AN OVERVIEW OF WORD REPRESENTATIONS

Language Reading Group

November 29, 2018

Étienne

LIP6 - Sorbonne Université

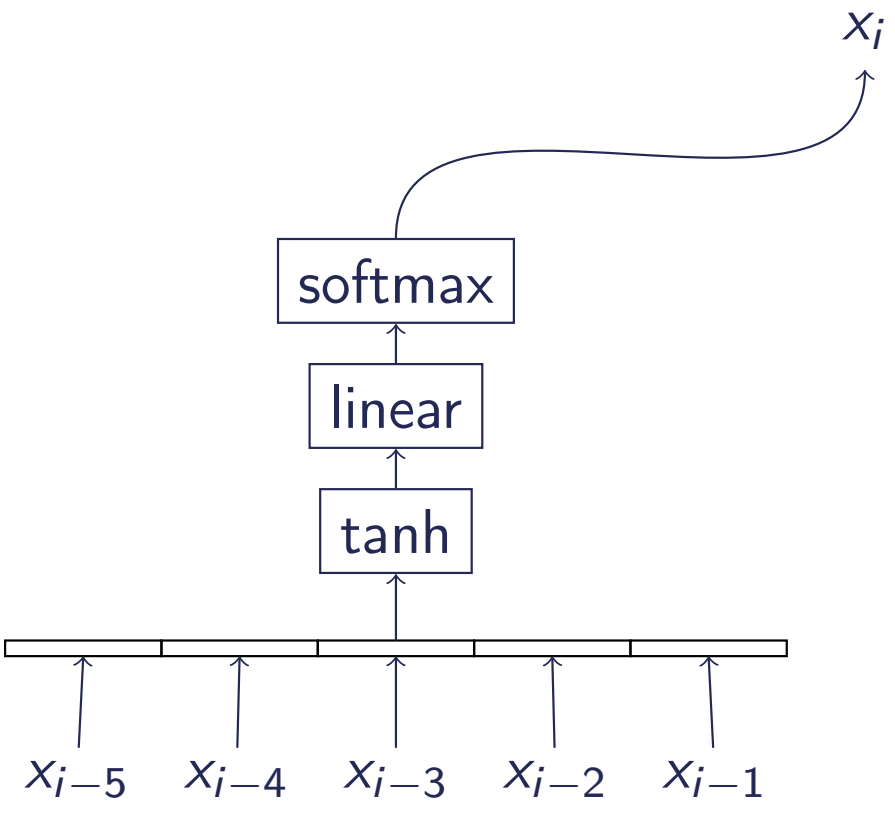
Overview

1. Classic Word Representations
   - Préhistoire
   - Word2Vec
   - GloVe
   - FastText
2. Contextualized Word Representations
   - CoVe
   - ELMo
   - GPT
   - BERT

Classic Word Representations

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model.

Journal of machine learning research, 3(Feb):1137–1155, 2003



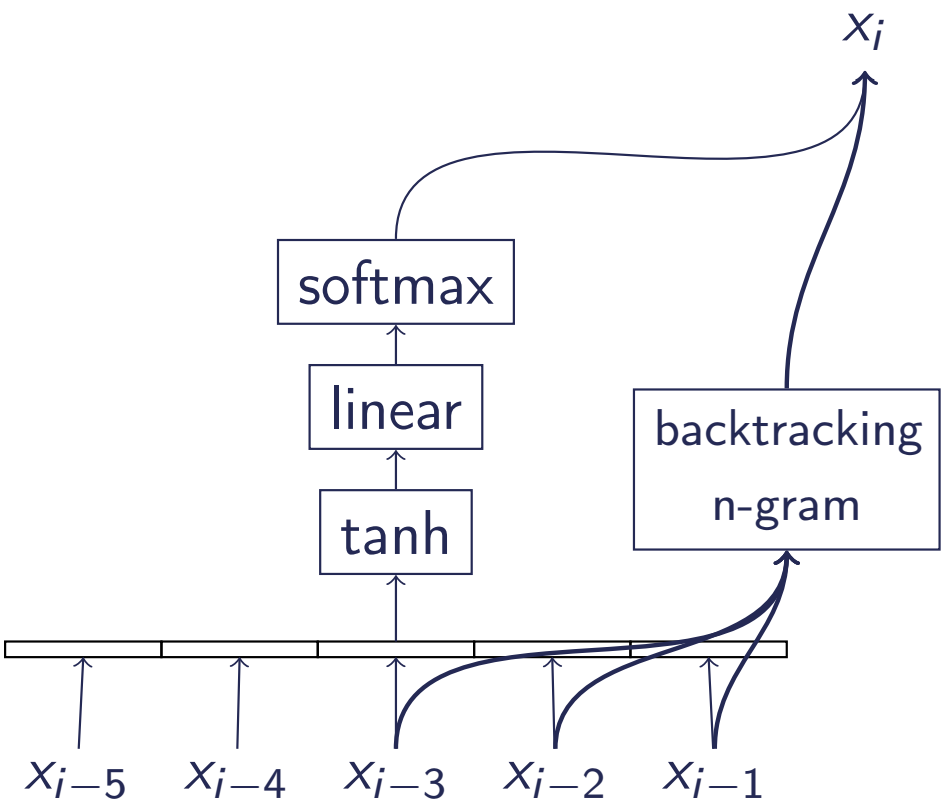
$$p(w|\boldsymbol{c}) = \frac{\exp s(w, \boldsymbol{c})}{\sum_{\overline{w} \in V} \exp s(\overline{w}, \boldsymbol{c})}$$

Language modeling

	ppl
bt n-gram	312
Bengio	276

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model.

Journal of machine learning research, 3(Feb):1137–1155, 2003



$$p(w|\boldsymbol{c}) = \frac{\exp s(w, \boldsymbol{c})}{\sum_{\overline{w} \in V} \exp s(\overline{w}, \boldsymbol{c})}$$

Language modeling

	ppl
bt n-gram	312
Bengio	276
mixture	252

$$p(c \mid w) = \frac{\exp s(w, c)}{\sum_{\overline{c} \in V} \exp s(w, \overline{c})}$$

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space.

arXiv preprint arXiv:1301.3781, 2013a

$$p(c \mid w) = \frac{\exp s(w, c)}{\sum_{\overline{c} \in V} \exp s(w, \overline{c})}$$

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space.

arXiv preprint arXiv:1301.3781, 2013a

Ne décrit pas vraiment ce qu'il fait...

$$p(c \mid w) = \frac{\exp s(w, c)}{\sum_{\overline{c} \in V} \exp s(w, \overline{c})}$$

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space.

arXiv preprint arXiv:1301.3781, 2013a

Ne décrit pas vraiment ce qu'il fait...

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean.

Distributed representations of words and phrases and their compositionality.

In *Advances in neural information processing systems*, pages 3111–3119, 2013b

Noise Contrastive Estimation (NCE)

Discriminer entre des vrais contexte et k contextes tirés aléatoirements:

$$p(D = 1 \mid w, c) = \frac{p(c \mid w)}{p(c \mid w) + kP_{\mathfrak{U}}(c)}$$

Noise Contrastive Estimation (NCE)

Discriminer entre des vrais contexte et k contextes tirés aléatoirements:

$$p(D = 1 \mid w, c) = \frac{p(c \mid w)}{p(c \mid w) + kP_{\mathfrak{U}}(c)}$$

$$\mathrm{J}_{\mathrm{NCE}}(w, c) = \log p(D = 1 \mid w, c) + \sum_{i=1}^{k} \mathop{\mathbb{E}}_{\overline{c}_i \sim \mathfrak{U}} \log p(D = 0 \mid w, \overline{c}_i)$$

Noise Contrastive Estimation (NCE)

Discriminer entre des vrais contexte et k contextes tirés aléatoirements:

$$p(D = 1 \mid w, c) = \frac{p(c \mid w)}{p(c \mid w) + kP_{\mathfrak{U}}(c)}$$

$$\mathrm{J}_{\mathrm{NCE}}(w, c) = \log p(D = 1 \mid w, c) + \sum_{i=1}^{k} \mathop{\mathbb{E}}_{\overline{c}_i \sim \mathfrak{U}} \log p(D = 0 \mid w, \overline{c}_i)$$

Negative Sampling

$$p(D = 1 \mid w, c) = \frac{p(c \mid w)}{p(c \mid w) + 1}$$

Noise Contrastive Estimation (NCE)

Discriminer entre des vrais contexte et k contextes tirés aléatoirements:

$$p(D = 1 \mid w, c) = \frac{p(c \mid w)}{p(c \mid w) + kP_{\mathfrak{U}}(c)}$$

$$\mathrm{J}_{\mathrm{NCE}}(w, c) = \log p(D = 1 \mid w, c) + \sum_{i=1}^{k} \mathop{\mathbb{E}}_{\overline{c}_i \sim \mathfrak{U}} \log p(D = 0 \mid w, \overline{c}_i)$$

Negative Sampling

$$p(D = 1 \mid w, c) = \frac{p(c \mid w)}{p(c \mid w) + 1}$$

$$\mathrm{J}_{\mathrm{NEG}}(w, c) = \log \sigma(w^{\intercal} c) + \sum_{i=1}^{k} \mathop{\mathbb{E}}_{\overline{c}_i \sim \mathfrak{U}} \log \sigma(-w^{\intercal} \overline{c}_i)$$

Accuracy sur un dataset Google News:

Model	Semantic	Syntactic	Total	Training Time
Bengio	34.2	64.5	50.8	14 days
Word2Vec	66.1	65.1	65.6	2.5 days

Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation.

In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014

Soit X_{ij} le nombre de co-occurrences du mot i avec le mot j :

$$X_{ij} = \sum_{d} \frac{\text{Nombre de fois que } j \text{ apparait à distance } d \text{ de } i}{d}$$

On cherche à factoriser X par le produit de deux matrices de rang faible w (mots) et c (contexte).

$$\mathrm{J}_{\mathrm{GloVe}} = \sum_{i,j=1}^{|V|} \left(w_i^{\mathsf{T}} c_j + b_i^w + b_j^c - \log(1 + X_{ij}) \right)^2$$

À la fin GloVe $= w + c$

Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation.

In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014

Soit X_{ij} le nombre de co-occurrences du mot i avec le mot j :

$$X_{ij} = \sum_{d} \frac{\text{Nombre de fois que } j \text{ apparait à distance } d \text{ de } i}{d}$$

On cherche à factoriser X par le produit de deux matrices de rang faible w (mots) et c (contexte).

$$\mathrm{J}_{\mathrm{GloVe}} = \sum_{i,j=1}^{|V|} \left(w_i^{\mathsf{T}} c_j + b_i^w + b_j^c - \log(1 + X_{ij}) \right)^2$$

À la fin GloVe $= w + c$

Global = fenetre de taille 10.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016

Word2Vec avec les facteurs du mots. Le mot `where` est découpé en:

`<wh,` `whe,` `her,` `ere,` `re>` et `<where>`

Les mots du contexte restent non-découpés.
Chaque facteur correspond à un sample lors de l'apprentissage.

Les embeddings des facteurs sont additionnés pour produire une représentation du mot.

Contextualized Word Representations

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*, pages 6294–6305, 2017

Entrainer un bi-LSTM avec attention sur une tache de traduction, soit MT-LSTM l'encoder.

$$\text{CoVe}(w) = \text{MT-LSTM}(\text{GloVe}(w))$$

La concaténation $[\text{CoVe}(w); \text{GloVe}(w)]$ est utilisé comme input d'autre tâches.

L'embedding du mot w dépend de toute la phrase, d'où le **contextualized** word representation.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner,

Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations.

arXiv preprint arXiv:1802.05365, 2018

Entrainer un bi-LSTM à L couches ($\boldsymbol{h}_j$ pour $1 \leq j \leq L$) pour du language modeling.

On remarque que:

- $\boldsymbol{h}_L$ encode la sémantique des mots.
- $\boldsymbol{h}_1$ encode la morphologie des mots.

$$\text{ELMo} = \gamma^{\text{task}} \sum_{j=0}^{L} s_j^{\text{task}} \boldsymbol{h}_j$$

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner,

Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations.

arXiv preprint arXiv:1802.05365, 2018

Entrainer un bi-LSTM à L couches ($\boldsymbol{h}_j$ pour $1 \leq j \leq L$) pour du language modeling.

On remarque que:

- $\boldsymbol{h}_L$ encode la sémantique des mots.
- $\boldsymbol{h}_1$ encode la morphologie des mots.

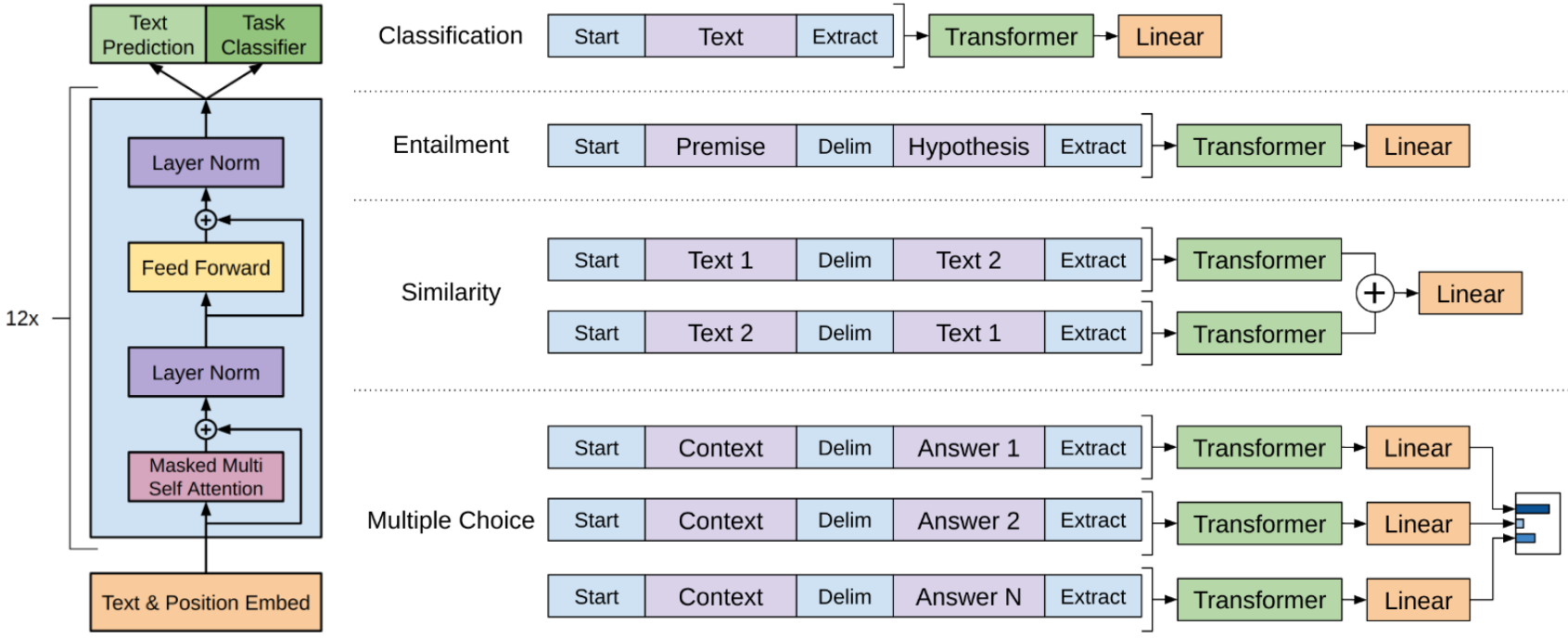
$$\text{ELMo} = \gamma^{\text{task}} \sum_{j=0}^{L} s_j^{\text{task}} \boldsymbol{h}_j$$

$L = 2$

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya

Sutskever. Improving language understanding by generative pre-training.

Preprint OpenAI, 2018



Fine-tune un transformer (entrainé pour du language modeling) avec un auxiliary objective en changeant seulement le linéaire final.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina

Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding.

arXiv preprint arXiv:1810.04805, 2018

Similaire à GPT mais pre-entrainé sur des tâches différentes:

1. Au lieu de prédire le prochain token comme dans un LM classique. Masque 15% des tokens et essaye de les prédires.
2. prédit si deux phrases données se succèdent l'une l'autre.

Model	GLUE average (%)
BiLSTM+ELMo+Attn	71.0
GPT	75.2
BERT	81.9