Online Similarity Learning with Feedback for Invoice Line Item Matching

Chandresh Kumar Maurya
Assistant Professor

Eötvös Loránd University, Budapest, Hungary

June 16, 2019

Publications (Journals)

1. C. Maurya; D. Toshniwal; G. Venkoparao, "Distributed Sparse Class-Imbalance Learning and its Applications," in **IEEE Transactions on Big Data** , vol.PP, no.99, pp.1-1 https://doi.org/10.1109/TBDATA.2017.2688372
2. Chandresh Kumar Maurya, Durga Toshniwal, Gopalan Vijendran Venkoparao, Online sparse class imbalance learning on big data, **Neurocomputing**, Volume 216, 5 December 2016, Pages 250-260, ISSN 0925-2312, http://doi.org/10.1016/j.neucom.2016.07.040. (IF 3.317)
3. Chandresh Kumar Maurya, Durga Toshniwal, Large-Scale Distributed Sparse Class-Imbalance Learning with Application to Anomaly Detection, **Information Sciences**, Volume 456, 4 May 2018, Pages 1-12, ISSN 0020-0255, Elsevier, https://doi.org/10.1016/j.ins.2018.05.004 (IF 4.832)
4. Creative Tagline Generation Framework for Product Advertisement, Chandresh Kumar Maurya et al., in **IBM Journal of Research & Development**) (SCIE IF 0.6)
5. **Online Similarity Learning with Feedback for Invoice Line Item Matching** (submitted to IEEE TKDE)

Publications (Conferences/Workshops)

1. Prediction of Invoice Payment Status in Account Payable Business Process. Tarun Tater, Sampath Dechu, Senthil Mani, and Chandresh Kumar Maurya, **International Conference on Service-Oriented Computing (ICSOC)** 2018, China.
2. Anomaly Detection via Distributed Sparse Class-Imbalance Learning. Chandresh Kumar Maurya, Durga Toshniwal, and Vishal Agarwal, (presented in International Conference on Machine Learning, ICML 2016 workshop on Anomaly detection, NY, USA)
3. Online Anomaly Detection via Class-Imbalance Learning, Chandresh Kumar Maurya and Durga Toshniwal, in **International Conference on Contemporary Computing (IC3)** , organised jointly by JIIT Noida and University of Florida, USA, Sep 2015.
4. Anomaly Detection in Nuclear Power Plant Data using Support Vector Data Description, Chandresh Kumar Maurya and Durga Toshniwal, in IEEE TechSym at IIT Kharagpur, Feb 2014.
5. Fuzzy Inference System for Internet Traffic Load Forecasting, Chandresh Kumar Maurya and Sonajharia Minz. In the Proceedings of National Conference of Computing & Communications (NCCCS)- 2012. DOI:10.1109/NCCCS.2012.6413010. .
6. Anomaly Detection in Streaming Data using Online Non-negative Matrix Factorization, Chandresh Kumar Maurya, Arun Chauhan and Durga Toshniwal, poster presentation at International workshop on machine learning & Text analytic(MLTA 2013), South Asian University, New Delhi. (Best presentation Award)

Patents While @IBM

1. A System and Method for Automatic Adjustment of Brightness of Mobile Devices based on Visual Sight (**under FILE**)
2. A System and Method for Recommending Popular Features for a Product based on Crowd-source Reviews of the Product and Competitor Product (**Rated search-2 by IBM, under search**)
3. A System and Method for Finding Best Accommodation using Deep Asthetic Features in Multimodal Data from Social Networks. (**Defensive publication**)
4. A System and Method for Automatic Compliance Checking in Clinical Process using Deep Multi-task Learning. (**Defensive publication**)
5. A system and Method for Similarity Learning with Heterogeneous Catalogues and Taxonomies for Invoice and PO Line Item Matching (under review).
6. An Intelligent and Proactive Reminder System using Multi-Modal Data through AI (under review).
7. A System and Method for Consistency Verification of Products Information on E-commerce Portals (under review).

Outline

Similarity Learning with Feedback for Invoice Line Item Matching

Procure to Pay process

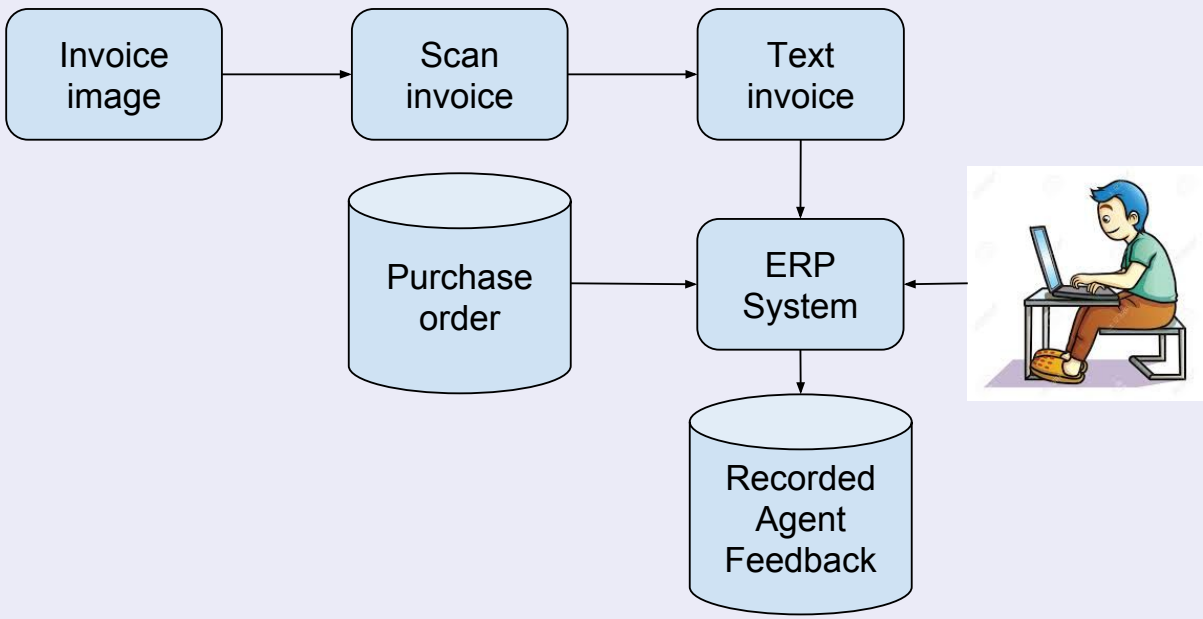


Figure: Invoice line item matching in P2P business process

The problem

Table: An example of line item matching

Invoice	PO
TRES 739mL CD KER Smooth	1. TRES 0.739L CD KER Smth 2. Tres Soya Smooth Conditioner 150 gm
5x200ml Fruit Juice 100% - Tropicana, Apple	1. Tropicana 100% Apple Juice - 1L 2. Fruit Juice 500ml - Tropicana, Custard Apple
Battery Distilled Water Replacement	1. Battery Maintenance Services 2. Battery Warranty extension

Our Contribution

- Two approaches are proposed to match descriptions using domain knowledge captured in the users feedback. First approach learns similarity rank when recorded users feedback has relative ranking of description matches and second approach uses binary classification when users recorded feedback is absolute match/no-match between pair of descriptions.
- To circumvent the issue of hierarchical relationship among items in the invoice and PO line items, we present an algorithm that makes use of product tax- onomy and catalog.
- Proposed approaches are evaluated on real-world description datasets e.g. invoice data from internal clients, publicly available product description datasets and compare the results with the state-of-the-art approaches applied to natural language sentences and show limitations of the existing work. Additionally, we also evaluate the proposed approachs using different kinds of description representations such as character n-gram (n=2 to 5), directly encoding the description assuming them as sentence via pre-trained models from Infersent [4] and Googles universal sentence encoder.

Related Work

- In [Chechik et al., 2010], the author present an online relative similarity learning task for images. They learn a metric W based on the triples of the images within the passive-aggressive learning framework [Crammer et al., 2006]
- In [Liu et al., 2015], the author uses support vector regression with various features such as WordNet-Based features, corpus-based features, Word2Vec-based feature, Alignment-based features, and Literal-based features to predict the similarity between short English sentences.
- In [Kashyap et al., 2016] proposes a robust distributional word similarity component that combines the LSA and ML augmented data from several linguistic resources.
- In [Kutiyanawala et al., 2018, Hu et al., 2018] the author propose matching query to items in the product catalog.

Proposed Algorithm

Algorithm 1: Similarity Learning with Feedback (SLF) Algorithm

Input: Aggressiveness parameter C/learning rate η, Invoice and PO line items.

Output: W_T

1 . **Initialize:** $W_0 = I$(identity Matrix) or weight vector $\mathbf{w}_0 = \mathbf{0}$.

2 **for** $t := 1, ..., T$ **do**

3 Apply Lexical Normalization as discussed in sec. 3 to the query string (e.g. Invoice string).

4 Receive K strings via fuzzy matching from the pool for query string s.

5 Extract noun phrases from string pair. If noun phrases did not match, return fuzzy matching score 0.

6 Present the pair of strings (s, s_i) to the agent where s_i is the best fuzzy matching string.

7 if the agent did not like the pair and gives negative vote, randomly sample a string s_j from the remaining pool of strings.

8 if the agent prefers the pair (s, s_j) more than the pair (s, s_i), we form triple of strings (s, s_i, s_j). If the agent labels pair (s, s_i) as dissimilar and the pair (s, s_j) as similar, we form data for binary classification.

9 **Update:**

$$\text{Ranking Similarty} \begin{cases} W_{t+1} = W_t + \tau_t U_t \\ \tau_t = \min(C, \ell_t^1 / \|U_t\|^2) \\ U_t = [s^1(s_j - s_i)..s^d(s_j - s_i)]^T \end{cases}$$

OR

10 Classification Similarity

11 **end**

Online Metric Similarity Learning

We want to learn a function $f(\cdot, \cdot)$ that assigns high score to pairs (s, s_j) than the pair (s, s_i) whenever the agent prefers (s, s_j) more than (s, s_i). Assume that the function f has a bilinear form shown in (1).

$$f_W(s_i, s_j) := s_i^T W s_j \tag{1}$$

where the matrix $W \in R^{d \times d}$. Our objective is to find the function $f(\cdot, \cdot)$ such that all the triplet strings satisfy the contraint in (2).

$$f_W(s, s_j) \geq f_W(s, s_i) + 1 \tag{2}$$

Contd...

The constraint in (2) leads to the following loss function.

$$\ell_t^1(s, s_i, s_j) = \max(0, 1 - f_W(s, s_j) + f_W(s, s_i)) \tag{3}$$

Following [Crammer et al., 2006], we can plug the above loss in passive-agressive algorithm as shown in (4).

$$\begin{aligned} W_{t+1} &= \text{argmin}_W \|W - W_t\|_{fro} + C\xi \\ \text{s.t.} \quad & \ell_t^1(s, s_i, s_i) \leq \xi \quad \text{and} \quad \xi \geq 0 \end{aligned} \tag{4}$$

Datasets

Table: Summary of datasets used in the experiment

Dataset	#Train	# Test	#Features
Invoice	370	184	3649
Amazon Electronics	9368	4683	35327
Amazon Automative	21107	10553	40123
Amazon Home	21887	10943	46453
Flipkart	9417	4708	19400
SNLI	121895	60947	55956
SICK	3865	1932	18379
STS	1426	713	16110

Preprocessing

Invoice data consists of invoice strings (s). We have the corresponding PO strings (s_j) a.k.a second string) as well. Since, there is no third string (s_i) available, we manually curated and generated third string from the second one (PO string) under the following assumption so that third string is less similar to the invoice string compared to the PO string. The following rules are derived during manual curation of the invoice data:

1. Common antonyms such as men vs women.
2. Small delta Numeric addition or deletion - for second string
3. Large delta Numeric addition or deletion - for third string
4. Replace Brand names, if applicable
5. Replace Product names, if applicable
6. String Manipulation such as insertion/deletion/substitution of random character and shuffle words

Contd...

An example of string triple from invoice data look like as follows:

s: 12z Dove Men US 2in1 FRts

s_j: 11z Dove Men US 2in1 Frts

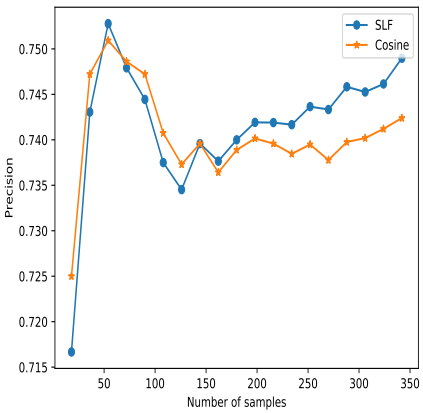
s_i: 12z Dove women US 2in1 Shampoo

Experimental Testbed and Setup

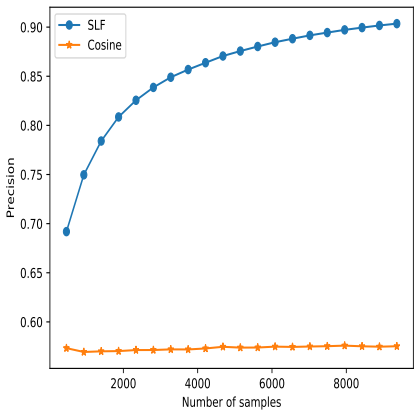
Proposed approach is compared against the following methods:

- Cosine Similarity
- Li's method [Li et al., 2006]
- UMBC [Han et al., 2013]
- Siamese method [Neculoiu et al., 2016]

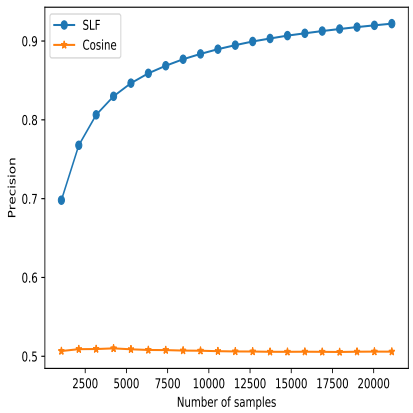
Learning behavior of SLFR



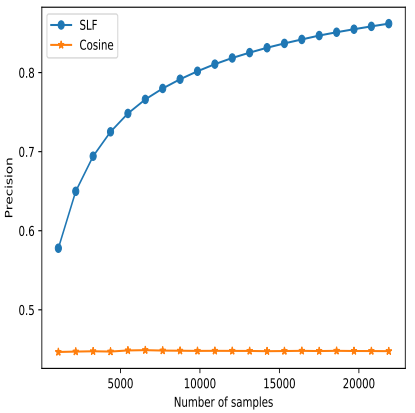
(a) Invoice



(b) Amazon Electronics



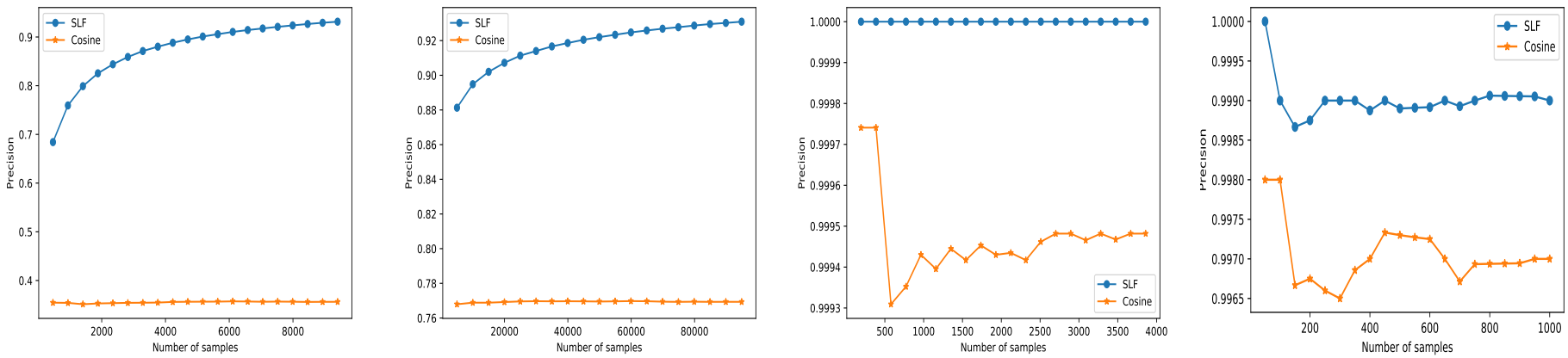
(c) Amazon Automotive



(d) Amazon Home

Figure: Evaluation of online average of *recall* over various benchmark data sets. (a) invoice (b) Amazon electronics (c) Amazon Automotive (d) Amazon Home

Contd...



(a) Flipkart (b) SNLI (c) SICK (d) STS

Figure: Evaluation of online average of *recall* over various benchmark data sets. (a) Flipkart (b) SNLI (c) SICK (d) STS

Generalization Performance of SLFR

Table: Evaluation on the test data (average precision over 20 runs) using three encoding schemes

[]DataEncoding	Tf-idf	Infersent	USE
Invoice	**77.77**	62.22	62.77
Amazon Electronics	**80.70**	65.62	75.53
Amazon Automotive	**90.75**	82.77	50.61
Amazon Home	**86.35**	82.62	77.04
Flipkart	86.83	**87.74**	84.57
SNLI	94.34	94.36	**99.19**
SICK	**99.89**	95.15	99.58
STS	**99.34**	95.60	98.40

Comparative Performance Evaluation of SLFR

Table: Comparative evaluation of average precision on test data

[]DataAlgo	SLFR	Cosine	Li	DKPRo	Siamese
Invoice	**77.77**	73.33	65.64	52.17	13.61
Amazon Electronics	**80.70**	58.29	40.90	41.34	9.64
Amazon Automotive	**90.75**	51.31	39.64	40.64	9.71
Amazon Home	**86.35**	45.70	40.01	35.56	10.78
Amazon Flipkart	**86.83**	28.35	24.67	27.71	5.60
SNLI	**94.34**	76.92	75.61	20.08	23.63
SICK	99.89	**99.94**	99.94	79.34	9.98
STS	**99.34**	99.20	99.21	79.34	8.23

Comparative Performance Evaluation of SLFC

Table: Average Precision, Recall and F-score from GradientBoost classifer on tf-idf, Infersent and USE encoded data

	Tf-df			Infersent		
Dataset	Precision	Recall	F-score	Precision	Recall	F-scor
Invoice	0.29	0.29	0.29	**0.30**	**0.30**	**0.30**
SICK	0.81	0.81	0.81	**0.83**	**0.85**	**0.85**
STS	**0.86**	**0.85**	**0.85**	0.60	0.60	0.60
SNLI	0.73	0.73	.073	0.75	0.75	0.75
Amazon Automotive	0.76	0.75	0.75	**0.88**	**0.87**	**0.87**
Amazon Electronics	**0.72**	**0.72**	**0.72**	0.55	0.55	0.55
Amazon Home	0.74	0.74	0.73	**0.89**	**0.89**	**0.89**
Flipkart	0.87	0.86	0.86	**0.93**	**0.93**	**0.93**

Research Problems

Problem 1 Finding Best Accommodation using Deep Aesthetic features in Multimodal Data from Social Networks

Problem 2 Recommending Popular Features for a Product based on Crowd-source Reviews of the Product and Competitor Product

Problem 3 Intelligent Reminder System using Multimodal Data

Problem 4 Generating image advertisement given a catchy tagline

Bibliography I

Chechik, G., Sharma, V., Shalit, U., and Bengio, S. (2010).
Large scale online learning of image similarity through ranking.
Journal of Machine Learning Research, 11(Mar):1109–1135.

Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., and Singer, Y. (2006).
Online passive-aggressive algorithms.
Journal of Machine Learning Research, 7(Mar):551–585.

Han, L., Kashyap, A. L., Finin, T., Mayfield, J., and Weese, J. (2013).
Umbc_ebiquity-core: semantic textual similarity systems.
In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, volume 1, pages 44–52.

Hu, Y., Da, Q., Zeng, A., Yu, Y., and Xu, Y. (2018).
Reinforcement learning to rank in e-commerce search engine: Formalization, analysis, and application.
arXiv preprint arXiv:1803.00710.

Kashyap, A., Han, L., Yus, R., Sleeman, J., Satyapanich, T., Gandhi, S., and Finin, T. (2016).
Robust semantic text similarity using lsa, machine learning, and linguistic resources.
Language Resources and Evaluation, 50(1):125–161.

Bibliography II


Kutiyanawala, A., Verma, P., et al. (2018).
Towards a simplified ontology for better e-commerce search.
arXiv preprint arXiv:1807.02039.


Li, Y., McLean, D., Bandar, Z. A., Crockett, K., et al. (2006).
Sentence similarity based on semantic nets and corpus statistics.
IEEE Transactions on Knowledge & Data Engineering, (8):1138–1150.


Liu, Y., Sun, C., Lin, L., Wang, X., and Zhao, Y. (2015).
Computing semantic text similarity using rich features.
In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, pages 44–52.


Neculoiu, P., Versteegh, M., and Rotaru, M. (2016).
Learning text similarity with siamese recurrent networks.
In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 148–157.