Large Scale Item Categorization in e-Commerce Using Multiple Recurrent Neural Networks

Sadia Chowdhury and Md Tahmid Rahman Laskar

Monday, 26th November, 2018

Outline

- Introduction
- Related Works
- Description of Item Metadata Attributes
- Deep Categorization Network (DeepCN) Model
- Dataset and Parameters
- Performance Measure and Comparison
- Advantages and Limitations
- Conclusion and Future Work

2

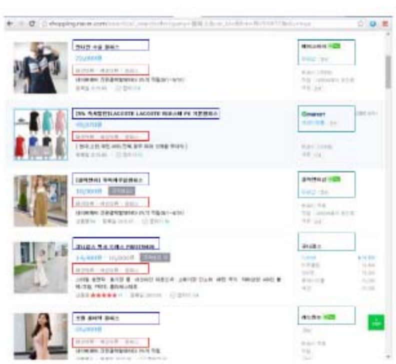
Introduction

- Recent advances in web and mobile technologies have increased the e-commerce markets.
- Precise item categorization in large e-commerce websites such as *eBay, Amazon,* or *Naver shopping* is a big challenge.
- Each item in e-commerce websites is represented by metadata attributes such as *title, category, image, price, etc.*
- Most metadata of items are represented as textual features.
- Item categorization is a text classification problem. It can be done automatically from the textual metadata information.
  - Automatic item categorization can reduce time and economic costs.
  - Categorization accuracy has large influences on customer satisfaction.

YORK UNIVERSITÉ UNIVERSITY

3

Example of an E-Commerce Website (NAVER Shopping)



- Red boxes denote the category name.
- Blue boxes denote the item name.
- Green boxes denote the shopping mall name of the item.

YORK UNIVERSITÉ UNIVERSITY

4

Challenges in Item Categorization in E-Commerce Websites

- Data distribution can have a long tail:
  - Many leaf categories which include only a few items.
  - Leaf categories in a long-tail position are difficult to categorized correctly. (Imbalanced data problem)
- Metadata may include noisy information:
  - Sellers may give incorrect metadata information.
- Scalability issue:
  - A model might initially show good performance, but accuracy could decrease with the addition of new items.

For the above reasons, applying text classification technique for item categorization in e-commerce is more challenging that the traditional text classification problem.

5

Related Works

- Algorithms applied for item categorization:
  - Support Vector Machines (SVM)
  - Naïve Bayes Classifier
  - Decision Trees
  - Latent Dirichlet Allocations (LDA)
  - Limitations of these algorithms: Scalability, sparsity, skewness.
- Other approaches with their limitations:
  - Hierarchical item categorization method based on unigram:
    - Limitation: Sparsity problem, difficult to understand the meaning of given word sequences.
  - Taxonomy-based approach:
    - Limitation: Prior knowledge of taxonomy of item categories are required.

6

Description of Item Metadata Attributes

- Sellers often register data by omitting many attributes. In this study, therefore, only **six essential attributes** are considered.
- An item ***d*** consisting of its leaf category label ***y*** and attribute vector ***x*** can be represented as following:

$$d = \{\mathbf{x}, y\} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(6)}, y\}$$

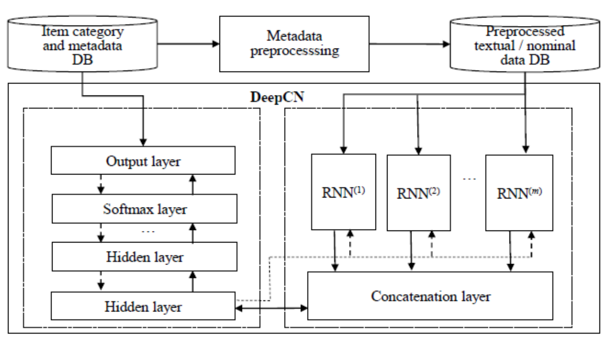
- By treating all the nominal values as textual words, the metadata attribute of an item **i** can be defined as the sequence of textual words as following:

$$\mathbf{x}^{(i)} = \{x_1^{(i)} x_2^{(i)} \ldots x_n^{(i)}\}$$

Var	Attributes	Values	Example
$\mathbf{x}^{(1)}$	Item name	Word sequence	Stylish wallet
$\mathbf{x}^{(2)}$	Brand name	Word sequence	Louis quatorze
$\mathbf{x}^{(3)}$	High-level category	Word sequence	Miscellaneous goods / Women goods
$\mathbf{x}^{(4)}$	Shopping Mall ID	Nominal	A023012
$\mathbf{x}^{(5)}$	Maker	Word sequence	Louis quatorze
$\mathbf{x}^{(6)}$	Image signature	Nominal	38720307
y	Leaf category	Nominal	3423(Women's wallet)

Proposed Model

- **Deep Categorization Network (DeepCN) Model**

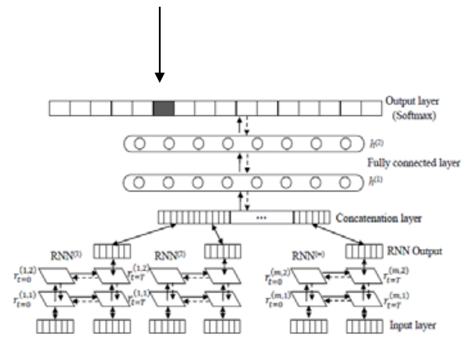


- The Output layer will produce the probability of the leaf category for the given textual metadata.

Deep Categorization Network (DeepCN) Model

- **DeepCN** consists of multiple RNNs and fully connected layers, a concatenation layer, one softmax layer and an output layer.
- Each **RNN** is dedicated to one attribute of the metadata. So, for **m** attributes, there are **m** RNNs.
- The RNNs generate **real-valued feature vector** from the given textual metadata represented by word sequences.
- All the outputs generated from the RNNs are concatenated into one vector by the **concatenation layer,** which then moves to the **fully connected layers**.
- Each node in the **output layer** contains the probability of each leaf category.
- The **Softmax** function provides the probability of each output node in the output layer.

The leaf category having maximum probability



9

Deep Categorization Network (DeepCN) Model

- Activation function of the *m*-th RNN for *n*-th hidden layer:
  - The number of the RNN: **m,** Weight matrix between the *(n-1)*-th layer and the *n*-th layer: **W**, The number of the layer: **n,** Activation function: **f,** Timestamp: **t,** Bias Unit: **b**

$$ {}^{R}_{m}\mathbf{h}_t^{(n)} = {}^{R}_{m}f^{(n)}\left({}^{R}_{m}W^{(n-1)n}\,{}^{R}_{m}\mathbf{h}_t^{(n-1)} + {}^{R}_{m}W^{nn}\,{}^{R}_{m}\mathbf{h}_{t-1}^{(n)} + {}^{R}_{m}\mathbf{b}^{(n)}\right) $$

- Activation function of the *m*-th RNN for the 1st hidden layer:
  - Input Vector: **x**

$$ {}^{R}_{m}\mathbf{h}_t^{(1)} = {}^{R}_{m}f^{(1)}\left({}^{R}_{m}W^{x1}\mathbf{x} + {}^{R}_{m}W^{11}\,{}^{R}_{m}\mathbf{h}_{t-1}^{(1)} + {}^{R}_{m}\mathbf{b}^{(1)}\right) $$

10

Deep Categorization Network (DeepCN) Model

- The Output vector **u** in the concatenation layer:

$$\mathbf{u} = {}^{R}_{1}\mathbf{h}^{(n)}_{T_1} \circ \cdots \circ {}^{R}_{m}\mathbf{h}^{(n)}_{T_m}$$

- The Activation function of the ***a*-th** layer of the Fully connected layer **F:**

$$^{F}\mathbf{h}^{(a)} = {}^{F}f^{(a)}\left({}^{F}W^{a(a-1)}\,{}^{F}\mathbf{h}^{(a-1)} + {}^{F}\mathbf{b}^{(a)}\right)$$

- The Activation function of the **1st** layer of the Fully connected layer **F:**

$$^{F}\mathbf{h}^{(1)} = {}^{F}f^{(1)}\left({}^{F}W^{21}\mathbf{u} + {}^{F}\mathbf{b}^{(1)}\right)$$

- The Softmax function ***y*** in the ***k***-th output node for the ***l***-th fully connected layer:

$$P(y_k \mid {}^{F}\mathbf{h}^{(l)}) = \frac{\exp\left(\mathbf{w}_k\,{}^{F}\mathbf{h}^{(l)}\right)}{\sum_{y\in Y}\exp\left(\mathbf{w}_y\,{}^{F}\mathbf{h}^{(l)}\right)}$$

YORK U UNIVERSITÉ UNIVERSITY

11

Deep Categorization Network (DeepCN) Model

- **Hyperbolic tangent function** is used for both RNN and Fully Connected Layer as it performs better than the sigmoid function in RNN learning [1].

- **Categorization error:**

*One-hot-encoding vector of the real category of the **n-th** item*

$$E = \frac{1}{2}\sum_{n=1}^{N}\left\{y^{(n)} - \hat{y}^{(n)}\right\}^2 \equiv \frac{1}{2}\sum_{n=1}^{N}\left\|\mathbf{y}^{(n)} - \hat{\mathbf{y}}^{(n)}\right\|^2$$

The calculated softmax probability vector

[1] Jozefowicz, R., Zaremba, W., and Sutskever, I. 2015. An Empirical Exploration of Recurrent Network Architectures. In *Proceedings of the 32nd International Conference on Machine Learning* (*ICML-15*), 2342-2350.

YORK U UNIVERSITÉ UNIVERSITY

12

Deep Categorization Network (DeepCN) Model

- Weight updates in the fully connected layer:

$$\mathbf{w}_i = \mathbf{w}_i - \eta \frac{\partial E}{\partial \mathbf{w}_i} \text{ and } \frac{\partial E}{\partial \mathbf{w}_i} = \delta_i \mathbf{x}$$

$$\delta_i = \begin{cases} \left(y_i^{(n)} - \hat{y}_i^{(n)}\right)\left(1 - \tanh^2(net_i)\right), \text{ if } i \in \mathbf{o} \\ \left(\sum_{j \in J} \delta_j w_{ij}\right)\left(1 - \tanh^2(net_i)\right), \text{ if } i \in \mathbf{h} \end{cases}$$

o denotes the node set of output layer

h denotes the node set of hidden layer

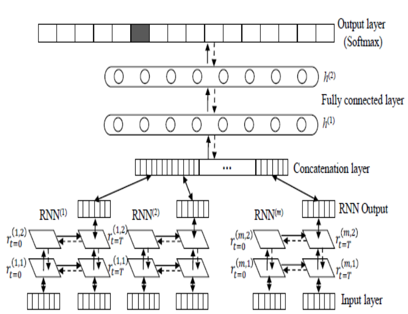
- Weight updates in the RNN:
  - All the weights of the RNNs are updated by backpropagation through time (BPTT) [2].

$$\mathbf{w}_i^{(n-1)n} = \mathbf{w}_i^{(n-1)n} - \eta \sum_{t=1}^{T} \delta_i(t)\, {}^{R}\mathbf{h}_t^{(n-1)}$$

[2] Werbos, P. J. 1990. Backpropagation through Time: What It Does And How to Do It. In *Proceedings of the IEEE*, 78, 10, 1550-1560.

YORK UNIVERSITÉ UNIVERSITY

DeepCN Algorithm



Algorithm 1: Learning of DeepCN

D' / D^{te}: Minibatch trainng dataset / test dataset
M: The number of RNNs (= # of used metadata attributes)
U: The set of concatenated vectors
E: Categorization error
${}^{R}E_m$: Backpropagated error of the m-th RNN
V_m: Word sequence embedding vector set by the m-th RNN
S: The number of separated minibatch datasets
${}^{F}\theta^i$ / ${}^{R}\theta_m^i$: Model parameters of FC layers and the m-th RNN at the i-th iteration
$MAXITER$: The number of maximum iteration for learning

$({}^{R}\theta_1^0, \ldots, {}^{R}\theta_M^0, {}^{F}\theta^0) \leftarrow$ Initialize(M)
for i =1 **to** $MAXITER$
  for j =1 **to** S
    $D' \leftarrow$ GetMiniBatch(D, $|D'|$);
    for m=1 **to** M
      $V_m \leftarrow$ DeepCN_RNN_Forward(D', ${}^{R}\theta_m^{i-1}$);
    endfor
    U $\leftarrow$ Concatenate($V_1, \ldots, V_M$)
    $E \leftarrow$ DeepCN_FC_Softmax_Forward(U, ${}^{F}\theta^{i-1}$);
    $({}^{F}\theta^i, {}^{R}E) \leftarrow$ DeepCN_ FC_Softmax_Backward(E, ${}^{F}\theta^{i-1}$);
    for m=1 **to** M
      ${}^{R}E_m \leftarrow$ SeparateError(${}^{R}E$, m)
      ${}^{R}\theta_m^i \leftarrow$ DeepCN_RNN_Backward(${}^{R}E_m$, ${}^{R}\theta_m^{i-1}$);
    endfor
  endfor
  EvaluateDeepCN(D^{te}, ${}^{R}\theta_1^0, \ldots, {}^{R}\theta_M^0, {}^{F}\theta^0$)
endfor

YORK UNIVERSITÉ UNIVERSITY

Dataset and Parameters

Dataset:

- Large data set: 94.8 million items; 4,116 leaf categories and 11 high level categories; collected from “NAVER SHOPPING”.
- Training data ratio: 8/11
- Validation data ratio: 2/11
- Test data ratio: 1/11
- Preprocessing: Removed rare words, parenthesis, quotation, period etc.

Parameters:
Parameters are selected based on experimental analysis.

- Learning rate: 0.001
- Momentum: 0.9
- Minibatch size: 100

15 Stochastic Gradient Descent with Momentum: https://arxiv.org/abs/1609.04747

Dataset Overview for each High Level Category

High-level category	# of leaf categories	Training (8)	Validation (2)	Test (1)	Total	Data size / leaf category
Fashion clothing	103	7,201,594	1,800,399	900,200	9,902,193	**96,138**
Miscellaneous goods	255	12,798,623	3,199,656	1,599,829	17,598,108	**69,012**
Cosmetic / Beauty	156	2,595,834	648,959	324,480	3,569,273	22,880
Digital & home electronic	642	13,341,844	3,335,462	1,667,731	18,345,037	28,575
Furniture / Interior	335	5,159,216	1,289,805	644,903	7,093,924	21,176
Childbirth / Infant care	473	5,162,732	1,290,684	645,342	7,098,758	15,008
Foods	432	1,610,087	402,522	201,261	2,213,870	**5,125**
Sports / Leisure	459	5,255,248	1,313,813	656,907	7,225,968	15,743
Life / Health	1,115	15,104,282	3,776,071	1,888,036	20,768,389	18,626
Trip / Culture	61	748,506	187,127	93,564	1,029,197	16,872
Tax free goods	85	25,196	6,299	3,150	34,645	**408**
Total	4,116	69,003,162	17,250,797	8,625,403	94,879,362	23,051

Bold faces denote high-level categories with a very skewed data size per leaf category. Values in parenthesis of table header mean the ratios of data separated for learning deepCN.

16

Performance Measurement and Comparison

- **Performance measurement:** Relative Accuracy.
- **Relative accuracy** of a model Θ for given data *D* is defined as the ratio of an *estimated accuracy* to *basis accuracy.*
- **Basis accuracy** is the accuracy of the model using all metadata attributes.

$$\text{Relative Accuracy} = \tilde{\psi}(D;\theta) = \frac{\psi(D;\theta)}{\bar{\psi}}$$

Estimated Accuracy

Basis Accuracy

Comparison: Compared with two other approaches.
- **DCN - 1R :** Deep Categorization Network with Single RNN.
- **BN_BoW :** Bayesian Network using Bag of Words.

17

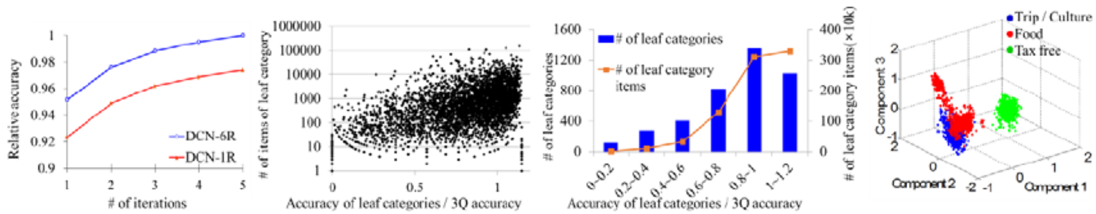
Relative Accuracies of Three Methods for Various High Level Categories

High Level Category	DCN – 6R	DCN – 1R	BN_BOW
Fashion Clothing	**1.004**	0.984	0.696
Miscellaneous goods	**0.895**	0.866	0.443
Cosmetic / Beauty	**1.011**	0.976	0.823
Digital & home electronic	**1.108**	1.091	0.852
Furniture / Interior	**0.983**	0.952	0.665
Childbirth / Infant care	**0.956**	0.926	0.740
Food	**1.033**	1.005	0.791
Sports / Leisure	**1.016**	0.985	0.713
Life / Health	**0.992**	0.963	0.628
Trip / Culture	**1.197**	1.193	0.930
Tax free goods	**1.027**	1.008	0.101

*Red values denote the poorest accuracies.

18

Categorization Performance



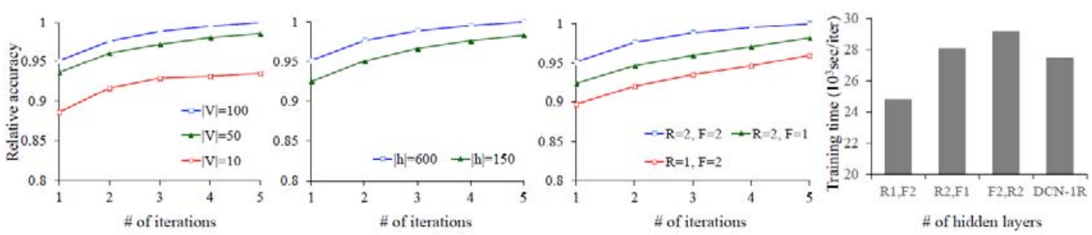
(a) Learning curve
(b) Distribution of item number of leaf category for accuracy
(c) Distribution of the number of leaf categories and leaf category items for accuracy
(d) PCA plot of word-embedding vectors of metadata

a) Relative Accuracy of DCN – 6R is better than DCN – 1R.
b) Leaf categories having # of items more than 10000 produce more accurate result.
c) Accuracy improves with the increase of # of items in a leaf category.
d) Concatenated word embedding vectors of metadata are separately scattered in a three dimensional space.

YORK UNIVERSITÉ UNIVERSITY

19

Categorization Performance



(a) Word vector size
(b) Number of hidden nodes
(c) Number of hidden layers
(d) Training time

Effects on **relative accuracy (a), (b), (c)** and **training time (d)** based on variations of **word vector size**, **number of hidden nodes**, **number of hidden layers** in **RNN layers** and **Fully Connected layers**.

YORK UNIVERSITÉ UNIVERSITY

20

Effects on Accuracy after Excluding some Attributes

High Level Category	Excluding Image Signatures	Excluding Image Signatures + Shopping mall id
Fashion Clothing	0.983	0.901
Miscellaneous goods	0.958	**0.870**
Cosmetic / Beauty	0.981	0.908
Digital & home electronic	0.975	0.929
Furniture / Interior	0.965	0.885
Childbirth / Infant care	0.970	0.882
Food	0.980	0.890
Sports / Leisure	0.977	0.888
Life / Health	0.966	0.884
Trip / Culture	0.997	0.974
Tax free goods	0.994	**0.814**

*Bold values denote the poorest accuracies.

YORK UNIVERSITÉ UNIVERSITY

21

Advantages and Limitations

- **Advantages:**
  - DCN – 6R performs **significantly** better than Bayesian-BoW.
  - DCN – 6R also performs better than DCN – 1R.
- **Limitations:**
  - Performances for very long-tail leaf categories are not satisfactory. Can be improved using LSTM or GRU.

YORK UNIVERSITÉ UNIVERSITY

22

Conclusion and Future Work

- In summary, DeepCN consists of multiple RNNs and fully connected layers, a concatenation layer, one softmax layer and an output layer.
- Each metadata item has a dedicated RNN.
  - Ambiguity emerging from concatenation of semantically heterogeneous word sequences have been overcome.
  - Keeps the length of word sequences short.
- Number of RNN layers has more effects than number of Fully Connected Layers in terms of categorization accuracy and learning time.
- Metadata attributes such as image signatures and shopping mall id have effect on categorization.
- DeepCN can be applied to various text classifications such as sentiment analysis and document classification.
- CNN for item images can further improve the performance instead of using image signatures.

YORK UNIVERSITÉ UNIVERSITY