Interpreting Deep Models for Text Analysis via Optimization and Regularization Methods

AUTHOR: YUAN, H., CHEN, Y., HU, X., & JI, S.

PRESENTER: JASON YAO

Motivation

- Interpreting deep neural networks is of great importance to understand and verify deep models for natural language processing (NLP) tasks

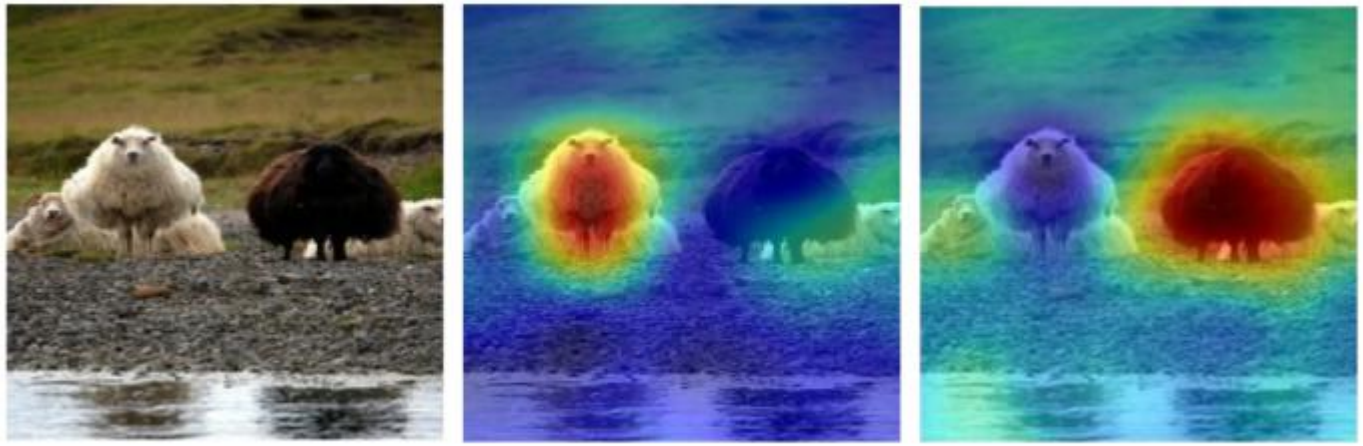
- Most existing approaches for interpreting NLP models only investigate the relationships between input sentences and output decisions to explore which input words are more important to make decisions

- there are no related studies focusing on the interpretation of hidden neurons of NLP models

Background - Visualization Technique

Saliency Map

- Initially used in Computer Vision, study which input pixels are more important to the final decision
- The importance of different pixels can be approximated by the gradient of output score with respect to the inputs
- In NLP, such technique only provide word-level interpretation while different words are highly correlated to convey a meaning



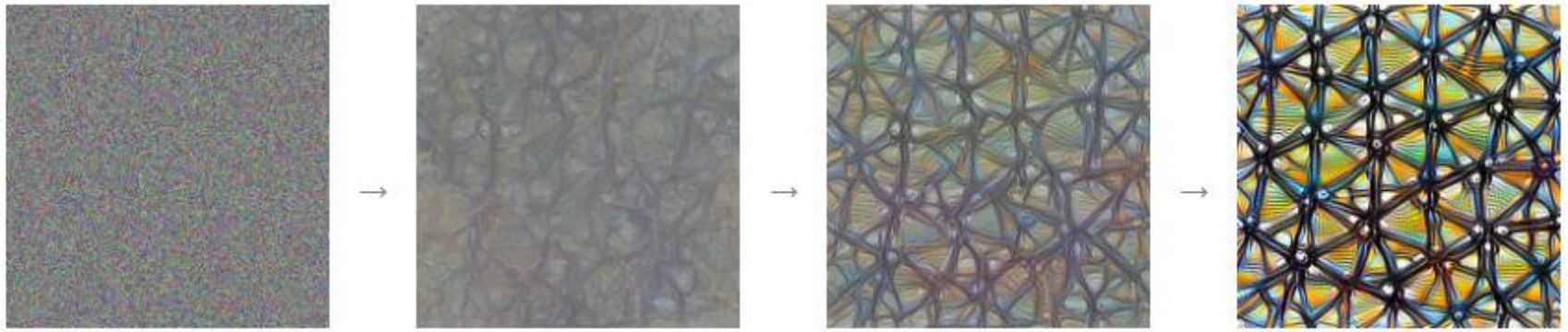
(a) Sheep - 26%, Cow - 17% (b) Importance map of '*sheep*' (c) Importance map of '*cow*'

Background - Visualization Technique

Optimization

- Investigates what pattern the hidden neurons of a model try to detect
- The key idea is to iteratively update a randomly initialized input towards a certain behavior, which is an objective function
- The optimized input can then be visualized as abstracted images to reflect the meaning
- In NLP models, the optimized input is a sequence of abstract vector representations and cannot be visualized as abstracted texts
- For example, optimize an image to maximize a particular neuron's activation and see what this neuron is looking at

Starting from random noise, we optimize an image to activate a particular neuron (layer mixed4a, unit 11).



Background - Visualization Technique

Optimization Example

Dataset Examples show us what neurons respond to in practice

Optimization isolates the causes of behavior from mere correlations. A neuron may not be detecting what you initially thought.

Baseball—or stripes?
mixed4a, Unit 6

Animal faces—or snouts?
mixed4a, Unit 240

Clouds—or fluffiness?
mixed4a, Unit 453

Buildings—or sky?
mixed4a, Unit 492

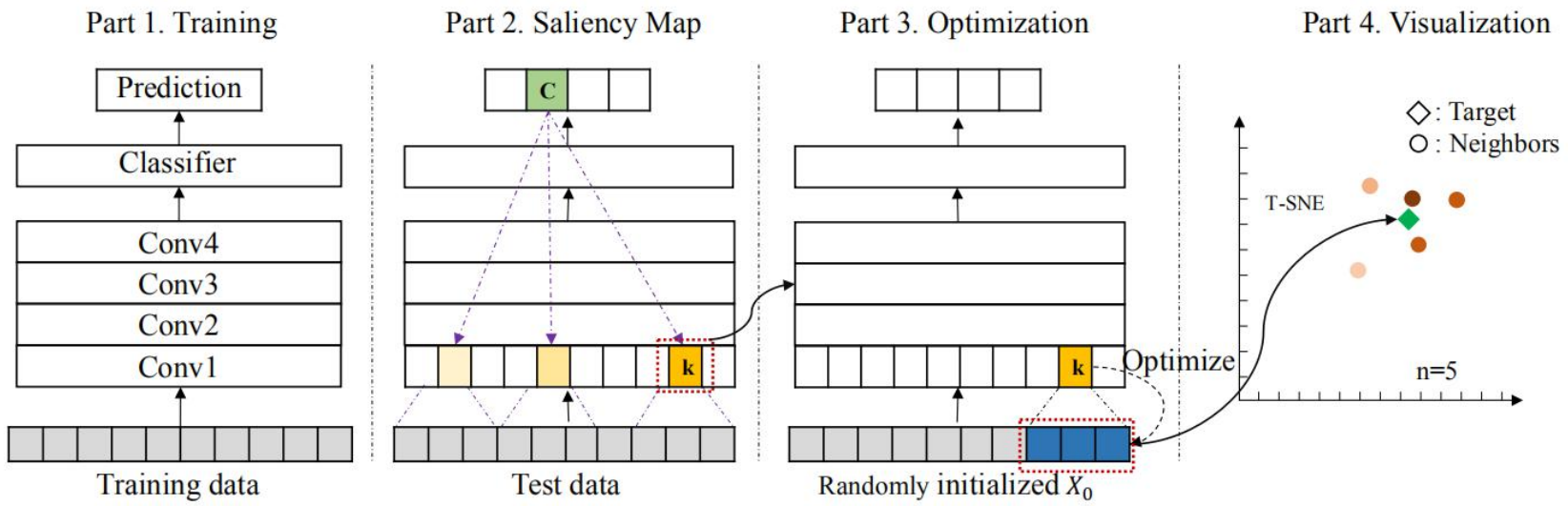
Background

- Neural Network Text Classification
  - Text Classification or Text Categorization is the activity of labeling natural language texts with relevant categories from a predefined set
  - CNN for Text Classification: treat word as a word embedding, for sentence we have a embedding matrix.
    - pretrain word embedding: each word in vocabulary is converted to a pretrained vector, which capture several features of words
  - Text Classification Example: Sentiment Analysis, a type of text classification, classify a text sentence into positive, negative or neural as the sentiment of the sentence

Methods

Overall Pipeline

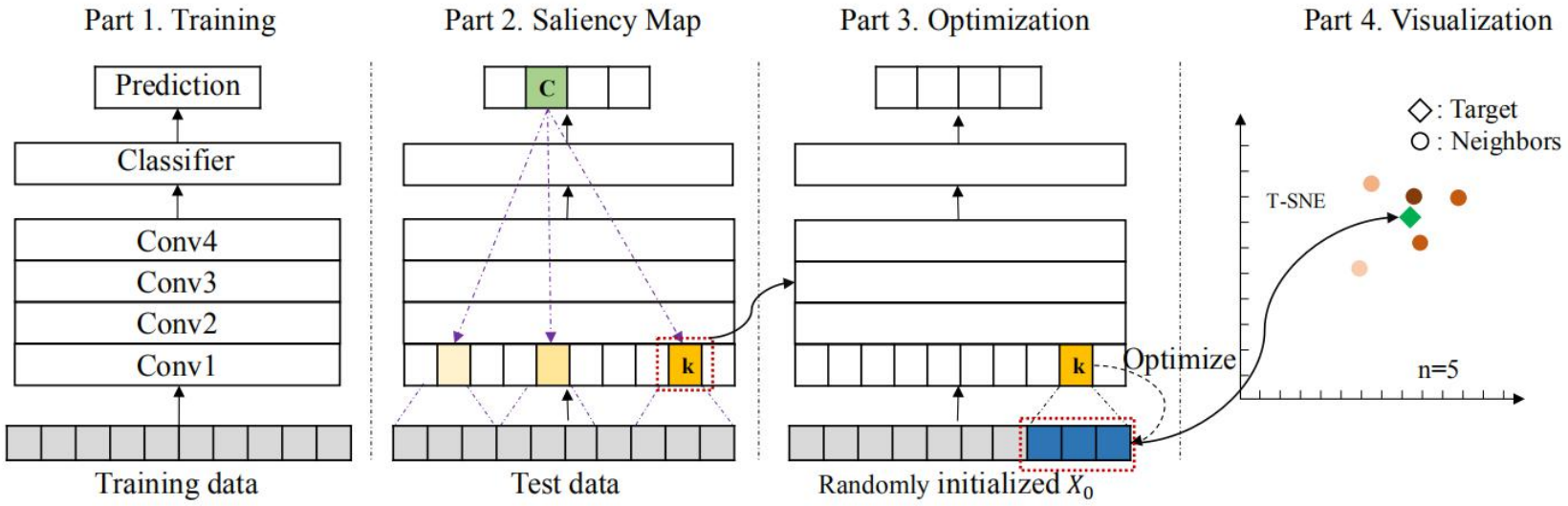
- Training a CNN for Sentence Classification: a general CNN model for text classification architecture
- Saliency Maps for Hidden Units: use saliency map to select spatial locations(hidden unit in the CNN) with high influence on the final decision
- Input Generation via Optimization: use optimization techniques to maximizing the activation value for selected location



Methods

Overall Pipeline

- Regularization: add regularization term to objective function to help convergence and visualization
- Visualization of Optimized Inputs: map optimize input back to word, present word with 2d T-SNE



Methods

- Saliency Maps for Hidden Units
  - In Li et al. 2015, saliency score is computed for each input word or word embedding
    - Given embedding E for input words with gold label c, we have a class score Sc(E) for pair (E, c), the goal is to decide which unit of E , which is e, make most contribution to Sc(E)
    - Approximate Sc(E) with a linear function of e by computing the first-order Taylor expansion
    - The magnitude (absolute value) of the derivative indicates the sensitiveness of the final decision to the change in e, telling us how much e contributes to the final decision
    - The saliency score is given by the absolute value of the derivative $S(e) = |w(e)|$
  - We want to compute saliency score for each hidden spatial location in the convolution neural net

Methods

- Saliency Maps for Hidden Units
  - To compute saliency score for each hidden spatial location in the convolution neural net
  - For class score Sc of give input sentence X and predicted class c, let aij represent the activation vector of ith neuron in layer j, let matrix Aj denotes the activation of layer j, we have similar approximation

$$S_c \approx \mathrm{Tr}(w(A_i)^T A_j) + b,$$

Tr(.) denotes the trace of a matrix

$$w(A_j) = \frac{\partial S_c}{\partial A_j}.$$

The gradient of class score with respect to the layer j

$$\mathrm{Score}_c(X)_{i,j} = w(A_j)_i \cdot a_{ij},$$

the saliency score with of location i in layer j

  - Gradient only reflect the sensitivity of the class score when there is a small change in the location
  - Activation value aij helps to measure how much neuron i contributes to final class score

Methods

- Input Generation via Optimization
  - To understand why these neurons are important
  - The activation value of each neuron shows the strength of the pattern detected from inputs
  - For neuron k in location i of layer j, let $\overline{X}_{ijk}$ be the optimized input and a_{ijk} be the activation value
    - $\overline{X}_{ij} = \sum_{k=1}^{n} a_{ijk} \overline{X}_{ijk}$, represent what is detected in the spatial location
    - Hard to optimize because each neuron have an optimized input and number of neurons can be large
  - let a'_{ij} be the activation vector for random input X_0, a_{ij} for input sentence $\check{X}$,
    - define objective function as $\max a_{ij} \cdot a'_{ij}$
    - Incorporate the activation vector of a spatial location and optimize the input for the whole spatial location

Methods

- Regularization
  - To converge the updating procedure
  - Let $\widehat{X_0}$ denote the receptive field of the spatial location, and l and r are the leftmost and rightmost indices in the field, we have $\widehat{X_0} = [x_{0l}, \cdots, x_{0i}, \cdots, x_{0r}]$
  - By adding regularization terms, objective function become $\max a_{ij} \cdot a'_{ij} - \lambda_1 \left\| \widehat{X_0} \right\|_2^2 + \lambda_2 Sim(\widehat{X_0})$
    - L2 regularization, encourage features with high contribution to maximization increase faster
    - Similarity regularization, encourage different vector representations in X0 to be semantically similar

$$Sim(\widehat{X_0}) = \frac{1}{N} \sum_{\forall i,j} \frac{x_{0i}}{\|x_{0i}\|_2} \cdot \frac{x_{0j}}{\|x_{0j}\|_2}, \qquad (7)$$

where $\cdot$ refers to dot product of vectors, $N = r - l + 1$ and $i, j \in [l, r]$.

Methods

- Visualization of Optimized Inputs
  - Mapping the optimized input back to words to interpret
  - Optimized input are discrete numerical vector that can not be mapped to word directly
  - Use Cosine Similarity find words with highest similarity to optimize input
  - Define overall approximation as $$x_{overall} = \frac{1}{N} \sum_{i=l}^{r} x_{0i}.$$
  - Visualize vector representation and its neighbor by dimension reduction technique like 2d t-SNE or principal component analysis

Experimental Studies

- Dataset
  - MR Dataset, movie review data with sentiment label “positive” “negative”
  - AG’s News Dataset, constructed from AG’s corpus of news articles with topic label “World”, “Sports”, “Business” or “Sci/Tech”
- Experimental Setup
  - CNN model, see Table 2
  - Interpretation, find top 3 spatial location in the first layer
  - Preprocessing, similar to NLP application(Kim 2014) , did not convert words to lower case

	MR	**AG’s News**
Length	56	195
Conv num	3	4
Kernel size	5	5
Conv channel	128, 64, 32	512,256,128,64
Activation	Relu	Relu
Embedding	300	300
Pre-train	Word2vec	Word2vec
Learning rate	2e-4	5e-4
Batch size	128	64

Table 2: The CNN models we used for the MR dataset and AG’s News dataset. Different columns refer to the network settings for different dataset. *Length*: the length of input sentence; *Conv num*: the number of 1D convolutional layers in the model; *Conv channel*: the number of channels for convolutional layers; *Activation*: activation function in convolutional layers; *Embedding*: dimension of word embedding; *Pre-train*: the type of pre-trained word embedding employed.

Experimental Studies

- Visual Interpretation Results
  - Accuracy

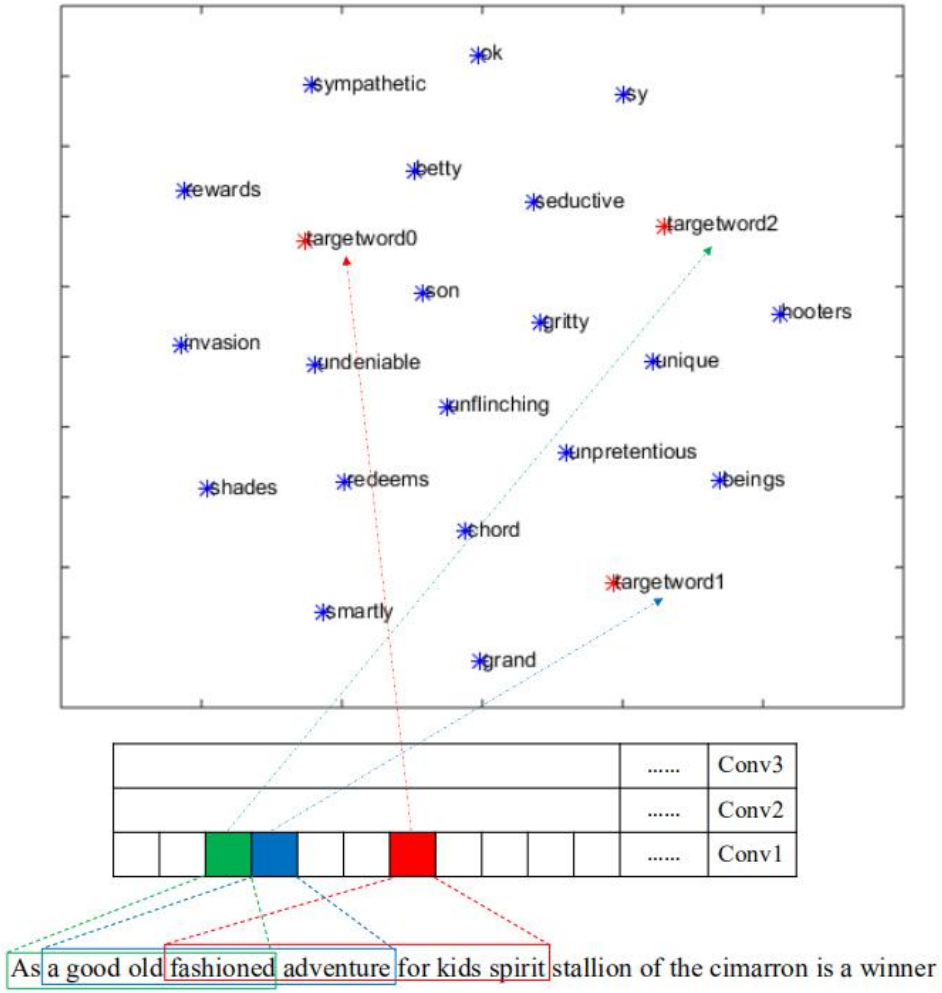
Dataset	MR	AG's News
Our CNN model	79.96%	92.05%
Baseline CNN model	81.50%	91.45%

Table 3: Comparison of prediction accuracy between the CNN models we build and the baseline CNNs.

  - MR dataset
    - "As a good old fashioned adventure for kids spirit stallion of the Cimarron is a winner";
    - "Plays like one of those conversations that comic book guy on the simpsons has"
  - AG's News Dataset
    - "Looking at his ridiculously developed upper body, with huge biceps and hardly an ounce of fat, it's easy to see why Ty Law, arguably the best cornerback in football, chooses physical play over finesse. That's not to imply that he's lacking a finesse" and
    - "Jet Propulsion Lab – Scientists have discovered irregular lumps beneath the icy surface of Jupiter's largest moon, Ganymede"

Experimental Studies

- Visual Interpretation Results
  - Spatial location selected and corresponding words
  - where red, blue, green means contribution from high to low
  - The bounding boxes correspond to the receptive field of that location.
  - The top part shows the t-SNE visualization.
- MR dataset
  - Mostly adjectives and adverbs
  - Interpreting: the information detected by these spatial locations is positive and these spatial locations have high contribution to the final decision so that the final prediction is positive



Experimental Studies

- Visual Interpretation Results
- AG's News Dataset
  - Mostly noun
- Interpreting: Most of them are highly related to the topic "sports", for example, "Toni", "Elarton" and "Fahrenhorst" are names of famous players; "Toulouse " and "Newcastle" are names of famous sports teams

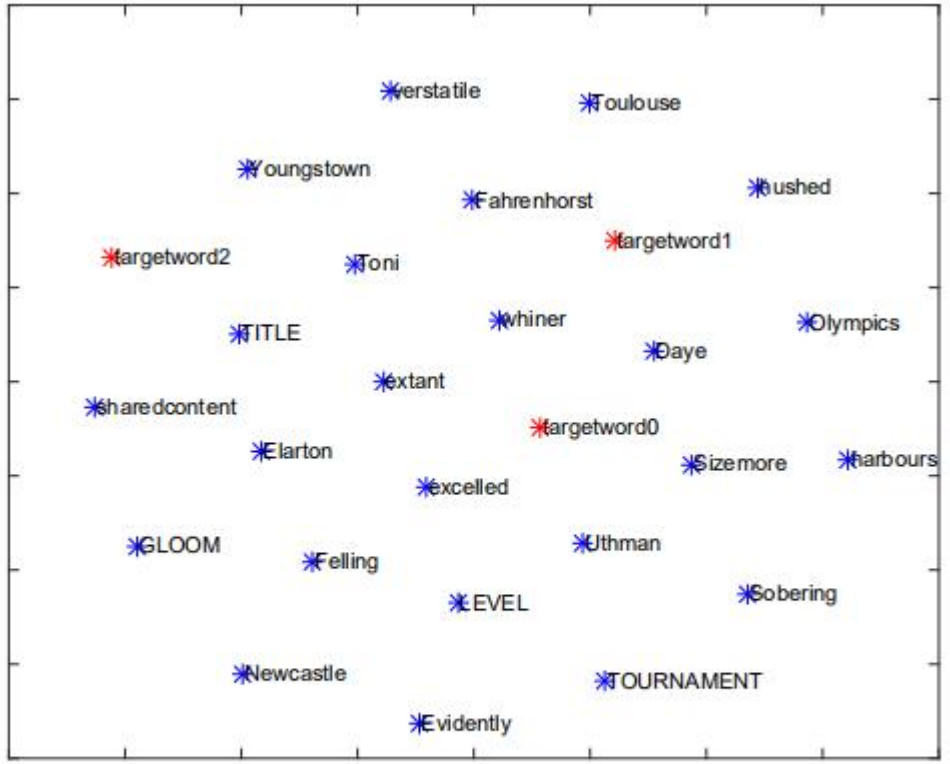


Figure 4: The visualization interpretation result of the first example for the AG's News dataset.

Experimental Studies

- Evaluation of Interpretability
  - the hidden layers word representation should convey similar high-level meaning compared with the original input sentence
  - Given input sentence X with label c, we obtain k spatial location with highest contribution, for each location we use m neighbor words to represent its meaning, so totally we have km words to interpret the hidden layer, denoted as X'
  - By feeding X' into the same network , we get a label c', if c' equals to c then we get similar high-level meaning, call it "matching"
  - Evaluation result:

Dataset	MR	AG's News
Matching rate	0.934	0.843

Table 4: The matching rates for the MR dataset and AG's News dataset.

Conclusion

- An approach to understand the meaning of CNN hidden location in NLP models
- Instead of exploring the contribution of word individually, choosing to explore the importance of different hidden spatial locations
- Use saliency map to find high contribution location, use optimization to explore what this location is looking at
- Propose new way to represent optimized word vector by finding words with similar vector

Discussion

- This paper use 2d word representation, can consider the possibility of 3d
- No clear explaination about why focus on spatial location of convolution neural nets rather than layer, neuron, word input
- No explaination about parameter, why choose 3 spatial location and 5 similar words, why only choose the first layer
- It's not very clear about the similarity regularization term, the similar word can already be calculated by cosine similarity, why we need the optimized input to have similar word vector?
- Why use the dot product of two activation vectors as the main objective in the optimization? Is there any other better way to do that?
- Why use Cosine similarity instead of other?
- There is a interesting relationship between the receptive field and the approximate X' sentence no explore
- We can find diversity even we have optimized input in one direction, the diversity can help capture the result
- There must be a relationship between the embedding and the network which is also very important