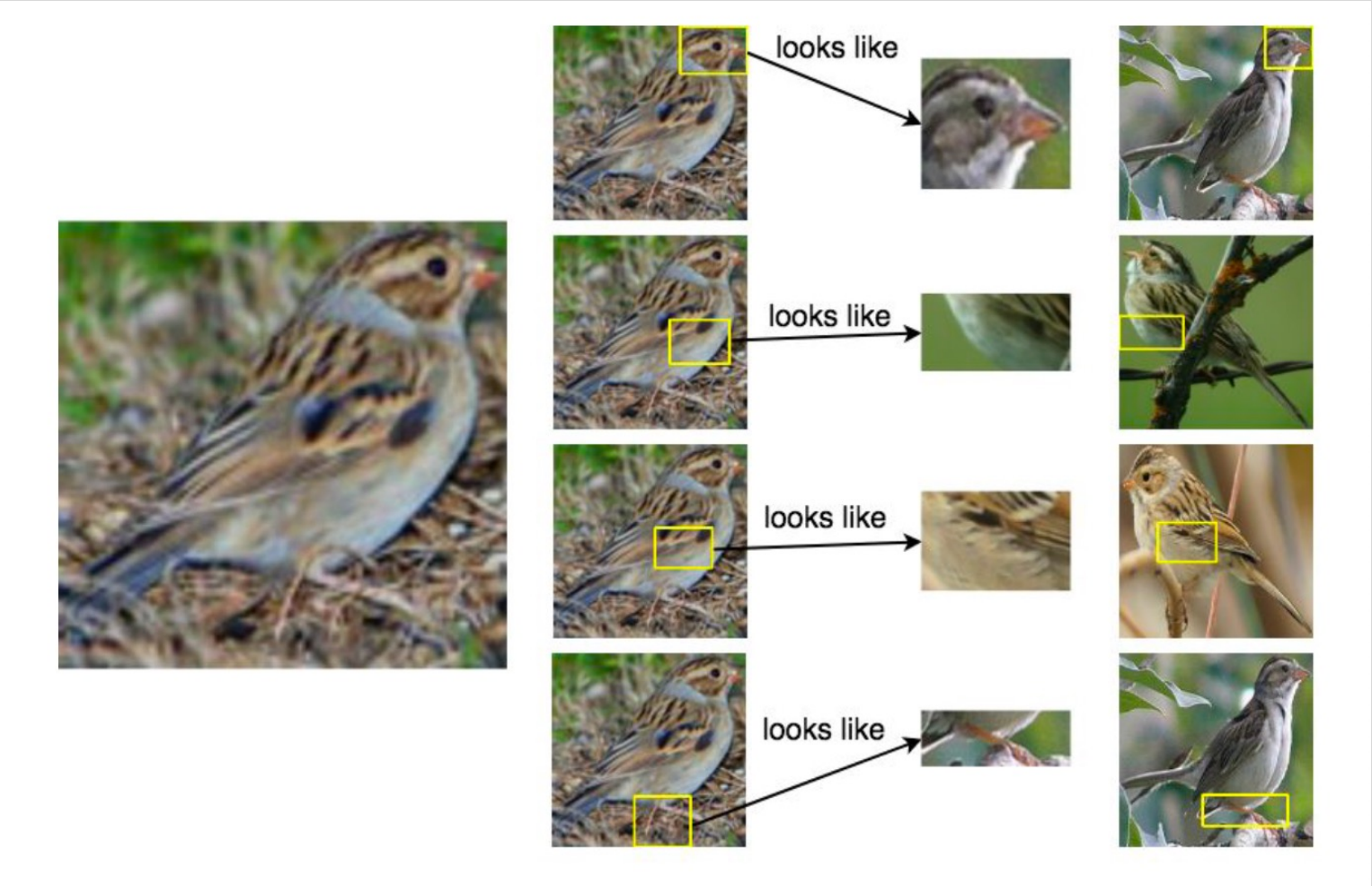
Interpretable Sequence Classification Via Prototype Trajectory

Dat Hong, Stephen S. Baek, Tong Wang

ProtoryNet

Plan

1. L'**interprétabilité** des modèles de ML selon Cynthia Rudin & Co
2. **ProSeNet** : Un modèle de classification à l'aide de prototypes pour des textes courts.
3. **ProtoryNet** : Un modèle de classification à l'aide de prototypes pour des paragraphes de plusieurs phrases.



L'interprétabilité des modèles de ML selon Cynthia Rudin

" If we do not succeed at these efforts, it is possible that black box models will continue to be permitted when it is not safe to use them. Since the definition of what constitutes a viable explanation is unclear, even strong regulations such as "right to explanation" can be undermined with less-than-satisfactory explanations. […] This may conti decisions throughout our crimi incorrect safety guidance for a incomprehensible loan deci widespread societal problems. "

∃ Rashomon Sets de modèles ML qu'on ne peut trouver par simple optimisation. Voir p.ex. page 45 de réf 4.

référence 3

référence 4

Des principes (très) généraux

1. Un modèle de ML interprétable vérifie des contraintes spécifiques à un domaine afin d'en faciliter la compréhension par les humains : parcimonie, monotonicité, décomposabilité en sous-problèmes, désintrication, contraintes physiques, singer le **raisonnement par analogie**,…
2. Un modèle interprétable ne crée pas nécessairement la confiance mais permet à un humain de prendre une décision éclairée
3. La nécessité de sacrifier la précision à l'interprétabilité d'un modèle de ML est une légende urbaine !
4. Les métriques de performance et d'interprétabilité doivent être définies par itérations successives

Des questions ouvertes par catégorie de modèles

- Modèles logiques parcimonieux (arbres de décision, forme DNF,…)
- Systèmes de scoring
- Modèles additifs généralisés (GAM ⊃ GLM)
- **Modern case-based reasoning** : cette présentation
- Désintrication des RN
- **Caractérisation des Rashomon Sets** : intéressant !

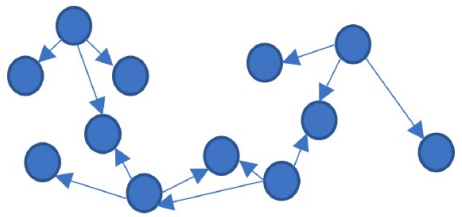
Le case-based reasoning et l'idée de prototype

Case-based reasoning

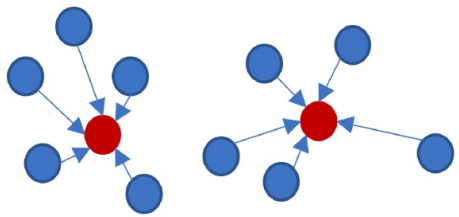
- Un paradigme pour résoudre des problèmes à partir de solutions connues de problèmes similaires
- Une simulation d'un raisonnement par analogie qui a l'avantage d'être interprétable
- De nombreux cas d'applications pratiques

kNN

- Une méthode non paramétrique sans entraînement
- Chercher les k plus proches voisins, au sens d'une distance ℓ^2 dans un espace approprié
- On peut utiliser l'espace latent d'un RN pour créer des notions de distances appropriées
- Attribuer un label identique à celui de la majorité des k plus proches voisins voisins



kNN



prototype reasoning

Prototype Reasoning

- Pour éviter la comparaison de chaque observation à toutes les autres on construit quelques prototypes représentatifs bien choisis
- Les voisins pourraient ne pas être représentatifs
- Un prototype peut être conçu comme un exemple « idéal »
- Un prototype n'appartient pas nécessairement au train set
- Si les prototypes sont interprétables et peu nombreux le modèle sera interprétable.
- Récemment intégré à aux approches par DL

ProSeNet : classification interprétable de textes courts

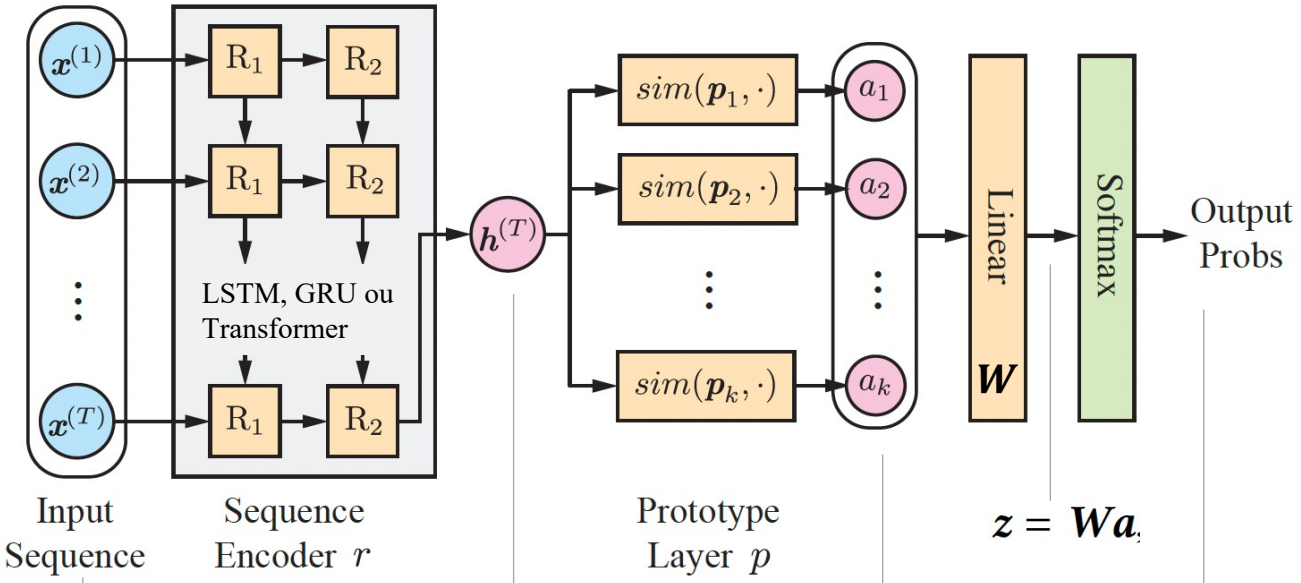
- Un modèle de classification interprétable de textes courts (<25 mots)
- Une combinaison d'un simple RNN avec l'idée d'apprentissage par prototypes

interprétabilité

- Des prototypes **simples** : ne contiennent que les mots clés du train set $\mathcal{D}$
- Des prototypes **diversifiés** : éviter la redondance
- Des représentations **parcimonieuses** : nombre k de prototypes actifs faible

BONUS

- Possibilité d'incorporer du savoir à priori : faire choisir des prototypes $\boldsymbol{p}_i$ par des experts plutôt que les apprendre à l'entraînement.



$(\boldsymbol{x}^{(t)})_{t=1}^{T}$

$\boldsymbol{x}^{(t)} \in \mathbb{R}^n$

$\boldsymbol{e} = r((\boldsymbol{x}^{(t)})_{t=1}^{T})$

$a_i = \exp(-d_i^2)$

similarités

$d_i^2 = \|\boldsymbol{e} - \boldsymbol{p}_i\|_2^2$

$\hat{\boldsymbol{y}} = \mathrm{softmax}(\boldsymbol{z})$

train set $\mathcal{D} = \{((\boldsymbol{x}^{(t)})_{t=1}^{T}, y)\}$

classes $y \in \{1, \ldots, C\}$

prototype apprenable

ProSeNet : fonction de coût et contraintes

Cross-entropie binaire : $CE(\Theta, \mathcal{D}) = \sum_{((\boldsymbol{x}^{(t)})_{t=1}^{T}, \boldsymbol{y}) \in \mathcal{D}} \boldsymbol{y} \log(\hat{\boldsymbol{y}}) + (1 - \boldsymbol{y}) \log(1 - \hat{\boldsymbol{y}})$

Favoriser la **diversité** des prototypes : $R_d(\Theta) = \sum_{i=1}^{k} \sum_{j=i+1}^{k} \max\left(0, d_{\min} - \|\boldsymbol{p}_i - \boldsymbol{p}_j\|_2\right)^2$

Favoriser les représentations **parcimonieuses** et **additives** : $\ell_{\text{sparse}}(\Theta) = \|\boldsymbol{W}\|_1$ et $W_{ij} > 0$

Favoriser le **clustering** autour des prototypes : $R_c(\Theta, \mathcal{D}) = \sum_{(\boldsymbol{x}^{(t)})_{t=1}^{T} \in \mathcal{X}} \min_{i=1}^{k} \left\| r\left((\boldsymbol{x}^{(t)})_{t=1}^{T}\right) - \boldsymbol{p}_i \right\|_2^2$

Favoriser l'**interprétabilité** de chaque prototype : $R_e(\Theta, \mathcal{D}) = \sum_{i=1}^{k} \min_{(\boldsymbol{x}^{(t)})_{t=1}^{T} \in \mathcal{X}} \left\| \boldsymbol{p}_i - r\left((\boldsymbol{x}^{(t)})_{t=1}^{T}\right) \right\|_2^2$

Fonction de coût complète :

$$\ell_{\text{total}}(\Theta, \mathcal{D}) = CE(\Theta, \mathcal{D}) + \lambda_c R_c(\Theta, \mathcal{D}) + \lambda_e R_e(\Theta, \mathcal{D}) + \lambda_d R_d(\Theta, \mathcal{D}) + \lambda_{l_1} \|\boldsymbol{W}\|_1$$

ProSeNet : entraînement

Entraînement usuel par DGS + …

… **projection** des prototypes sur des embeddings les plus proches après quelques époques

$$\boldsymbol{p}_i \leftarrow \underset{\boldsymbol{e} \in r(\mathcal{X})}{\arg\min} \left\| \boldsymbol{e} - \boldsymbol{p}_i \right\|_2$$

Garantit l'interprétabilité du prototype $\boldsymbol{p}_i$ tout en évitant le coût d'une opération complexe de décodage d'un auto-encodeur

… **simplification** des prototypes avec un beam search

$$\mathrm{seq}_i = \underset{\mathrm{seq} \subset \mathrm{sub}(\chi)}{\arg\min} \left(\| r(\mathrm{seq}) - \boldsymbol{p}_i \|_2 \right)$$

$$\boldsymbol{p}_i \leftarrow r(\mathrm{seq}_i)$$

$$O(2^T N) \xrightarrow{\text{beam search}} O(w\, T^2 N)$$

Longueur des séquences

Taille du train set

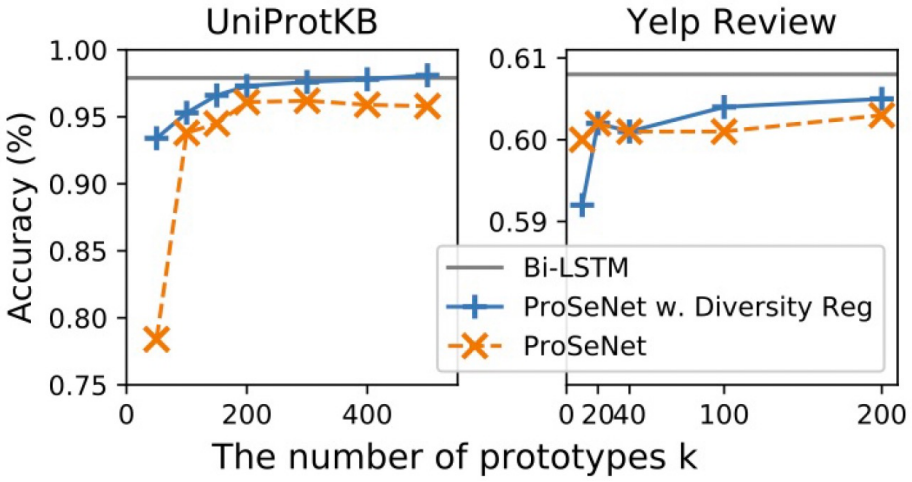
Projection sur la meilleure sous-séquence du trainset plutôt que sur une phrase complète

… **mise à jour** des prototypes grâce à une expertise humaine

ProSeNet : résultats

Un exemple d'interprétation

Input: pizza is good but service is extremely slow
Prediction: Negative
Explanation: 0.69 * good food but worst service (Negative 2.1)
+ 0.30 * service is really slow (Negative 1.1)



Importance du nombre de prototypes et de la régularisation de diversité R_d.

Comparaison avec les modèles en boîte noire

Model	Binary	Fine-grained
ProSeNet	95.5 (0.1)	60.7 (0.2)
$\text{ProSeNet}_{\text{Bi-LSTM}}$	95.5 (0.2)	61.0 (0.3)
LSTM	95.6 (0.1)	60.7 (0.3)
Bi-LSTM	95.7 (0.1)	61.1 (0.3)
ResNet	95.3 (0.3)	60.6 (0.4)

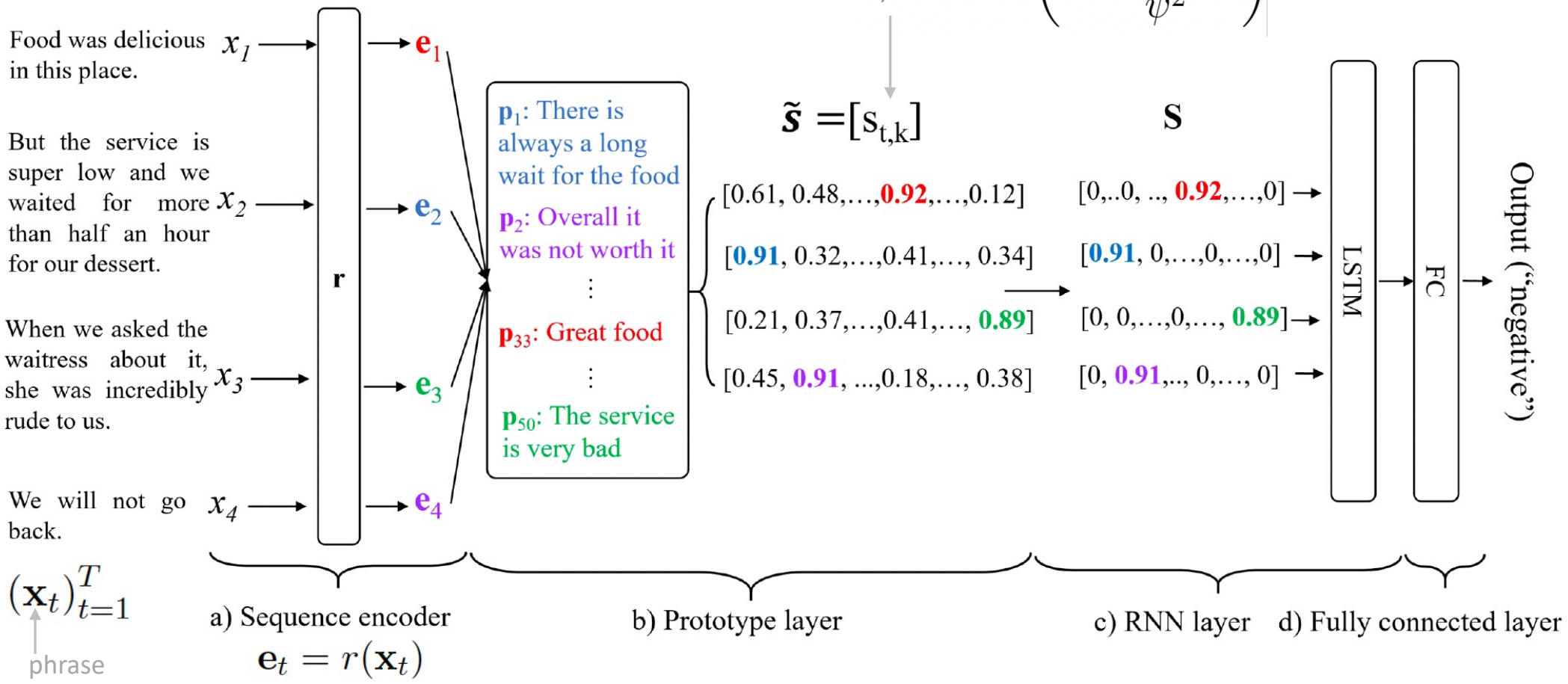
Accrédite l'idée qu'il existe des modèles interprétables aussi bons que les modèles en boite noire !

ProtoryNet : prototypes pour classer des paragraphes

- Une variante du modèle ProSeNet
- S'applique à des paragraphes vus comme des suites de phrases
- Associe un prototype à chaque phrase
- Part du constat qu'il est impossible de trouver un petit nombre prototypes pertinents pour évaluer la tonalité d'un paragraphe
- Classe un paragraphe en fonction de la séquence des prototypes associés à chaque phrase
- Conforte, là encore, encore l'idée qu'il existe des modèles interprétables aussi précis que les modèles en boîte noire

ProtoryNet : l'architecture

$$s_{t,k} := \exp\left(-\frac{d(\mathbf{e}_t, \mathbf{p}_k)}{\psi^2}\right)$$



$$\mathbf{e}_t = r(\mathbf{x}_t)$$

$$\mathbf{X} = (\mathbf{x}_t)_{t=1}^{T}$$

phrase

paragraphe

train set $\mathcal{D} = \{(\mathbf{X}^{(i)}, \mathbf{y}^{(i)}) : i = 1, \ldots, N\}$

un paragraphe = une suite de phrases $\mathbf{X}^{(i)}$

un multi label $\mathbf{y}^{(i)}$

ProtoryNet : fonction de coût et contraintes

Maximisation de la précision

$$\mathcal{L}_{\text{acc}}(\mathcal{D}) := \frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{y}^{(i)} - \hat{\mathbf{y}}^{(i)}\right\|^2$$

labels multi hot encodés

Favoriser la **diversité** des prototypes :

$$d_{\min} := \min_{k_1,k_2} d(\mathbf{p}_{k_1}, \mathbf{p}_{k_2})$$

$$\mathcal{L}_{\text{div}}(\mathcal{D}) := \sigma\left(\eta(\delta - d_{\min})\right)$$

Favoriser l'**interprétabilité** de chaque prototype :

$$\mathcal{L}_{\text{proto}} = \frac{1}{M}\sum_{\mathbf{X}\in\mathcal{D}}\sum_{\mathbf{x}_t\in\mathbf{X}} \min_k d(\mathbf{r}(\mathbf{x}_t), \mathbf{p}_k)$$

$M =$ nombre total de phrases

Fonction de coût complète :

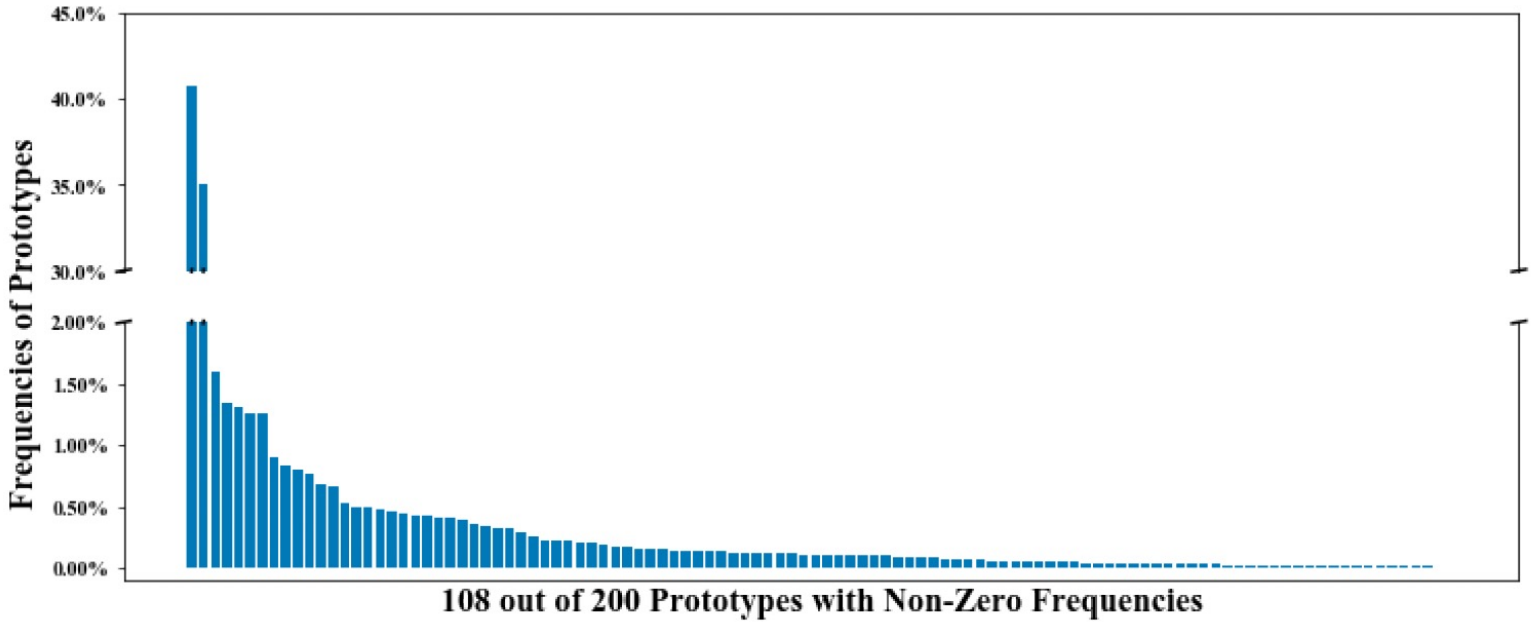
$$\mathcal{L} = \mathcal{L}_{\text{acc}} + \alpha\mathcal{L}_{\text{div}} + \beta\mathcal{L}_{\text{proto}}$$

Favoriser le **clustering** autour des prototypes à l'initialisation de l'entraînement :

clustering des embeddings des phrases puis utilisation des centroïdes pour initialiser les prototypes $\mathbf{p}_k$

ProtoryNet : projection et élagage

$$\mathbf{s}_k = \underset{\mathbf{x}_t \in X^{(i)}, \forall X^{(i)} \in \mathcal{D}}{\operatorname{argmin}} d(\mathbf{r}(\mathbf{x}_t), \mathbf{p}_k), \qquad k \in [1, K]$$



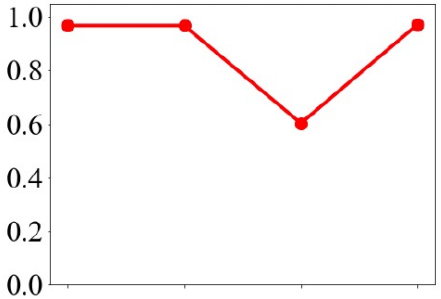
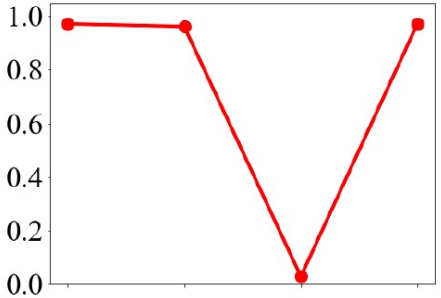
Data set	$\hat{K}$
IMDB	8
Amazon Reviews	14
Yelp Reviews	15
Rotten Tomatoes	23
Hotel Reviews	7
Steam Reviews	17

ProtoryNet : résultats (1)

Data set	DistilBERT	**ProtoryNet** (Fine-tuned)	**ProtoryNet** (Not fine-tuned)	ProSeNet	Vanilla LSTM	Bag-of-words
IMDB	0.931	0.914	0.871	0.863	0.871	0.877
Amazon Reviews	0.940	0.918	0.890	0.875	0.884	0.830
Yelp Reviews	0.967	0.962	0.941	0.932	0.952	0.908
Rotten Tomatoes	0.903	0.881	0.771	0.869	0.877	0.785
Hotel Reviews	0.976	0.961	0.949	0.930	0.949	0.905
Steam Reviews	0.955	0.924	0.876	0.834	0.864	0.844

ProtoryNet : résultats (2)

Table 5: Two Positive Examples

	Input Text	Prototype	Trajectory
Example 1	①This was our first visit to Paradise Bakery and all I can say is YUM	3	
	② It was so good, I went back for lunch the next day	3	
	③ The dining room is very pleasant and clean, the service is great and the sandwiches are super yummy	6	
	④ I love having this fairly close to our house	1	
Example 2	①This place is fantastic	1	
	② Impeccable service, great atmosphere and outstanding food	4	
	③Yes, it's pricey but well worth it.	9	
	④I've been here a couple of times and it never disappoints	1	

Conclusion

- **Définition** (C. Rudin) : *Un modèle de ML interprétable vérifie des contraintes spécifiques à un domaine qui en facilite la compréhension par les humains.*

- L'utilisation de prototypes en ML ne constitue pas une révolution de l'IA ou du NLP.
- C'est une idée simple et opérationnelle à rajouter dans notre besace d'outils NLP.
- Un exemple d'intégration d'intuitions linguistiques à des modèles DL au prix d'un peu d'imagination et de bricolage?
- Accrédite l'idée que des modèles interprétables n'ont pas nécessairement à payer un prix important en termes de précision si on les compare aux modèles en boîte noire usuels.

- **Question** : Quelle formalisation mathématique pour l'idée des Rashomon Sets ?